О синтезе и сложности формул с ограниченной глубиной альтернирования

С. А. Ложкин, В. А. Коноводов

lozhkin@cs.msu.su, vkonovodov@gmail.com

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Рассматриваются (см., например, [1]) схемы из функциональных элементов (СФЭ) и формулы в стандартном базисе Б_0, состоящем из функциональных элементов (ФЭ) &, $\vee$ и $\neg$ веса 1, которые реализуют функции алгебры логики (ФАЛ) $x_1 \cdot x_2$, $x_1 \vee x_2$ и $\bar{x}_1$ соответственно. При этом, как обычно, формулами считаются те одновыходные СФЭ, в которых выход любого ФЭ либо поступает на вход ровно одного (другого) ФЭ, либо является выходом схемы.

Под сложностью $L(S)$ СФЭ или формулы S понимается, как обычно, число ФЭ в ней. Формулу C, которая состоит из r, $r \geqslant 1$, ФЭ $\mathcal{E}^{(1)}, \ldots, \mathcal{E}^{(r)}$ и в которой выход ФЭ $\mathcal{E}^{(i)}$, $i = 1, \ldots, r-1$, является входом ФЭ $\mathcal{E}^{(i+1)}$, будем называть *нетривиальной цепью*, а число r — её *длиной* или *глубиной*. Если при этом последовательность $\varphi^{(1)}, \ldots, \varphi^{(r)}$, где $\varphi^{(i)} \in \text{Б}_0$ — тип базисного ФЭ $\mathcal{E}^{(i)}$, $i = 1, \ldots, r$, имеет вид $\varphi_1, \ldots, \varphi_1, \varphi_2, \ldots, \varphi_2, \ldots, \varphi_a, \ldots, \varphi_a$, где $\varphi_j \neq \varphi_{j+1}$ при всех j, $j = 1, \ldots, a-1$, а число c равно 1 в случае $\varphi_1 = \neg$ и равно 0 в остальных случаях, то разность $a - c$ будем называть *глубиной альтернирования* указанной цепи C. Формулу, которая состоит из единственной вершины, являющейся как её входом, так и её выходом, будем считать *тривиальной цепью*, а глубину и глубину альтернирования такой цепи положим равными 0.

Для СФЭ S определим её глубину $D(S)$ (глубину альтернирования $A(S)$) как максимальную глубину (соответственно глубину альтернирования) цепей, являющихся подсхемами S.

Отметим, что в ряде работ величина $A(\mathcal{F})$ называлась глубиной формулы $\mathcal{F}$ (см., например, [2]), при этом в работе [1] величина $(A(\mathcal{F}) - 1)$ называлась альтернированием формулы $\mathcal{F}$. Что же касается величины $D(\mathcal{F})$, то в большинстве работ, так же как и в данной работе, она считается глубиной формулы $\mathcal{F}$.

Заметим, что любая элементарная конъюнкция (элементарная дизъюнкция), то есть конъюнкция (соответственно дизъюнкция) переменных или их отрицаний, имеет глубину альтернирования 1. При этом глубина альтернирования любой КНФ или ДНФ, которая отлична как от элементарной конъюнкции, так и от элементарной дизъюнкции, равна 2.

Для любого $a \geqslant 2$ определим сложность $L^{(a)}(f)$ ФАЛ f как минимальную из сложностей тех реализующих её формул, глубина альтернирования которых не больше, чем a. Введем далее функцию Шеннона $L^{(a)}(n)$ как максимальную из сложностей $L^{(a)}(f)$, где максимум берется по всем ФАЛ от булевых переменных (БП) $x_1, \ldots, x_n$. Известно, что (см., например, [1]) $L^{(2)}(n) = \frac{3}{2} n \cdot 2^{n-1} - 1$. При этом в работе [2] доказано, что при любом $a \geqslant 3$ справедливы неравенства (здесь и далее все логарифмы — двоичные)[1]

$$\frac{2^n}{\log n}\left(1 - O\left(\frac{1}{\log n}\right)\right) \leqslant L^{(a)}(n) \leqslant \frac{2^n}{\log n}\left(1 + \frac{2\log\log n + O(1)}{\log n}\right),$$

которые устанавливают поведение функции Шеннона $L^{(a)}(n)$ с относительной погрешностью вида $O\left(\frac{\log\log n}{\log n}\right)$. Основным результатом данной работы является получение при любом фиксированном a, $a \geqslant 3$, асимптотических оценок высокой степени точности для функции Шеннона $L^{(a)}(n)$, устанавливающих её поведение с относительной погрешностью $O\left(\frac{1}{\log n}\right)$. Пусть $\log^{[a]} x \stackrel{\text{def}}{=} \underbrace{\log \ldots \log}_{a \text{ раз}} x$.

Теорема 1. *При любых a, $a \geqslant 3$, и достаточно больших n*

$$L^{(a)}(n) = \frac{2^n}{\log n}\left(1 + \frac{\log^{[a-1]} n \pm O(1)}{\log n}\right).$$

Пусть $\log^* x$ — итерированный логарифм числа x, то есть такое наименьшее k, что $\log^{[k]} x \leqslant 1$.

Лемма 1. *Пусть $a \geqslant 3$, тогда при достаточно больших n и любом натуральном L, таком, что $\log^*(L+1) \geqslant a$ и $\log^{[a-1]}(L+1) \leqslant n$, число попарно не эквивалентных формул сложности не больше чем L, имеющих глубину альтернирования не больше чем a и реализующих*

---
[1]Верхняя оценка получена при более точной оценке сложности построенных формул.

функции от БП $x_1, \ldots, x_n$, не превосходит величины

$$\left(\frac{c_a n}{\log^{[a-1]}(L+1)}\right)^{L+1},$$

где c_a — некоторое число, зависящее только от a.

Применением мощностного метода с использованием этой леммы устанавливается

Лемма 2. *Для всех a, $a \geqslant 3$, при достаточно больших n*

$$L^{(a)}(n) \geqslant \frac{2^n}{\log n}\left(1 + \frac{\log^{[a-1]} n - O(1)}{\log n}\right).$$

Пусть $B = \{0,1\}$ и B^n, $n = 1, 2, \ldots,$ — единичный n-мерный куб, а $P_2(n)$ — множество всех ФАЛ $f = f(x_1, \ldots, x_n) : B^n \xrightarrow{f} B$.

Рассмотрим разбиение Δ куба $B^q(x_1, \ldots, x_q)$ на непересекающиеся подмножества $\delta_1, \ldots, \delta_{2^{q-m}}$, где m и q — натуральные числа ($m \leqslant q$). Будем говорить, что разбиение Δ моделирует функции из множества G, $G \subseteq P_2(m)$, с помощью булевых переменных или их отрицаний тогда и только тогда, когда для любой функции g, $g \in G$, и для каждого i, $i = 1, \ldots, 2^{q-m}$, существует переменная x_j, где $1 \leqslant j \leqslant q$, и константа σ, $\sigma \in B$, такие, что $g \equiv x_j^{\sigma}$ на компоненте δ_i. При этом δ_i — «хорошая» компонента, если для каждой такой функции g указанная константа σ равна 1.

Пусть f — произвольная ФАЛ из $P_2(n)$ и пусть $x' = (x_1, \ldots, x_q)$, $x'' = (x_{q+1}, \ldots, x_n)$. Для получения верхней оценки рассматриваются специальные разбиения куба $B^q(x')$ на компоненты $\delta'_1, \ldots, \delta'_{2^{q-m}}$, а куба $B^{n-q}(x'')$ — на компоненты $\delta''_1, \ldots, \delta''_{2^{n-q-t}}$. При этом все элементарные дизъюнкции от переменных x'' моделируются переменными или отрицаниями переменных на любой компоненте куба $B^{n-q}(x'')$, а каждая подфункция функции f от переменных x' на компонентах куба $B^q(x')$ моделируется специальной формулой $\mathcal{F}_{\varphi}(x_{j_1}^{\sigma_1}, \ldots, x_{j_N}^{\sigma_N})$ от некоторых переменных или их отрицаний, имеющей ограниченную глубину альтернирования. Во втором случае разбиения строятся так, что почти все компоненты — «хорошие». Тогда функция f

представима в виде формулы

$$\mathcal{F}_f = \bigvee_{i=1}^{2^{q-m}} \bigvee_{j=1}^{2^{n-q-t}} \mathfrak{B}_\mathfrak{i}(x')\mathfrak{B}_\mathfrak{j}(x'') \mathop{\&}_{\sigma'' \in \delta''_j} \left(x_k^\tau \vee \mathcal{F}_\varphi(x_{j_1}^{\sigma_1}, \ldots, x_{j_N}^{\sigma_N})\right),$$

где $\tau, \sigma_1, \ldots, \sigma_N \in \{0,1\}$, $\mathfrak{B}_i$ и $\mathfrak{B}_j$ — КНФ характеристических функций компонент разбиения δ'_i и δ''_j соответственно, а k, τ, $j_1, \ldots, j_N$, $\sigma_1, \ldots, \sigma_N$ зависят от i, j, σ''. С помощью указанного разложения доказывается справедливость следующей леммы.

Лемма 3. *При достаточно больших n для любой функции $f \in P_2(n)$ и любого a, $a \geqslant 3$, существует формула $\mathcal{F}_f$, $A(\mathcal{F}_f) = a$, реализующая эту функцию, такая, что*

$$L(\mathcal{F}_f) \leqslant \frac{2^n}{\log n}\left(1 + \frac{\log^{[a-1]} n + O(1)}{\log n}\right).$$

Теорема 1 следует непосредственно из лемм 2 и 3.

Работа выполнена при поддержке РФФИ, проект 09-01-00817-а.

Список литературы

[1] *Ложкин С. А.* Лекции по основам кибернетики: Учебное пособие. — М.: Изд. отдел факультета ВМиК МГУ им. М. В. Ломоносова, 2004.

[2] *Лупанов О. Б.* О реализации функций алгебры логики формулами из конечных классов (формулами ограниченной глубины) в базисе $\&, \vee, \neg$ // Проблемы кибернетики. Вып. 6. — 1961. — С. 5–14.

Условия непрерывной реберной раскрашиваемости ассоциированных графов

А. М. Магомедов

magomedtagir1@yandex.ru

Дагестанский государственный университет, Махачкала

1. Рассматривается следующая задача теории расписаний. Исходные данные к составлению расписания обслуживания множества требований $N = \{1, \ldots, n\}$ в системе приборов $L = \{1, \ldots, l\}$ представлены в виде семейства предписаний $\Omega = \{\omega_1, \ldots, \omega_l\}$, где каждое

ω_i содержит точно два элемента из N (необязательно различных), $r_{i,j}$ — количество вхождений требования $j \in N$ в предписание $\omega_i \in \Omega$, $m = \max\limits_{j \in L} \sum\limits_{i=1}^{l} r_{i,j}$, $m > 2$. Каждый прибор $i \in L$ должен выполнить с каждым требованием $j \in N$ точно $r_{i,j}$ операций; длительность каждой операции равна единице, условия предшествования отсутствуют, в любой момент времени каждый прибор и каждое требование могут участвовать лишь в одной операции.

Расписание обслуживания длительности m будем называть *непрерывным*, если каждый прибор выполняет операции в последовательные моменты времени. Вопрос: существует ли непрерывное расписание?

2. Двудольный граф $G = (X, Y, E)$, $X = \{x_1, \ldots, x_n\}$, $Y = \{y_1, \ldots, y_l\}$, где вершины $x_j \in X$ и $y_i \in Y$ соединены $r_{i,j}$ ребрами, будем называть *ассоциированным* с семейством Ω. Очевидно, наибольшая степень вершин множества X равна m, степень каждой вершины множества Y равна 2. Сюръекцию $f\colon E \to \{1, \ldots, m\}$ будем называть *правильной* рёберной раскраской графа G, если $f(a) \neq f(b)$ для любых смежных рёбер a и b. Правильную рёберную раскраску графа G будем называть *непрерывной*, если для каждой вершины из Y инцидентные рёбра a и b удовлетворяют условию:

$$|f(a) - f(b)| = 1.$$

Задача о существовании непрерывного расписания и задача о существовании непрерывной рёберной раскраски ассоциированного графа эквивалентны.

3. В [1] показано, что в случае чётного m непрерывное расписание всегда существует. Для $m = 5$ задача рассмотрена в [2]. В формулировке следующей теоремы $\Gamma(S)$ — множество вершин, смежных с вершинами S.

Теорема 1. *При нечётном m для существования непрерывной раскраски ассоциированного графа $G = (X, Y, E)$ необходимо и достаточно, чтобы выполнялись условия:*

$$|S| \leqslant \lfloor m/2 \rfloor \cdot |\Gamma(S)| \ \ \forall\, S \subseteq Y. \tag{1}$$

Теорема 2. *Проверка условий (1) осуществима за полиномиальное время.*

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России» № 2010–1.3.2–111–017/12.

Список литературы

[1] *Магомедов А. М.* Матрица расписания с двумя ненулевыми элементами в строке // Вестник ДГУ. — 1999. — Вып. 4.

[2] *Магомедов А. М., Сапоженко А. А.* Условия существования непрерывных расписаний длительности пять // Вестник МГУ, сер. Вычислительная математика и кибернетика. — 2010. — Т. 34, № 1. — С. 39–44.

О стационарных классах функций трехзначной логики

А. А. Мазуров

anat-mazurov@mail.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Булевы и k-значные функции широко применяются в кибернетике и криптографии. Исследованию их свойств посвящено множество работ. В работах [1] и [2] исследовано преобразование между двумя представлениями булевых функций: в виде вектора значений и в виде полинома Жегалкина. В настоящей работе представлено обобщение некоторых полученных в [1] и [2] результатов на случай трехзначной логики.

Пусть k – натуральное число, $k \geqslant 2$. Множество всех натуральных чисел от 0 до $k-1$ обозначим E_k: $E_k = \{0, \dots, k-1\}$. *Функцией* k-значной логики от n переменных называется отображение $\varphi:\ E_k^n \to E_k$.

Множество всех функций k-значной логики от n переменных обозначается $P_k(n)$. Множество всех функций k-значной логики (от любого количества переменных) обозначается P_k.

Вектором значений функции f, зависящей от переменных $x_1, \dots, x_n$, называется последовательность значений функции на всех наборах от $(0, \dots, 0)$ до $(k-1, \dots, k-1)$ в лексикографическом порядке.

Полиномом в k-значной логике называется формула вида

$$f(x_1, \ldots, x_n) = \sum_{\tilde{\alpha} \in E_k^n} c_\alpha x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n} \pmod{k},$$

где

$$x^\alpha = \begin{cases} 1, & \alpha = 0, \\ \underbrace{x \cdot x \cdot \ldots \cdot x}_{\alpha}, & \alpha \neq 0, \end{cases} \qquad c_\alpha \in E_k, \qquad \tilde{\alpha} = (\alpha_1, \ldots, \alpha_n).$$

Здесь и далее сложение и умножение элементов из E_k (а именно, значений переменых и значений функций, коэффициентов полинома) ведется по модулю k. Числа c_α называются *коэффициентами полинома. Вектором коэффициентов* полинома функции называется последовательность

$$c_{(0,\ldots,0)}, \ldots, c_{(k-1,\ldots,k-1)},$$

где индексы расположены в лексикографическом порядке.

Для любого простого числа k и любой функции $f \in P_k$ существует и единственно представление этой функции в виде полинома с точностью до перестановки слагаемых [3].

Преобразование Мёбиуса — это отображение, переводящее вектор значений функции в вектор коэффициентов соответствующего ей полинома.

Преобразование Мёбиуса — это линейное преобразование в пространстве векторов размерности k^n [4].

Теорема 1 [4, 3]. *В P_k преобразование Мёбиуса обратимо, если k — простое, причем в P_2 $\mu^{-1}(f) \equiv \mu(f)$.*

Функцию $f \in P_k$ назовем *стационарной функцией относительно μ^p, $p \geqslant 1$, с константой m*, если $\mu^p(f) = m \cdot f$, где $m \in E_k \backslash \{0\}$.

Стационарным классом (*функций n переменных*) *с константой m относительно μ^p* назовем множество всех стационарных функций k-значной логики (зависящих от n переменных) с константой m относительно μ^p.

Обозначим $Q_m^p(n) = \{f \in P_3(n) \mid \mu^p(f) \equiv m \cdot f\}$.

В результате исследования получены следующие результаты.

Иерархия стационарных классов

Теорема 2. $\forall f \in P_3 \quad \mu^8(f) \equiv f$.

Из этого следует, что достаточно рассмотреть только преобразования μ в степенях от 1 до 8.

Теорема 3. *В* P_3

$$Q_1^8(n) = P_3(n),$$
$$Q_2^8(n) = \varnothing,$$
$$Q_1(n) = Q_1^3(n) = Q_1^5(n) = Q_1^7(n),$$
$$Q_2(n) = Q_2^3(n) = Q_2^5(n) = Q_2^7(n),$$
$$Q_1^2(n) = Q_1^6(n),$$
$$Q_2^2(n) = Q_2^6(n).$$

Теорема 4.

$$Q_1(n) \subset Q_1^2(n) \subset Q_1^4(n) \subset Q_1^8(n) = P_3(n),$$
$$Q_2(n) \subset Q_1^2(n),$$
$$Q_2^2(n) \subset Q_1^4(n),$$
$$Q_1^4(n) \subset Q_1^8(n) = P_3(n).$$

Теорема 5. *В* P_3

$$|Q_1^4(n)| = 3^{3^{n-1}} \times |Q_2^4(n-1)|,$$
$$|Q_2^4(n)| = 3^{3^{n-1}} \times |Q_1^4(n-1)|.$$

Следствие 1. *В трехзначной логике*

$$|Q_1^4(n)| = \begin{cases} 3^{\frac{1}{2}(3^n-1)}, & \text{если } n \text{ — нечетное,} \\ 3^{\frac{1}{2}(3^n+1)}, & \text{если } n \text{ — четное,} \end{cases}$$

$$|Q_2^4(n)| = \begin{cases} 3^{\frac{1}{2}(3^n-1)}, & \text{если } n \text{ — четное,} \\ 3^{\frac{1}{2}(3^n+1)}, & \text{если } n \text{ — нечетное.} \end{cases}$$

Теорема 6. *В трехзначной логике для любого натурального* n

$$|Q_1^2(n)| = |Q_2^4(n-1)| \times |Q_1^2(n-1)|.$$

Теорема 7. *В трехзначной логике для любого натурального* n

$$|Q_2^2(n)| = |Q_2^4(n-1)| \times |Q_2^2(n-1)|.$$

Следствие 2. *В трехзначной логике*

$$|Q_1^2(n)| = \begin{cases} 3^{\frac{1}{4}(3^n+1)}, & \textit{если } n \textit{ — нечетное}, \\ 3^{\frac{1}{4}(3^n+5)}, & \textit{если } n \textit{ — четное}, \end{cases}$$

$$|Q_2^2(n)| = \begin{cases} 3^{\frac{1}{4}(3^n-3)}, & \textit{если } n \textit{ — нечетное}, \\ 3^{\frac{1}{4}(3^n+1)}, & \textit{если } n \textit{ — четное}. \end{cases}$$

Обозначим $P_i(n) = \{f \in P_3(n) \mid \mu^2(f) + i\mu(f) = f\}$, где $i = 0, 1, 2$.

Теорема 8. *В трехзначной логике для любого натурального* n

$$|Q_1(n)| = |Q_1(n-1)| \times |P_1(n-1)|.$$

Теорема 9. *В трехзначной логике для любого натурального* n

$$|Q_2(n)| = |Q_2(n-1)| \times |P_2(n-1)|.$$

Список литературы

[1] *Josef Pieprzyk, Xian-Mo Zhang.* Computing möbius transforms of boolean functions and characterising coincident boolean functions // Boolean Functions: Cryptography and Applications, 2007 (BFCA'07).

[2] *Josef Pieprzyk, Huaxiong Wang, Xian-Mo Zhang.* Möbius-α commutative functions and partially coincident functions // Boolean Functions: Cryptography and Applications, 2008 (BFCA'08).

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

[4] *Логачев О. А., Сальников А. А., Ященко В. В.* Булевы функции в кодированиии и криптологии. — М.: МЦНМО, 2004.

Исследование метода главных компонент с дополнительными метрическими ограничениями

А. И. Майсурадзе

maysuradze@cs.msu.su

Московский государственный университет им. М. В. Ломоносова, Факультет вычислительной математики и кибернетики

Для работы с классическими признаковыми описаниями объектов предложены различные подходы, в числе которых выделяют методы понижения размерности описаний. Один из наиболее известных — это метод главных компонент. Классический МГК [1] может быть сформулирован как задача аппроксимации числовых признаковых описаний объектов, причем на признаки не накладываются никакие дополнительные ограничения.

Описание конечного набора объектов набором метрик называется метрическим. При этом возникают и требуют решения проблемы, аналогичные проблемам признаковых описаний (например, см. [2]). Предлагается, формализуется и исследуется метод сокращения размерности метрических описаний объектов, названный метрическим МГК. Формализация метода аналогична классическому методу главных компонент с дополнительными метрическими требованиями. Доказано, что наложение дополнительных требований не ухудшает качества аппроксимации и не нарушает полезные свойства МГК.

Основные обозначения, формализация задачи метода

Через q обозначим число объектов обрабатываемой выборки. Через $[1, q]$ обозначим множество натуральных чисел от 1 до q.

Определение. Функция $\hat{r} : [1, q]^2 \to \mathbb{R}$ называется *конфигурацией сходства*, если для любых x и y из $[1, q]$ она удовлетворяет условиям $\hat{r}(x, x) = 0$ и $\hat{r}(x, y) = \hat{r}(y, x)$.

Можно считать, что конфигурация сходства $\hat{r}$ — вектор размерности $t = \binom{q}{2} = q(q-1)/2$, $\hat{r} \in \mathbb{R}^t$. Признак $\mathbf{f}$ — вектор размерности q, $\mathbf{f} \in \mathbb{R}^q$.

В работе используются стандартные определения метрики и полуметрики. Если для конфигурации сходства $\hat{r}$ выполнены все аксиомы полуметрики, то говорим, что конфигурация удовлетворяет метрическим требованиям, и пишем $\hat{r} \in \mathrm{MET}$. Через $\hat{1}$ обозначим метрическую конфигурацию, соответствующую метрике пространства изолированных точек, т. е. расстояние между любыми объектами равно 1.

При работе с признаковыми описаниями один объект описывается вектором значений разных признаков. При работе с метрическими описаниями пара объектов описывается вектором различных расстояний [3]. Через n обозначим число конфигураций сходства в метрических описаниях. Метрические описания выборки представляются матрицей данных вида $Y = (\hat{\rho}_1 \ldots \hat{\rho}_n) = (\mathbf{y}_1 \ldots \mathbf{y}_t)'$, где конфигурация сходства $\hat{\rho}_j \in \mathbb{R}^t$ и метрическое описание пары объектов $\mathbf{y}_i \in \mathbb{R}^n$.

Задача (метрического МГК). Для заданной матрицы метрических описаний Y требуется найти такие $\mathbf{a}_0, \ldots, \mathbf{a}_k \in \mathbb{R}^n$, определяющие линейное многообразие

$$L_k = \{\mathbf{a}_0 + g_1\mathbf{a}_1 + \cdots + g_k\mathbf{a}_k \mid g_1, \ldots, g_k \in \mathbb{R}\},$$

и такие $\hat{r}_1, \ldots, \hat{r}_k \in \mathrm{MET}$, которые доставляют минимум функционалу ошибки аппроксимации

$$Q = \left\| (\hat{\rho}_1 \ldots \hat{\rho}_n) - (\hat{1}\hat{r}_1 \ldots \hat{r}_k) \times (\mathbf{a}_0\mathbf{a}_1 \ldots \mathbf{a}_k)' \right\|_2^2 .$$

Конфигурации $\hat{r}_1, \ldots, \hat{r}_k$ называются новыми конфигурациями сходства и составляют новое метрическое описание. Вектор $\mathbf{a}_0$ называется свободным членом модели, векторы $\mathbf{a}_1, \ldots, \mathbf{a}_k$ называются главными компонентами.

Необходимость свободного члена

Если свободный член в модели классического МГК отсутствует, то возникают сложности с определением подходящей размерности многообразия, а главные компоненты существенно реагируют на направление на начало координат. В общем случае при работе с признаковыми описаниями нет основания считать нулевое описание (начало координат) типичным представителем искомого многообразия и, следовательно, нецелесообразно пренебрегать свободным членом.

Однако при работе с метрическими описаниями возникает гипотеза, что свободный член не оказывает существенного влияния на результат факторизации. Действительно, нулевое метрическое описание можно считать типичным, поскольку оно соответствует парам совпадающих объектов. Эксперименты на модельных наборах и реальных данных подтвердили, что наличие свободного члена в модели МГК практически не влияет на результаты применения МГК к метрическим описаниям.

Основные результаты

Результаты работы классического МГК обладают рядом важных свойств.

1. Решение задачи оптимизации известно в аналитическом виде. Нет необходимости применять численную оптимизацию и бороться с проблемами локальных оптимумов.
2. Решение в определённом смысле может быть получено сразу для всех размерностей аппроксимации. Главные компоненты упорядочиваются, и оптимальным решением для любой размерности будет выбор первых компонент.
3. Главные компоненты ортонормированы. Это очень важно, поскольку позволяет построить формулы перехода от старых описаний к новым без знания всех компонент. Можно вычислять не все главные компоненты.
4. Можно в ходе оптимизации вычислять лишь главные компоненты, после чего новые признаковые описания будут получены быстро и однозначно.
5. Есть вычислительно эффективные методы расчёта лишь нескольких первых компонент.
6. Новые признаки ортогональны. Это интерпретируется как декорреляция признаков. Можно быть уверенными в линейной независимости новых признаков.

В задаче метрического МГК введены дополнительные требования. Следовательно, можно опасаться, что качество аппроксимации упадёт, а решение утратит перечисленные свойства. Последнее свойство (6) будет утрачено неминуемо. В [4] показано, что ортогональность несовместима с метрическими требованиями.

Теорема 1. *Метрический МГК даёт ту же самую аппроксимацию исходных метрических описаний, что и классический МГК.*

Следовательно, качество аппроксимации не ухудшается. Сами новые метрические описания будут отличаться, поскольку результат классического МГК ортогонален.

Теорема 2. *Решение, даваемое метрическим МГК, содержит те же самые главные компоненты* $\mathbf{a}_1, \ldots, \mathbf{a}_k$, *что и решение, даваемое классическим МГК. Следовательно, свойства* (3) *и* (5) *сохраняются.*

Свойство (1) сохраняется. Вид решения известен, есть процедура прямого вычисления оптимального решения. Свойство (2) практически сохраняется. Разница заключается в том, что если требуется пригодный для всех размерностей набор главных компонент $\mathbf{a}_0, \ldots, \mathbf{a}_n$, то придётся вычислить и все новые конфигурации сходства. Свойство (4) сохраняется частично. В ходе оптимизации можно искать лишь главные компоненты. Новые конфигурации сходства можно получить потом и однозначно, однако процедура их получения требует больших вычислительных затрат.

Выводы

Хотя в задачу МГК были добавлены метрические требования, качество аппроксимации не ухудшилось, а из всех важных полезных свойств решения классического МГК было утрачено лишь то, которое было неминуемо утратить. Вычислительные затраты возросли.

Как и в классическом МГК, решение метрического МГК неоднозначно. Это позволяет исследовать возможность введения в задачу метрического МГК других дополнительных ограничений.

Работа выполнена при поддержке РФФИ, проект 10-01-00131-а.

Список литературы

[1] *Jolliffe T.* Principal Component Analysis (2nd ed.). — New York: Springer-Verlag Inc, 2002.

[2] *Майсурадзе А. И.* О поиске оптимального коллективного слагаемого для набора метрических конфигураций // Искусственный интеллект. — 2006, № 2. — С. 146–150.

[3] *Майсурадзе А. И.* О оптимальных разложениях конечных метрических конфигураций в задачах распознавания образов // ЖВМ и МФ. — 2004. — Т. 44, № 9. — С. 1697–1707.

[4] *Майсурадзе А. И.* Гомогенные и ранговые базисы в пространствах метрических конфигураций // ЖВМ и МФ. — 2006. — Т. 46, № 2. — С. 344–361.

Об универсальных свойствах многогранника разрезов

А. Н. Максименко

maksimenko_a_n@mail.ru

Ярославский государственный университет им. П. Г. Демидова

Семейство многогранников разрезов является одним из наиболее хорошо исследованных в классе комбинаторных многогранников. В частности, описанию свойств и приложений этого семейства посвящена монография [1] объемом более 700 страниц.

Приведем формулировку классической задачи о максимальном разрезе. Пусть $G(V,E)$ — полный граф с вершинами $V = \{v_1, v_2, \ldots, v_n\}$ и ребрами $E = \{e_{ij} = (v_i, v_j) : \ 1 \leqslant i < j \leqslant n\}$. Рассмотрим произвольное подмножество вершин $W \subseteq V$ (W может быть пустым). *Разрезом* $\mathrm{cut}(W)$ называется множество всех таких ребер графа G, для каждого из которых ровно одна инцидентная ему вершина принадлежит множеству W, а другая, соответственно, — множеству $V \setminus W$. (Следовательно, $\mathrm{cut}(W) = \mathrm{cut}(V \setminus W)$.)

Задача о максимальном разрезе. Дан граф $G(V,E)$, ребрам e_{ij} которого приписаны веса $c_{ij} \in \mathbb{R}$. Требуется найти подмножество вершин $W \subseteq V$, такое, чтобы разрез $\mathrm{cut}(W)$ был максимален по суммарному весу входящих в него ребер.

Многогранник, ассоциированный с задачей о максимальном разрезе, определяется следующим образом. Каждому ребру e_{ij} графа G сопоставим координату x_{ij}, $1 \leqslant i < j \leqslant n$, вектора $x = (x_{ij}) \in \mathbb{R}^m$, $m = \frac{n(n-1)}{2}$. Каждому разрезу $\mathrm{cut}(W)$ графа G поставим в соответствие вектор $x(W)$ с координатами

$$x_{ij}(W) = \begin{cases} 1, & \text{если } e_{ij} \in \mathrm{cut}(W), \\ 0, & \text{в остальных случаях.} \end{cases}$$

Выпуклая оболочка всех таких векторов называется *многогранником разрезов*:

$$CUT_n = \mathrm{conv}\{x(W) : \ W \subseteq V\}.$$

По аналогии с многогранником CUT_n определяются и многие другие многогранники, ассоциированные с оптимизационными задачами на графах. Это прежде всего задача коммивояжера, задача о длиннейшем (кратчайшем) пути, задача о назначениях, задача о дереве Штейнера и многие другие. Исследованию свойств их многогранников уделяется зачастую не менее пристальное внимание. Другой тип комбинаторных многогранников связан с задачами, допускающими более естественную переформулировку в виде задачи целочисленного линейного программирования. Это, в первую очередь, задача о рюкзаке, задачи о покрытиях, разбиениях и упаковках множества, задача о вершинном покрытии в графе и некоторые другие.

В качестве примера приведем описание многогранника упаковок [2]. Пусть $\mathbf{A}$ — $0/1$-матрица размера $m \times d$, $\bar{1}$ — m-мерный вектор, все координаты которого равны 1. Выпуклая оболочка множества всех векторов $x \in \{0,1\}^d$, удовлетворяющих неравенству $\mathbf{A}x \leqslant \bar{1}$, называется *многогранником упаковок*.

С целью сравнения комбинаторных характеристик упомянутых выше многогранников введем для них отношение порядка.

Определение 1. В случаях когда многогранник p аффинно эквивалентен либо самому многограннику q, либо некоторой его грани, будем говорить, что p *не сложнее* q. Обозначение: $p \leqslant q$.

Заметим, что если $p \leqslant q$, то

1) число вершин p не превосходит числа вершин q,
2) граф многогранника p является подграфом графа многогранника q,
3) число фасет p не превосходит числа фасет q.

Если же требуется сравнивать семейства комбинаторных многогранников, то полезным оказывается понятие аффинной сводимости.

Определение 2. Семейство многогранников P *аффинно сводится* к семейству многогранников Q, если при некотором $r > 0$ для каждого многогранника $p \in P$ найдется $q \in Q$, такой, что $p \leqslant q$, причем $\dim q = O\left((\dim p)^r\right)$. Обозначение: $P \propto_A Q$.

Отметим, что это определение отличается от определения аффинной сводимости, введенного ранее в работе [3], в сторону усиления условий. Перечислим некоторые свойства нового понятия.

Пусть $P \propto_A Q$, тогда семейство Q наследует от P:

1) суперполиномиальное число вершин и фасет,
2) NP-полноту проверки несмежности вершин,
3) суперполиномиальное кликовое число графа.

Ранее (см. [4]) для нескольких семейств многогранников, обладающих свойством 2, были установлены следующие зависимости.

Теорема 1 [4]. *Многогранники двойных покрытий аффинно сводятся к многогранникам коммивояжера, многогранникам задачи о рюкзаке, многогранникам покрытий, многогранникам задачи* 3-*выполнимости и многогранникам кубических подграфов.*

Кроме того, нетрудно показать, что многогранники разрезов аффинно сводятся к многогранникам двойных покрытий. Причем так как задача распознавания несмежности вершин для CUT_n тривиальна (каждые две его вершины образуют собственное ребро многогранника [1, 3]), то ни одно из перечисленных в теореме 1 семейств не может быть аффинно сведено к многогранникам разрезов. Оказывается, те же рассуждения справедливы и в отношении многих других известных комбинаторных многогранников. Объединяя их с теоремой 1, получаем:

Теорема 2. *Многогранники разрезов аффинно сводятся к многогранникам покрытий, разбиений и упаковок множества, многогранникам коммивояжера, многогранникам задачи о рюкзаке, многогранникам задачи* 3-*выполнимости, многогранникам кубических подграфов, многогранникам вершинных покрытий, многогранникам* 3-*сочетаний, многогранникам раскрасок графа, многогранникам деревьев Штейнера.*

Причем, по указанным выше причинам, аффинное сведение в обратную сторону невозможно. Эти обстоятельства позволяют говорить о многогранниках разрезов как о наиболее простых (в смысле комбинаторных свойств) среди многогранников, ассоциированных с труднорешаемыми задачами.

Работа проводилась при финансовой поддержке Федерального агентства по науке и инновациям, ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг.

Список литературы

[1] *Деза М. М., Лоран М.* Геометрия разрезов и метрик. — М.: МЦНМО, 2001.

[2] *Емеличев В. А., Ковалев М. М., Кравцов М. К.* Многогранники, графы, оптимизация. — М.: Наука, 1981.

[3] *Бондаренко В. А., Максименко А. Н.* Геометрические конструкции и сложность в комбинаторной оптимизации. — М.: ЛКИ, 2008.

[4] *Максименко А. Н.* О комбинаторных свойствах многогранника двойных покрытий // Информационный бюллетень Ассоциации математического программирования № 12 (Тезисы докладов XIV Всероссийской конференции «Математическое программирование и приложения»). — Екатеринбург: УрО РАН, 2011. — С. 197–198.

Конечно определенные минимальные сложные классы графов для задачи о реберном списковом ранжировании

Д. С. Малышев

dsmalyshev@rambler.ru

Национальный исследовательский университет
«Высшая школа экономики» (Нижегородский филиал);
Нижегородский государственный университет им. Н. И. Лобачевского

Исследуются элементы границы между «простыми» и «сложными» классами графов для некоторой задачи в семействе *наследственных классов графов*, т. е. классов графов, замкнутых относительно удаления вершин.

Формализуем понятия «простого» и «сложного» класса графов. Пусть Π — какая-либо NP-полная задача на графах. Наследственный класс графов назовем Π-*простым*, если задача Π для графов из этого класса полиномиально разрешима, и Π-*сложным* в противном случае. Далее везде предполагаем справедливость неравенства $P \neq NP$ и не включаем его явно в формулировки полученных результатов.

Естественной идеей решения задачи демаркации является поиск *максимальных* Π-*простых* и *минимальных* Π-*сложных классов*, т. е. тупиковых классов графов соответствующей сложности из рассматриваемой решетки. К сожалению, использование понятия максимального простого класса графов оказывается безрезультатным. Так, В. Е. Алексеев в работе [1] установил, что ни один Π-простой класс не является максимальным простым (правда, в [1] это утверждается только про задачу о независимом множестве, но все рассуждения из данной работы легко переносятся на общий случай). Вместе с тем, до недавнего времени про минимальные сложные классы ничего не было известно.

Первый результат для подобного рода классов был получен автором в работе [2]. Там рассматривалась задача распознавания принадлежности наследственному классу графов $\mathbf{X}$ (задача РП[$\mathbf{X}$]) и было доказано следующее утверждение: *для любого наследственного класса* $\mathbf{X}$ *минимальных* РП[$\mathbf{X}$]*-сложных классов не существует*. С другой стороны, в работах [2, 3, 4] было показано, что определенные классы графов являются минимальными сложными для некоторых обобщений задач о раскраске — задач о списковом ранжировании (реберного и вершинного вариантов). В этой публикации исследуются минимальные сложные классы для задачи о реберном списковом ранжировании.

Задача о реберном списковом ранжировании (далее — задача РСР) заключается в следующем. Пусть заданы граф G с множеством ребер E и множество $\mathcal{L} = \{L(e) : \ e \in E\}$, где каждое $L(e)$ — конечное множество натуральных чисел (цветов, в которые разрешается покрасить ребро e). *$\mathcal{L}$-ранжированием ребер* графа G называется такая раскраска c его вершин, что:

1) $c(e) \in L(e)$ для каждого ребра e;
2) если $c(e_1) = c(e_2)$, $e_1 \neq e_2$, то каждый путь, соединяющий e_1 и e_2, содержит такое ребро e_3, что $c(e_3) > c(e_1)$.

Задача РСР состоит в том, чтобы по данным G и $\mathcal{L}$ определить, существует ли $\mathcal{L}$-ранжирование ребер графа G. Уточним, что под РСР-простым классом графов далее понимается такой наследственный класс, что задача РСР решается для графов из этого класса за полиномиальное время при любом множестве $\mathcal{L}$. В формулировке задачи о вершинном списковом ранжировании слово «ребро» заменено словом «вершина».

К настоящему времени полное описание множества минимальных РСР-сложных классов неизвестно. Вместе с тем, по-видимому, движение к получению результата такого рода предполагает изучение минимальных РСР-сложных классов с дополнительными ограничениями. В качестве такого ограничения в настоящей статье предлагается рассмотреть количество запрещенных порожденных подграфов. Напомним, что любой наследственный класс графов $\mathbf{X}$ может быть задан множеством своих запрещенных порожденных подграфов $\mathbf{S}$, при этом принята запись $\mathbf{X} = Free(\mathbf{S})$. Минимальное по включению множество $\mathbf{S}$ с таким свойством существует, единственно и обозначается через $Forb(\mathbf{X})$. Если $Forb(\mathbf{X})$ конечно, то $\mathbf{X}$ называется *конечно определенным*, а если $|Forb(\mathbf{X})| = k$, то $\mathbf{X}$ называется *k-определенным*.

Итак, целью данной работы является исследование строения множества минимальных PCP-сложных классов, определяемых наименьшим количеством запрещенных порожденных подграфов, затем определяемых следующим за этим наименьшим количеством таких подграфов и т. д. Первый результат такого рода был получен в работе [5].

Теорема 1. *Класс полных графов* **Clique** *является единственным 1-определенным минимальным PCP-сложным классом.*

В той же работе [5] было установлено, что существует единственный минимальный PCP-сложный подкласс класса всех полных двудольных графов **BComplete**. Обозначим этот подкласс через **BC**. Хотя остается неизвестным, совпадает ли **BC** с **BComplete**, можно с уверенностью утверждать, что **BC** либо 2-определенный, либо 3-определенный. Имеет место следующее утверждение [5].

Теорема 2. *Множество 2-определенных минимальных PCP-сложных классов либо пусто, либо состоит из одного класса* **CBiparite**. *Существует минимальный PCP-сложный класс, определяемый двумя или тремя запрещенными порожденными подграфами.*

Интерес к исследованию минимальных сложных классов именно для реберного варианта задачи о списковом ранжировании (а не для вершинного) обусловлен следующим фактором. При некотором специальном преобразовании (называемом продолжением), введенном в работе [6], сохраняется NP-полнота задачи PCP. Вполне возможно, что полученный при такой операции класс будет либо минимальным PCP-сложным, либо близким к PCP-минимальному. Класс $\mathbf{Y}$ называется *продолжением* класса $\mathbf{X}$, если выполняются следующие два условия:

— для любого графа $G \in \mathbf{X}$ существует такой граф $H \in \mathbf{Y}$, что G является остовным подграфом графа H;

— в любом графе из $\mathbf{Y}$ существует остовный подграф, принадлежащий классу $\mathbf{X}$.

В статье [6] доказано следующее утверждение.

Теорема 3. *Пусть* $\mathbf{Y}$ *— произвольный класс графов, являющийся продолжением класса* $\mathbf{X}$ *с NP-полной задачей PCP. Тогда задача PCP NP-полна в классе* $\mathbf{Y}$.

Очевидно, что из теоремы 3 незамедлительно следует *NP*-полнота задачи PCP в классе **Clique**. Вместе с тем, продолжение не сохраняет *NP*-полноту задачи о вершинном списковом ранжировании, поскольку для полных графов она полиномиально разрешима (методом поиска наибольшего паросочетания в двудольном графе). Теорема 3 также позволяет получать новые k-определенные минимальные PCP-сложные классы при больших значениях k. Например, в [6] получен (путем применения теоремы 3 к классу $\mathbf{X} = \mathbf{Star}$) минимальный PCP-сложный класс **Camomile**, определенный семью запрещенными порожденными подграфами. Класс **Star** — совокупность графов, являющихся порожденными подграфами для графов из $\bigcup\limits_{i=1}^{\infty}\{S_i\}$ (S_i — граф, получаемый подразбиением каждого ребра графа $K_{1,i}$). Класс **Camomile** составляют графы, являющиеся порожденными подграфами в графах из $\bigcup\limits_{i=1}^{\infty}\{Camomile(i)\}$, где $Camomile(i)$ получается из S_i добавлением всех ребер, инцидентных вершине степени i и всем листьям. Более того, в этой работе доказано, что, применяя теорему 3 к классу графов $\mathbf{X} = \mathbf{Star}$, можно получить только минимальные PCP-сложные классы **Clique**, **BC**, **Camomile**.

Работа поддержана РФФИ, проекты № 10-01-00357-а и № 11-01-00107-а, и ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2012 гг., № ГК 16.740.11.0310.

Список литературы

[1] *Alekseev V. E.* On easy and hard classes of graphs with respect to the independent set problem // Discrete Applied Mathematics. — 2004. — V. 132, № 3. — P. 17–26.

[2] *Малышев Д. С.* О минимальных сложных классах графов // Дискретный анализ и исследование операций. — 2009. — Т. 16, № 6. — С. 43–51.

[3] *Малышев Д. С.* О минимальных сложных элементах решетки наследственных классов графов // Материалы VII молодежной научной шко-

лы по дискретной математике и ее приложениям. Часть II. — М.: ИПМ РАН, 2009. — С. 12–16.

[4] *Малышев Д. С.* О тупиковых по вычислительной сложности наследственных классах графов // Материалы X Международного семинара «Дискретная математика и ее приложения». — М.: Изд-во механико-математического ф-та МГУ, 2010. — С. 314–316.

[5] *Малышев Д. С.* Последовательные минимумы решетки наследственных классов графов для задачи о реберном списковом ранжировании // Вестник Нижегородского университета им. Н .И. Лобачевского. — 2010. — № 4. — С. 70–76.

[6] *Малышев Д. С.* Минимальные сложные классы графов для задачи о реберном списковом ранжировании // Дискетный анализ и исследование операций. — 2011. — Т. 17, № 1. — С. 133–136.

О сложности периодических функций трехзначной логики в классе поляризованных полиномов

Н. К. Маркелов

nord_rk@bk.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

В настоящей заметке предложены семейства периодических функций трехзначной логики, которые являются сложными в классе поляризованных полиномов.

Пусть $k \geqslant 2$, $E_k = \{0, 1, \dots, k-1\}$. Функция $f(\tilde{x}^n)$ называется *функцией k-значной логики*, если на каждом наборе $\tilde{\alpha} \in E_k^n$ ее значение содержится в E_k. Множество всех функций k-значной логики от n переменных обозначается P_k^n.

Поляризованным по вектору поляризации $\sigma = (\sigma_1, \dots, \sigma_n) \in E_k^n$ *полиномом* назовем выражение вида

$$\sum_{\alpha=(a_1,\dots,a_n)\in E_k^n} c_f^{\sigma}(\tilde{\alpha}) \cdot (x_1+\sigma_1)^{a_1} \cdot \dots \cdot (x_n+\sigma_n)^{a_n},$$

в котором $c_f^\sigma(\alpha) \in E_k$ — некоторые коэффициенты, $(x_i+\sigma_i)^{a_i}$ — степени, то есть

$$(x_i+\sigma_i)^{a_i} = \underbrace{(x_i+\sigma_i)\cdot(x_i+\sigma_i)\cdot\dots\cdot(x_i+\sigma_i)}_{a_i \text{ раз}}, \quad (x_i+\sigma_i)^0 = 1,$$

все суммы и произведения берутся по $\bmod k$.

Если и только если число k – простое, для каждого вектора поляризации $\sigma = (\sigma_1, \dots, \sigma_n) \in E_k^n$ для каждой функции $f(\tilde{x}^n) \in P_k^n$ существует однозначный поляризованный по вектору σ полином, ее задающий [1, 2].

Введем характеристику сложности функций k-значной логики в классе поляризованных полиномов. *Сложностью* $l(P^\sigma)$ *полинома* P^σ, поляризованного по вектору σ, назовем число его слагаемых с ненулевыми коэффициентами. *Сложность функции* k-значной логики f в классе поляризованных полиномов определяется как минимальная по всем поляризациям сложность полинома, реализующего эту функцию: $L(f) = \min\limits_{\sigma\in E_k^n, P^\sigma = f} l(P^\sigma)$. *Функция Шеннона* сложности k-значных функций в классе поляризованных полиномов $L_k(n)$ определяется как сложность самой сложной функции k-значной логики от n переменных: $L_k(n) = \max\limits_{f\in P_k^n} L(f)$.

Известны следующие оценки функций Шеннона сложности k-значных функций в классах поляризованных полиномов:

$L_2(n) = \left[\frac{2^{n+1}}{3}\right]$ [3];
$L_k(n) \leqslant \frac{k(k-1)}{k(k-1)+1}k^n$ при простых $k \geqslant 3$ [2];
$L_k(n) \gtrsim \frac{k-1}{k}k^n$ при простых $k \geqslant 3$ [4].

В настоящей заметке предлагаются семейства сложных в классе поляризованных полиномов функций трехзначной логики.

Функция k−значной логики $f(\tilde{x}^n) \in P_k^n$ называется *периодической периода* $T < k^n$, если ее вектор значений $\tilde{\alpha}_f$ обладает следующим свойством:

$$\alpha_i = \alpha_{i+T} \ (i = 0, \dots, k^n - T).$$

С каждой периодической функцией от n переменных однозначно связывается вектор $(\alpha_0, \dots, \alpha_{T-1}) \in E_k^T$, который назовем ее *периодом*.

Теорема 1. *Пусть* $f_i(\tilde{x}^n) \in P_3^n$, $i = 1, \ldots, 4$, *– периодические функции с периодами* $p_i \in E_3^4$, *где*

$$p_1 = (1122),\ p_2 = (2211),\ p_3 = (1221),\ p_4 = (2112).$$

Тогда $L(f_i) \geqslant \frac{3}{4}3^n - \frac{1}{4}$.

Соответствующая нижняя оценка сложности функций f_i, $i = 1, \ldots, 4$, для значений $n \leqslant 10$ была найдена экспериментально Е. Н. Денисовым [5].

Работа выполнена при поддержке РФФИ, проект 09-01-00731-а.

Список литературы

[1] *Яблонский С. В.* Введение в дискретную математику. — Высшая школа, 2001. — С. 9–42.

[2] *Селезнева С. Н.* О сложности представления функций многозначных логик поляризованными полиномами // Дискретная математика. — 2002. — Т. 14, вып. 2. — С. 48–53.

[3] *Перязев Н. А.* Сложность булевых функций в классе полиномиальных поляризованных форм // Алгебра и логика. — 1995 — Т. 34, вып. 3. — С. 323–326.

[4] *Алексеев В. Б., Селезнева С. Н., Вороненко А. А.* О сложности реализации функций k-значной логики поляризованными полиномами // Труды V Международной конференции «Дискретные модели в теории управляющих систем» (Ратмино, 26–29 мая 2003 г.). — М., МГУ. — 2003. — С. 8-9.

[5] *Денисов Е. Н.* О сложности функций трехзначной логики в классе поляризованных полиномов. Дипломная работа, кафедра математической кибернетики, ВМиК МГУ, 2006.

Аффинные преобразования геометрических образов конечных автоматов

Д. О. Матов

dmitrymatov@gmail.com

Саратовский государственный университет им. Н. Г. Чернышевского

Задание поведения конечных детерминированных автоматов геометрическими образами дает новые инструменты для решения различных задач [1]. Значительный интерес представляет взаимосвязь геометрических преобразований образов и свойств автоматов в их классическом представлении. В данной статье представлены некоторые результаты исследования применения аффинных преобразований к геометрическим образам, порождаемым некоторыми классами автоматов. Эти исследования были начаты при рассмотрении поведения динамических систем, определяемых геометрическими образами автоматов [2]. Однако сейчас данные вопросы рассматриваются с точки зрения их последующего применения для построения различных композиций автоматов с использованием языка геометрических образов.

Пусть автомат $A = (S, X, Y, \delta, \lambda)$, где S — множество состояний, X, Y — входной и выходной алфавиты, $\delta : S \times X \to S$, $\lambda : S \times X \to Y$ — функции переходов и выходов соответственно. Пусть $|X| = n$, $|Y| = m$. С инициальным автоматом (A, s), $s \in S$, связано автоматное отображение $\Lambda_A^s : X^* \to Y^*$. Геометрическое пространство Γ для автомата (A, s) определяется по следующему алгоритму [1]:

1. Сопоставим элементам множества X натуральные числа от 1 до n, т. е. осуществим взаимно-однозначное отображение $f : X \to \{1, 2, \ldots, n\}$.
2. Определим координатную ось абсцисс $\tilde{X}$ для пространства Γ как отрезок числовой оси $[0, n+1]$.
3. Каждому слову $p = x_{i_1} \ldots x_{i_k}$ поставим в соответствие вектор $\omega = \big(f(x_{i_1}), \ldots, f(x_{i_k})\big)$, т. е. осуществим взаимно-однозначное соответствие $g : X^* \to V_N$, где V_N — пространство конечномерных векторов, элементами которых являются натуральные числа.
4. Каждому такому вектору $\omega = (\omega_1, \omega_2, \ldots, \omega_k)$ взаимно-однозначно сопоставим точку $\tilde{x} \in Q$ на оси абсцисс:

$$\tilde{x} = \frac{\omega_1}{(n+1)^0} + \frac{\omega_2}{(n+1)^1} + \frac{\omega_3}{(n+1)^2} + \ldots + \frac{\omega_k}{(n+1)^{k-1}}.$$

Аналогично определяются нумерация элементов множества Y, ось ординат $\tilde{Y}$ пространства Γ и отображение $h : Y^* \to V_N$. Каждой паре $(p, q) \in \Lambda_A^s$ в пространстве Γ сопоставляется точка с координатами $(\tilde{x}, \tilde{y})$, где

$$\tilde{x} = \sum_{i=1}^{|p|} \frac{c_i}{(n+1)^{i-1}}, \quad (c_1, c_2, \ldots, c_{|p|}) = g(p),$$

$$\tilde{y} = \sum_{i=1}^{|q|} \frac{b_i}{(m+1)^{i-1}}, \quad (b_1, b_2, \ldots, b_{|q|}) = h(q).$$

Множество таких пар $(\tilde{x}, \tilde{y})$ понимается под геометрическим образом Ω_A^s автомата (A, s).

Класс автоматов, у которых $|S| = N$, $|X| = L$ и $|Y| = M$, будем обозначать $K(N, L, M)$.

Два геометрических образа называются *аффинно-эквивалентными*, если множество точек одного образа может быть преобразовано в множество точек другого образа некоторым поточечным аффинным преобразованием. Аффинное преобразование задается формулами

$$\tilde{x}' = c_{11}\tilde{x} + c_{12}\tilde{y} + r_1, \quad \tilde{y}' = c_{21}\tilde{x} + c_{22}\tilde{y} + r_2, \quad \begin{vmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{vmatrix} \neq 0,$$

выражающими для каждой данной точки $M(\tilde{x}, \tilde{y})$ координаты преобразованной точки $M'(\tilde{x}', \tilde{y}')$ (в той же системе координат).

Зафиксируем некоторый класс автоматов $K(N, L, M)$ и рассмотрим множество Ω геометрических образов всевозможных инициальных автоматов из данного класса. Будем рассматривать те аффинные преобразования образов, путем применения которых можно некоторый образ $\Omega_i \in \Omega$ преобразовать в другой образ $\Omega_j \in \Omega$. В этой статье рассматривается следующее аффинное преобразование: параллельный перенос вдоль оси ординат одновременно с растяжением или сжатием относительно оси абсцисс. Данное преобразование имеет вид:

$$\tilde{x}' = \tilde{x}, \quad \tilde{y}' = a\tilde{y} + b. \tag{1}$$

Будем говорить, что образы Ω_i, Ω_j *совместимы* выбранным видом аффинного преобразования, если при применении преобразования (1) ко всем точкам из Ω_i получается Ω_j.

Рассмотрим бинарное отношение $\rho \subset \Omega^2$, образованное парами совместимых образов. Легко показать, что оно рефлексивно, симметрично и транзитивно, поэтому является отношением эквивалентности на множестве Ω и задает разбиение этого множества на классы эквивалентности. Множество всех различных геометрических образов всевозможных инициальных автоматов из класса $K(N, L, M)$ будем обозначать $\Omega(N, L, M)$. Множество пар коэффициентов преобразований образов из $\Omega(N, L, M)$ будем обозначать $F(N, L, M)$.

Для выбранного вида аффинного преобразования был сформулирован и доказан критерий аффинной эквивалентности пары геометрических образов:

Теорема 1. *Геометрические образы $\Omega_A, \Omega_B \in \Omega(N, L, M)$ совместимы преобразованием вида (1) тогда и только тогда, когда преобразованием этого вида совместимы их конечные части, составленные из точек, относящимся к входным и выходным словам длины не более N^2.*

Этот критерий дает способ эффективной непосредственной проверки аффинной эквивалентности.

Следующие две теоремы первоначально были сформулированы и доказаны для автономных автоматов [3], а затем их удалось обобщить и на случай произвольной мощности входного алфавита.

Теорема 2. *Если $(a, b) \in F(N, L, M)$, то*

$$a = \frac{p_a}{q_a}, \quad b = \frac{p_b}{q_b}, \quad p_a, q_a, p_b, q_b \in Z,$$

$$1 \leqslant p_a \leqslant M, \quad 1 \leqslant q_a \leqslant M, \quad 0 \leqslant |p_b| \leqslant M^2 - 1, \quad 1 \leqslant q_b \leqslant M.$$

Следствие. $|F(N, L, M)| \leqslant 2M^5$.

Эту оценку можно улучшить до следующей: $|F(N, L, M)| \leqslant M^4$.

Теорема 3.

$$F(N, L, M) = F(2, 1, M)\ \forall N \geqslant 2,\ \forall L \geqslant 1.$$

Список литературы

[1] *Тяпаев Л. Б.* Решение некоторых задач для конечных автоматов на основе анализа их поведения // Изв. Сарат. ун-та. Сер. Математика. Механика. Информатика. — 2006. — Т. 6, вып. 2. — С. 121–133.

[2] *Тяпаев Л. Б., Матов Д. О.* Базисы геометрических образов для динамических систем, определяемых некоторыми классами автоматов // Компьютерные науки и информационные технологии: Материалы Междунар. науч.конф. — Саратов: Изд-во Сарат. ун-та, 2009. — С. 201–204.

[3] *Матов Д. О.* Классы аффинной эквивалентности геометрических образов автономных автоматов // Компьютерные науки и информационные технологии: Материалы науч. конф. — Саратов: Изд-во Сарат. ун-та, 2010. — С. 103–108.

Оценка обобщающей способности для монотонных алгоритмов классификации

Г. А. Махина

gmakhina@yandex.ru

Таврический национальный университет им. В. И. Вернадского, Симферополь

Предлагается один метод доопределения частично заданной монотонной булевой функции. В терминах теории обучения машин данный метод решает задачу классификации на два непересекающихся класса, в которой объекты описываются n бинарными признаками. Требование монотонности накладывается на алгоритм классификации в тех случаях, когда имеется априорная информация о монотонном характере зависимости класса от признаков [1] либо когда строится монотонная корректирующая операция [2]. Верхняя оценка обобщающей способности для произвольных монотонных алгоритмов классификации была получена в [1]. В данной работе предлагается точная оценка для монотонных алгоритмов классификации, основанных на принципе ближайшего соседа.

Определения и обозначения

Множество всех наборов $(\alpha_1, \ldots, \alpha_n)$, таких, что $\alpha_i \in \{0, 1\}$, $i = 1, \ldots, n$, называется *n-мерным единичным кубом* и обозначается через B^n. Будем говорить, что вектор $\alpha = (\alpha_1, \ldots, \alpha_n)$ предшествует вектору $\beta = (\beta_1, \ldots, \beta_n)$ или $\alpha \preceq \beta$, если $\alpha_i \leqslant \beta_i$ для всех $i\colon 1 \leqslant i \leqslant n$. Булева функция $f\colon B^n \to B^1$ называется *монотонной*, если для любых двух векторов α и β, таких, что $\alpha \preceq \beta$, выполняется отношение $f(\alpha) \leqslant f(\beta)$.

Пусть задано множество $X^L = \{x_i \in B^n\}_{i=1}^L$, называемое *генеральной выборкой*, и бинарная функция $y(x)$, возвращающая истинный класс объекта x. Обозначим $y_i = y(x_i)$. Будем полагать, что $y(x)$ является монотонной булевой функцией. Введём *множество нулей* $X_0^L = \{x_i \in X^L\colon y_i = 0\}$ и *множество единиц* $X_1^L = \{x_i \in X^L\colon y_i = 1\}$.

Пусть $A = \bigl\{a\colon B^n \to B^1\bigr\}$ — множество монотонных булевых функций; будем называть их *алгоритмами классификации* или *классификаторами*. Зададим бинарную функцию $I\colon A \times X^L \to \{0, 1\}$, называемую *индикатором ошибки*: $I(a, x_i) = 1$, если $a(x_i) \neq y_i$. Для каждого алгоритма $a \in A$ определим частоту его ошибок ν на произвольной выборке $X \subseteq X^L$: $\nu(a, X) = \frac{1}{|X|} \sum\limits_{x \in X} I(a, X)$.

Пусть ℓ — фиксированное натуральное число $\ell < L$. Обозначим через $[X]^\ell$ множество всех ℓ-элементных подмножеств генеральной выборки X^L, $\bigl|[X]^\ell\bigr| = \binom{L}{\ell}$. Методом обучения называется отображение $\mu\colon [X]^\ell \to A$, которое произвольной обучающей выборке $X \in [X]^\ell$ ставит в соответствие некоторый алгоритм $a \in A$. Выборка $\bar{X} = X^L \setminus X$ называется контрольной.

Монотонный классификатор, основанный на принципе ближайшего соседа

Нижней тенью набора α будем называть множество $\{x \in B^n \colon x \preceq \alpha\}$. *Верхней тенью* набора α будем называть множество $\{x \in B^n \colon x \succeq \alpha\}$. Определим расстояние $R_{\mathrm{up}}(\alpha, u)$ от набора α до верхней тени набора u как минимальное число нулевых координат набора α, которые надо инвертировать, чтобы полученный в результате набор α' попал в верхнюю тень u, т. е. $R_{\mathrm{up}}(\alpha, u) = \|\alpha \vee u\| - \|\alpha\|$. Определим расстояние $R_{\mathrm{dn}}(\alpha, z)$ от набора α до нижней тени набора z как минимальное число единичных координат набора α, которые надо инвертировать, чтобы полученный в результате данного преобразования набор α' попал в нижнюю тень z, т. е. $R_{\mathrm{dn}}(\alpha, z) = \|\alpha\| - \|\alpha \wedge z\|$.

Рассмотрим алгоритм классификации $a(x; X)$, основанный на принципе ближайшего соседа

$$a(x; X) = y(\arg\min_{x' \in X} \rho(x, x')) \text{ для всех } x \in X, \tag{1}$$

где функция расстояния ρ определяется через расстояния до верхних теней объектов из множества X_1^L и нижних теней объектов из множества X_0^L:

$$\rho(x, x') = \begin{cases} \|x \vee x'\| - \|x\|, & \text{если } x' \in X_1^L; \\ \|x\| - \|x \wedge x'\|, & \text{если } x' \in X_0^L. \end{cases} \tag{2}$$

Метод обучения μ является в данном случае *ленивым* (lazy learning), т. е. сводится к запоминанию обучающей выборки X, которая затем может быть использована для классификации произвольных объектов x по формуле (1).

При условии монотонности исходной выборки результатом применения метода ближайшего соседа является монотонный алгоритм. Доказательство этого факта можно найти в [2].

Оценка обобщающей способности

Обобщающая способность метода обучения μ на конечной генеральной выборке X^L характеризуется функционалом полного скользящего контроля (complete cross-validation, CCV) [3], который определяется как средняя по всем разбиениям $X \bigcup \bar{X} = X^L$ частота ошибок на контрольной выборке:

$$\mathrm{CCV}(\mu, X^L) = \frac{1}{\binom{L}{\ell}} \sum_{X \in [X]^\ell} \nu(\mu(X), \bar{X}).$$

Лемма. *Для произвольных μ и X^L*

$$\mathrm{CCV}(\mu, X^L) = \frac{1}{k\binom{L}{\ell}} \sum_{i=1}^{L} N_i^\ell, \qquad N_i^\ell = \sum_{X \in [X]^\ell} [x_i \in \bar{X}]\, I(\mu(X), x_i), \quad (3)$$

где N_i^ℓ — число разбиений генеральной выборки X^L на обучающую X и контрольную $\bar{X}$, при которых объект x_i оказывается в контрольной выборке и алгоритм μX допускает на нем ошибку.

Доказательство леммы можно найти в [3].

Каждому объекту $x_i \in X^L$ сопоставим две функции: $t_i(m)$ есть число объектов $x' \in X^L$, таких, что $\rho(x_i, x') < m$ и $x' \neq x_i$; $s_i(m)$ есть число объектов $x' \in X^L$, таких, что $\rho(x_i, x') = m$ и $y(x') \neq y(x_i)$ для всех $m = 0, \ldots, n$.

Алгоритмы вида (1), примененные к монотонной выборке, отличаются классификацией только объектов, равноудаленных от классов единиц и нулей. Для получения верхней оценки $\mathrm{CCV}(\mu, X^L)$ возьмём пессимистичный метод обучения μ_p, который по любой выборке X строит алгоритм $\mu_p(X)$, неправильно классифицирующий все такие объекты.

Теорема. *Пусть X^L — монотонная выборка, $\mu_p(X)$ — пессимистичный алгоритм вида $(1), (2)$. Тогда*

$\mathrm{CCV}(\mu_p, X^L) =$

$$= \frac{1}{k}\frac{1}{\binom{L}{\ell}}\sum_{i=0}^{L-1}\sum_{m=0}^{n}\left(\binom{L-t_i(m)-1}{\ell}-\binom{L-t_i(m)-s_i(m)-1}{\ell}\right).$$

Доказательство. Предположим, что пессимистичный алгоритм μ_p допускает ошибку на объекте x_i и расстояние до ближайшего объекта равно m. Это значит, что все объекты с $\rho(x_i, x') < m$ вошли в контрольную выборку, а в обучающей выборке присутствует по крайней мере один объект с $\rho(x_i, x') = m$ из противоположного класса. Число разбиений генеральной выборки, удовлетворяющих требуемому условию, равно

$$N_{im}^l = \binom{L-t_i(m)-1}{\ell}-\binom{L-t_i(m)-s_i(m)-1}{\ell}.$$

Пробегая по всем возможным расстояниям, получаем $N_i^l = \sum\limits_{m=0}^{n} N_{im}^l$. Подставляя это значение N_i^l в (3), получаем требуемое. ∎

Сложность вычисления CCV не превышает $O(nL^2)$ операций.

Автор выражает признательность д.ф.-м.н. К. В. Воронцову за постановку проблемы и ценные замечания.

Список литературы

[1] *Воронцов К. В.* Комбинаторный подход к оценке качества обучаемых алгоритмов // Матем. вопр. кибернетики. — 2004. — Вып. 13. — С. 5–36.

[2] *Воронцов К. В.* Оптимизационные методы линейной и монотонной коррекции в алгебраическом подходе к проблеме распознавания // ЖВМ и МФ. — 2000. — Т. 40, № 1. — С. 166–176.

[3] *Mullin M., Sukthankar R.* Complete cross-validation for nearest neighbour classifiers // Proceedings of International Conference on Machine Learning. — 2000. — P. 639–646.

Итерации конечных языков и недетерминированные конечные автоматы

Б. Ф. Мельников

B.Melnikov@tltsu.ru

Тольяттинский государственный университет

Введение. Равенство бесконечных итераций языков

В работе [1] рассматривались бесконечные итерации конечных языков. А именно, мы рассматривали ω-языки вида A^ω, где A – конечный язык над заданным алфавитом Σ. (Мы предполагаем, что $|\Sigma| \geqslant 2$ и $A \not\ni \varepsilon$.) Для двух конечных языков A и B мы рассматривали их равенство $A^\omega = B^\omega$, которое равносильно выполнению специального отношения эквивалентности, также определённого в [1]: $A^* \equiv_\infty B^*$ тогда и только тогда, когда

$$\begin{cases} (\forall\, u \in A^*)\ (\exists\, v \in B^*)\ (u \in \mathrm{Pref}\,(v)) \\ (\forall\, v \in B^*)\ (\exists\, u \in A^*)\ (v \in \mathrm{Pref}\,(u)). \end{cases} \tag{1}$$

(Здесь $\mathrm{Pref}\,(u)$ – множество всех префиксов слова u, включая ε и u. Аналогично для языков: $\mathrm{Pref}\,(A) = \{\, u \mid u \in \mathrm{Pref}\,(v),\ v \in A \,\}$.) Ниже будем писать $A \equiv B$ вместо обозначения $A^* \equiv_\infty B^*$, использовавшегося в [1].

Достаточное условие равенства

Достаточное условие равенства $A \equiv B$ формулируется в виде следующих 3 шагов.

1. Рассматриваем *новый* конечный алфавит Δ и некоторый максимальный префиксный код [2] над этим алфавитом. При этом условие $A \equiv B$ выполнено, поскольку $A^\omega = B^\omega = \Delta^\omega$.
   В качестве примера рассмотрим алфавит $\Delta = \{\, 0, 1 \,\}$ и два таких языка: $A = \{\, 0, 10, 11 \,\}$ и $B = \{\, 00, 01, 1 \,\}$.
2. Добавляем некоторые слова к языкам A и B. (Для A при этом достаточно рассматривать только такие слова u, для которых $u \notin \mathrm{Pref}\,(A)$; аналогично для B.) При этом равенство $A \equiv B$ тоже выполняется. Заметим, что полученные языки, вообще говоря, не являются префиксными.

В качестве продолжения предыдущего примера рассмотрим языки $A = \{ 0, 10, 11, 1101 \}$ и $B = \{ 00, 01, 1, 111, 1110 \}$.

3. Для полученных языков рассмотрим морфизм $h : \Delta^* \to \Sigma^*$; заметим, что язык $h(\Delta) = \{ h(c) \mid c \in \Delta \}$ тоже, вообще говоря, не является префиксным. Для полученных языков (их тоже будем обозначать A и B) условие $A \equiv B$ также выполняется.

   Продолжим рассматривать предыдущий пример. Для (непрефиксного) морфизма $h(0) = ab$, $h(1) = aba$ получаем следующие языки:

$$A = \{ ab, \quad aba\,ab, \quad aba\,aba, \quad aba\,aba\,ab\,aba \} \quad \text{и}$$
$$B = \{ ab\,ab, \quad ab\,aba, \quad aba, \quad aba\,aba\,aba, \quad aba\,aba\,aba\,ab \}$$

Итак, очевидное достаточное условие таково: если языки A и B получены описанным здесь способом, то $A \equiv B$.

Формулировка необходимого условия равенства

Теорема 1. *Сформулированное в предыдущем разделе достаточное условие является необходимым и достаточным.*

То есть *каждая* пара конечных языков A и B, для которых выполнено условие $A \equiv B$, может быть сконструирована с помощью алгоритма, описанного в предыдущем разделе.

Приведённая нами формулировка близка к доказанной в [1] теореме 1. Иными словами, это утверждение для рассматриваемых языков A и B (таких, что $A \equiv B$) может быть переформулировано следующим образом. Существует конечный язык D (пусть $D = \{ u_1, \ldots, u_n \}$), для которого выполняется следующее условие. Для некоторого нового алфавита $\Delta = \{ c_1, \ldots, c_n \}$ существуют (конечные) языки $A', B' \subseteq \Delta^*$, такие, что:

— A' и B' содержат (вообще говоря, различные) максимальные префиксные коды над Δ;

— для морфизма

$$h : \Delta^* \to \Sigma^*, \quad \text{где} \quad h(c_1) = u_1\,, \ \ldots, \ h(c_n) = u_n\,,$$

   выполнены условия $h(A') = A$ и $h(B') = B$.

Этот язык D (для некоторого рассматриваемого нами языка A) будем называть «минимальным».

Итерации языков и конечные автоматы

В этом разделе рассмотрим несложный алгоритм свед́ения рассмотренной выше проблемы равенства ω-языков вида A^ω к конечным автоматам. (Автоматы и связанные с ними объекты определяются согласно [3].) Отметим, что альтернативный алгоритм для проверки равенства (1) рассмотрен в [4].

Для языка $A = \{u_1, \dots, u_n\}$, где $u_i = a_{i,1} \dots a_{i,l_i}$ для $i = 1, \dots, n$, рассмотрим недетерминированный конечный автомат $\mathcal{K}(A) = (Q, \Sigma, \delta, \{s\}, Q)$, где

$$Q = \{s\} \cup \bigcup_{i \leqslant n,\ j < l_i} q_{i,j}\ ,$$

со следующей функцией переходов δ (мы рассматриваем приведённые далее определения для каждого $i = 1, \dots, n$). Если $|u_i| = 1$, то $\delta(s, a_{i,1}) \ni s$; иначе

$$\delta(s, a_{i,1}) \ni q_{i,1}\ , \qquad \delta(q_{i,l_i-1}, a_{i,l_i}) = \{s\}$$
$$\text{и} \quad \delta(q_{i,j-1}, a_{i,j}) = \{q_{i,j}\} \quad \text{для каждого возможного} \quad j \leqslant l_i - 1\ .$$

Очевидно, что такой автомат определяет язык $\mathrm{Pref}\,(A^*)$; поэтому условие (1) можно проверить, ответив на вопрос об эквивалентности автоматов $\mathcal{K}(A)$ и $\mathcal{K}(B)$.

Для ответа на этот вопрс мы можем использовать канонические автоматы, эквивалентные $\mathcal{K}(A)$ и $\mathcal{K}(B)$ и определённые согласно [3]. Однако основной аргумент для рассмотрения канонических автоматов такой: *каждое* состояние соответствующего канонического автомата соответствует некоторому подмножеству множества состояний рассматриваемого автомата, например автомата $\mathcal{K}(A)$.

Такое соответствие также может быть построено на основе [3]. Фактически им является рассматриваемая в той статье функция $(\varphi^{in})^{-1}$. Это соответствие означает следующее. Для некоторого состояния $\widetilde{q}$ канонического автомата и некоторого слова v входного языка этого состояния (т. е. $\mathcal{L}^{in}(\widetilde{q})$) существуют некоторые состояния $q_1, \dots, q_m$ автомата $\mathcal{K}(A)$, такие, что

$$v \in \mathcal{L}^{in}(q_1),\ \dots,\ v \in \mathcal{L}^{in}(q_m)\,.$$

Поэтому если для «минимального» языка D и некоторого слова $v \notin D^*$ выполнено условие $D \equiv D \cup \{v\}$, то

$$q_i \neq s \quad \text{для} \quad i = 1, \ldots, m; \qquad \text{а также} \qquad \bigcup_{i \leqslant m} \mathcal{L}^{out}(q_i) \supseteq \mathrm{Pref}\,(D^*).$$

План возможного доказательства равенства P=NP

Вследствие изложенного выше, для доказательства равенства P=NP достаточно доказать два следующих факта.

1. Проблема определения эквивалентности определённых выше автоматов $\mathcal{K}(A)$ и $\mathcal{K}(B)$ (для заданных конечных языков A и B) принадлежит классу NP. (По-видимому, доказательство этого факта может быть получено аналогично проблемам AL1 и AL9 монографии [5].)
2. Существует полиномиальный алгоритм «построения инверсного морфизма». То есть для заданного конечного языка $A \in \Sigma^*$ можно за полиномиальное время получить «минимальный» (в сформулированном выше смысле) язык D, такой, что $A \equiv D$.

Список литературы

[1] *Melnikov B.* The equality condition for infinite catenations of two sets of finite words // Int. J. of Found. of Comp. Sci. — 1993. — V. 4, № 3. — P. 267–274.

[2] *Лаллеман Ж.* Полугруппы и комбинаторные приложения. — М.: Мир, 1985.

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

[4] *Мельников Б.* Алгоритм проверки равенства бесконечных итераций конечных языков // Вестник Московского унив. Сер. 15. — 1996. — № 4. — С. 25–28.

[5] *Гэри М., Джонсон Д.* Вычислительные машины и труднорешаемые задачи. — М.: Мир, 1982.

О вычислении сложности по Арнольду двоичных слов

Ю. В. Мерекин

merekin@math.nsc.ru

Институт математики им. С. Л. Соболева СО РАН, Новосибирск

Для двоичных слов длины 2^n, $n \geqslant 1$, покажем возможность ускоренного вычисления сложности по Арнольду.

Пусть $w = x_1 x_2 \ldots x_{2^n}$, $w \neq v^k$, $k \geqslant 2$, — произвольное непериодическое двоичное слово длины $|w| = 2^n$, $n \geqslant 1$. Через (w) обозначим бесконечное периодическое слово $(w) = ww\ldots$. Далее под термином «слово (w)» мы будем понимать бесконечное периодическое слово (w).

Рассмотрим схему (цепочку слов)

$$(w) = (w_1), (w_2), \ \ldots \ , (w_s) = (v), \tag{1}$$

в которой первое слово произвольно, а каждое слово $(w_i) = y_1 y_2 \ldots$ порождает очередное слово $(w_{i+1}) = z_1 z_2 \ldots$, $1 \leqslant i \leqslant s-1$, с помощью оператора

$$F(\cdot, h_i) : (w_i) \mapsto (w_{i+1}) : \ z_j = y_j \oplus y_{j+h_i}, \tag{2}$$

где $j \geqslant 1$, $1 \leqslant h_i = 2^{n_i}$, $0 \leqslant n_i \leqslant n$, $\oplus$ является сложением по модулю два, то есть $F(w_i, h_i) = (w_{i+1})$. Число h_i назовём *рангом оператора* (2), а число s — *длиной схемы* (1).

Тип схемы, в которой (w) — первое, (v) — последнее слово, h — максимальный ранг используемых операторов, s — длина схемы, обозначим через $S(w, v, h, s)$. Для всякого (w) существует такое минимальное s, что в схеме $S(w, 0, 1, s)$ все слова попарно различны и $F(v, 1) = (0)$. Такую схему назовём *сложностной схемой*. Любое слово (w) имеет единственную сложностную схему. Число $s-1$ назовём *сложностью* двоичного слова (w) и обозначим через $A(w)$. Это понятие введено В. И. Арнольдом в более общем виде [1]. Оно совпадает с нашим определением сложности для слов длины 2^n. Сложность периодического слова (w) равна сложности конечного слова w.

В произвольной схеме $S(w, v, 1, s)$ выделим цепочку слов

$$(w) = (w_1), (w_{1+l_1}), \ \ldots \ , (w_{1+l_1+\ldots+l_t}) = (v), \tag{3}$$

где $l_i \geqslant 1$, $1 \leqslant i \leqslant t \leqslant s-1$. Если в (3) каждое слово $(w_{1+l_1+\ldots+l_i})$, $1 \leqslant i \leqslant s-1$, совпадает с $F(w_{1+l_1+\ldots+l_{i-1}}, h_i)$, $h_i = l_i$, то цепочка слов (3) является схемой $S(w, v, h, s_t)$, $s_t = 1+t$, которую назовём *эквивалентной* схеме $S(w, v, 1, s)$. В работе [2] доказана

Теорема 1. *Любая схема* $S(w, v, 1, 2^r+1)$, $r \geqslant 0$, *эквивалентна схеме* $S(w, v, 2^r, 2)$.

В слове $w = x_1x_2\ldots x_{2^n}$, $n \geqslant 1$, выделим 2^{n-m}, $0 \leqslant m \leqslant n-1$, позиций, таких, что расстояние между двумя соседними выделенными позициями одинаково и равняется 2^m. Используя выделенные позиции, образуем слово $u = x_ix_{i+2^m}\ldots x_{i+2^n-2^m}$ длины 2^{n-m}. Бесконечное слово $(u) = uu\ldots$ обозначим через $(w)^{x_i}_{2^{n-m}}$ и назовём *прореженным словом*. Число 2^{n-m} назовём *шагом* прореженного слова $(w)^{x_i}_{2^{n-m}}$. Заметим, что каждое прореженное слово является линейно упорядоченным множеством номеров позиций слова w.

Для любого слова (w) при фиксированном значении m существуют 2^m различных прореженных слов

$$(w)^{x_1}_{2^{n-m}}, (w)^{x_2}_{2^{n-m}}, \ \ldots\ , (w)^{x_{2^m}}_{2^{n-m}}, \quad n \geqslant 1, \quad 0 \leqslant m \leqslant n-1. \qquad (4)$$

Определим операцию объединения прореженных слов, которую будем обозначать через «$*$». Из определения прореженного слова следует, что каждая из позиций $x_1, x_2, \ldots, x_{2^n}$ встречается в прореженных словах (4) один и только один раз, поскольку является объединением арифметических прогрессий с разностями степени двойки. Справедливо представление слова (w) в виде

$$(w) = (w)^{x_1}_{2^{n-m}} * (w)^{x_2}_{2^{n-m}} * \ldots * (w)^{x_{2^m}}_{2^{n-m}}, \quad n \geqslant 1, \quad 0 \leqslant m \leqslant n-1. \quad (5)$$

Для произвольного слова (w) определим его принадлежность к одному из типов слов: чётному или нечётному. Слово $(w) = ww\ldots$, где $w = x_1x_2\ldots x_{2^n}$, $x_i \in \{0,1\}$, $1 \leqslant i \leqslant 2^n$, $n \geqslant 0$, назовём *чётным*, если $x_1 \oplus x_2 \oplus \ldots \oplus x_{2^n} = 0$, и *нечётным*, если $x_1 \oplus x_2 \oplus \ldots \oplus x_{2^n} = 1$. Существует простой алгоритм вычисления чётности всех прореженных слов $(w)^{x_i}_{2^{n-m}}$, $|w| = 2^n$, $n \geqslant 1$, $0 \leqslant m \leqslant n-1$, слова $(x_1x_2\ldots x_{2^n})$.

Например, для $n = 3$ имеем:

$$(w)^{x_1}_{2^3} = (x_1x_2x_3x_4x_5x_6x_7x_8), \qquad m = 0;$$

$$(w)^{x_1}_{2^{3-1}} = (x_1x_3x_5x_7), \quad (w)^{x_2}_{2^{3-1}} = (x_2x_4x_6x_8), \qquad m = 1;$$

$$(w)^{x_1}_{2^{3-2}} = (x_1x_5), \quad (w)^{x_2}_{2^{3-2}} = (x_2x_6),$$
$$(w)^{x_3}_{2^{3-2}} = (x_3x_7), \quad (w)^{x_4}_{2^{3-2}} = (x_4x_8), \qquad m = 2.$$

Теорема 2. *Для любого двоичного слова (w), длина периода которого равна 2^n, $n \geqslant 1$, все прореженные слова $(w)^{x_i}_{2^{n-m}}$, $0 \leqslant m \leqslant n-1$, $1 \leqslant i \leqslant 2^m$, нечётны тогда и только тогда, когда сложность слова (w) равна $A(w) = 2^n - 2^m + 1$.*

Рассмотрим пример вычисления сложности $A(w)$, в котором используются теорема 1 и теорема 2. Слова (w), $|w| = 2^n$, $n \geqslant 1$, расположены на вершинах ориентированного двоичного дерева (см. рис. 1).

На каждом ярусе дерева расположены все слова, сложность которых равна номеру яруса. В этом дереве выделим некоторый ориентированный путь, содержащий $A(w)$ рёбер и заканчивающийся в вершине нулевого яруса.

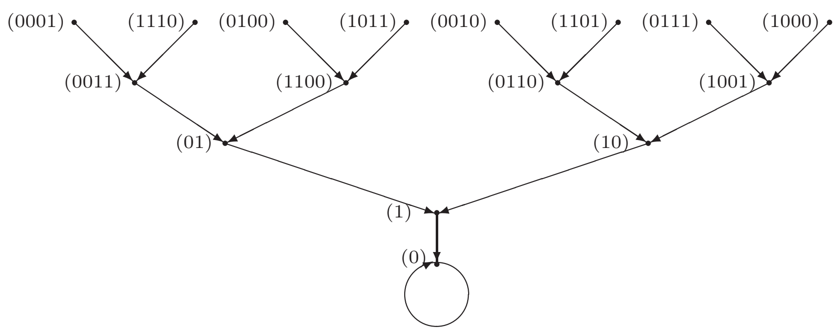

Рис. 1

Например, путь из вершины 16-го яруса: $(w_{16}) \mapsto (w_{15}) \mapsto \ldots \mapsto (w_1) \mapsto (w_0)$.

Для слов $(w_{16}), (w_{15}), (w_{13}), (w_9), (w_8), (w_7), (w_5), (w_4), (w_3), (w_2)$ значение сложности $A(w_i)$ согласно теореме 2 вычисляется подсчётом чётности прореженных слов.

Для слов $(w_{14}), (w_{12}), (w_{11}), (w_{10}), (w_6)$ используем теорему 1, согласно которой применение оператора вида (2) ранга 2^r, $r \geqslant 0$, эквивалентно движению по выделенному пути на 2^r ярусов.

В таблице 1 приведены результаты вычисления сложности $A(w)$ для всех слов (w), $|w| = 2^n$, $1 \leqslant n \leqslant 4$.

Таблица 1

(w_i)	$A(w_i)$	$F(w_i, 2^r) : (w_i) \mapsto (w_j)$	$A(w_j)$	$A(w_i) = 2^r + A(w_j)$
(w_{16})	$2^4 - 2^0 + 1$			
(w_{15})	$2^4 - 2^1 + 1$			
(w_{14})		$F(w_{14}, 2^0) : (w_{14}) \mapsto (w_{13})$	$A(w_{13}) = 2^4 - 2^2 + 1$	$A(w_{14}) = 2^0 + (2^4 - 2^2 + 1)$
(w_{13})	$2^4 - 2^2 + 1$			
(w_{12})		$F(w_{12}, 2^2) : (w_{12}) \mapsto (w_8)$	$A(w_8) = 2^3 - 2^0 + 1$	$A(w_{12}) = 2^2 + (2^4 - 2^2 + 1)$
(w_{11})		$F(w_{11}, 2^1) : (w_{11}) \mapsto (w_9)$	$A(w_9) = 2^4 - 2^3 + 1$	$A(w_{11}) = 2^1 + (2^4 - 2^3 + 1)$
(w_{10})		$F(w_{10}, 2^0) : (w_{10}) \mapsto (w_9)$	$A(w_9) = 2^4 - 2^3 + 1$	$A(w_{10}) = 2^0 + (2^4 - 2^3 + 1)$
(w_9)	$2^4 - 2^3 + 1$			
(w_8)	$2^3 - 2^0 + 1$			
(w_7)	$2^3 - 2^1 + 1$			
(w_6)		$F(w_6, 2^0) : (w_6) \mapsto (w_5)$	$A(w_5) = 2^3 - 2^2 + 1$	$A(w_6) = 2^0 + (2^3 - 2^2 + 1)$
(w_5)	$2^3 - 2^2 + 1$			
(w_4)	$2^2 - 2^0 + 1$			
(w_3)	$2^2 - 2^1 + 1$			
(w_2)	$2^1 - 2^0 + 1$			
(w_1)	2^0			

Следовательно, для вычисления сложности по Арнольду любого из 2^{2^4} слов, сложность которых не превосходит величины 2^4, достаточно применить оператор (2) не более одного раза.

Работа выполнена при поддержке РФФИ, проект № 11–01–00997.

Список литературы

[1] *Арнольд В. И.* Топология и статистика формул арифметики // Успехи мат. наук. — 2003. — Т. 58, № 4. — С. 1–26.

[2] *Merekin Yu. V.* On the Computational Complexity of the Arnold Complexity of Binary Words // Asian-European Journal of Mathematics. — 2009. — V. 2, № 4. — P. 641–648.

О замкнутых классах функций трехзначной логики, порожденных периодическими симметрическими функциями

А. В. Михайлович

avmikhailovich@gmail.com

Московский государственный университет леса

Рассматривается задача базируемости и конечной порожденности для классов функций трехзначной логики, порожденных симметрическими функциями. Известно [1], что все замкнутые классы булевых функций имеют конечный базис. В [2] показано, что при всех $k \geqslant 3$ в P_k существуют как замкнутые классы со счетным базисом, так и классы, не имеющие базиса. В [3, 4] рассмотрены некоторые семейства замкнутых классов, порожденных симметрическими функциями; для них приведены критерии базируемости и конечной порожденности. В данной работе рассматриваются классы, порожденные периодическими симметрическими функциями. Показано, что такой класс имеет базис тогда и только тогда, когда порождающая система содержит лишь конечное число функций. Все необходимые определения можно найти в [3–5].

Обозначим через R множество всех функций из P_3, принимающих значения только из множества $\{0, 1\}$ и равных нулю на наборах, содержащих хотя бы одну нулевую компоненту. Нетрудно показать, что для функций из множества R выполняется следующее утверждение.

Утверждение 1. *Пусть Φ — некоторая формула над* R, Φ_1 — *произвольная нетривиальная подформула формулы* Φ, $\widetilde{\alpha}, \widetilde{\beta}$ — *наборы из* E^n, *такие, что* $\Phi_1(\widetilde{\alpha}) = 0$, $\Phi(\widetilde{\beta}) = 1$. *Тогда* $\Phi(\widetilde{\alpha}) = 0$ *и* $\Phi_1(\widetilde{\beta}) = 1$.

Пусть $\widetilde{\alpha} \subseteq \{1, 2\}^n$. Число единиц в наборе $\widetilde{\alpha}$ обозначим через $|\widetilde{\alpha}|$. Пусть $f \subseteq$ R. Будем обозначать через N_f множество всех наборов из E_3^n, на которых функция f принимает значение 1. Число единиц и число двоек в слое с наибольшим числом единиц из N_f обозначим через e_f и d_f соответственно. Пусть Φ — формула над R. Множество всех функций, функциональные символы которых используются при построении формулы Φ, обозначим через $\Theta(\Phi)$. Множество всех наборов из E_3^n, которые получаются друг из друга перестановкой компонент, называется слоем. Слой из $\{1, 2\}^n$, содержащий e единиц и d двоек, $e + d = n$, обозначим через $\mathcal{L}(e, d)$. Функция $f(x_1, \ldots, x_n)$ из R называется периодической симметрической функцией с периодом t, если для некоторых e_f, d_f, таких, что $e_f + d_f = n$, $0 \leqslant d_f < t$, выполняется соотношение $N_f = \bigcup \mathcal{L}(e_f - it, d_f + it)$, где объединение берется по всем i, $0 \leqslant i \leqslant (n - d_f)/t$. Множество всех периодических симметрических функций будем обозначать PS. Множество всех периодических симметрических функций с периодом t будем обозначать PS^t. Обозначим через $i_n(x_1, \ldots, x_n)$, $n \geqslant 1$, функцию из PS, принимающую значение 1 на всех наборах из $\{1, 2\}^n$. Легко видеть, что $\mathrm{PS}^1 = \cup\{i_n\}$, где объединение берется по всем $n > 0$. Отметим следующее свойство множества PS^1.

Утверждение 2. *Для любого* $n \geqslant 2$ *выполняется равенство*

$$\mathrm{PS}^1 = [\{i_n\}].$$

Для доказательства основного результата нам потребуются утверждения 3–5.

Утверждение 3. *Пусть* $f(x_1, \ldots, x_n) \in \mathrm{PS}^t$, $t > 1$, *существует набор* $\widetilde{\alpha} \in N_f$, *такой, что* $0 < |\widetilde{\alpha}| < n$. *Пусть* Φ — *формула над* R, *реализующая функцию* f, Φ_1 — *подформула формулы* Φ, *имеющая вид* $g(\mathcal{B}_1, \ldots, \mathcal{B}_m)$, *где* $g \in \mathrm{PS}^r$, $r \in \mathbb{N}$, $\mathcal{B}_1, \ldots, \mathcal{B}_m$ — *формулы над* R. *Пусть среди* $\mathcal{B}_1, \ldots, \mathcal{B}_m$ *символы переменных* $x_1, \ldots, x_n$ *встречаются* $q_1, \ldots, q_n$ *раз соответственно,* $q_1, \ldots, q_n \in \mathbb{Z}^+$. *Тогда для любых* i, j, $1 \leqslant i, j \leqslant n$, *существует* $l \in \mathbb{Z}^+$, *такое, что* $|q_i - q_j| = lr$.

Для доказательства утверждения 3 достаточно рассмотреть значения формул Φ_1, $\mathcal{B}_1, \ldots, \mathcal{B}_m$ на паре наборов из одного слоя из N_f, различающихся в i-й и j-й компонентах.

Утверждение 4. *Пусть* $f(x_1, \ldots, x_n) \in \mathrm{PS}^t$, $t > 1$, Φ *— формула над* R, *реализующая функцию* f, Φ_1 *— подформула формулы* Φ, *имеющая вид* $g(\mathcal{B}_1, \ldots, \mathcal{B}_m)$, *где* $g \in \mathrm{PS}^r$, $r \in \mathbb{N}$, $\mathcal{B}_1, \ldots, \mathcal{B}_m$ *— формулы над* R. *Если существуют такие* i, $1 \leqslant i \leqslant n$, $k_i \in \mathbb{Z}^+$, *что среди формул* $\mathcal{B}_1, \ldots, \mathcal{B}_m$ *переменная* x_i *встречается* $k_i r$ *раз, то формула* Φ_1 *эквивалентна формуле* $i_m(\mathcal{B}_1, \ldots, \mathcal{B}_m)$.

Действительно, используя утверждение 3 получаем, что для каждого $j = 1, \ldots, n$ существует $k_j \in \mathbb{Z}^+$, такое, что среди формул $\mathcal{B}_1, \ldots, \mathcal{B}_m$ переменная x_j встречается $k_j r$ раз. Эквивалентность формул Φ_1 и $i_m(\mathcal{B}_1, \ldots, \mathcal{B}_m)$ доказывается с помощью утверждения 1.

Следствие 1. *Пусть* $G \subseteq \mathrm{PS}^t$, $t > 1$, $G^+ = \{g(x_1, \ldots, x_m) \mid m \geqslant n\}$, $G^- = \{g(x_1, \ldots, x_m) \mid m < n\}$, $f(x_1, \ldots, x_n) \in \mathrm{PS} \cap [G]$. *Тогда* $f \in [G^+ \cup \mathrm{PS}^1]$ *и* $f \notin [G^- \cup \mathrm{PS}^1]$.

Следствие 2. *Пусть* G *— конечное множество функций из* PS^t. *Тогда множество* $[G] \cap (\mathrm{PS} \setminus \mathrm{PS}^1)$ *содержит лишь конечное число попарно неконгруэнтных функций.*

Утверждение 5. *Пусть* $t > 1$, $f(x_1, \ldots, x_n)$, $g(x_1, \ldots, x_m) \in \mathrm{PS}^t$, $d_f = d_g$ *и существует такое* $k \in \mathbb{Z}^+$, *что* $m - n = kt$. *Тогда* $f \in [\{g\}]$.

Это утверждение следует из равенства

$$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_{n-1}, \underbrace{x_n, \ldots, x_n}_{kt+1}).$$

Теорема. *Пусть* $t > 1$, G *— множество попарно неконгруэнтных функций из* PS^t, $F = [G]$. *Следующие условия равносильны:*

1) *множество* G *конечно;*
2) *класс* F *имеет конечный базис;*
3) *класс* F *имеет базис.*

Доказательство. Очевидно, что из 1) следует 2) и из 2) следует 3). Покажем, что из 3) следует 1).

Пусть F имеет базис $\mathfrak{A}$. Предположим, что множество G содержит счетное число попарно неконгруэнтных функций. Для каждой функции f из $\mathfrak{A}$ зафиксируем некоторую формулу Υ_f над G, реализующую функцию f. Обозначим через $\mathfrak{B}$ множество всех таких функций g из G, что для любой функции h из G, отличной от g, выполняется соотношение $g \notin [\{h\}]$. Используя утверждение 5, легко показать, что множество $\mathfrak{B}$ содержит не более $t^2 + 1$ различных функций. Следовательно, существует конечное множество $\mathfrak{A}_B \subseteq \mathfrak{A}$, такое, что $\mathfrak{B} \subseteq [\mathfrak{A}_B]$. Положим $G_B = (G \backslash \mathrm{PS}^1) \cap [\mathfrak{B}]$. В силу следствия 2 множество G_B конечно.

Рассмотрим множество $F \cap \mathrm{PS}^1$. В силу утверждения 2 существует конечное множество $\mathfrak{A}_I$, такое, что $F \cap \mathrm{PS}^1 \subseteq [\mathfrak{A}_I]$.

Положим $\widehat{\mathfrak{A}} = \mathfrak{A}_B \cup \mathfrak{A}_I$, $\widehat{G} = \cup \Theta(\Upsilon_h)$, где объединение берется по всем функциям h из множества $\widehat{\mathfrak{A}}$. Покажем, что $\mathfrak{A} \subseteq \widehat{\mathfrak{A}}$. Предположим, что существует функция $f(x_1, \ldots, x_n) \in \mathfrak{A} \backslash \widehat{\mathfrak{A}}$. Обозначим через n_0 максимальное число существенных переменных у функций из множества $\widehat{G} \cup \Theta(\Upsilon_f)$. Пусть Υ_f — формула над G, реализующая функцию f. Если для любой функции $g \in \Theta(\Upsilon_f)$ выполняется соотношение $g \in G_B$, то $f \in [\mathfrak{A}_B] \subseteq \mathfrak{A} \backslash \{f\}$, что противоречит определению базиса.

Рассмотрим функцию $g \in \Theta(\Upsilon_f) \backslash G_B$. Поскольку $g \notin \mathfrak{B}$ и множество G содержит счетное число попарно неконгруэнтных функций, то существует функция $h(x_1, \ldots, x_s) \in G$, такая, что $g \in [\{h\}]$ и $s > n_0$. В силу следствия 1 выполняется соотношение $h \notin [\widehat{G}]$. Следовательно, $h \notin [\widehat{\mathfrak{A}} \cup \{f\}]$. А значит, $h \in [\mathfrak{A} \backslash (\widehat{\mathfrak{A}} \cup \{f\})]$. Поэтому $f \in [\mathfrak{A} \backslash \{f\}]$. Это противоречит тому, что $\mathfrak{A}$ — базис. ∎

Автор выражает благодарность проф. А. Б. Угольникову за обсуждение результатов работы.

Работа выполнена при финансовой поддержке РФФИ (проект 11–01–00508) и программы фундаментальных исследований ОМН РАН «Алгебраические и комбинаторные методы математической кибернетики и информационные системы нового поколения».

Список литературы

[1] *Post E. L.* The two-valued iterative systems of mathematical logic. — Annals of Math. Studies. Princeton Univ. Press, 1941.

[2] *Янов Ю. И., Мучник А. А.* О существовании k-значных замкнутых классов, не имеющих конечного базиса // ДАН СССР. — 1959. — Т. 127, № 1. — С. 44–46.

[3] *Михайлович А. В.* О замкнутых классах трехзначной логики, порожденных симметрическими функциями // Вестник Моск. ун-та. Сер. 1. Математика. Механика. — 2008. — № 4. — С. 54–57.

[4] *Михайлович А. В.* О классах трехзначной логики, порожденных монотонными симметрическими функциями // Вестник Моск. ун-та. Сер. 1. Математика. Механика. — 2009. — № 1. — С. 33–37.

[5] *Михайлович А. В.* О замкнутых классах многозначной логики, порожденных функциями со специальными свойствами // Материалы XVII Междунар. школы-семинара «Синтез и сложность управляющих систем» (Новосибирск, 27 октября – 1 ноября 2008 г.) — Новосибирск: Изд-во Института математики, 2008. — С. 117–122.

Метод принятия решения комитетом нейронных сетей при решении плохо обусловленных задач восстановления функциональных зависимостей

О. А. Мишулина, И. А. Круглов

mishulina@gmail.com, i-kruglov@yandex.ru

Национальный исследовательский ядерный университет «МИФИ», Москва

Постановка задачи

В работе рассматривается задача оценивания значений параметров объекта по результатам измерений. Допустим, имеется некоторый исследуемый объект (например, физический процесс или явление), интересующие свойства которого могут быть описаны двумя различными векторами характеристик. Обозначим их $\mathbf{x}$ и $\mathbf{y}$. Как правило, один из векторов является более удобным и доступным для измерений (будем считать, что это $\mathbf{y}$), а второй — для описания теоретических свойств объекта. В качестве конкретного примера задачи такого класса можно привести определение значений механических характеристик металла [1] ($\mathbf{x}$) по диаграмме индентирования ($\mathbf{y}$) [2]. Поскольку оба вектора характеристик описывают один и тот же объект, можно определить два отображения $\mathbf{y} = \mathbf{f}(\mathbf{x})$ и $\mathbf{x} = \mathbf{f}^{-1}(\mathbf{y})$, представляющие собой *прямую* и *обратную* задачи оценивания параметров объекта соответственно. Будем считать, что оба отображения существуют и являются взаимно-однозначными. Однако их строгий теоретический вид на практике может быть неизвестен. Мы остановимся именно на этом случае, когда требуется определить зависимость между параметрами объекта $\mathbf{x}$ и определяемыми по результатам наблюдений параметрами $\mathbf{y}$ (т.е. решить обратную задачу) на основе имеющейся выборки полученных экспериментально пар значений $\left(\mathbf{x}^{(p)}, \mathbf{y}^{(p)}\right)$, $p = 1, 2, \ldots, P$. В данной ситуации наиболее целесообразным является использование моделей на основе нейронных сетей [3]. Особый интерес представляет собой случай, когда обратная задача является плохо обусловленной (или некорректной) [4]. Следуя критериям, приведенным в монографии [4], в данной работе под плохой обусловленностью обратной задачи будем понимать большие ошибки определения значений вектора $\mathbf{x}$ при незначительных ошибках измерения вектора $\mathbf{y}$. В такой ситуации использование единственной нейронной сети для построения модели зависимости $\mathbf{x} = \mathbf{f}^{-1}(\mathbf{y})$ является недостаточным из-за недопустимо большой погрешности получаемого решения. В предыдущей работе авторов [5] для повышения точности нейросетевого решения плохо обусловленной обратной задачи было предложено применить комитет нейронных сетей [6], использующий решение прямой задачи. Целью настоящей работы является совершенствование метода принятия решения на основании результатов обработки данных нейросетями комитета.

Метод принятия решения в комитете нейронных сетей

Описание работы комитета. Комитет нейронных сетей, реализующий отображение $\mathbf{x} = \mathbf{f}^{-1}(\mathbf{y})$, включает $K \in \mathbb{N}$ многослойных нейронных сетей (МНС), каждая из которых решает обратную задачу оценивания параметров $\mathbf{x}$, одну МНС, решающую прямую задачу, и модуль принятия решения. Первая группа сетей реализует преобразование

$$\widetilde{\mathbf{x}}^{(k)} = \widetilde{\mathbf{f}}_k^{-1}(\mathbf{y}), \quad k = 1, 2, \ldots, K,$$

где $\widetilde{\mathbf{f}}_k^{-1}(\mathbf{y})$ — функция, моделируемая k-й сетью решения обратной задачи. Это преобразование может быть представлено в виде

$$\widetilde{\mathbf{x}}^{(k)} = \mathbf{f}^{-1}(\mathbf{y}) + \varepsilon_k(\mathbf{y}), \quad k = 1, 2, \ldots, K, \tag{1}$$

где $\mathbf{f}^{-1}(\mathbf{y})$ — истинное (неизвестное) решение обратной задачи, а $\varepsilon_k(\mathbf{y})$ — ошибка нейросетевого решения обратной задачи. Все сети первой группы обучаются независимо друг от друга на отдельных (возможно, пересекающихся) выборках данных. В связи с этим при предъявлении комитету конкретного входного вектора признаков $\mathbf{y}$ некоторые из значений $\widetilde{\mathbf{x}}^{(k)}$ будут ближе к искомому вектору $\mathbf{x}$, чем другие. Поэтому возникает необходимость в определении оценки ошибки значений $\widetilde{\mathbf{x}}^{(k)}$ для выборки наиболее точных из них. Для этих целей используется нейросетевое решение хорошо обусловленной прямой задачи, которое может быть применено для любой оценки $\widetilde{\mathbf{x}}^{(k)}$, $k = 1, 2, \ldots, K$:

$$\widetilde{\mathbf{y}}^{(k)} = \widetilde{\mathbf{f}}(\widetilde{\mathbf{x}}^{(k)}) = \mathbf{f}(\widetilde{\mathbf{x}}^{(k)}) + \delta(\widetilde{\mathbf{x}}^{(k)}), \tag{2}$$

где $\mathbf{f}(\widetilde{\mathbf{x}}^{(k)})$ — истинное (неизвестное) решение прямой задачи, $\delta(\widetilde{\mathbf{x}}^{(k)})$ — ошибка нейросетевого решения прямой задачи.

Метод принятия решения. Рассмотрим идеальную ситуацию, в которой отсутствуют ошибки работы сетей, т.е. $\varepsilon_k(\mathbf{y}) = \delta(\widetilde{\mathbf{x}}^{(k)}) = 0$ для $k = 1, 2, \ldots, K$. В таком случае моделируемые сетями функции являются истинными решениями обратной и прямой задач, что приводит к $\widetilde{\mathbf{x}}^{(k)} = \mathbf{x}$ и $\widetilde{\mathbf{y}}^{(k)} = \mathbf{f}(\widetilde{\mathbf{x}}^{(k)}) = \mathbf{f}(\mathbf{x}) = \mathbf{y}$. Таким образом, если ввести в рассмотрение метрику ρ, с помощью которой будет измеряться расстояние между векторами $\widetilde{\mathbf{y}}^{(k)}$ и $\mathbf{y}$, то в данной ситуации $\rho\big(\widetilde{\mathbf{y}}^{(k)}, \mathbf{y}\big) = 0$. В реальных условиях мы всегда имеем $\varepsilon_k(\mathbf{y}) \neq 0$ и $\delta(\widetilde{\mathbf{x}}^{(k)}) \neq 0$. Однако, используя разложение в ряд Тейлора до первого члена в окрестности $\mathbf{y}$ и учитывая (1) и (2), нетрудно показать, что $\rho\big(\widetilde{\mathbf{y}}^{(k)}, \mathbf{y}\big) \propto \varepsilon_k(\mathbf{y}) + \delta(\widetilde{\mathbf{x}}^{(k)})$. Причём из-за того, что обратная задача является плохо обусловленной, величина $\varepsilon_k(\mathbf{y})$ значительно превосходит $\delta(\widetilde{\mathbf{x}}^{(k)})$, поэтому последней можно пренебречь. Таким образом, получаем, что величина ошибки $\varepsilon_k(\mathbf{y})$ k-й ($k = 1, 2, \ldots, K$) сети, решающей обратную задачу, может быть оценена при помощи значения $\rho\big(\widetilde{\mathbf{y}}^{(k)}, \mathbf{y}\big)$. Изложенный подход к оцениванию ошибки работы нейронных сетей первой группы лежит в основе предлагаемого метода принятия решения в комитете нейронных сетей.

Этапы обучения блока принятия решения комитетом нейросетей:

1. Для каждого обучающего примера, состоящего из пар значений $\big(\mathbf{y}^{(p)}, \mathbf{x}^{(p)}\big)$, $p = 1, 2, \ldots, P$ (P — число обучающих примеров), определить значения $\widetilde{\mathbf{x}}^{(k,p)}$, $k = 1, 2, \ldots, K$, формируемые сетями, решающими обратную задачу.

2. Вычислить $\rho_0^{(p)} = \rho\left(\mathbf{y}^{(p)}, \widetilde{\mathbf{f}}(\mathbf{x})^{(p)}\right)$ для $p = 1, 2, \ldots, P$, которые отражают степень неточности результатов, формируемых сетью решения прямой задачи.
3. Используя значения $\rho\left(\mathbf{y}^{(p)}, \widetilde{\mathbf{f}}\left(\widetilde{\mathbf{x}}^{(k,p)}\right)\right)$, $k = 1, 2, \ldots, K$, построить эмпирическую функцию распределения вероятности $F_p^*(\rho)$ и вычислить её значение при $\rho = \rho_0^{(p)}$: $\alpha^{(p)} = F_p^*(\rho_0^{(p)})$.
4. Наконец, определить значение $\alpha = \frac{1}{P}\sum_{p=1}^{P} \alpha^{(p)}$.

Функционирование обученного блока принятия решения комитетом:

1. Для заданного входного вектора $\mathbf{y}$ при помощи сетей решения обратной задачи определить значения $\widetilde{\mathbf{x}}^{(k)}$, $k = 1, 2, \ldots, K$.
2. Построить эмпирическую функцию распределения $F_p^*(\rho)$ по значениям $\rho\left(\mathbf{y}, \widetilde{\mathbf{f}}\left(\widetilde{\mathbf{x}}^{(k)}\right)\right)$, $k = 1, 2, \ldots, K$, и рассчитать ρ_α из условия $F_p^*(\rho_\alpha) = \alpha$.
3. Вычислить решение комитета $\mathbf{x}^{(k^*)}$, где

$$k^* = \arg\min \left|\rho\left(\mathbf{y}, \widetilde{\mathbf{f}}(\mathbf{x}^{(k)})\right) - \rho_\alpha\right|.$$

Работа комитета с предложенным статистическим методом принятия решения была проверена на примере задачи определения значений двух прочностных характеристик ($\mathbf{x} = (x_1, x_2)^T$) группы сталей по результатам кинетического индентирования. Было установлено, что комитет позволяет добиться существенного повышения точности по сравнению с одиночными МНС даже на зашумленных данных (см. таблицу 1).

Таблица 1 Результаты тестирования для данных с наложенным шумом амплитудой в 5% от незашумленного значения

Относительная ошибка	Одиночная МНС		Комитет	
	x_1	x_2	x_1	x_2
Средняя	14.0%	23.4%	10.4%	16.5%
Максимальная	47.8%	110.8%	26.4%	45.7%

Список литературы

[1] *Tabor D.* The hardness of metals. — Oxford University Press, 2000. — 175 p.

[2] *Бакиров М. Б., Потапов В. В., Фролов И. В.* Разработка расчётно-экспериментальных методик получения механических характеристик на основе метода кинетического индентирования // Мир измерений. — 2006. — № 8. — С. 5–11.

[3] *Хайкин С.* Нейронные сети: полный курс, 2-е издание. — М.: Издательский дом «Вильямс», 2006. — 1104 с.

[4] *Тихонов А. Н., Арсенин В. Я.* Методы решения некорректных задач. Изд. 3-е, испр. — М.: Наука, 1986. — 288 с.

[5] *Мишулина О. А., Бакиров М. Б., Круглов И. А.* Анализ точности нейросетевого решения задачи восстановления механических характеристик металла // XIII Всероссийская научно-техническая конференция «Нейроинформатика-2011». Сборник научных трудов. Часть 2. — М: НИЯУ МИФИ, 2010. — С. 163–171.

[6] *Терехов С. А.* Гениальные комитеты умных машин // IX Всероссийская научно-техническая конференция «Нейроинформатика-2007». Лекции по нейроинформатике. Часть 2.— М: МИФИ, 2007. — С. 11–42.

Упаковки и покрытия 3-путей

Д. Б. Мокеев

MokeevDB@mail.ru

Нижегородский государственный университет им. Н. И. Лобачевского

Характеризуется класс графов, у которых максимальное число непересекающихся порожденных подграфов P_3 равно минимальному числу вершин, представляющих все такие подграфы, и это равенство наследуется всеми порожденными подграфами.

Введение

Пусть $\mathbf{X}$ — множество графов. $\mathbf{X}$-*упаковкой* графа G называется множество его непересекающихся порожденных подграфов, каждый из которых изоморфен какому-нибудь графу из $\mathbf{X}$. Наибольшее число подграфов в $\mathbf{X}$-упаковке графа G будем обозначать через $pack(\mathbf{X}; G)$. $\mathbf{X}$-*покрытием* графа G называется множество вершин, после удаления которых получается граф, не содержащий порожденных подграфов, принадлежащих $\mathbf{X}$. Наименьшее число вершин в $\mathbf{X}$-покрытии графа G будем обозначать через $cover(\mathbf{X}; G)$. В случае когда $\mathbf{X}$ состоит из единственного графа H, будем говорить просто об H-покрытии и H-упаковке. В частности, K_2-упаковки — это паросочетания, а K_2-покрытия известны как вершинные покрытия.

Очевидно неравенство $pack(\mathbf{X}; G) \leqslant cover(\mathbf{X}; G)$. Теорема Кёнига утверждает, что для двудольных графов имеет место равенство $pack(P_2; G) = cover(P_2; G)$. Верно и в известном смысле обратное утверждение: если это равенство выполняется для графа G и любого его порожденного подграфа, то этот граф — двудольный.

Граф G будем называть *кёниговым* графом относительно множества $\mathbf{X}$, если для любого его порожденного подграфа H выполняется равенство $pack(\mathbf{X}; H) = cover(\mathbf{X}; H)$. Класс всех кёниговых графов относительно $\mathbf{X}$ обозначим через $\mathbf{K}(\mathbf{X})$.

Класс $\mathbf{K}(\mathbf{X})$ при любом $\mathbf{X}$ является наследственным и, следовательно, может быть описан множеством минимальных запрещенных (порожденных) подграфов. Для P_2 такую характеризацию дает теорема Кёнига вместе с известным критерием двудольности. Кроме этой классической теоремы нам известен еще только один результат такого рода для обыкновенных графов — в работе [1] описаны все запрещенные подграфы для класса $\mathbf{K}(\mathbf{C})$, где $\mathbf{C}$ — множество всех простых циклов.

Цель настоящей работы — охарактеризовать класс $\mathbf{K}(P_3)$. Мы применяем «конструктивный» подход к описанию этого класса: показываем, как можно построить любой граф из этого класса с помощью операций подразбиения ребер и замены вершин кликами.

Далее вместо $pack(P_3; G)$ и $cover(P_3; G)$ пишем просто $pack(G)$ и $cover(G)$ соответственно, под покрытием и упаковкой подразумеваем P_3-покрытие и P_3-упаковку, а под кёниговым графом — кёнигов граф относительно P_3.

Расширение графов

Заметим, что граф кёнигов тогда и только тогда, когда каждая его компонента связности — кёнигов граф. Поэтому мы будем рассматривать только связные графы.

Операция замены вершины x t-кликой состоит в том, что эта вершина удаляется из графа, к нему добавляются t новых вершин, все они попарно соединяются между собой и каждая из них соединяется ребром с каждой вершиной, с которой была смежна x. Граф, получаемый из графа G заменой некоторых его вершин степени 1 и 2 кликами (возможно, разного размера), назовем расширением графа G, а клики, на которые были заменены вершины, будем называть секциями. Каждая вершина, не подвергавшаяся замене, считается отдельной секцией.

Лемма 1. *Каждое наименьшее покрытие любого расширения любого графа состоит из целых секций.*

Лемма 2. *Любое расширение любого дерева является кёниговым графом.*

Два подкласса кёниговых графов

Связные графы из класса $\mathbf{K}(P_3)$ удобно разделить на две категории: расширенные циклы и все остальные графы (их будем называть ординарными). Множество расширенных циклов, принадлежащих $\mathbf{K}(P_3)$, обозначим через $\mathbf{KC}$, а множество ординарных графов — через $\mathbf{KR}$.

Пусть H — мультиграф без петель. Каждое цикловое ребро (ребро, принадлежащее какому-нибудь циклу) этого мультиграфа подразобьем двумя вершинами. Эти вершины будем называть новыми. Заменим каждую новую вершину и каждую вершину степени 1 или 2, не принадлежащую циклу, какой-нибудь кликой. Полученный таким образом граф будем называть 2-расширением исходного мультиграфа.

Теорема 3. *Следующие утверждения равносильны для связного графа G:*

1) *G — ординарный кёнигов граф;*
2) *G является 2-расширением некоторого мультиграфа, отличного от простого цикла.*

Что касается расширения циклов, то для них задача полного описания остаётся нерешенной. Можно, однако, сформулировать теорему для случаев, когда в расширяемом цикле не более 9 вершин:

Теорема 4. *Любое расширение циклов C_6 и C_9 является кёниговым графом.*

Список литературы

[1] *Ding G., Xu Z., Zang W.* Packing cycles in graphs. II // J. Combinatorial Theory. Ser. B. — 2003. — V. 87. — P. 244–253.

О тестах относительно локальных линейных слипаний входов схем

Е. В. Морозов, Д. С. Романов

antake@yandex.ru, romanov@cs.msu.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Пусть n, k — натуральные числа ($2 \leqslant k \leqslant n$), $f(\tilde{x}^n)$ — булева функция и $\Phi = \Phi(\tilde{y}^k) = \left\{\varphi_i(\tilde{y}^k) | i = 1, \ldots, 2^{k+1}\right\}$ — система всевозможных попарно неравных линейных булевых функций k переменных. Обозначим через $\Psi = \Psi_{n,k,f}(\tilde{x}^n)$ систему функций, в которую входит функция $f(\tilde{x}^n)$ и всевозможные функции

$$\psi_{j,i}(\tilde{x}^n) =$$

$$f(x_1, \ldots, x_{j-1}, \underbrace{\varphi_i(x_j, \ldots, x_{j+k-1}, \ldots, \varphi_i(x_j, \ldots, x_{j+k-1}}_{k \text{ раз}}, x_{j+k}, \ldots, x_n),$$

где $1 \leqslant j \leqslant n - k + 1$, $i \in \left\{1, \ldots, 2^{k+1}\right\}$. Тест для системы $\Psi_{n,k,f}(\tilde{x}^n)$ будем называть полным тестом относительно локальных k-кратных линейных слипаний переменных функции f (не введенные в этой работе понятия можно найти, например, в [1, 2]). Длину теста T (число наборов в нем) будем обозначать через $l(T)$ или $|T|$. Традиционным образом введем функции Шеннона $L_k^{\mathrm{check}}(n)$ и $L_k^{\mathrm{diagn}}(n)$ длины полного проверяющего и полного диагностического (соответственно) теста относительно локальных k-кратных линейных слипаний переменных как максимум по всем булевым функциям $f(\tilde{x}^n)$ минимальной длины теста требуемого типа для булевой функции f. В настоящей работе приводятся оценки для этих функций Шеннона.

Таблицу с 2^k строками и 2^{k+1} столбцами, состоящую из столбцов значений всех функций из Φ, обозначим через $M(\Phi)$, а поэлементное отрицание этой таблицы через $\overline{M(\Phi)}$. Подкубом $B_{k,j}^{\tilde{\alpha},\tilde{\beta}}$, где $\tilde{\alpha} = (\alpha_1, \ldots, \alpha_{j-1})$, а $\tilde{\beta} = (\beta_{j+k}, \ldots, \beta_n)$, будем называть грань Γ_γ, где $\gamma = (\tilde{\alpha}, 2, \ldots, 2, \tilde{\beta})$, двойки стоят на местах переменных $x_j, \ldots, x_{j+k-1}$ и обозначают произвольность значений этих переменных. Зафиксируем число j $(1 \leqslant j \leqslant n-k+1)$. Рассмотрим таблицу неисправностей, построенную для произвольной булевой функции $f(\tilde{x}^n)$ на всех (следующих в лексикографическом порядке) наборах подкуба $B_{k,j}^{\tilde{\alpha},\tilde{\beta}} = (\tilde{\alpha}^{j-1}, \tilde{2}^k, \tilde{\beta}^{n-j-k+1})$. Вектор значений функции $f'(\tilde{x}^n)$ на подкубе $B_{k,j}^{\tilde{\alpha},\tilde{\beta}}$ обозначим через $f'(B_{k,j}^{\tilde{\alpha},\tilde{\beta}})$. Заметим, что если $f(\tilde{\alpha}^{j-1}, \tilde{0}^k, \tilde{\beta}^{n-j-k+1}) = f(\tilde{\alpha}^{j-1}, \tilde{1}^k, \tilde{\beta}^{n-j-k+1}) = \sigma$, то для любой функции $\psi_{j,i}$ вектор $\psi_{j,i}(B_{k,j}^{\tilde{\alpha},\tilde{\beta}})$ равен $(\tilde{\sigma}^{2^k}) = (\sigma, \ldots, \sigma)$. При этом, если существует такой булев набор $\tilde{\delta}^k$, что $f(\tilde{\alpha}^{j-1}, \tilde{\delta}^k, \tilde{\beta}^{n-j-k+1}) \neq \sigma$, то этот набор проверяет все функции неисправностей $\psi_{j,1}(\tilde{x}^n), \ldots, \psi_{j,2^{k+1}}(\tilde{x}^n)$. В этом случае тройку наборов $(\tilde{\alpha}^{j-1}, \tilde{0}^k, \tilde{\beta}^{n-j-k+1})$, $(\tilde{\alpha}^{j-1}, \tilde{1}^k, \tilde{\beta}^{n-j-k+1})$, $(\tilde{\alpha}^{j-1}, \tilde{\delta}^k, \tilde{\beta}^{n-j-k+1})$, такую, что

$$f(\tilde{\alpha}^{j-1}, \tilde{0}^k, \tilde{\beta}^{n-j-k+1}) = f(\tilde{\alpha}^{j-1}, \tilde{1}^k, \tilde{\beta}^{n-j-k+1}) \neq f(\tilde{\alpha}^{j-1}, \tilde{\delta}^k, \tilde{\beta}^{n-j-k+1}),$$

назовем замечательной тройкой для x_j, а набор $(\tilde{\alpha}^{j-1}, \tilde{\delta}^k, \tilde{\beta}^{n-j-k+1})$ — особым набором для x_j. Проверкой переменной x_j назовем процесс построения множества наборов, обнаруживающих всевозможные линейные локальные k-кратные слипания переменных $x_j, \ldots, x_{j+k-1}$. Если $f(\tilde{\alpha}^{j-1}, \tilde{0}^k, \tilde{\beta}^{n-j-k+1}) = 0$, $f(\tilde{\alpha}^{j-1}, \tilde{1}^k, \tilde{\beta}^{n-j-k+1}) = 1$, то фрагмент таблицы неисправностей, связанный с проверкой переменной x_j, на подкубе $B_{k,j}^{\tilde{\alpha},\tilde{\beta}}$ имеет вид $M(\Phi)$. Если же $f(\tilde{\alpha}^{j-1}, \tilde{0}^k, \tilde{\beta}^{n-j-k+1}) = 1$, $f(\tilde{\alpha}^{j-1}, \tilde{1}^k, \tilde{\beta}^{n-j-k+1}) = 0$, то фрагмент таблицы неисправностей, связанный с проверкой переменной x_j на подкубе $B_{k,j}^{\tilde{\alpha},\tilde{\beta}}$, имеет вид $\overline{M(\Phi)}$. В последних двух случаях для проверки функций неисправностей $\psi_{j,1}, \ldots, \psi_{j,2^{k+1}}(\tilde{x}^n)$ достаточно взять некоторые $(k+1)$ наборов подкуба $B_{k,j}^{\tilde{\alpha},\tilde{\beta}}$ и еще один некоторый набор. Вектор $f[B_{k,j}^{\tilde{\alpha},\tilde{\beta}}] = (\xi_1, \ldots, \xi_{2^k})$ (где $\xi_1 \neq \xi_{2^k}$) назовем перспективным, если в нем найдутся две неравные компоненты, отстоящие друг от друга на 2^{k-1}; соответствующую этим компонентам пару входных наборов функции f назовем перспективной парой.

Теорема 1. $n-k+1+\left\lceil \frac{n-k+1}{k-1} \right\rceil \leqslant L_k^{\text{check}}(n) \leqslant \left\lceil \frac{n-k+1}{k} \right\rceil 4k.$

Доказательство. Пусть проверяется переменная x_j. Рассмотрим различные случаи.

1. Для переменной x_j существует замечательная тройка. Тогда особый набор из нее включаем в тест, переходим к следующему шагу.

2. Замечательных троек нет, но есть перспективная пара наборов. Рассмотрим k подкубов $B_{k,j}^{\tilde{\alpha}^{(1)},\tilde{\beta}^{(1)}}, \ldots, B_{k,j-k+1}^{\tilde{\alpha}^{(k)},\tilde{\beta}^{(k)}}$, содержащих эту пару. Каждый подкуб имеет неконстантный вектор. Подкуб с нижним индексом (k, m) будем использовать для проверки переменной x_m. Выберем набор $\tilde{\gamma} = (\gamma_1, \ldots, \gamma_n)$, принадлежащий всем подкубам. Добавим в тест его и все соседние наборы по переменным $x_{j-k+1}, \ldots, x_{j+k-1}$. Тогда для любой линейной функции, подающейся вместо переменных $x_m, \ldots, x_{m+k-1}$, $m = \overline{j-k+1, j}$, в тесте будут $(k+1)$ наборов, позволяющих отличить эту функцию от других линейных функций k переменных. Добавим еще по одному некоторому набору для каждой из переменных $x_{j-k+1}, \ldots, x_j$. Итого, сделано k шагов, не более $3k$ наборов добавлено в тест.

Можно показать, что и во всех остальных случаях за каждые k шагов в тест добавляется не более чем $4k$ наборов. Подытоживая, получаем верхнюю оценку.

Докажем теперь нижнюю оценку. Будем рассматривать функцию $h(\tilde{x}^n) = x_1 x_2 \cdots x_n$. Пусть T — некоторое неупорядоченное множество наборов, $1 \leqslant j \leqslant n-k+1$. Всякий булев набор вида $(1, \ldots, 1, \alpha_{j-k+1}, \ldots, \alpha_j, 1, \ldots, 1)$ назовем x_j-набором. Пусть $\chi(\tilde{\alpha})$ есть число тех переменных x_j, для которых набор $\tilde{\alpha} \in B^n$ является x_j-набором. Пусть, далее, $\nu_j(T)$ есть число различных x_j-наборов, входящих в множество T. Нетрудно показать, что если найдется $j' \in \{1, 2, \ldots, n-k+1\}$, такое, что $\nu_{j'}(T) < k+2$, то T не является полным проверяющим тестом относительно локальных линейных k-кратных слипаний переменных для $h(\tilde{x}^n)$. Обозначим $\hat{\nu}_j(T) = \min(\nu_j(T), k+2)$. Тогда ясно, что для того чтобы множество T было тестом указанного типа, необходимо, чтобы величина $I(T) = \sum\limits_{j=1}^{n-k+1} \hat{\nu}_j(T)$ была бы не меньше $(n-k+1)(k+2)$. А следовательно, и величина $J(T) = \sum\limits_{\tilde{\alpha} \in T} \chi(\tilde{\alpha})$ должна быть не меньше $(n-k+1)(k+2)$ (так как $J(T) = \sum\limits_{j=1}^{n-k+1} \nu_j(T)$). Заметим, что $\chi(\tilde{1}^n) = n-k+1$, что при любом j $(1 \leqslant j \leqslant n-k+1)$ $\chi(\tilde{1}^{j-1}, 0, \tilde{1}^{n-j}) = k$ и что для всех остальных наборов $\tilde{\alpha} \in B^n$: $\chi(\tilde{\alpha}) \leqslant k-1$. Теперь легко видеть, что для выполнения неравенства $J(T) \geqslant (n-k+1)(k+2)$ необходимо, чтобы $|T| \geqslant (n-k+1) + \left\lceil \frac{(n-k+1)(k+2)-(n-k+1)k-(n-k+1)}{k-1} \right\rceil = n-k+1+\left\lceil \frac{n-k+1}{k-1} \right\rceil$. ∎

Теорема 2 [3]. *Пусть $n, k \in N$, $2 \leqslant k \leqslant n$. Пусть, далее, $l(\Phi)$ — длина минимального диагностического теста для матрицы, составленной из всех столбцов системы Φ. Тогда, начиная с некоторого n, имеет место неравенство $L_k^{\mathrm{diagn}}(n) \geqslant (l(\Phi)-1)(n-k+1)$.*

В случае линейных локальных слипаний $l(\Phi) = k+1$. Поэтому получаем нижнюю оценку функции Шеннона $L_k^{\mathrm{diagn}}(n) \geqslant k(n-k+1)$.

Лемма [1]. *Если строки теста T для матрицы M линейно зависимы относительно операций покоординатного сложения строк по модулю 2 и умножения строки на число 0 или 1, то тест T не является тупиковым.*

Через $M^+(\Phi)$ будем обозначать матрицу, полученную из $M(\Phi)$ добавлением снизу строки из единиц, через $w(M)$ — максимальное число линейно независимых строк M.

Теорема 3. *Пусть $n, k \in N$, $2 \leqslant k \leqslant n$. Тогда имеет место неравенство $L_k^{\mathrm{diagn}}(n) \leqslant w(M^+(\Phi))(n-k+1)+1$.*

Доказательство. Образуем линейное пространство R как линейную оболочку всех строк матрицы $M^+(\Phi)$ относительно операций покоординатного сложения строк по модулю 2 и умножения строки на число 0 или 1, а линейное пространство r — как пространство из двух элементов $0, 1$ относительно тех же операций. Ясно, что $\dim R = w(M^+(\Phi))$, $\dim r = 1$. Рассмотрим произвольную булеву функцию $f(\tilde{x}^n)$. Легко видеть, что каждая строка таблицы неисправностей имеет вид $(\theta, \tilde{\xi}^{(1)}, \ldots, \tilde{\xi}^{(1)})$, где $\theta \in r$, $\tilde{\xi}^{(j)} \in R$ $(j = \overline{1; n-k+1})$. Можно показать, что размерность линейной оболочки строк такого вида равна $1 + w(M^+(\Phi))(n-k+1)$, откуда с учетом предыдущей леммы следует требуемое. ∎

Теперь из двух предыдущих теорем имеем:

Теорема 4. $k(n-k+1) \leqslant L_k^{\mathrm{diagn}}(n) \leqslant (k+1)(n-k+1)+1$.

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг.

Список литературы

[1] *Соловьев Н. А.* Тесты (теория, построение, применение). — Новосибирск: Наука (Сибирское отделение), 1978. — 192 с.

[2] *Кузнецов И. А., Романов Д. С.* О полных проверяющих тестах относительно локальных слипаний переменных в булевых функциях // Ученые записки Казанского государственного университета. Серия Физико-математические науки. — 2009. — Т. 151, книга 2. — С. 90–97.

[3] *Романов Д. С., Казачёк А. Е.* Об оценках функций Шеннона длин локальных тестов относительно слипаний // Материалы XVI Международной школы-семинара «Синтез и сложность управляющих систем». — М.: Изд-во мех.-мат. ф-та МГУ, 2006. — С. 91–97.

.

О сложности некоторых функций для вероятностных вычислений

Р. Г. Мубаракзянов

rustam@ksu.ru

Казанский (Приволжский) федеральный университет

Рассматриваются различные модели вероятностных вычислений булевых функций. Показана иерархия классов сложности.

Определения

Ветвящейся программой [1] на переменных X называется ориентированный ациклический граф с одной начальной вершиной (корнем) и двумя финальными вершинами, помеченными 0 и 1. Каждая нефинальная вершина этого графа помечена переменной из X, и из нее выходят ровно 2 дуги, помеченные 0 и 1. Процесс вычисления при фиксировании значений переменных из X сводится к следующему. Вычисление начинается в корне s программы. Значение вычисляемой программой функции – пометка той финальной вершины, в которую приводит вычисление. Если на каждом вычислительном пути любая переменная читается не более одного раза, программа называется *один раз читающей* ($BP1$), если при этом переменные читаются в соответствии с фиксированным порядком, то программа называется *упорядоченной один раз читающей*, или $OBDD$.

Вероятностная ветвящаяся программа имеет в дополнение к детерминированным вероятностные узлы, помеченные дополнительными переменными. Значение такой переменной выбирается случайно, с одинаковой вероятностью $1/2$ — значение 0 или 1. Для такой программы B значения детерминированных переменных $(a_1, a_2, \ldots, a_n)$ определяют вероятность $c_B(a_1, a_2, \ldots, a_n)$ попадания в финальную 1-вершину. Как и в классическом случае, различают несколько типов вероятностных вычислений. Вероятностная бинарная программа B вычисляет функцию f:

1) *без ограничения на ошибку*, если

$$f(a_1, a_2, \ldots, a_n) = 1 \Leftrightarrow c_B(a_1, a_2, \ldots, a_n) > 1/2,$$

$$f(a_1, a_2, \ldots, a_n) = 0 \Leftrightarrow c_B(a_1, a_2, \ldots, a_n) < 1/2;$$

2) *с ограничением на ошибку*, если существует константа $\varepsilon < 1/2$, такая, что

$$f(a_1, a_2, \ldots, a_n) = 1 \Leftrightarrow c_B(a_1, a_2, \ldots, a_n) \geqslant 1 - \varepsilon,$$

$$f(a_1, a_2, \ldots, a_n) = 0 \Leftrightarrow c_B(a_1, a_2, \ldots, a_n) \leqslant \varepsilon.$$

Количество вершин ветвящейся программы называется ее *сложностью*, или размером.

Результаты

Лемма. *Вычисление конечным вероятностным автоматом (КВА) [2] является частным случаем вычисления вероятностной $OBDD$.*

Теорема 1. *Класс функций, вычислимых вероятностными с ограничением на ошибку $OBDD$ полиномиального размера, является строгим подмножеством класса функций, вычислимых вероятностными без ограничения на ошибку $OBDD$ полиномиального размера.*

Доказательство основано на представлении функции, для вычисления которой существует вероятностная без ограничения на ошибку $OBDD$ линейного размера, а любая вычисляющая ее вероятностная с ограничением на ошибку $OBDD$ имеет экспоненциальный размер.

Теорема 2. *Класс функций, вычислимых вероятностными с ограничением на ошибку $OBDD$ полиномиального размера, является строгим подмножеством класса функций, вычислимых вероятностными без ограничения на ошибку $BP1$ полиномиального размера.*

Список литературы

[1] *Wegener I.* Branching Programs and Binary Desicion Diagrams: Theory and Applications. SIAM Monographs on Discrete Mathematics and Applications. — Philadelphia, 2000.

[2] *Бухараев Р. Г.* Основы теории вероятностных автоматов. — М.: Наука, 1985.

О свойствах теоретико-множественных операций над предполными классами трехзначной логики

А. С. Нагорный

anagorny@list.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Введение

Пусть k — натуральное число, $k \geqslant 2$, $E_k = \{0, 1, \dots, k-1\}$, P_k — множество всех конечноместных функций на E_k. Элементы множества P_k будем называть функциями k-значной логики или k-значными функциями.

Определения используемых ниже операции суперпозиции, понятий замыкания, замкнутого класса, полноты и базиса можно найти в [1, §§ 2–7].

Замкнутый (относительно суперпозиции) класс H функций k-значной логики назовем предполным в P_k, если $H \neq P_k$, но для любой функции $f \in P_k \setminus H$ замыкание множества $H \cup \{f\}$ совпадает с P_k.

Все 5 предполных классов в P_2 были найдены Э. Постом в [2, 3], все 18 предполных классов в P_3 были описаны С. В. Яблонским в [4]. Затем С. В. Яблонским и А. В. Кузнецовым было установлено, что для любого $k \geqslant 3$ число предполных в P_k классов конечно [5, § 10].

Пусть $k \geqslant 2$. Обозначим через q_k количество предполных классов в P_k. Пронумеруем эти предполные классы числами от 1 до q_k в некотором порядке. Каждой k-значной функции f можно поставить в соответствие строку $\tilde{\gamma}^f = (\gamma_1^f, \gamma_2^f, \dots, \gamma_{q_k}^f)$, состоящую из «плюсов» и «минусов», такую, что $\gamma_i^f =$ «$-$» тогда и только тогда, когда f не лежит в i-м предполном классе. Строку $\tilde{\gamma}^f$ назовем *строкой принадлежности* функции f.

Коротко о свойствах предполных классов в P_2

Пусть $k = 2$. Поскольку в P_2 ровно 5 предполных классов (это классы Поста T_0, T_1, L, S и M), строка принадлежности каждой двузначной (булевой) функции состоит из 5 элементов. Ограничения на комбинации знаков «+» и «−» в строках принадлежности булевых функций полностью описываются следующим утверждением (это прямое следствие результатов Э. Поста [2, 3]).

Теорема 1. *В P_2 справедливы следующие вложения:*

$$\begin{array}{ll} M \subseteq T_0 \cup T_1, & M \subseteq L \cup (T_0 \cap T_1), \\ S \cap T_1 \subseteq T_0, & S \cap T_0 \subseteq T_1, \\ L \cap T_0 \cap T_1 \subseteq S, & L \subseteq T_0 \cup T_1 \cup S, \end{array}$$

причем строка из 5 знаков «+» и «−» является строкой принадлежности некоторой булевой функции тогда и только тогда, когда эта строка удовлетворяет перечисленным выше свойствам.

Данный факт позволяет, например, легко доказать, что любой базис в P_2 содержит не более 4 функций.

Некоторые свойства предполных классов функций трехзначной логики

Пусть теперь $k = 3$. Обозначим 18 предполных классов трехзначной логики следующим образом:

S — класс функций, самодвойственных относительно перестановки (120);

M_0, M_1, M_2 — классы функций, монотонных относительно порядка $2 < 0 < 1$, $0 < 1 < 2$ и $1 < 2 < 0$ соответственно;

L — класс линейных функций (над полем $GF(3)$);

U_0, U_1, U_2 — классы функций, сохраняющих разбиение $\{\{0\}, \{1, 2\}\}$, $\{\{1\}, \{2, 0\}\}$ и $\{\{2\}, \{0, 1\}\}$, соответственно;

T_0, T_1, T_2 — классы функций, сохраняющих константу 0, 1 и 2, соответственно;

T_{12}, T_{02}, T_{01} — классы функций, сохраняющих множество $\{1, 2\}$, $\{0, 2\}$ и $\{0, 1\}$, соответственно;

C_0, C_1, C_2, где C_σ — класс функций, сохраняющих 2-местный предикат $((x = y) \vee \sigma \in \{x, y\})$, для любого $\sigma \in E_3$;

B — класс Слупецкого (все трехзначные функции, имеющие не более одной существенной переменной либо принимающие не более двух значений).

В P_3, как и в P_2, не любая комбинация знаков «+» и «−» в строках принадлежности возможна.

Теорема 2. *В P_3 справедливы следующие включения:*

$$\begin{aligned}
&M_1 \cap M_2 \cap C_1 \cap C_2 \subseteq M_0,\\
&L \cap (M_1 \cup M_2) \subseteq M_0,\\
&U_1 \cap U_2 \cap (M_1 \cup M_2 \cup L \cup U_0) \subseteq M_0,\\
&(M_1 \cup M_2) \cap (U_1 \cap C_1 \cup U_2 \cap C_2) \subseteq M_0,\\
&L \cap U_0 \cap (U_1 \cup U_2) \subseteq M_0,\\
&L \cap U_0 \cap (C_1 \cup C_2) \subseteq M_0,\\
&L \cap C_0 \cap (U_1 \cup U_2) \subseteq M_0,\\
&L \cap (U_1 \cap C_2 \cup U_2 \cap C_1) \subseteq M_0,\\
&L \cap (C_0 \cap C_1 \cup C_0 \cap C_2 \cup C_1 \cap C_2) \subseteq M_0,\\
&U_0 \cap C_0 \cap (U_1 \cap C_1 \cup U_2 \cap C_2) \subseteq M_0,\\
&C_1 \cap C_2 \cap (U_1 \cap U_2 \cup C_0) \subseteq M_0.
\end{aligned}$$

Теорема 3. *Пусть A — замкнутый класс функций трехзначной логики, содержащийся в классе M_0. Тогда A является пересечением некоторых предполных классов из множества $\{M_1,\ M_2,\ L,\ U_0,\ U_1,\ U_2,\ C_0,\ C_1,\ C_2\}$ тогда и только тогда, когда A совпадает с одним из 7 следующих классов функций из P_3:*

1) $M_0 \cap M_1 \cap U_2 \cap C_2$,
2) $M_0 \cap M_2 \cap U_1 \cap C_1$,
3) $M_0 \cap M_1 \cap U_0 \cap U_2 \cap C_1 \cap C_2$,
4) $M_0 \cap M_2 \cap U_0 \cap U_1 \cap C_1 \cap C_2$,
5) $M_0 \cap M_1 \cap U_1 \cap U_2 \cap C_0 \cap C_2$,
6) $M_0 \cap M_2 \cap U_1 \cap U_2 \cap C_0 \cap C_1$,
7) $M_0 \cap M_1 \cap M_2 \cap L \cap U_0 \cap U_1 \cap U_2 \cap C_0 \cap C_1 \cap C_2 \cap B$.

Теорема 4. *В P_3 справедливы следующие включения:*

$$
\begin{aligned}
&M_0 \cap (M_1 \cap C_1 \cup M_2 \cap C_2) \subseteq U_0,\\
&L \cap (M_0 \cup M_1 \cup M_2) \subseteq U_0,\\
&M_1 \cap M_2 \subseteq U_0,\\
&M_1 \cap U_1 \cap C_1 \cup M_2 \cap U_2 \cap C_2 \subseteq U_0,\\
&L \cap U_1 \cap U_2 \subseteq U_0,\\
&L \cap (U_1 \cap C_2 \cup U_2 \cap C_1) \subseteq U_0,\\
&L \cap C_0 \subseteq U_0,\\
&C_1 \cap C_2 \subseteq U_0.
\end{aligned}
$$

Теорема 5. *Пусть A — замкнутый класс функций трехзначной логики, содержащийся в классе U_0. Тогда A является пересечением некоторых предполных классов из множества $\{M_0, M_1, M_2, L, U_1, U_2, C_0, C_1, C_2\}$ тогда и только тогда, когда A совпадает с одним из 13 следующих классов функций из P_3 :*

1) $U_0 \cap C_1 \cap C_2$,
2) $M_0 \cap U_0 \cap C_1 \cap C_2$,
3) $M_1 \cap U_0 \cap C_1 \cap C_2$,
4) $M_2 \cap U_0 \cap C_1 \cap C_2$,
5) $U_0 \cap U_1 \cap C_1 \cap C_2$,
6) $U_0 \cap U_2 \cap C_1 \cap C_2$,
7) $M_0 \cap M_2 \cap U_0 \cap U_1 \cap C_1 \cap C_2$,
8) $M_0 \cap M_1 \cap U_0 \cap U_2 \cap C_1 \cap C_2$,
9) $M_1 \cap M_2 \cap U_0 \cap C_0$,
10) $M_1 \cap M_2 \cap U_0 \cap U_1 \cap C_0 \cap C_2$,
11) $M_1 \cap M_2 \cap U_0 \cap U_2 \cap C_0 \cap C_1$,
12) $L \cap U_0 \cap C_0 \cap B$,
13) $M_0 \cap M_1 \cap M_2 \cap L \cap U_0 \cap U_1 \cap U_2 \cap C_0 \cap C_1 \cap C_2 \cap B$.

Вложения в двойственные классы M_σ и U_σ $(\sigma \in \{1, 2\})$ аналогичны.

Доказав еще ряд свойств для предполных классов S, L, T_σ, T_{ij}, C_σ и B, автору удалось полностью построить решетку (по вложению) всех замкнутых классов в P_3, которые являются пересечениями некоторых предполных классов, а также получить в качестве следствия известный результат о том, что максимальный базис в P_3 содержит 6 функций.

Работа выполнена при поддержке РФФИ, проект № 09-01-00701.

Список литературы

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

[2] *Post E. L.* Introduction to a general theory of elementary propositions // Amer. J. Math. — 1921. — V. 43, № 4. — P. 163–185.

[3] *Post E. L.* Two-valued iterative systems of mathematical logic // Annals of Math. Studies. Princeton Univ. Press, 1941. — V. 5.

[4] *Яблонский С. В.* О функциональной полноте в трехзначном исчислении // ДАН СССР. — 1954. — Т. 95, № 6. — С. 1153–1156.

[5] *Яблонский С. В.* Функциональные построения в k-значной логике // Тр. МИАН СССР им. В. А. Стеклова. — 1958. — Т. 51. — С. 5–142.

О применении антиунификации подстановок для проверки эквивалентности программ

Т. А. Новикова, В. А. Захаров

taniaelf@mail.ru, zakh@cs.msu.su

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Одна из задач реорганизации программ (program refactoring) — это поиск подобных фрагментов программ и выделение процедуры, реализующей эти фрагменты (см. [1]). Эту задачу можно сформулировать так: для пары программ, реализующих функции $F_1(x)$ и $F_2(x)$, требуется построить такую программу, реализующую функцию $F_0(x)$, для которой равенства $F_1(x) = \psi_1(F_0(\varphi_1(x)))$ и $F_2(x) = \psi_2(F_0(\varphi_2(x)))$ выполняются для некоторых функций $\varphi_i, \psi_i,\ i = 1, 2$, вычислимых очень простыми программами.

Задача «обобщения» программ, родственная задаче выделения процедуры, возникала ранее при построении суперкомпиляторов [2], и для ее решения были предложены алгоритмы антиунификации (см., например, [2, 3]), вычисляющие наиболее специальные общие шаблоны выражений (термов, формул). Однако для решения обеих задач необходимо иметь эффективные средства сравнения функций, вычисляемых программами, что является трудной (вообще говоря, алгоритмически неразрешимой) проблемой. Для преодоления этой трудности предлагается следующий подход.

1. Выбрать такую формальную модель программ, в которой функциональной характеристикой каждой программы является регулярное множество выражений, допускающих применение к ним операции антиунификации.
2. Используя упомянутые операции, разработать алгоритм проверки эквивалентности программ в выбранной модели.
3. Модифицировать предложенный алгоритм проверки эквивалентности программ для вычисления наиболее общего примера (унификация) и наиболее специального шаблона (антиунификация) двух программ.

В настоящей заметке изложены результаты выполнения двух первых пунктов предложенного подхода.

Пусть X — некоторое множество переменных, σ — некоторая сигнатура языка логики предикатов первого порядка. Над сигнатурой σ обычным образом определяются множество термов $Term$ и атомарных формул $Atom$. Подстановкой называется всякое отображение $\theta : X \to Term$. Множество всех подстановок обозначим записью $Subst$. Обычным образом определяется операция $E\theta$ применения подстановки θ к выражению (терму или атому) E и операция композиции $\theta\eta$ подстановок θ и η так, что $E(\theta\eta) = (E\theta)\eta$. На множестве $Subst$ определяется отношение сравнения $\preceq$: $\theta \preceq \eta$ тогда и только тогда, когда для некоторой подстановки ρ выполняется равенство $\eta = \theta\rho$. В статье [4] показано, что $(Subst, \preceq)$ является полной квазирешеткой, обладающей свойством обрыва возрастающих цепей. Обозначим символом $\vee$ операцию антиунификации, вычисляющую точную нижнюю грань $\theta_1 \vee \theta_2$ подстановок θ_1 и θ_2. В [4] показано, что композиция подстановок дистрибутивна относительно операции антиунификации: $\theta(\eta_1 \vee \eta_2) = (\theta\eta_1) \vee (\theta\eta_2)$.

Моделями программ служат конечные размеченные системы переходов $\pi(X) = \langle V, entry, exit, B, T\rangle$, в которых V — непустое множество точек программы, $entry$ и $exit$ — точки входа и выхода, $B : V \to Atom$ — функция разметки, $T : (V \setminus \{exit\}) \times \{0, 1\} \to (V \setminus \{entry\}) \times Subst$ — 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** $\{\theta_1;$ **goto** $u_1\}$ **else** $\{\theta_0;$ **goto** $u_0\}$. Все стандартные схемы программ моделируются системами переходов указанного вида. Далее для обозначения атома $B(v)$, приписанного точке v, будем использовать запись A_v, а для изображения переходов из точки v будет использоваться запись $v \xrightarrow{\delta,\theta} u$ в случае $T(v,\delta) = (u,\theta)$. Не ограничивая общности, будем предполагать также, что через каждую точку программы проходит маршрут из точки $entry$ в точку $exit$.

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 в истории $lth(tr)$ состоит из примера атома $A_{v_i}\eta_i$, приписанного точке v_i этой трассы, 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 логико-термально (л.-т.) эквивалентными, если $\det(\pi_1(X)) = \det(\pi_2(X))$. В статье [5] предложен алгоритм полиномиальной сложности, проверяющий л.-т. эквивалентность последовательных программ.

Мы предлагаем гораздо более простой алгоритм проверки л.-т. эквивалентности, опирающийся лишь на операции с подстановками. Сначала в анализируемой паре программ $\pi_1(X)$ и $\pi_2(X)$, где $X = \{x_1, \ldots, x_n\}$, проводится переименование переменных для того, чтобы получить пару программ $\pi_1(X')$ и $\pi_2(X'')$ с непересекающимися, но соответствующими друг другу множествами переменных X' и X''. Далее строится граф совместных вычислений Γ_{π_1,π_2} следующего вида. Вершинами этого графа являются пары точек (v_1, v_2) анализируемых программ. Для любой пары точек v_1, v_2 в программах $\pi_1(X')$ и $\pi_2(X'')$ и для любого $\delta,\ \delta \in \{0,1\}$, если $T_1(v_1,\delta) = (\theta', u_1)$ и $T_2(v_2,\delta) = (\theta'', u_2)$, то в графе Γ_{π_1,π_2} из вершины (v_1, v_2) исходит дуга e в вершину (u_1, u_2), помеченная подстановкой $\theta_e = \theta' \cup \theta''$. Для каждого пути α, исходящего из вершины $(entry, entry)$ в графе Γ_{π_1,π_2} и проходящего по дугам $e_1, e_2, \ldots, e_n$, обозначим записью $\theta[\alpha]$ композицию подстановок $\theta_{e_n} \ldots \theta_{e_2}\theta_{e_1}\lambda_0$, где θ_{e_i} — подстановка, приписанная дуге e_i, а $\lambda_0 = \{x_1'/x_1, \ldots, x_n'/x_n, x_1''/x_1, \ldots, x_n''/x_n\}$ — переименование, отождествляющее соответствующие переменные программ $\pi_1(X')$ и $\pi_2(X'')$. Множество всех маршрутов в графе Γ_{π_1,π_2}, ведущих из вершины $(entry, entry)$ в вершину (v_1, v_2), обозначим записью $Pt(v_1, v_2)$.

Теорема 1. *Программы $\pi_1(X)$ и $\pi_2(X)$ не являются л.-т. эквивалентными тогда и только тогда, когда в графе совместных вычислений Γ_{π_1,π_2} существует вершина $w = (v_1, v_2)$, удовлетворяющая одному из следующих двух условий:*

A: $Pt(w) \neq \varnothing$ и при этом лишь одна из точек v_1, v_2 является точкой выхода $exit$;

B: существует такой маршрут α, $\alpha \in Pt(w)$, что $A_{v_1}\theta[\alpha] \neq A_{v_2}\theta[\alpha]$.

Для каждой вершины w в графе совместных вычислений Γ_{π_1,π_2} рассмотрим подстановку $\eta[w] = \bigvee\limits_{\alpha \in Pt(w)} \theta[\alpha]$, являющуюся точной нижней гранью множества подстановок, вычислимых на всех маршрутах, ведущих из вершины $(entry, entry)$ в вершину w.

Теорема 2. *Условие B теоремы 1 выполняется для вершины $w = (v_1, v_2)$ в том и только том случае, если выполняется соотношение $A_{v_1}\eta[w] \neq A_{v_2}\eta[w]$.*

Таким образом, проверка л.-т. эквивалентности программ сводится к задаче вычисления подстановок $\eta[w]$ в графе совместных вычислений Γ_{π_1,π_2}. Для решения этой задачи каждой вершине w графа сопоставим переменную Z_w, принимающую значения из множества $Subst$, а также выделим множество $In(w)$ пар вида (u, θ): пара (u, θ) входит в $In(w)$ тогда и только тогда, когда из вершины u в вершину w ведет дуга e, помеченная подстановкой θ_e. Далее для каждой вершины w, $w \neq (entry, entry)$, графа Γ_{π_1,π_2} сформируем уравнение $\mathcal{E}_w : \ Z_w = \bigvee\limits_{(u,\theta) \in In(w)} \theta Z_u$. Уравнение $\mathcal{E}_{(entry,entry)}$, соответствующее вершине $(entry, entry)$ графа Γ_{π_1,π_2}, имеет вид $Z_{(entry,entry)} = \lambda_0$, где λ_0 — упомянутое выше переименование переменных.

Теорема 3. *Система уравнений $\{\mathcal{E}_w : w$ — вершина графа $\Gamma_{\pi_1,\pi_2}\}$ всегда имеет решение, и множество подстановок $\{\eta[w] : w$ — вершина графа $\Gamma_{\pi_1,\pi_2}\}$ является ее единственным решением.*

Решение системы уравнений, указанной в теореме 3, можно осуществить итеративно, методом последовательных приближений, выбрав в качестве начального значения каждой переменной Z_w наибольший элемент τ квазирешетки $(Subst, \preceq)$. Закон дистрибутивности $\theta_0(\theta_1 \vee \theta_2) = (\theta_0\theta_1) \vee (\theta_0\theta_2)$ обеспечивает корректность, а свойство обрыва убывающих цепей гарантирует завершаемость итеративной процедуры уточнения последовательных приближений решений системы. Используя алгоритм антиунификации, предложенный в [3], при помощи такой процедуры можно получить решение рассматриваемой в теореме 3 системы уравнений за время, полиномиальное относительно размеров анализируемых программ $\pi_1(X)$ и $\pi_2(X)$.

Работа выполнена при поддержке РФФИ, проект № 09-01-00277.

Список литературы

[1] *Komondoor R., Horwitz S.* Semantics-preserving procedure extraction // Proc. 27-th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages. — 2000. — P. 155–169.

[2] *Sorensen M. H., Gluck R.* An algorithm of generalization in positive supercompilation // Proc. 1995 International Symposium on Logic Programming. — MIT Press, 1995. — P. 465–479.

[3] *Bulychev P. E., Kostylev E. V., Zakharov V. A.* Anti-unification algorithms and their applications in program analysis // LNCS. V. 5947. — 2009.

[4] *Eder E.* Properties of substitutions and unifications // J. Symbolic Computations. — V. 1. — 1985. — P. 31–46.

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

Распознавание подклассов марковских автоматов на основе последовательностей состояний конечной длины

А. Р. Нурутдинова, С. В. Шалагин

alsu124@mail.ru, Sshalagin@mail.ru

Казанский государственный технический университет им. А. Н. Туполева

Цепи Маркова (ЦМ) используются для моделирования реальных процессов, событий, явлений, имеющих вероятностную (стохастическую) природу [1, 2]. ЦМ можно рассматривать как процессы, получаемые на выходе вероятностных моделей автоматного типа [3, 4].

Необходимость в задачах распознавания этих объектов возникает при решении вопросов анализа эффективности функционирования марковских моделей сложных систем, статистического распознавания автоматов, передачи и защиты информации и т. п. В данной работе ставится и решается задача распознавания марковского автомата (МА), определенного на основе эргодических стохастических матриц класса циклических, не приведенных к нормальному виду [2], по реализуемым ими ЦМ. Признаки распознавания сформированы на базе характеристического признака циклической стохастической матрицы и позволяют определить вероятность принадлежности МА к заданному подклассу на основе последовательности определенной длины, генерируемой МА.

Определим основные понятия, на основе которых будет осуществляться исследование МА.

Определение 4. *Простая однородная цепь Маркова задана в виде* [1]

$$(S,\ P,\ \pi_0)\,. \tag{1}$$

Определение 5. *Автономным вероятностным автоматом будем называть систему* [3]

$$\left(\bar{U},\ S,\ P,\ \delta(u,\ s)\right), \tag{2}$$

где $\bar{U}$ *— дискретная случайная величина, принимающая на входе значения* $u_0,\ u_1, ...,\ u_{h-1}$ *с распределением* $\bar{p} = (p_0, p_1, ..., p_{h-1})$, h *— размер имплицирующего вектора,* $\sum\limits_{i=0}^{l-1} p_i = 1,\ \ 0 \leqslant p_i \leqslant 1$; $\delta(u,\ s)$ *— функция переходов, ставящая в соответствие паре* $(u,\ s)$ *однозначно новое состояние* $s' \in S$.

Модель устройства, заданного на базе (1) и (2), определена как АММ.

АММ реализуема в виде последовательной композиции [4] — генератора дискретной случайных величин $\bar{U}$=$\{\mathrm{u}_1, \mathrm{u}_2, \ldots, \mathrm{u}_l\}$ и конечного детерминированного автомата (КДА). КДА реализует функцию перехода $\delta(u, s)$ в соответствии с ЭСМ P, таким образом, позволяя моделировать ЦМ вида (1)

Рассмотрим класс АММ SC_r, определенных на базе ЦСМ$_r$. Пусть $\left(\overline{S}_1, \overline{S}_2, ..., \overline{S}_r\right)$ — некоторое разбиение множества S, определенного в

(1), на r непересекающихся подмножеств, такое, что ЦМ, находясь в какой-либо момент времени в состоянии подмножества $\overline{S}_k$, в следующий момент неизбежно переходит в группу $\overline{S}_{k+1}$ или, находясь в группе $\overline{S}_r$ переходит в группу$\overline{S}_1$. Обозначим n-размерность ЦСМ$_r$ через n_i, $n_i = \left|\overline{S}_i\right|$, $i = \overline{1, r}$, $\sum\limits_{i=1}^{r} n_i = n$.

Пусть для ЦМ, заданной согласно (1), справедливо следующее:

$$\forall t : t \bmod r = l, \left(\sum_{i:\mathrm{S}_i \in \overline{S}_l} \pi_i(t) = 1 \right) \& \left(\sum_{j:\mathrm{S}_j \in \overline{S}_{l-1}} \pi_i(t-1) = 1 \right), \quad (3)$$

где π_i — i-й элемент вектора распределения вероятности состояния ЦМ (1) в момент времени t.

ЭСМ P из (1), для которой справедливо условие (3), является ЦСМ$_r$ [2].

Рассматривая АММ, определенные на основе ЦСМ$_r$, как объекты исследования, решена задача распознавания МА, заданного на базе ЦСМ$_r$, не приведенного к нормальному циклическому виду [2], принадлежащего к подклассу SC$_r$. Для решения задачи требуется определить признаки для распознавания МА на основе реализаций ЦМ конечной длины N.

Обозначим r возможных групп вариантов последовательностей, генерируемых на основе ЦСМ$_r$, $\chi_k = \left\{ s^k_{i_1}, s^k_{i_2}, \ldots, s^k_{i_r} \right\}$, $s^k_{i_l} \in \overline{S}_{l+k}$, $l = \overline{1,\ r-k}$, $s^k_{i_{r-k+1}} \in \overline{S}_j$, $j = \overline{1,\ k}$, $k = \overline{0,\ r-1}$.

Характеристический признак ЦСМ$_r$, определяемый индексом $r-$ c_r, $r > 1$, позволяет определить признаки, представленных в виде функционалов на ЦМ — $\tilde{c}_r$.

Если ЦСМ$_r$ имеет индекс r, $r > 1$, то признак c_r, позволяющий распознать АММ вида (2), заданную ЦСМ$_r$, определяется выражением:

$$c_r = \begin{cases} 1, \text{ если } \bigvee\limits_{i=0}^{r-1} \left(\mathop{\&}\limits_{j=0}^{m-1} s_{t_{i+j+1}} \in \overline{S}_{(i+j) \bmod r+1} \right) \\ 0, \text{ иначе} \end{cases}, \quad m = \overline{1,\ r}. \quad (4)$$

Признаки c_r, определенные согласно (4), являются теоретическими оценками величин $\tilde{c}_r$ вычисление которых производится путем подсчета циклически повторяющихся подмножеств состояний ЦМ определенной длины N:

$$\tilde{c}_r = \begin{cases} \alpha, \text{ если } \underset{i=0}{\overset{r-1}{\vee}} \left(\underset{j=0}{\overset{m-1}{\&}} \left(s_{t_{i+j+1}} \in \overline{S}_{(i+j) \bmod r+1} \right) \right) \\ 0, \text{иначе} \end{cases}, \quad m = \overline{1,\ r}.$$

где α — вероятность отнесения ЦМ к заданному подклассу SC_r. Пусть величина D, $D>0$, на основе которой заданы элементы ЦСМ$_r$ $p_{ij} = D \cdot z$, где z — целое положительное число, есть дискретность положительных элементов ЭСМ.

Тогда минимальная вероятность того, что в момент времени t ЦМ, определяемая ЦСМ$_r$, находясь в состоянии $s \in \overline{S_i}$, в следующий момент времени $(t+1)$ перейдет в состоянии $s \in \overline{S_p}$,

$$p = \begin{cases} 1 : i = r \\ i+1 : i < r \end{cases},$$

равна $\frac{D}{n}$.

Максимальная вероятность того, что ЦМ единичной длины будет ошибочно отнесена к классу SC_r, равна $\overline{p}^{(r)}_{\max} = 1 - \frac{D}{n} = \frac{n-1}{n}D$.

Для ЦМ длины N максимальная вероятность ее ошибочного отнесения к классу SC_r равна $\overline{p}^{(r)}_{\max} = \left(\frac{n-1}{n}D\right)^N$.

На основе вышеизложенного справедливо

Утверждение 1. *Минимальная вероятность безошибочного распознавания ЦМ вида (1), $|\,S\,| = n$ и длины N, определяемой на основе ЦСМ$_r$, положительные элементы которой определены с дискретностью D, равна:*

$$\alpha = 1 - \overline{p}^{(r)}_{\max} = 1 - \left(\frac{n-1}{n}D\right)^N.$$

Обозначим $\mathrm{A}(\chi_k)$ — КДА, распознающий последовательность χ_k, либо ее первые m элементов, $k = \overline{0, (r-1)}$. Выход $\mathrm{A}(\chi_k)$ определен значением булевой переменной

$$\underset{j=0}{\overset{m-1}{\&}} \left(s_{t_{i+j+1}} \in \overline{S}_{(i+j) \bmod r+1} \right).$$

Задача оценки сложность распознавания принадлежности АММ вида (1) к подклассу SC_r сводится к определению количества автоматов типа $\mathrm{A}(\chi_k)$, распознающих всевозможные варианты последовательностей χ_k, $k = \overline{0, (r-1)}$. Оценку временной сложности работы автомата $\mathrm{A}(\chi_k)$ обозначим T, а оценку его емкостной сложности — Q. Примем величины T и Q в качестве мер временной и емкостной сложности, соответственно

Количество способов разбиения множества из n состояний на r непустых подмножеств определяем по формуле числа Стирлинга второго рода [5]:

$$S(n,r) = \frac{1}{r!}\sum_{j=0}^{r}(-1)^{r+j}\cdot S\left(r,j\right)\cdot j^{n},$$

$$S(j+1,\ 0) = 0, \quad S(j,\ j) = 1, \quad j = \overline{0,\ r}.$$

На основе вышеизложенного сформулировано

Утверждение 2. *Оценки временной и емкостной сложности распознавания принадлежности АММ вида (1), заданной на основе ЦСМr, не приведенной к нормальному циклическому виду, к подклассу SC_r, равны, соответственно: T и $Q = rS(n,k)$.*

Список литературы

[1] *Кемени Дж., Снелл Дж.* Конечные цепи Маркова. — М.: Наука, 1970.

[2] *Романовский В. И.* Дискретные цепи Маркова. — М.: Гостехиздат, 1949.

[3] *Бухараев Р. Г.* Основы теории вероятностных автоматов. — М.: Наука, 1985.

[4] *Поспелов Д. А.* Вероятностные автоматы. — М.: Энергия, 1970.

[5] *Андерсон Джеймс А.* Дискретная математика и комбинаторика: пер. с англ. — М.: Вильямс, 2003.

О некоторых свойствах одноместных монотонных функций многозначной логики

Д. Ю. Панин

pank.dm@gmail.com

Московский государственный университет им. М. В. Ломоносова,
Механико-математический факультет

Рассматривается некоторое множество одноместных функций многозначной логики, монотонных относительно частичного порядка специального вида, и замыкание этого множества относительно операции суперпозиции [1, 2]. Получен критерий полноты для рассматриваемой функциональной системы (см. также [3]). Подобные вопросы возникают при изучении свойств предполных классов монотонных функций, не имеющих конечных порождающих систем [4–7], а также при решении задачи о полноте систем функций многозначной логики (см., например, [8–10]).

Пусть $n \geqslant 1$. Положим $Q_k = \{0, a_1, a_1', \ldots, a_k, a_k'\}$, где $0 \leqslant k \leqslant n$, $Q = Q_n$, $a_0 = a_0' = 0$. Введем на элементах множества Q отношение частичного порядка $\leqslant$ следующим образом:

1) $\varepsilon_i \leqslant \varepsilon_j$ для всех $\varepsilon_i, \varepsilon_j$, таких, что $\varepsilon_i \in \{a_i, a_i'\}$, $\varepsilon_j \in \{a_j, a_j'\}$, $0 \leqslant i < j \leqslant n$;
2) $\varepsilon \leqslant \varepsilon$ для всех $\varepsilon \in Q$.

Пусть $\alpha, \beta \in Q$. Если для этих элементов выполняется по крайней мере одно из соотношений $\alpha \leqslant \beta$, $\beta \leqslant \alpha$, то эти элементы называются *сравнимыми*, в противном случае — *несравнимыми*.

Будем обозначать через F множество всех функций $f(x)$, определенных на множестве Q, принимающих значения из Q, монотонных относительно частичного порядка $\leqslant$ и таких, что $f(\delta) \leqslant \delta$ для всех $\delta \in Q$. Введем на множестве F операцию композиции. Пусть $f(x)$, $g(x) \in F$. *Композицией функций f и g* будем называть функцию $(f \circ g)(x)$, значение которой на любом элементе $\alpha \in Q$ определяется равенством $(f \circ g)(\alpha) = f(g(\alpha))$.

Пусть $\mathfrak{A} \subseteq F$. Определим по индукции понятие *формулы над* $\mathfrak{A}$, а также понятие *функции, реализуемой формулой*.

1. Выражение вида $f(x)$, где $f(x) \in \mathfrak{A}$, является формулой над $\mathfrak{A}$. Такая формула называется *тривиальной* и реализует функцию $f(x)$.
2. Пусть Φ_1 — формула над $\mathfrak{A}$, реализующая функцию $f_1(x)$, а Φ_2 — формула над $\mathfrak{A}$, реализующая функцию $f_2(x)$. Тогда выражение $\Phi_1 \circ \Phi_2$ является формулой над $\mathfrak{A}$ и реализует функцию $(f_1 \circ f_2)(x)$.
3. Для удобства будем считать, что пустая формула реализует функцию $e(x)$, такую, что $e(\alpha) = \alpha$ для всех $\alpha \in Q$.

При этом предполагается, что других формул над $\mathfrak{A}$ нет.

Замыканием множества $\mathfrak{A} \subseteq F$ будем называть множество всех функций, которые могут быть реализованы формулами над $\mathfrak{A}$ (обозначение $[\mathfrak{A}]$). Очевидно, что $[\mathfrak{A}] \subseteq F$. Систему $\mathfrak{A}$ будем называть *полной*, если $[\mathfrak{A}] = F$.

Цепью длины m, $1 \leqslant m \leqslant n+1$, будем называть последовательность элементов $b_0, b_1, b_2, \ldots, b_{m-1} \in Q$, таких, что $b_0 = 0$, $b_i \in \{a_i, a_i'\}$ для всех $i = 1, \ldots, m-1$.

Пусть $\Omega \subseteq Q$. Положим

$$S_\Omega = \{f \in F \mid f(\delta) = \delta \text{ для всех } \delta \in \Omega\}.$$

В частности, $S_{\{a_n\}}$ — множество всех функций из F, сохраняющих элемент a_n.

Будем обозначать через G_1 множество всех функций g из F, для которых найдутся номер k, $2 \leqslant k \leqslant n$, и цепь Ω длины $k-1$, такие, что выполнены следующие условия:

1) $g \in S_\Omega$,
2) $\{g(a_k'), g(a_k)\} = \{a_{k-1}, a_{k-1}'\}$.

Будем обозначать через G_2 множество всех функций g из F, для которых найдутся номер k, $1 \leqslant k \leqslant n$, элементы $\varepsilon, \varepsilon' \in Q$ и цепь Ω длины $k-1$, такие, что выполнены следующие условия:

1) $\{\varepsilon, \varepsilon'\} = \{a_k, a_k'\}$,
2) $g \in S_\Omega$,
3) $g(\alpha) = \alpha$ для всех $\alpha \in Q$, таких, что $\varepsilon \leqslant \alpha$,
4) $g(\varepsilon') \neq \varepsilon'$.

Будем обозначать через $H(m, d)$, $0 \leqslant d < m \leqslant n$, множество всех функций из F, для которых выполнены следующие условия:

1) $\{f(a_m), f(a'_m)\} = \{a_{m-d}, a'_{m-d}\}$;
2) элементы $f(a_j)$ и $f(a'_j)$ сравнимы для всех $j = 1, \ldots, m-1$.

Имеет место следующий критерий полноты для систем функций из F.

Теорема. *Система $G \subseteq F$ является полной тогда и только тогда, когда $G_1 \subseteq G$ и $G_2 \subseteq G$.*

Доказательство теоремы опирается на следующие леммы.

Лемма 1. *Пусть $f_1, f_2 \in F$, $g \in G_1 \cup G_2$ и $g = f_2 \circ f_1$; тогда $f_2 = g$ или $f_1 = g$.*

Лемма 2. *Пусть $h \in H(m, d)$, где $2 \leqslant d < m \leqslant n$. Тогда существуют функции $h_0 \in H(m, 0)$, $g \in H(m, 1)$ и $f \in H(m-1, d-1)$, такие, что $h = f \circ (g \circ h_0)$.*

Работа выполнена при финансовой поддержке РФФИ (проект №11-01-00508) и программы фундаментальных исследований Отделения математических наук РАН «Алгебраические и комбинаторные методы математической кибернетики и информационные системы нового поколения» (проект «Задачи оптимального синтеза управляющих систем»).

Список литературы

[1] *Яблонский С. В.* Введение в дискретную математику. — М.: Высшая школа, 2008.

[2] *Яблонский С. В.* Функциональные построения в k-значной логике // Труды Математического института АН СССР. Т. 51. — 1958. — С. 5–142.

[3] *Панин Д. Ю.* О порождении одноместных монотонных функций многозначной логики // Вестник Московского университета. Математика. Механика. — 2010. — № 6. — С. 52–55.

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

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

[6] *Дудакова О. С.* О конечной порожденности предполных классов монотонных функций многозначной логики // Математические вопросы кибернетики. Вып. 17 — М.: Физматлит, 2008. — С. 13–104.

[7] *Дудакова О. С.* О классах функций k-значной логики, монотонных относительно множеств ширины два // Вестник Московского университета. Математика. Механика. — 2008. — № 1. — С. 31–37.

[8] *Slupecki J.* Kriterium pelnosci wielwartosciowych systemow logiki zdaṙi // C.R. Séanc. Soc. Sci. Varsovie, Cl. III. — 1939. — V. 32. — P. 102–109.

[9] *Salomaa A.* Some completeness criteria for sets of functions over a finite domain I, II // Turun Ylopiston Jalkaisuja Annales Universitatis Turkuensis, sarja A 53. — 1962. — P. 1–9; A 63. — 1963. — P. 1–19. (Русский перевод: *Саломаа А.* Некоторые критерии полноты для множеств функций многозначной логики // Кибернетический сборник. Вып. 8. — М.: Мир, — 1964. — С. 7–32.)

[10] *Мальцев А. И.* Об одном усилении теорем Слупецкого и Яблонского // Алгебра и логика. — 1967. — Т. 6, № 3. — С. 61–74.

Об одной последовательности мультиклонов

В. И. Пантелеев

v_panteleyev@mail.ru

Восточно-Сибирская государственная академия образования, Иркутск

Пусть A — конечное множество, содержащее k элементов; n-местной мультифункцией на множестве A называется отображение $f : A^n \to 2^A$, а n-местной мультипроекцией — отображение $e_n^i : (x_1, \ldots, x_n) \to \{x_i\}$, $1 \leqslant i \leqslant n$.

Суперпозиция $f(f_1, \ldots, f_m)$ с внешней мультифункцией f и внутренними мультифункциями $f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n)$ определяет мультифункцию $h_1(x_1, \ldots, x_n)$ следующим образом:

$$h_1(a_1, \ldots, a_n) = \begin{cases} \varnothing, & \text{если существует } i \in \{1, \ldots, m\}, \text{ такое,} \\ & \text{что } f_i(a_1, \ldots, a_n) = \varnothing; \\ \bigcup\limits_{b_i \in f_i(a_1, \ldots, a_n)} f(b_1, \ldots, b_m) & \text{иначе} \end{cases}.$$

Замечание 1. Определение суперпозиции мультифункций фактически определяет вложение множества всех мультифункций на множестве A в множество функций 2^k-значной логики, поэтому n-местную мультифункцию можно рассматривать как n-местную функцию 2^k-значной логики, однозначно определяемую своими значениями на наборах, построенных из одноэлементных подмножеств.

С учетом замечания 1, во-первых, отождествляется одноэлементное подмножество с элементом этого подмножества, а во-вторых, понятие сохранения предиката функцией k-значной логики естественным образом распространяется на мультифункции (предикаты при этом задаются на множестве всех подмножеств множества A).

Замечание 2. Множество мультифункций, сохраняющих некоторый предикат, не обязательно является замкнутым относительно суперпозиции.

Мультиклоном (частичным гиперклоном) называется множество мультифункций, содержащее все мультипроекции и замкнутое относительно суперпозиции. Замыкание множества мультифункций и максимальные мультиклоны определяются обычным образом.

Определим m-местный предикат G_m $(m \geqslant 1)$ следующим образом: G_m состоит из всех наборов $(\alpha_1, \ldots, \alpha_m)$ длины m, таких, что либо для некоторого $i \in \{1, \ldots, m\}$ выполняется $\alpha_i = \varnothing$, либо для всех $i \in \{1, \ldots, m\}$ выполняется $\alpha_i \in A$ и при этом $\alpha_i \neq \alpha_j$, если $i \neq j$.

Пусть F_m — множество мультифункций, сохраняющих предикат G_m.

Теорема. *Справедливы следующие утверждения:*
1) F_m — *мультиклон;*
2) $F_m \subseteq F_n$ *при* $m \leqslant n$*;*
3) F_k — *максимальный мультиклон.*

Доказательство. Приведем основные идеи доказательства. Справедливость первого утверждения следует из равенства:

$$f(f_1, \ldots, f_m)(\alpha_1, \ldots, \alpha_n) = f(f_1(\alpha_1, \ldots, \alpha_n), \ldots, f_m(\alpha_1, \ldots, \alpha_n)),$$

справедливого для любого набора $(\alpha_1, \ldots, \alpha_n)$ из A^n, и из того, что любой набор, содержащий $\varnothing$, принадлежит предикату.

Справедливость второго утверждения получается несложным образом из определения предиката.

Для доказательства того, что F_k — максимальный мультиклон, заметим, что этому множеству принадлежат, во-первых, одноместные мультифункции, принимающие в качестве своих значений все элементы множества A, во-вторых, одноместные мультифункции, принимающие при одном значении переменной $\varnothing$ и равные некоторому непустому подмножеству при всех остальных значениях переменной.

Пусть теперь некоторая функция f не сохраняет G_k, т. е. на некоторых наборах из предиката она возвращает набор $(A_1, \ldots, A_k)^t$, не принадлежащий предикату. Тогда найдется элемент d из A, такой, что d не принадлежит ни одному из тех A_i $(1 \leqslant i \leqslant m)$, мощность которых равна 1. Используя одноместную функцию $h(x)$, которая равна $\varnothing$ при $x = d$ и 0 при остальных значениях переменной, а также одноместную функцию $g(x) = (A_1, \ldots, A_k)$, можно получить константу 0.

Подставив в трехместную функцию $u(x, y, z)$, которая равна $max(y, z) + 1$ при $x = 0$ и $\varnothing$ при остальных значениях переменной, константу 0, получим базис мультифункций, принимающих в качестве своих значений только элементы множества A.

Затем можно получить все мультифункции, принимающие в качестве своих значений пустое множество или элементы множества A, и после этого — все мультифункции.

Заметим, что при $m > k$ множество F_m совпадает с множеством всех мультифункций. ∎

Максимальными мультиклонами являются также множества мультифункций T_E [1]:

$$T_E = \{f \mid f(a_1, \ldots, a_n) = B,\ a_i \in E,\ B \subseteq E\},$$

где E — непустое подмножество множества A, и $T^*_\alpha (\alpha \in A)$ [2]:

$$T^*_\alpha = \{f(x_1, \ldots, x_n) \mid \alpha \in f(\alpha, \ldots, \alpha) \text{ или } f(x_1, \ldots, x_n) \equiv \varnothing\}.$$

Список литературы

[1] *Пантелеев В. И.* О некоторых максимальных частичных гипер- и ультраклонах // Материалы XVIII Международной школы-семинара «Синтез и сложность управляющих систем» имени академика О. Б. Лупанова (Пенза, 28 сентября – 3 октября 2009 г.) / Под ред. О. М. Касим-Заде. — М.: Изд-во механико-математического факультета МГУ, 2009. — С. 73–75.

[2] *Пантелеев В. И.* Об одном семействе максимальных частичных гиперклонов // Материалы 3-й Российской школы-семинара, посвященной 80-летию со дня рождения профессора А. И. Кокорина (Иркутск, 10–14 августа 2010 г.). — Иркутск: Изд-во ГОУ ВПО «Восточно-Сибирская государственная академия образования», 2010. — С. 74–75.

Применение дополнений паросочетаниями для задачи коммивояжера

А. В. Панюков, Т. А. Панюкова

a_panyukov@mail.ru, qwark@mail.ru

Южно-Уральский государственный университет, Челябинск

Введение

В статье изложен подход к приближенному решению задачи коммивояжера, основанный на дополнении частичного тура паросочетаниями подграфа открытых вершин. Проведено аналитическое исследование, показавшее, что алгоритм, основанный на непосредственном применении данного подхода, во-первых, имеет вычислительную сложность не более $O(n^3)$, n — число городов, во-вторых, не улучшает гарантированные оценки точности известных алгоритмов. Предложена модификация алгоритма Сердюкова, имеющая оценку вычислительной сложности $O(n^3)$ и наилучшую гарантированную оценку точности. Представлены результаты вычислительного эксперимента, позволяющие выдвинуть гипотезу об асимптотической точности алгоритма для достаточно широкого класса задач.

Пусть $G = (V, E)$ — полный неориентированный граф с множеством вершин $V : |V| = n$ и множеством ребер E. Пусть для каждого ребра $e \in E$ задан его вес $c(e)$. Для множества ребер $E' \subseteq E$ будем обозначать $c(E') = \sum_{e \in E'} c(e)$. Гамильтонов цикл — это цикл, проходящий через каждую вершину графа ровно один раз. Задача коммивояжера на максимум (MAX TSP) заключается в нахождении в графе G гамильтонова цикла максимального веса.

В данной работе для решения задачи MAX TSP представлены анонсированные ранее алгоритм дополнения паросочетаниями и его применение в модификации алгоритма Сердюкова [1, 2]. Данные алгоритмы предварительно строят специальные частичные туры, а затем дополняют их до гамильтонова цикла паросочетаниями подграфов открытых вершин. Гарантированные оценки точности предложенного алгоритма: 2/3 — для симметрической TSP и 7/9 — для метрической. Оценка точности модификации алгоритма Сердюкова — 3/4 для симметрической TSP и 5/6 — для метрической. Вычислительная сложность всех алгоритмов не превосходит величины $O(n^3)$.

Алгоритм дополнения паросочетаниями

В основе разработанного алгоритма лежит идея дополнения частичного тура до гамильтонова цикла ребрами паросочетаний подграфа на открытых вершинах.

Поясним эту идею. Первым шагом алгоритма является построение 2-фактора максимального веса. После этого производится разрыв полученных циклов удалением ребра минимального веса. Далее выполняется процедура дополнения полученного частичного тура паросочетанием максимального веса на подграфе графа G индуцированным множеством открытых вершин этого частичного тура. В результате будет получен 2-фактор с числом компонент связности, по крайней мере в 2 раза меньшим, чем у первоначального 2-фактора. Алгоритм выполняет указанную процедуру до тех пор, пока очередной 2-фактор не будет иметь одну компоненту связности. В этом случае полученный цикл будет гамильтоновым.

Алгоритм А1 (Алгоритм дополнения паросочетаниями).

Вход: полный неориентированный граф $G = (V, E)$ с весовой функцией c.

Выход: гамильтонов цикл.

Шаг 1. Вычислить 2-фактор максимального веса $C = C_1, \dots, C_s$.

Шаг 2. Если $s = 1$, *вернуть* C_1.

Шаг 3. Для каждого C_i, $i = 1, \dots, s$, найти $e_i = \{u_i, v_i\} = \arg\min_{e \in C_i} c(e)$. Положить $C_i = C_i \setminus \{e_i\}$.

Шаг 4. Пусть множество $S = \bigcup_{i=1,\dots,s} \{u_i, v_i\}$. Пусть H — подграф графа G, образованный вершинами множества S. Положить $E(H) = E(H) \setminus \left(\bigcup_{i=1,\dots,s} e_i\right)$.

Шаг 5. Найти в графе H паросочетание максимального веса M.

Шаг 6. Пусть $C = C \cup M$. Обозначить непересекающиеся циклы, образованные ребрами множества C, через $C_1, \ldots, C_s$. Перейти на *Шаг* 2.

Вычислительная сложность алгоритма

В наихудшем случае 2-фактор будет состоять из наибольшего возможного числа циклов длины 3, и после разрыва получится $\lfloor 2 \cdot n/3 \rfloor$ вершины, которым инцидентно только одно ребро частичного тура (так называемые открытые вершины). В этом случае количество компонент связности частичного тура равно $\lfloor n/3 \rfloor$. В наихудшем случае в результате одной процедуры построения дополнительного графа количество открытых вершин будет уменьшено лишь вдвое. Так как алгоритм заканчивает работу, когда не останется открытых вершин, то количество выполнения шагов 3–5 алгоритма будет не более $\lceil \log_2 \lfloor 2 \cdot n/3 \rfloor \rceil$. Поскольку при каждом последующем выполнении шагов 3–5 размерность задачи понижается как минимум в два раза, то время выполнения этих шагов не будет превосходить величины

$$\sum_{k=1}^{\lceil \log_2 \lfloor 2 \cdot n/3 \rfloor \rceil} O\left(\frac{n}{2^k}\right)^3 = O(n^3).$$

Таким образом, алгоритм **A1** является полиномиальным и при эффективной реализации подзадач нахождения максимальных паросочетаний и 2-фактора имеет вычислительную сложность $O(n^3)$.

Оценка точности алгоритма

Теорема 1. *Пусть opt — вес оптимального гамильтонова цикла в неориентированном взвешенном графе G при решении задачи на максимум, а T — цикл, найденный с помощью алгоритма* **A1** *дополнения подграфами. Тогда выполняется следующее неравенство: $c(T) \geqslant 2/3 \cdot opt$. Данная оценка неулучшаема.*

Теорема 2. *Пусть opt — вес оптимального гамильтонова цикла в метрическом неориентированном взвешенном графе G при решении задачи на максимум, пусть T — цикл, найденный с помощью алгоритма* **A1**. *Тогда выполняется следующее неравенство:*

$$c(T) > \frac{7}{9} \cdot opt.$$

Данный алгоритм может быть эффективно использован вместе со многими другими алгоритмами для дополнения частичного тура до гамильтонова, в частности, в модификации алгоритма Сердюкова.

Заключение

Предложенный в статье алгоритм приближенного решения задачи коммивояжера на максимум, основанный на дополнении частичного тура паросочетаниями подграфа открытых вершин, имеет вычислительную сложность $O(n^3)$ и оценки точности $2/3$ для симметрической MAX TSP и $7/9$ — для метрической. Несмотря на то, что известные гарантированные оценки не улучшены, вычислительный эксперимент показал, что получаемые данным алгоритмом решения являются одними из самых точных среди приближенных алгоритмов. В связи с этим достаточно эффективной оказалась связка разработанного алгоритма с алгоритмом, имеющим наиболее высокую из известных гарантированную оценку точности, — алгоритмом Сердюкова. Предложенная в работе модификация алгоритма Сердюкова по результатам вычислительного эксперимента показала наилучшие результаты среди известных полиномиальных алгоритмов для MAX TSP. Полученные результаты вычислительного эксперимента позволяют выдвинуть гипотезу об асимптотической точности алгоритма для достаточно широкого класса задач.

Список литературы

[1] *Гимади Э. Х., Сердюков А. И.* О некоторых результатах для задачи коммивояжера на максимум // Дискретный анализ и исследование операций. — 2001. — № 1. — С. 22–39.

[2] *Гимади Э. Х., Тычинин С. А.* Исследование реализаций алгоритма Сердюкова для задачи MAX TSP // Российская конференция «Дискретная оптимизация и исследование операций»: Материалы конф. (Владивосток, 7–14 сентября 2007). — Новосибирск: Изд-во Ин-та математики, 2007. — С. 132.

Теория Галуа для клонов и суперклонов

Н. А. Перязев

nikolai.baikal@gmail.com

Восточно-Сибирская государственная академия образования, Иркутск

Исследования замкнутых классов функциональных систем в нашей стране начал С. В. Яблонский [1]. Алгебраический подход к изучению функциональных систем впервые был предложен А. И. Мальцевым [2].

Среди алгебр функций наибольшее распространение получили клоны — алгебры операций, замкнутые относительно суперпозиции и содержащие операции проекций. В работах [3] устанавливается связь Галуа для клонов и ко-клонов — алгебр отношений.

Рассматривая алгебры над множествами мультиопераций с одной дополнительной операцией разрешимости простейшего уравнения, получаем алгебру, называемую суперклоном. Так как решетки суперклонов и ко-клонов изоморфны [4], то должна существовать связь Галуа для клонов и суперклонов. В данной работе эта связь устанавливается.

Предварительные результаты

В дальнейшем будем предполагать, что A — фиксированное конечное множество. Отображение из A^n в A называется n-местной операцией на A (будем допускать случай $n = 0$). Множество всех n-местных операций на A обозначаем через P_A^n, а множество всех операций на A через

$$P_A = \bigcup_{n \geqslant 0} P_A^n.$$

Пусть $F \subseteq P_A$. Алгебра $\mathfrak{F} = \langle F; *, \zeta, \tau, \Delta, \varepsilon \rangle$ типа $\langle 2, 1, 1, 1, 0 \rangle$ с ниже определенными операциями подстановки $(f * g)$, циклической перестановки аргументов (ζf), транспозиции аргументов (τf), отождествления аргументов (Δf) и операцией ε, выделяющей операцию $e_1^2 \in P_A^2$, называется *клоном* над A.

Если $f \in F \cap P_A^n$ и $g \in F \cap P_A^m$, то

$$(f * g)(a_1, ..., a_{n+m-1}) = f(g(a_1, ..., a_m), a_{m+1}, ..., a_{n+m-1}) \text{ при } n \geqslant 1;$$

$$(f * g)(a_1, ..., a_m) = f \text{ при } n = 0;$$

$$(\zeta f)(a_1, ..., a_n) = f(a_2, ..., a_n, a_1) \text{ при } n > 1 \text{ и } (\zeta f) = f \text{ при } n \leqslant 1;$$

$$(\tau f)(a_1, ..., a_n) = f(a_2, a_1, a_3, ..., a_n) \text{ при } n > 1 \text{ и } (\tau f) = f \text{ при } n \leqslant 1;$$

$$(\Delta f)(a_1, ..., a_{n-1}) = f(a_1, a_1, a_2, ..., a_{n-1}) \text{ при } n > 1;$$

$$(\Delta f) = b, \text{ если } f(a) = b \text{ для всех } a \in A, \text{ иначе } (\Delta f) = f \text{ при } n = 1;$$

$$(\Delta f) = f \text{ при } n = 0;$$

$$\varepsilon = e_1^2, \text{ где } e_1^2(a_1, a_2) = a_1.$$

Пусть $B(A)$ — множество всех подмножеств A, в том числе $\varnothing$. Отображение из A^n в $B(A)$ называется n-местной мультиоперацией на A (будем допускать случай $n = 0$). Для множества всех n-местных мультиопераций на A используем обозначение M_A^n, а для множества всех мультиопераций на A обозначение

$$M_A = \bigcup_{n \geqslant 0} M_A^n.$$

Пусть $S \subseteq M_A$. Алгебра $\mathscr{S} = \langle S; *, \zeta, \tau, \Delta, \mu, \varepsilon \rangle$ типа $\langle 2, 1, 1, 1, 1, 0 \rangle$ с ниже определенными операциями подстановки $(f * g)$, циклической перестановки аргументов (ζf), транспозиции аргументов (τf), отождествления аргументов (Δf), разрешимости (μf) и операцией ε, выделяющей бинарную мультиоперацию — проекцию по первому аргументу, называется *суперклоном* над A.

Если $f \in S \cap M_A^n$ и $g \in S \cap M_A^m$, то

$$(f * g)(a_1, \ldots, a_{n+m-1}) = \{a \mid \text{существует } a_0 \in g(a_1, \ldots, a_m) \\ \text{такой, что } a \in f(a_0, a_{m+1}, \ldots, a_{m+n-1})\} \text{ при } n \geqslant 1;$$

$$(f * g)(a_1, \ldots, a_m) = f(), \text{ если } g(a_1, \ldots, a_m) \neq \varnothing \text{ и}$$

$$(f * g)(a_1, \ldots, a_m) = \varnothing, \text{ если } g(a_1, \ldots, a_m) = \varnothing \text{ при } n = 0;$$

$$(\zeta f)(a_1, \ldots, a_n) = f(a_2, \ldots, a_n, a_1) \text{ при } n > 1, (\zeta f) = f \text{ при } n \leqslant 1;$$

$$(\tau f)(a_1, \ldots, a_n) = f(a_2, a_1, a_3, \ldots, a_n) \text{ при } n > 1, (\tau f) = f \text{ при } n \leqslant 1;$$

$$(\Delta f)(a_1, \ldots, a_{n-1}) = f(a_1, a_1, a_2, \ldots, a_{n-1}) \text{ при } n > 1;$$

$$(\Delta f) = \{a | a \in f(a)\} \text{ при } n = 1, (\Delta f) = f \text{ при } n = 0;$$

$$(\mu f)(a_1, \ldots, a_n) = \{a | a_1 \in f(a, a_2, \ldots, a_n)\} \text{ при } n \geqslant 1;$$

$$(\mu f) = \varnothing \text{ при } n = 0;$$

$$\varepsilon = e, \text{ где } e(a_1, a_2) = \{a_1\}.$$

Мультиоперации на всех наборах, равные одноэлементным множествам, ничем не отличаются от операций. Поэтому такие мультиоперации будем отождествлять с соответствующими им операциями.

Используя главные операции суперклона, можно определить частичную операцию суперпозиции

$$(f * (f_1, \ldots, f_n))(a_1, \ldots, a_m) = \bigcup_{b_i \in f_i(a_1, \ldots, a_m)} f(b_1, \ldots, b_n).$$

При этом если $f, f_1, \ldots, f_n$ являются операциями, то получается общепринятое понятие суперпозиции операций.

Лемма. *Для суперпозиции мультиопераций выполняется полутождество суперассоциативности*

$$(g_0 * (g_1, ..., g_s)) * (h_1, ..., h_m) \subseteq g_0 * (g_1 * (h_1, ..., h_m), ..., g_s * (h_1, ..., h_m)).$$

Если $h_1, \ldots, h_m$ — операции, то выполняется тождество суперассоциативности

$$(g_0 * (g_1, ..., g_s)) * (h_1, ..., h_m) = g_0 * (g_1 * (h_1, ..., h_m), ..., g_s * (h_1, ..., h_m)).$$

Эта лемма используется в доказательстве основных результатов.

Основные результаты

Если для любых элементов $a_1^1, \ldots, a_m^1, \ldots, a_1^n, \ldots, a_m^n$ из A выполняется включение

$$f * (g(a_1^1, ..., a_m^1), ..., g(a_1^n, ..., a_m^n)) \subseteq g * (f(a_1^1, ..., a_1^n), ..., f(a_m^1, ..., a_m^n)),$$

то говорим, что для f и g верно *полутождество перестановочности*, а также, что f *стабильна относительно* g, а g *нормальна относительно* f.

Если $g \in M_A$, то

$$S(g) = \{f \mid f \in P_A \text{ и } f \text{ стабильна относительно } g\}$$

называется *стабилизатором* g.

Если $f \in P_A$, то

$$N(f) = \{g \mid g \in M_A \text{ и } g \text{ нормальна относительно } f\}$$

называется *нормализатором* f.

Если $R \subseteq M_A$, то $S(R) = \bigcap\limits_{g \in R} S(g)$ — *стабилизатор* множества мультиопераций R.

Если $K \subseteq P_A$, то $N(P) = \bigcap\limits_{f \in K} N(f)$ — *нормализатор* множества K.

Теорема 1.
а) *Стабилизатор $S(R)$ является клоном для любого $R \subseteq M_A$.*
б) *Нормализатор $N(K)$ является суперклоном для любого $K \subseteq P_A$.*

Стандартным образом определяется алгебраическое замыкание множества операций K и множества мультиопераций R:

$[\,K\,] = \bigcap\limits_{K \subseteq K_i} K_i$, где K_i являются клонами;

$\langle\, R\,\rangle = \bigcap\limits_{R \subseteq R_i} R_i$, где R_i являются суперклонами.

Замыканием Галуа множества операций K и множества мультиопераций R называются, соответственно, $S(N(K))$ и $N(S(R))$.

Теорема 2. *Алгебраические замыкания совпадают с замыканиями Галуа:*
а) $[\,K\,] = S(N(K))$*;*
б) $\langle\, R\,\rangle = N(S(R))$.

Теорема 3. *Отображения $S(x)$ и $N(x)$ являются взаимно-обратными антиизоморфизмами решетки клонов и решетки суперклонов.*

Таким образом, отображения $S(x)$ и $N(x)$ определяют совершенную связь Галуа [5] между клонами и суперклонами.

Стабилизатор множества операций будем называть *централизатором*. При этом вместо полутождества перестановочности получаем тождество перестановочности [2]. Отметим, что централизатор определяет на множестве клонов двойственное себе соответствие Галуа [5], которое не является совершенным.

Работа выполнена при поддержке РФФИ, проект № 09-01-00476.

Список литературы

[1] *Яблонский С. В.* Функциональные построения в k-значной логике // Труды Матем. ин-та АН СССР им. В. А. Стеклова. — 1958. — Т. 51. — С. 5–142.

[2] *Мальцев А. И.* Итеративные алгебры Поста. — Новосибирск: Новосибирский государственный университет, 1976.

[3] *Бондарчук В. Г., Калужнин Л. А., Котов В.Н., Ромов Б. А.* Теория Галуа для алгебр Поста // Кибернетика. — 1969. — № 3. — С. 1–10; № 5. — С. 1–9.

[4] *Перязев Н. А.* Клоны, ко-клоны, гиперклоны и суперклоны // Ученые записки Казанского государственного университета. Серия Физико-математические науки. — Казань, 2009. — Т. 151. Книга 2. — С. 120–125.

[5] *Оре О.* Теория графов. — М.: Наука, 1980.

Построение графов-обструкций ограниченного ориентируемого рода

В. И. Петренюк

petrenjukvi@rambler.ru

Кировоградский национальный технический университет

Задача. Построить из нескольких простых графов новые графы G ограниченного ориентируемого рода $\gamma(G)$, у которых все рёбра существенны относительно рода при операции удаления произвольного ребра u, $\gamma(G\backslash u) = \gamma(G)-1$, т. е. графы-обструкции ограниченного эйлерова рода $\gamma(G)$. В $[3, 4]$ исследованы свойства графов G без вершин степени 2, с заданным эйлеровым родом $\gamma(G)$ и заданным разбиением на два 2-связных подграфа G_i меньшего рода γ_i, $i = 1, 2$. Граф G будем называть apex-m-склейкой графов G_i, полагая, что граф G_2 с $\{e_i\}_1^m$ ребрами подобен квазизвезде $St_m(G_2)$ с графом G_2 вместо центральной вершины и m рёбрами-лучами, инцидентными висячим вершинам g_i. Будем полагать, что такая квазизвезда приклеивается к графу G_1 путём отождествления пар точек (g_i, a_i) в вершину a_i^* графа G, где подмножества $M_1 = \{a_i\}_1^m$, $M_2 = \{g_i\}_1^m$ принадлежат графам G_1, $St_m(G_2)$, соответственно.

Результат решения задачи аналогичен приведенной в [2] оценке рода apex-графа G, представленного как φ-образ графа $(G\backslash v)$ +

$St_n(v)$ при φ-преобразовании путём попарного отождествления каждой вершины, инцидентной v, с концевыми вершинами звезды $St_n(v)$, где v — произвольная вершина графа G. Использовался метод φ-преобразований графов, предложенный в 1973 г. Н. П. Хоменко. Им было введено число достижимости $t_G(X), t_G(X) = t$, некоторого множества X точек графа G как наименьшее число 2-клеток s_j, $j = 1, \ldots, t$, на границы которых выходят все точки из X, где под точкой понимаем либо вершину, либо внутреннюю точку ребра. Для множества $\{s_j\}_{j=1}^t$ в [1] были предложены две новые характеристики $\theta, \partial\theta$ множества точек X графа G, роль которых заключается в измерении циклической и цепочной структур дуального графа.

Сформулируем основной результат.

Теорема 1. *Пусть имеют место следующие соотношения* А, Б, В*:*

А. *Каждое ребро u 2-связного графа G_i, $i = 1, 2$, удовлетворяет только одному из следующих условий:*
   1. *Существенно относительно рода $\gamma(G_i)$ при операции удаления ребра u, где $0 \leqslant \gamma(G_i \backslash u) < \gamma(G_i)$;*
   2. *Существенно относительно числа $t_{G_i}(M_i)$ при операции удаления ребра u, причём $\theta_{G_i \backslash u}(M_i) = \theta_{G_i}(M_i)$, $\partial\theta_{G_i \backslash u}(M_i) = \partial\theta_{G_i}(M_i)$, где $t_{G_i}(M_i) \geqslant 1$, $i = 1, 2$;*

Б. *Для каждого элемента x множества $x \in M_i$, где $M_1 = \{a_j\}_1^{m_1}$, $M_2 = \{g_j\}_1^{m_2}$, и хотя бы одного k, $k \in \{1, 2\}$, имеет место следующее неравенство: $t_{G_k}(M_k \backslash \{x\}) = t_{G_k}(M_k) - 1$;*

В. *Граф G представлен как φ-образ графа $G_1 + St_{m_2}(G_2)$, где $St_{m_2}(G_2)$ — квазизвезда с графом G_2 вместо центральной вершины с висячими рёбрами $(g_j, g_j^{'})$, где $g_j^{'}$ — висячие вершины, $M_2 = \{g_j\}_1^{m_2}$, при φ-преобразовании, заданном на множестве упорядоченных пар точек $(a_i, g_j^{'})$ из множеств M_1, M_2, соответственно, путём отождествления пары точек $(a_i, g_j^{'})$ в вершину a_{ij}^* для тех $i \in \{1, 2 ... m_1\}$, $j \in \{1, 2, \ldots, m_2\}$, что не порождают на множестве $\{a_{ij}^*\} \cup \{g_j^{'}\}$ новых частичных подграфов графа G, гомеоморфных графам K_4, $K_{2,3}$.*

Тогда имеют место следующие утверждения:

1) *каждое ребро u графа G существенно при операции удаления ребра относительно рода $\gamma(G)$;*

2) *род* $\gamma(G)$ *графа* G *удовлетворяет неравенству*

$$\gamma(G) \leqslant \sum_{i=1}^{2} \gamma(G_i) + \sum_{i=1}^{2} \big((t_{G_i}(M_i) - 1) - \theta_{G_i}(M_i) - \partial\theta_{G_i}(M_i)\big).$$

Доказательство. Рассмотрим минимальные вложения f_i графа G_i в замкнутое ориентируемое 2-многообразие σ_i рода $\gamma(G_i)$, при которых множества точек $M_i = \{a_j\}_1^{m_i}$ графа G_i, $i = 1, 2$, располагаются на границах 2-клеток из множества $S_{G_i}(M_i)$, где $S_{G_i}(M_i) = \sigma_i \setminus f_i(G_i)$, состоящего из $t_{G_i}(M_i)$ 2-клеток, причём циклическая и цепочная структуры множества $S_{G_i}(M_i)$ имеют числовые характеристики $\theta_{G_i}(M_i)$, $\partial\theta_{G_i}(M_i)$, соответственно. Выполним операции [2] присоединения к элементам множества $S_{G_i}(M_i)$ 2-ручек в количестве $t_{G_i}(M_i) - 1 - \theta_{G_i}(M_i) - \partial\theta_{G_i}(M_i)$ штук. Получим для $i = 1, 2$ два новых вложения $f_i^{'}$ графа G_i в замкнутое ориентируемое 2-многообразие $\sigma_i^{'}$ рода $\gamma_i^{'}$, где $\gamma_i^{'} = \gamma(G_i) + t_{G_i}(M_i) - 1 - \theta_{G_i}(M_i) - \partial\theta_{G_i}(M_i)$, реализующие расположение на границе некоторой клетки s_i множества $f_i^{'}(M_i)$, где $s_i \in \sigma_i^{'} \setminus f_i^{'}(G_i)$. Выполним стандартную операцию по отождествлению границ пары дисков d_i из клеток s_i при условии, что их границы не имеют общих точек. В результате получим склеенные по границам регулярных клеток 2-многообразия $\sigma_i^{'}$ рода $\gamma_i^{'}$ в 2-многообразие $\sigma^{''}$ и вложение $f, f = f_1^{'} + f_2^{'}$, графа $\sum\limits_{i=1}^{2} G_i$ в замкнутое ориентируемое 2-многообразие $\sigma^{''}$ рода

$$\sum_{i=1}^{2} \gamma(G_i) + \sum_{i=1}^{2} ((t_{G_i}(M_i) - 1) - \theta_{G_i}(M_i) - \partial\theta_{G_i}(M_i)),$$

причём множество точек $\sum\limits_{i=1}^{2} f_i^{'}(M_i)$ будет вкладываться на границу ∂s клетки s, склеенной из клеток s_1, s_2, подобной плоскому кольцу.

Согласно соотношения B отождествим (приклеим) висячие вершины $g_j^{'}$ графа $St_{m_2}(G_2)$ с точками a_j графа G_1, где $j \in \{1, 2, \ldots, m_2\}$, $a_i \in M_1$, так, чтобы на множестве из всех отождествлённых вершин a_{ij}^* для $i \in \{1, 2 ... m_1\}$, $j \in \{1, 2 ... m_2\}$, не порождались частичные подграфы, гомеоморфные графам K_4,$K_{2,3}$. В результате получим граф G, у которого в силу соотношения B ориентируемый род $\gamma(G)$ удовлетворяет неравенству

$$\gamma(G) \leqslant \sum_{i=1}^{2} \gamma(G_i) + \sum_{i=1}^{2}((t_{G_i}(M_i) - 1) - \theta_{G_i}(M_i) - \partial\theta_{G_i}(M_i)).$$

Покажем, что каждое ребро u графа G существенно относительно рода $\gamma(G)$ при операции удаления ребра. Действительно, имеют место только два следующих варианта.

Вариант 1. Если φ-прообраз ребра u принадлежит одному из графов G_i, то в силу соотношения А при операции удаления ребра либо уменьшится на 1 род $\gamma(G_i)$, либо уменьшится на 1 число $t_{G_i}(M_i)$, без изменений остаются характеристики $\theta_{G_i}(M_i)$, $\partial\theta_{G_i}(M_i)$.

Вариант 2. Если φ-прообраз ребра u не принадлежит ни одному из графов G_i, то это висячее ребро $(g_j, g_j^{'})$ из $St_{m_2}(G_2)$, а в силу соотношения Б для каждого элемента x множества $x \in M_i$ имеем либо $x = g_j$, либо $x = a_{ij}^{*}$, уменьшение на 1 числа $t_{G_k}(M_k)$. В каждом из вариантов 1, 2 получим уменьшение на 1 верхней оценки для числа $\gamma(G)\backslash u$, что означает существенность каждого ребра u графа G при операции удаления ребра относительно рода $\gamma(G)$. ∎

Следствие 1. *Если граф G_2 вырождается в точку g_0 и каждое ребро u графа G_1 либо существенно относительно рода $\gamma(G_1)$, либо существенно относительно числа $t_{G_1}(M_1)$, $t_{G_1}(M_1) > 1$ при операции удаления ребра u и для каждого элемента x множества $x \in M_i$ имеет место неравенство $t_{G_i}(M_i) = t_{G_i}(M_i) - 1$, при условии, что $M_1 = \{a_i\}_1^m$, $St_m(g_0)\backslash\{g_0\} = M_2 = \{g_i\}_1^m$, а граф G представлен как φ-образ графа $(G_1 + St_m(g_0)$ при φ-преобразовании, заданном путём попарного отождествления пар точек (a_i, g_i) в вершину a_i^{*} для всех $i = 1, 2, \ldots, m$, то каждое ребро u графа G существенно при операции удаления ребра относительно рода $\gamma(G)$, удовлетворяющего неравенству $\gamma(G) \leqslant \gamma(G_1) + t_{G_1}(M_1) - \theta_{G_1}(M_1) - \partial\theta_{G_1}(M_1) - 1$.*

Следствие 2. *Пусть граф G построен так, как указано в формулировке теоремы 1. Если граф G имеет ориентированный род $\gamma(G)$, такой, что $\gamma(G) = \sum\limits_{i=1}^{2} \gamma(G_i) + \sum\limits_{i=1}^{2}((t_{G_i}(M_i) - 1) - \theta_{G_i}(M_i) - \partial\theta_{G_i}(M_i))$, то каждое ребро u графа G существенно относительно рода при операции удаления ребра u.*

Следствие 3. *Пусть граф G построен так, как указано в формулировке следствия 1. Если граф G имеет ориентированный род $\gamma(G)$, такой, что $\gamma(G) = \gamma(G_1) + t_{G_1}(M_1) - \theta_{G_1}(M_1) - \partial\theta_{G_1}(M_1) - 1$, то каждое ребро u графа G существенно относительно рода при операции удаления ребра u.*

Для иллюстрации приведём следующие примеры.

Пример 1. Рассмотрим граф G, построенный из трёх копий графа K_5 без одного ребра путём отождествления троек несмежных вершин в пару несмежных вершин. Применим теорему 1 к этой известной обструкции для тора. Нетрудно убедиться в справедливости соотношений А, Б, В, а именно в том, что граф G_1, построенный из двух копий графа K_5 без одного ребра e, $e = (1, 2)$, путём отождествления i-й пары несмежных вершин в вершину v_i, где $i = 1, 2$, имеет род $\gamma(G_1)$, $\gamma(G_1) = 1$, и множество M_1, $M_1 = \{v_1, v_2\}$, имеет только одну ненулевую характеристику $t_{G_1}(M_1)$, $t_{G_1}(M_1) = 1$; третью копию графа K_5 без одного ребра графа G_2 сделаем центром квазизвезды $St_2(G_2)$ с двумя висячими рёбрами (i, g_i), где $i = 1, 2$, множество M_2, $M_2 = \{1, 2\}$, имеет только одну ненулевую характеристику $t_{G_2}(M_2) = 2$. Пары (v_i, g_i) висячих вершин и v_i отождествим в вершину v_i^*, где $i = 1, 2$, и получим граф G. Тогда $\gamma(G) \leqslant 1+0+(1-1)+2-1$. Сжимая в точку рёбра (i, g_i), получим упомянутую выше известную обструкцию для тора.

Пример 2. Для графа из примера 1 можно применить следствие 1. Рассмотрим граф G, построенный из графа G_1 и звезды $St_9(v)$ путём отождествления каждой пары вершин (v_i, g_i) в v_i^*, где $i \in \{1, 2, \ldots, 9\}$. Граф G_1 состоит из трёх копий графа K_4, склеенных по одной вершине в точку сочленения v^* с множеством M_1, $M_1 = \{v_i\}_{i=1}^{9}$, у которого характеристики $t_{G_1}(M_1) = 3$, $\theta_{G_1}(M_1) = \partial\theta_{G_1}(M_1) = 0$, удовлетворяют соотношениям А, Б, В теоремы 1. В силу следствия 1 имеем неравенство $\gamma(G) \leqslant 0 + 3 - 1$.

Список литературы

[1] *Петренюк В. И.* Две характеристики дуального графа плоского графа // Мат. Межд. конф. «Искусст. интеллект–2004». — Кацивели: Наука и освіта, 2004. — С. 230–231.

[2] *Петренюк В. И.* Обобщённая оценка рода простого графа // Искусст. интеллект. — 2004. — Т. 4. — С. 34–45.

[3] *Петренюк В. И.* Оценка рода apex-m-склейки простых графов // Материалы XV Международной конференции «Проблемы теоретической кибернетики». — М.: Издательство МГУ, 2008. — С. 34.

[4] *Mohar Bojan.* Face covers and the genus problem for apex graphs // J. Combin. Theory. Ser. B. — 2001. — V. 82. — P. 102–117.

Конечные автоматы и алгебраические модели программ с позиции разрешимости проблемы эквивалентности

Р. И. Подловченко

rip@vvv.nivc.msu.ru

Научно-исследовательский вычислительный центр
Московского государственного университета им. М. В. Ломоносова

Рассматриваются два семейства математических моделей вычислений — конечные автоматы над конечными алфавитами и алгебраические модели программ. Первые берутся в их классическом определении, а вторые — в определении, данном в [1]. Для тех и других обсуждается проблема эквивалентности. Изучается методика решения этой проблемы для конечных автоматов, и демонстрируется ее влияние на методику распознавания эквивалентности в алгебраических моделях программ. Материалом для этого служат результаты исследований, выполненных в [2, 3].

Алгебраические модели программ строятся над двумя конечными алфавитами Y и P. Элементы алфавита Y называются операторными символами, а элементы алфавита P — логическими переменными, каждая из которых пробегает множество $\{0, 1\}$. Все алгебраические модели над Y и P имеют общее множество объектов, каковыми являются схемы программ, и одна модель отличается от другой отношением эквивалентности на множестве схем программ.

В [2] установлен следующий факт: проблема эквивалентности в любой алгебраической модели над Y, P сводится к проблеме эквивалентности в множестве матричных схем, принадлежащих модели. На этом основании в [2, 3] исследуется проблема эквивалентности в множестве матричных схем.

Матричная схема над Y, P представляет собой ориентированный и размеченный граф, вершины которого подразделяются на вход, выход, пустой цикл и преобразователи. Из каждой вершины, кроме выхода, исходят дуги в количестве, равном числу элементов в множестве X, где $X = \big\{x \mid x : P \to \{0,1\}\big\}$; каждая дуга помечена элементом из X, и разные дуги — различными элементами. Дуги из пустого цикла ведут в него же. Каждому преобразователю сопоставлен символ из Y.

Каждая матричная схема над Y, P является носителем отображения, в общем случае частичного, множества $\mathcal{L}$, состоящего из функций разметки над Y, P, в множество Y^*; элементы последнего называются операторными цепочками. Функцией разметки над Y, P называется всюду определенное отображение множества Y^* в множество X.

Реализуемое матричной схемой отображение осуществляется процедурой ее выполнения на функции μ из $\mathcal{L}$. Выполнение представляет собой обход схемы, который начинается в ее входе с пустой операторной цепочкой из Y^* и сопровождается накоплением операторной цепочки. Приход в преобразователь знаменуется приписыванием к текущей цепочке символа, сопоставленного преобразователю, и если h — полученная при этом цепочка из Y^*, то выход из преобразователя совершается по дуге с пометкой $\mu(h)$. При достижении выхода схемы ее выполнение на μ завершается, а результатом его считается операторная цепочка, с которой достигается выход.

Эквивалентность матричных схем индуцируется параметрами ν и L, где ν — эквивалентность в Y^*, а $L \subseteq \mathcal{L}$. По определению, две матричные схемы (ν, L)-эквивалентны тогда и только тогда, когда, какой бы ни была функция μ из L, всякий раз, когда одна из схем достигает на μ выхода, другая достигает тоже, и результаты их выполнений на μ — это ν-эквивалентные цепочки.

Множество матричных схем над Y, P с введенной в нем (ν, L)-эквивалентностью называется моделью $M(\nu, L)$. Эквивалентность ν всегда является разрешимой.

Обратимся теперь к методике распознавания эквивалентности конечных автоматов над алфавитом Σ. Ею предусматривается два этапа исследования.

Назначением первого этапа является следующее: сохраняя отношение эквивалентности автоматов, перейти от выполнения их на множестве всех слов над Σ к выполнению на конечном его подмножестве. В этих целях предписывается:

1) для любого натурального n ввести разрешимое отношение n-эквивалентности автоматов, удовлетворяющее требованию: при любом n из эквивалентности автоматов следует их n-эквивалентность;
2) для любой пары автоматов A_1, A_2 конструктивно определить такое число $N(A_1, A_2)$, что из $N(A_1, A_2)$-эквивалентности следует их эквивалентность.

Решение задачи 1) осуществляется так: n-эквивалентными считаются автоматы, принимающие одни те же слова длины, не превосходящей числа n.

При решении задачи 2) отыскивается такое n, что к n-эквивалентным автоматам применима техника следов. Последняя предписывает искать в маршрутах длины $n + 1$, прокладываемых общим для автоматов словом, повторяющуюся пару вершин, равноудаленных от начала маршрутов, и сокращать маршруты на витки обнаруживаемых циклов. Оказалось, что $N(A_1, A_2) = n_1 n_2$, где n_i — число вершин в автомате A_i, $i = 1, 2$.

По выполнении первого этапа методики получаем алгоритм, разрешающий эквивалентность автоматов над Σ, но сложность этого алгоритма экспоненциальна относительно размеров сравниваемых автоматов.

Вторым этапом методики преследуется цель: построить алгоритм, разрешающий эквивалентность автоматов с полиномиальной сложностью. Для этого методикой предлагается следующее: при выполнении сравниваемых на эквивалентность автоматов A_1, A_2 перебор слов длины $n_1 n_2$ заменить анализом маршрутов, прокладываемых в автоматах словами. На этом пути и получен известный алгоритм Мура, сложность которого оценивается как $O(n^2 \log n)$, где $n = \max(n_1, n_2)$.

Изложенная методика стала руководящей при разработке методики распознавания эквивалентности в алгебраических моделях программ. Сохраняем оба этапа исследований с теми же предназначениями, что и в случае конечных автоматов.

На первом этапе при определении N-эквивалентности схем из $M(\nu, L)$ множество L заменяется множеством N-префиксов функций из L, которые определены на операторных цепочках длины N. Чтобы выполнялось предписание 1), на параметры ν и L налагаются ограничения: ν-эквивалентные цепочки должны иметь одинаковую длину, а L — состоять из всех функций из $\mathcal{L}$, сохраняющих свое значение на классах ν-эквивалентности цепочек. $M(\nu, L)$ тогда называется уравновешенной. Доказывается, что предписание 2) выполняется для уравновешенной модели $M(\nu, L)$ при дополнительных требованиях к ν: ν — это полугрупповая эквивалентность с левым сокращением. Последнее свойство заключается в том, что в ν-эквивалентных цепочках при ν-эквивалентности их префиксов являются ν-эквивалентными и их суффиксы. Итак, на первом этапе выявлены уравновешенные полугрупповые модели $M(\nu, L)$ с левым сокращением. В каждой из них разрешима проблема эквивалентности; сложность этого экспоненциальна.

Второй этап исследований преследует те же цели, что и в случае конечных автоматов, — искать приемлемые по сложности разрешающие алгоритмы. Сначала для любой из выявленных моделей построим тем же подходом, что и в случае конечных автоматов, алгоритм разрешения эквивалентности, альтернативный тому, что получен на первом этапе. Так как и он имеет экспоненциальную сложность, последовали дополнительные ограничения на параметр ν. Установлено, что альтернативный алгоритм легко модифицируется в алгоритм полиномиальной сложности, если ν-эквивалентные цепочки сохраняют ν-эквивалентность, будучи одинаково продолженными. Этим сделан прорыв к алгоритмам, разрешающим эквивалентность в $M(\nu, L)$ с приемлемой сложностью.

Работа выполнена при поддержке РФФИ, проект № 09-01-00277.

Список литературы

[1] *Подловченко Р.И.* Иерархия моделей программ // Программирование. — 1981. — № 2. — С. 3–14.

[2] *Подловченко Р.И.* Техника следов в разрешении проблемы эквивалентности в алгебраических моделях программ // Кибернетика и системный анализ. — 2009. — № 5. — С. 25–37.

[3] *Подловченко Р.И.* К вопросу о полиномиальной разрешимости проблемы эквивалентности в алгебраических моделях программ // Кибернетика и системный анализ. — 2011. — № 4.

О двухленточных машинах, описывающих полугруппы с левым сокращением

В. В. Подымов, В. А. Захаров

Valdus@yandex.ru, zakh@cs.msu.su

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Для решения задач анализа, верификации и оптимизации программ в статьях [1, 2] была предложена абстрактная модель последовательных программ, семантика которых описывается при помощи полугрупп. Состояниями данных программ являются элементы полугруппы, а операторы программ выступают в роли образующих элементов полугруппы. Такой способ представления операционных семантик программ позволяет описывать алгебраические свойства программных операторов при помощи полугрупповых тождеств. В ряде последующих статей (см. [3, 4, 5]) было показано, что в подобных моделях программ можно построить эффективные алгоритмы решения задачи проверки эквивалентности программ.

В частности, в статье [6] был предложен один общий подход к решению этой задачи, опирающийся на описания полугрупп посредством детерминированных двухленточных машин (2-DM). Двухленточная машина представляет собой детерминированный автомат с конечным или бесконечным числом состояний, снабженный двумя считывающими головками, движущимися в одну сторону по двум лентам, на которых записана пара слов. Машины такого вида способны описывать бинарные отношения на множестве слов конечного алфавита, в т. ч. отношение равенства элементов полугруппы с конечным числом образующих. Основу предложенного метода проверки эквивалентности программ составляют следующие две теоремы из статьи [6].

Теорема 1. *Для любой полугруппы отношение равенства элементов полугруппы $x = y$ распознается некоторой 2-DM тогда и только тогда, когда полугруппа является упорядоченной* (*т. е. для любой пары ее элементов x, y верно соотношение $xy = x \;\Rightarrow\; y = e$, где e — нейтральный элемент*).

Теорема 2. *Если семантика программных операторов определяется полугруппой, отношение равенства элементов которой может быть* *описано 2-DM с рекурсивной функцией переходов и конечным множеством допускающих состояний, то проблема эквивалентности программ в указанной семантике разрешима.*

В частности, условиям теоремы 2 удовлетворяют упорядоченные полугруппы, обладающие свойством левого сокращения (т. е. для любой тройки элементов x, y, z верно соотношение $xy = xz \ \Rightarrow \ y = z$). Таким образом, теорема 2 обобщает результаты, полученные в [1, 4]. В статье [7] были установлены необходимые и достаточные условия, при которых семантика некоторых программных операторов удовлетворяет условиям теоремы 1.

Сложность проблемы эквивалентности программ, семантика операторов которых определяется упорядоченной полугруппой, существенно зависит от устройства 2-DM, описывающей отношение равенства элементов полугруппы. Поэтому важное значение придается задаче построения наиболее простых 2-DM, описывающих полугруппу с заданной системой определяющих соотношений (тождеств). Решению этой задачи посвящена настоящая заметка.

Пусть задана полугруппа с конечным множеством образующих Σ и множеством определяющих соотношений (тождеств) S. Для любой пары слов h_1, h_2 из Σ^* запись $h_1 =_{(\Sigma,S)} h_2$ обозначает равенство элементов полугруппы, порожденных этими словами. Символом $\circ$ далее будем обозначать полугрупповую операцию, а символом e — нейтральный элемент полугруппы.

Двухленточной машиной со вспомогательной памятью над алфавитом Σ (далее — 2-DM) будем называть систему $A = (Q_1, Q_2, Mem, \Delta, R, q_0, s_0, F)$, состоящую из двух конечных непересекающихся множеств состояний управления Q_1, Q_2, множества (возможно, бесконечного) состояний вспомогательной памяти Mem, функции переходов $\Delta : (Q_1 \cup Q_2) \times Mem \times \Sigma \to Q_1 \cup Q_2$ и функции преобразования памяти $R : (Q_1 \cup Q_2) \times Mem \times \Sigma \to Mem$. В машине выделены начальное состояние управления q_0, начальное состояние памяти s_0 и множество финальных состояний $F, F \subseteq (Q_1 \cup Q_2)$.

Функцию переходов Δ можно распространить на множество $\Sigma^* \times \Sigma^*$ пар слов в алфавите Σ следующим образом:

$$\Delta^*(q, s, h_1, h_2) = \begin{cases} q, \text{ если } h_1 = \lambda, h_2 = \lambda, \\ \Delta^*(\Delta(q, s, a), R(q, s, a), h_1', h_2), \ q \in Q_1, \ h_1 = ah_1', \\ \Delta^*(\Delta(q, s, a), R(q, s, a), h_1, h_2'), \ q \in Q_2, \ h_2 = ah_2', \\ \text{не определено иначе.} \end{cases}$$

2-DM A описывает полугруппу (Σ, S), если для любой пары слов u, v из Σ^* верно соотношение $u =_{(\Sigma,S)} v \iff \Delta^*(q_0, s_0, u, v) \in F$.

Исследуемая задача состоит в том, чтобы для заданной полугруппы (Σ, S) определить 2-DM, описывающую эту полугруппу.

Рассмотрим полугруппу с расширенным множеством образующих $\Sigma \cup \overline{\Sigma}$, где $\overline{\Sigma} = \{\bar{a} : a \in \Sigma\}$, и множеством определяющих тождеств $\overline{S} = S \cup \{\bar{u} = \bar{v} : u = v \in S\} \cup \{\bar{a}a = e : a \in \Sigma\}$, где запись $\bar{h}$ для слова $h = a_1 a_2 \ldots a_n \in \Sigma^*$ обозначает слово $\bar{a}_n \ldots \bar{a}_2 \bar{a}_1$.

Воспользуемся записями $N(h, \Sigma)$ и $N(h, \overline{\Sigma})$ для обозначения количества символов алфавитов Σ и $\overline{\Sigma}$, соответственно, в слове h, $h \in \Sigma \cup \overline{\Sigma}$, и для каждого элемента x полугруппы $(\Sigma \cup \overline{\Sigma}, \overline{S})$ положим

$$K(x, \Sigma) = \min\Big(N(h, \Sigma) \ : \ h \in (\Sigma \cup \overline{\Sigma})^*, \ h =_{\Sigma \cup \overline{\Sigma}, \overline{S}} x\Big).$$

Аналогичным образом определяется значение $K(x, \overline{\Sigma})$.

Тогда в качестве двухленточной машины, предназначенной для описания полугруппы (Σ, S), предлагается использовать следующую 2-DM $A_{(\Sigma,S)} = (Q_1, Q_2, Mem, \Delta, R, q_0, s_0, F)$, где

- $Q_1 = \{q_0^1, q_1^1\}$;
- $Q_2 = \{q_1^2\}$;
- множество состояний памяти Mem состоит из всех элементов полугруппы $(\Sigma \cup \overline{\Sigma}, \overline{S})$, представимых в виде $\bar{h}_1 h_2$, где $h_1, h_2 \in \Sigma^*$;
- функцию переходов Δ можно описать следующим образом:
  - $\Delta(q_0^1, e, a) = q_1^2$,
  - $\Delta(q_1^1, x, a) = q_1^1$, если $K(x \circ a, \overline{\Sigma}) > 0$,
  - $\Delta(q_1^1, x, a) = q_1^2$, если $K(x \circ a, \overline{\Sigma}) = 0$ и $K(x \circ a, \Sigma) > 0$,
  - $\Delta(q_1^1, x, a) = q_0^1$, если $K(x \circ a, \overline{\Sigma}) = K(x \circ a, \Sigma) = 0$,
  - $\Delta(q_1^2, x, a) = q_1^2$, если $K(\bar{a} \circ x, \Sigma) > 0$,
  - $\Delta(q_1^2, x, a) = q_1^1$, если $K(\bar{a} \circ x, \Sigma) = 0$ и $K(\bar{a} \circ x, \overline{\Sigma}) > 0$,

— $\Delta(q_1^2, x, a) = q_0^1$, если $K(\bar{a} \circ x, \Sigma) = K(\bar{a} \circ x, \overline{\Sigma}) = 0$;

— функцию преобразования памяти R можно описать следующим образом:
  — $R(q_0^1, e, a) = a$,
  — $R(q_1^1, x, a) = x \circ a$,
  — $R(q_1^2, x, a) = \bar{a} \circ x$;

— $q_0 = q_0^1$;

— $s_0 = e$;

— $F = \{q_0^1\}$.

Теорема 3. *Если для произвольного конечного множества образующих Σ и множества тождеств S полугруппа (Σ, S) упорядочена и обладает свойством левого сокращения, то 2-DM $A_{(\Sigma,S)}$ описывает указанную полугруппу.*

Работа выполнена при поддержке РФФИ, проект № 09-01-00277.

Список литературы

[1] *Глушков В. М., Летичевский А. А.* Теория дискретных преобразователей // Избранные вопросы алгебры и логики. — Новосибирск: Наука, 1973. — С. 5–39.

[2] *Подловченко Р. И.* Полугрупповые модели программ // Программирование. — 1981. — № 4. — С. 3–13.

[3] *Захаров В. А.* Быстрые алгоритмы разрешения эквивалентности пропозициональных операторных программ на упорядоченных полугрупповых шкалах // Вестник Московского ун-та. Сер. 15. Вычислительная математика и кибернетика. — 1999. — № 3. — С. 29–35.

[4] *Летичевский А. А.* Эквивалентность автоматов относительно полугрупп с сокращением // Проблемы кибернетики. Вып.27. — М.: Наука, 1973. — С. 195–212.

[5] *Подловченко Р. И., Захаров В. А.* Полиномиальный по сложности алгоритм, распознающий коммутативную эквивалентность схем программ // Доклады РАН. — 1998. — Т. 362, № 6. — С. 27–31.

[6] *Захаров В. А.* Проверка эквивалентности программ при помощи двухленточных автоматов // Кибернетика и системный анализ. — 2010. — № 4. — С. 39–48.

[7] *Подымов В. В., Захаров В. А.* Об одной полугрупповой модели программ, определяемой при помощи двухленточных автоматов // Научн. ведомости Белгородского гос. ун-та. Серия История. Политология. Экономика. Информатика. — 2010. — Вып. 14/1, №7. — С. 94–101.

О мощности компонент корреляционно-иммунных функций, совершенных раскрасок и кодов

В. Н. Потапов

vpotapov@math.nsc.ru

Институт математики им. С. Л. Соболева СО РАН, Новосибирск

Введение

Обозначим через E^n множество упорядоченных двоичных наборов (вершин) длины n. Булев n-куб E^n естественным образом наделяется структурой векторного пространства над полем $GF(2)$. Пусть $S \subset E^n$, через χ^S будем обозначать характеристическую функцию множества S. Функция χ^S называется *корреляционно-иммунной порядка* $n-m$, если для любой грани размерности m её пересечения с множеством S имеют одинаковую мощность.

Сферой радиуса 1 с центром в вершине x называется множество $F(x) = \{y \in E^n : d(x,y) = 1\}$, где d — расстояние Хэмминга. *Совершенной раскраской* булева n-куба в k цветов называется отображение $Col : E^n \to \{1,\dots,k\}$, удовлетворяющее следующему условию: мощность пересечения $|Col^{-1}(i) \cap F(x)|$ зависит только от цветов i и $Col(x)$, но не от вершины $x \in E^n$. Каждой совершенной раскраске соответствует матрица параметров $A = \{a_{ij}\}$, где a_{ij} — число вершин цвета j в сфере радиуса 1 с центром в вершине цвета i. В дальнейшем речь пойдёт только о раскрасках в два цвета, причём для удобства изложения будем считать, что множество цветов есть $\{0,1\}$. В этом случае функция Col является булевозначной и $Col = \chi^S$, где S — множество вершин цвета 1. *Совершенным кодом* с расстоянием 3 называется подмножество булева n-куба, пересекающееся с любым шаром радиуса 1 ровно по одной вершине. Нетрудно видеть, что характеристической функцией совершенного кода $C \subset E^n$ является совершенная раскраска χ^C с матрицей параметров вида $\begin{pmatrix} 0 & n \\ 1 & n-1 \end{pmatrix}$. *Компонентой* совершенной раскраски χ^S булева n-куба E^n называется такое множество $K \subset S$, что для некоторого множества $K' \subset E^n \setminus K$ функция $\chi^{(S \cap K') \setminus K}$ является совершенной раскраской с теми же параметрами, что и χ^S. Проще говоря, подмножество $K \subset S$ можно заменить на K' без изменения свойств раскраски χ^S. Аналогично определяются компоненты совершенных кодов и корреляционно-иммунных функций.

Вопрос о спектре мощностей компонент совершенных кодов был поставлен в статье [1] и рассматривался, в частности, в работах [2, 3, 4]. Минимальная мощность компоненты совершенного кода была известна ранее. Однако, проблема существования компонент мощности *промежуточной* между минимальной и удвоенной минимальной оставалась мало исследованной. Ниже на основе результатов статьи [5] установлены необходимые условия на промежуточную мощность компонент и при некоторых параметрах построены совершенные раскраски и корреляционно-иммунные функции с компонентами промежуточной мощности.

Основные результаты

Каждая булева функция $f : E^n \to E$ может быть представлена в виде *многочлена Жегалкина. Алгебраической степенью* $\deg(f)$ булевой функции f называется максимальная степень слагаемого в её многочлене Жегалкина. Алгебраической степенью множества $S \subset E^n$ будем называть алгебраическую степень его характеристической функции.

Справедливо, следующее

Утверждение 1 (см. [8]). *Пусть $f : E^n \to E$ — корреляционно-иммунная функция порядка $n - m$. Тогда $\deg(f) \leqslant m$ (неравенство Зигенталлера); алгебраическая степень компоненты корреляционно-иммунной функции f не превосходит m.*

Поскольку совершенная раскраска с матрицей параметров

$$\begin{pmatrix} n-b & b \\ c & n-c \end{pmatrix} \tag{1}$$

является корреляционно-иммунной функцией порядка $\frac{b+c}{2} - 1$ (см., например, [6, 7, 8]), а совершенный код длины n является корреляционно-иммунной функцией порядка $\frac{n-1}{2}$, из утверждения 1 вытекают (см. [9]) следствия.

Следствие 1. *Пусть $f : E^n \to E$ — совершенная раскраска с матрицей параметров (1). Тогда $\deg(f) \leqslant n - \frac{b+c}{2} + 1$; алгебраическая степень компоненты совершенной раскраски f не превосходит $n - \frac{b+c}{2} + 1$.*

Следствие 2. *Пусть $C \subset E^n$ — совершенный код. Тогда $\deg(\chi^C) \leqslant \frac{n+1}{2}$; алгебраическая степень компоненты совершенного кода C не превосходит $\frac{n+1}{2}$.*

Булевы функции $f : E^n \to E$ можно рассматривать как элементы булева куба размерности 2^n. Множество булевых функций алгебраической степени не выше m называется кодом Рида — Маллера типа $\mathcal{R}(m,n)$ в E^{2^n}. В [10] рассмотрен весовой спектр кодов Рида — Маллера и, в частности, имеются следующие утверждения.

Утверждение 2. [10, глава 13, теоремы 3 и 5] *Для любой булевой функции $f = \chi^S$ справедливо неравенство $|S| \geqslant 2^{n-\deg(f)}$. Если $|S| = 2^{n-\deg(f)}$, то множество S — линейный код.*

Утверждение 3. [5], [10, глава 15, теорема 10] *Пусть $f = \chi^S$ — булева функция в E^n, $\deg(f) \geqslant 2$ и $2^{n-\deg(f)+1} > |S| \geqslant 2^{n-\deg(f)}$. Тогда $|S| = 2^{n-\deg(f)+1} - 2^{n-\deg(f)+1-p}$, где $p \in \{1,\dots,\mu\}$, где $\mu = \max\{(n-\deg(f)+2)/2, \min\{n-\deg(f), \deg(f)\}\}$.*

Из утверждений 1–3 и следствий 1–2 получаем

Следствие 3. *Пусть множество $S \subset E^n$ есть компонента корреляционно-иммунной функции порядка $n-m$ и $2^{n-m+1} > |S|$. Тогда $|S| = 2^{n-m+1} - 2^p$, где $p \in \{1,\dots,n-m\}$. Более того, компонента мощности 2^{n-m} является линейным кодом.*

Следствие 4. *Пусть f — совершенная раскраска с матрицей параметров $\begin{pmatrix} n-b & b \\ c & n-c \end{pmatrix}$, множество $S \subset E^n$ есть компонента f и $2^{\frac{b+c}{2}} > |S|$. Тогда $|S| = 2^{\frac{b+c}{2}} - 2^p$, где $p \in \{1,\dots,\frac{b+c}{2}-1\}$. Более того, компонента мощности $2^{\frac{b+c}{2}-1}$ является линейным кодом.*

Следствие 5. *Пусть множество $S \subset E^n$ есть компонента совершенного кода $C \subset E^n$ и $2^{\frac{n+1}{2}} > |S|$. Тогда $|S| = 2^{\frac{n+1}{2}} - 2^p$, где $p \in \{1,\dots,\frac{n-1}{2}\}$. Более того, компонента мощности $2^{\frac{n-1}{2}}$ является линейным кодом.*

Теорема 4. *Пусть $p \in \{0,\dots,km-1\}$, $n = (2^s-1)k$, $m = 2^{s-2}$, $s \geqslant 2$, $km \geqslant 3$. Существует совершенная раскраска f с матрицей параметров $\begin{pmatrix} k & k(2^s-2) \\ 2k & k(2^s-3) \end{pmatrix}$, имеющая компоненту мощности $(2^{km} - 2^p)2^{km}$.*

При $k = 1$ такую совершенную раскраску рассматривают как двукратный совершенный код. Из теоремы 4, подставляя $m = 1$, получаем

Следствие 6. *Пусть $n = 3k + n'$, $r = 2k + n' - 1$, $k \geqslant 3$. Для любого $p \in \{0, \ldots, k-1\}$ найдётся корреляционно-иммунная функция $g : E^{n+n'} \to E$ порядка r, имеющая компоненту мощности $(2^k - 2^p)2^{k+n'}$.*

Работа выполнена при поддержке Российского фонда фундаментальных исследований (проекты 10-01-00424, 10-01-00616).

Список литературы

[1] *Etzion T., Vardy A.* Perfect binary codes and tilings: problems and solutions // SIAM J. Discrete Math. — 1998. — V. 11, № 2. — P. 205–223.

[2] *Avgustinovich S. V., Lobstein A. C., Soloveva F. I.* Intersection matrices for partitions by binary perfect codes // IEEE Trans. Inform. Theory. — 2001. — V. 47, № 4. — P. 1621–1624.

[3] *Avgustinovich S. V., Heden O., Soloveva F. I.* On intersection problem for perfect binary codes // Des. Codes Cryptogr. — 2006. — V. 39, № 3. — P. 317–322.

[4] *Heden O., Soloveva F. I., Mogilnykh I. Yu.* Intersections of perfect binary codes // Proc. of IEEE Int. Conf. on Computational Technologies in Electrical and Electronics Engineering. — Piscataway: IEEE, 2010. — P. 50–51.

[5] *Kasami T., Tokura N.* On the weight structure of Reed—Muller codes // IEEE Trans. Inform. Theory — 1970. — V. 16. — P. 752–759.

[6] *Фон-Дер-Флаасс Д. Г.* Совершенные 2-раскраски гиперкуба // Сибирский математический журнал — 2007. — Т. 48, № 4. — С. 923–930.

[7] *Fon-Der-Flaass D. G.* A bound of correlation immunity // Siberian Electronic Mathematical Reports — 2007. — V. 4. — P. 133–135.

[8] *Таранников Ю. В.* О корреляционно-иммунных и устойчивых булевых функциях // Математические вопросы кибернетики — Выпуск 11, М.: Физматлит, 2002. — С. 91–148.

[9] *Потапов В. Н.* О совершенных раскрасках булева n-куба и корреляционно-иммунных функциях малой плотности // Сибирские электронные математические известия — 2010. — Т. 7. — С. 372–382.

[10] *Мак-Вильямс Ф. Дж., Слоэн Н. Дж. А.* Теория кодов, исправляющих ошибки. — М.: Связь, 1979.

Алгебры языков, представимых в размеченных графах

Е. А. Пряничникова

pryanichnikovae@gmail.com

Государственный университет информатики и искусственного интеллекта, Донецк

Вводится система операций на формальных языках, которая, в частности, может использоваться в биологии, генетике, а также ДНК-вычислениях. Для этой системы операций вводится понятие регулярных выражений. Определяется понятие языка, допустимого в конечном автомате (в графе с отмеченными вершинами). Доказано, что язык L допустим в конечном автомате (в графе с отмеченными вершинами) тогда и только тогда, когда он описывается регулярным выражением во введенной системе операций.

Основные определения

Пусть X — конечный алфавит, X^* — множество всех слов конечной длины в алфавите X, X^+ — множество всех непустых слов конечной длины в алфавите X, X^n — множество всех слов длины n в алфавите X, $X^{\geqslant n}$ — множество всех слов конечной длины в алфавите X, длина которых больше или равна n , 2^{X^*} — множество всех языков в алфавите X. Обозначим пустое множество через $\varnothing$, пустое слово через λ. Конкатенацию двух слов $u \in X^+$ и $v \in X^+$ обозначим uv. Конкатенацию двух языков $L, R \in 2^{X^+}$ обозначим через $L \cdot R$ или LR и определим как $L \cdot R = \{uv | u \in L, v \in R\}$.

Определим на множестве X^* частичную бинарную операцию $\overset{n}{\circ}$ склеивания двух слов с параметром n следующим образом: для всех $w_1, w_2 \in X^*$

$$w_1 \overset{n}{\circ} w_2 = \begin{cases} xyz, & \text{если } w_1 = xy,\ w_2 = yz,\ y \in X^n; \\ \text{не определено}, & \text{в противном случае.} \end{cases}$$

В случае когда $n = 0$, введенная операция совпадает с операцией конкатенации слов.

Введем на языках L, $R \subseteq X^*$ следующие операции:

1) $L \cup R = \{w | w \in L \text{ или } w \in R\}$;

2) $L \overset{n}{\circ} R = \{w_1 \overset{n}{\circ} w_2 | w_1 \in L$ и $w_2 \in R\}$;
3) $L^{\overset{n}{+}} = \bigcup_{i=1}^{\infty} L^i$, где $L^1 = L$; $L^{i+1} = L^i \overset{n}{\circ} L$ для всех $i \geqslant 1$;
4) $L^{\overset{n}{*}} = \bigcup_{i=0}^{\infty} L^i$, где $L^0 = X^n$; $L^{i+1} = L^i \overset{n}{\circ} L$ для всех $i \geqslant 0$.

Введенные операции на формальных языках, в частности, могут использоваться в биологии, генетике, ДНК-вычислениях [1].

Операция $\overset{n}{\circ}$ ассоциативна при любом n, то есть $(2^{X^*}, \overset{n}{\circ})$ и $(2^{X^{\geqslant n}}, \overset{n}{\circ})$ — полугруппы. Нейтральный элемент по операции $\overset{n}{\circ}$ существует тогда и только тогда, когда она определена на множестве языков, в которых нет слов с длиной меньше n. Если нейтральный элемент существует, то он равен X^n. Таким образом, полугруппа $(2^{X^*}, \overset{n}{\circ})$ является моноидом только при $n = 0$.

Рассмотрим два семейства алгебр $(2^{X^*}, \overset{n}{\circ}, \cup, \overset{n}{+}, \varnothing)$ и $(2^{X^{\geqslant n}}, \overset{n}{\circ}, \cup, \overset{n}{*}, X^n, \varnothing)$.

Все алгебры $(2^{X^{\geqslant n}}, \overset{n}{\circ}, \cup, \overset{n}{*}, X^n, \varnothing)$ являются полукольцами. Алгебра $(2^{X^*}, \overset{n}{\circ}, \cup, \overset{n}{+}, \varnothing)$ будет иметь единицу по операции $\overset{n}{\circ}$ только в случае, когда $n = 0$ и операция $\overset{n}{\circ}$ совпадает с конкатенацией, а рассматриваемая алгебра является алгеброй регулярных языков. Во всех остальных случаях эти алгебры не будут полукольцами.

Регулярные выражения в алгебре $(2^{X^{\geqslant n}}, \overset{n}{\circ}, \cup, \overset{n}{*}, X^n, \varnothing)$ определим рекурсивно следующим образом:

1) $\varnothing$ является регулярным выражением и представляет язык $L(\varnothing) = \varnothing$;
2) x является регулярным выражением и представляет язык $L(x) = \{x\}$ для всех $x \in X^n \cup X^{n+1}$;
3) если R и Q — регулярные выражения, представляющие языки $L(R)$ и $L(Q)$ соответственно, то выражения $(R \overset{n}{\circ} Q)$, $(R \cup Q)$, $(R^{\overset{n}{*}})$ также являются регулярными, причем $L(R \overset{n}{\circ} Q) = L(R) \overset{n}{\circ} L(Q)$, $L(R \cup Q) = L(R) \cup L(Q)$, $L(R^{\overset{n}{*}}) = (L(R))^{\overset{n}{*}}$.

Регулярные выражения в алгебре $(2^{X^*}, \overset{n}{\circ}, \cup, \overset{n}{+}, \varnothing)$ определим следующим образом:

1) $\varnothing$ является регулярным выражением и представляет язык $L(\varnothing) = \varnothing$;

2) x является регулярным выражением и представляет язык $L(x) = \{x\}$ для всех $x \in \bigcup\limits_{0 \leqslant i \leqslant n+1} X^i$;
3) если R и Q — регулярные выражения, представляющие языки $L(R)$ и $L(Q)$ соответственно, то выражения $(R \overset{n}{\circ} Q)$, $(R \cup Q)$, $(R^{\overset{n}{+}})$ также являются регулярными, причем $L(R \overset{n}{\circ} Q) = L(R) \overset{n}{\circ} L(Q)$, $L(R \cup Q) = L(R) \cup L(Q)$, $L(R^{\overset{n}{+}}) = (L(R))^{\overset{n}{+}}$.

Графом с отмеченными дугами (конечным автоматом) назовем четверку $G = (Q, E, X, \mu)$, где Q — конечное множество вершин, $\mid Q \mid = n$, $E \subseteq Q \times Q$ — множество дуг, X — множество отметок, $\mu : E \to X$ — функция отметок дуг.

Графом с отмеченными вершинами назовем четверку $G = (Q, E, X, \mu)$, где Q — конечное множество вершин, $|Q| = n$, $E \subseteq Q \times Q$ — множество дуг, X — множество отметок вершин, $\mu : Q \to X$ — функция отметок вершин.

Пусть $I \subseteq Q$ — множество начальных вершин графа, $F \subseteq Q$ — множество финальных вершин. Множество отметок всех путей, начальной вершиной которых является вершина q_i, а конечная вершина $q_k \in F$, назовем языком, порожденным вершиной q_i. Отметки всех путей в графе G, начальные вершины которых принадлежат множеству I, а конечные — множеству F, назовем языком, допускаемым графом G, и обозначим $L(G)$.

В теории конечных автоматов одним из важнейших результатов является теорема Клини, в которой утверждается, что класс языков, распознаваемых конечными автоматами, совпадает с классом рациональных языков, представимых регулярными выражениями алгебры Клини [2]. Основная цель данной работы — доказать аналогичную теорему для более широкого класса размеченных графов и алгебр.

Основные результаты

Теорема 1. *Язык $L \subseteq X^*$ допустим в конечном автомате (в графе с отмеченными вершинами) тогда и только тогда, когда он описывается регулярным выражением любой алгебры из рассматриваемых двух семейств $(2^{X^*}, \overset{n}{\circ}, \cup, \overset{n}{+}, \varnothing)$ и $(2^{X^{\geqslant n}}, \overset{n}{\circ}, \cup, \overset{n}{*}, X^n, \varnothing)$.*

Эта теорема в некотором смысле аналогична широко известной теореме Клини для конечных автоматов. В случае когда $n = 0$ и рассматриваются только графы с отмеченными дугами, теорема 1 совпадает с теоремой Клини. На основе доказательства теоремы разработаны методы анализа и синтеза языков, представимых в графах с отмеченными вершинами.

Поскольку для описания одного и того же класса графов можно использовать различные алгебры, представляет интерес вопрос о связи таких алгебр между собой.

Теорема 2. *Для двух полуколец* $(2^{X^{\geqslant n_1}}, \overset{n_1}{\circ}, \cup, \overset{n_1}{*}, X_1^n, \varnothing)$ *и* $(2^{X^{\geqslant n_2}}, \overset{n_2}{\circ}, \cup, \overset{n_2}{*}, X_2^n, \varnothing)$ *в случае, когда* $n_1 < n_2$ *, существует такое отображение* $\varphi : 2^{X^{\geqslant n_1}} \to 2^{X^{\geqslant n_2}}$*, которое является гомоморфизмом. Если* $n_2 > n_1$*, то гомоморфизма нет.*

Рассматриваемое в теореме отображение является сюръекцией, поэтому в случае, когда $n_1 < n_2$, полукольцо $(2^{X^{\geqslant n_1}}, \overset{n_1}{\circ}, \cup, \overset{n_1}{*}, X_1^n, \varnothing)$ изоморфно вложимо в полукольцо $(2^{X^{\geqslant n_2}}, \overset{n_2}{\circ}, \cup, \overset{n_2}{*}, X_2^n, \varnothing)$, причем образ φ является подполукольцом в $(2^{X^{\geqslant n_2}}, \overset{n_2}{\circ}, \cup, \overset{n_2}{*}, X_2^n, \varnothing)$, а значит, все рассматриваемые полукольца входят в одно квазимногообразие, в которое входит алгебра регулярных языков.

Теорема 3. *Пусть* $\Re(n)$ *— множество всех регулярных выражений алгебры* $(2^{X^{\geqslant n}}, \overset{n}{\circ}, \cup, \overset{n}{*}, X^n, \varnothing)$*. Если* $n_1 < n_2$*, то существует такое отображение* $\psi : \Re(n_1) \to \Re(n_2)$*, которое сохраняет язык регулярного выражения, т. е. если* r *— регулярное выражение алгебры* $(2^{X^{\geqslant n_1}}, \overset{n_1}{\circ}, \cup, \overset{n_1}{*}, X_1^n, \varnothing)$*,* $L(r)$ *— язык, представляемый этим регулярным выражением, то* $\psi(r)$ *— это регулярное выражение алгебры* $(2^{X^{\geqslant n_2}}, \overset{n_2}{\circ}, \cup, \overset{n_2}{*}, X_2^n, \varnothing)$ *и* $L(\psi(r)) = L(r)$*.*

Заключение

В данной работе исследованы основные свойства семейства алгебр языков, допустимых в размеченных графах, найдена алгебраическая характеризация языков, представимых регулярными выражениями этих алгебр, разработаны методы анализа и синтеза рассматриваемых языков.

Список литературы

[1] *Anderson J.* Automata Theory with Modern Applications. — Cambridge: Cambridge University Press, 2006.

[2] *Капитонова Ю. В., Летичевский А. А.* Математическая теория проектирования вычислительных систем. — М.: Наука, 1988.

Структурные свойства комбинаторных систем, описываемых субмодулярными функциями

А. М. Ревякин

Arevyakin@mail.ru

Московский государственный институт электронной техники

Пусть $P(S)$ — множество всех подмножеств конечного множества S. Система $\mathcal{I} \subseteq P(S)$ подмножеств из S называется матроидом $M = (S, \mathcal{I})$, а множества из $\mathcal{I}$ — независимыми, если выполняются следующие условия: (i1) $\oslash \in \mathcal{I}$; (i2) если $A \subseteq B$ и $B \in \mathcal{I}$, то $A \in \mathcal{I}$; (i3) если $A, B \in \mathcal{I}$ и $|A| > |B|$, то найдется $a \in A \setminus B$, такое, что $B \cup \{a\} \in \mathcal{I}$.

Ранговой функцией матроида M на множестве S называется целочисленная функция $r(A)$, определенная для всех $A \subseteq S$, такая, что $r(A) = \max\{|X| : X \subseteq A \text{ и } X \in \mathcal{I}\}$.

Пара (S, r), где r — целочисленная функция (ранг), определенная на подмножествах конечного множества S, образует матроид, если для всех $A, B \subseteq S$ и любых $a, b \in S$ выполняются либо свойства: (r1) $0 \leqslant r(A) \leqslant |A|$; (r2) если $A \subseteq B$, то $r(A) \leqslant r(B)$; (r3) $r(A \cup B) + r(A \cap B) \leqslant r(A) + r(B)$; либо: (r4) $r(\varnothing) = 0$; (r5) $r(A) \leqslant r(A \cup \{a\}) \leqslant r(A) + 1$; (r6) если $r(A) = r(A \cup \{a\}) = r(A \cup \{b\})$, то $r(A) = r(A \cup \{a, b\})$. При этом семейство $\mathcal{I}$ подмножеств множества S, для которых $r(A) = |A|$, образует матроид $(S, \mathcal{I})$ с ранговой функцией r.

Конечное множество S с оператором замыкания $A \to \overline{A}$, определенным для всех $A \subseteq S$ (т.е. (cl1) $A \subseteq \overline{A}$ для всех $A \subseteq S$; (cl2) если $A, B \subseteq S$ и $A \subseteq B$, то $\overline{A} \subseteq \overline{B}$; (cl3) $\overline{A} = \overline{\overline{A}}$ для всех $A \subseteq S$) и удовлетворяющим свойству замены: (cl4) для любых $x, y \in S$ и всякого $A \subseteq S$ из $y \in \overline{A \cup \{x\}}$ и $y \notin \overline{A}$ следует, что $x \in \overline{A \cup \{y\}}$, образует матроид $(S, \overline{})$. Причем семейство $\mathcal{I}$ подмножеств $A \subseteq S$, таких, что из $x \in A$ следует, что $x \notin A \setminus \{x\}$, образует матроид $M = (S, \mathcal{I})$.

Если $\overline{A} = A$, то A называется замкнутым (или поверхностью) в M. Пара $(S, \mathcal{F})$, где $\mathcal{F}$ — семейство подмножеств (поверхностей) из S, образует матроид, если (f1) $S \in \mathcal{F}$; (f2) если $F_1, F_2 \in \mathcal{F}$, то

$F_1 \cap F_2 \in \mathcal{F}$; (f3) если $F_1, F_2, \ldots, F_k \in \mathcal{F}$ и F_i покрывает F (т. е. $F_i \supseteq F$ и не существует поверхности $F' \in \mathcal{F}$ такой, что $F_i \supset F' \supset F$) для всех $i, i = 1, 2, \ldots, k$, то $\{F_1 \setminus F, \ldots, F_k \setminus F\}$ — разбиение множества $S \setminus F$. Причем семейство $\mathcal{I}$ подмножеств $A \subseteq S$, таких, что для всех $a \in A$ найдется подмножество $F \in \mathcal{F}$, для которого $A \setminus F = \{a\}$, образует матроид $(S, \mathcal{I})$.

Если $G = (S, \overline{})$ — матроид и $\mathcal{L}(G)$ — множество всех его поверхностей, упорядоченных по включению, то $\mathcal{L}(G)$ является геометрической решеткой (см. [1, 2]), в которой $A \vee B = \overline{A \cup B}$ и $A \wedge B = A \cap B$, для всех $A, B \in \mathcal{L}(G)$.

Пусть K — дистрибутивная решетка с 0 (нулем) и 1 (единицей), где $x \vee y = \sup\{x; y\}$, $x \wedge y = \inf\{x; y\}$ для всех $x, y \in K$. Например, таковой является семейство всех подмножеств конечного множества S с операциями объединения и пересечения. Действительная функция μ, определенная на K, называется субмодулярной, если для всех x, y из K выполняется неравенство: $\mu(x) + \mu(y) \geqslant \mu(x \vee y) + \mu(x \wedge y)$. Если для всех x, y из K справедливо равенство: $\mu(x) + \mu(y) = \mu(x \vee y) + \mu(x \wedge y)$, то функция μ называется модулярной.

Субмодулярными функциями являются ранговые функции матроидов и полиматроидов, мощность множества, а модулярными — размерности подпространств в векторных пространствах и в проективных геометриях. Комбинаторные системы, описываемые в терминах субмодулярных функций, рассмотрены в [3, 4, 5].

Подрешетка L дистрибутивной решетки K называется μ-остовом, если субмодулярная функция μ, определенная на дистрибутивной решетке K, будет модулярной на L.

Пусть $\mu(x) = \sum\limits_{i=1}^{n} c_i \mu_i(x)$, где $\mu_i(x)$ — субмодулярные функции на дистрибутивной решетке K, c_i — положительные действительные коэффициенты и $i = 1, 2, \ldots, n$. Тогда $\mu(x)$ — также субмодулярная функция на K.

Рассмотрим задачу минимизации субмодулярной функции $\mu(x)$ на K. В таком виде, например, можно сформулировать задачи объединения и пересечения матроидов. Действительно, если (S, r_i) — матроиды на множестве S с ранговыми функциями r_i, где $i = 1, 2, \ldots, n$, то ранг объединения матроидов равен $\min\limits_{X\subseteq S}\{\sum\limits_{i=1}^{n} r_i(X)+|S\setminus X|\}$, а размер максимального независимого множества первых двух матроидов — $\min\limits_{X\subseteq S}\{r_1(X)+r_2(S\setminus X)\}$.

Теорема 1. *Семейство L всех элементов дистрибутивной решетки K, на которых субмодулярная функция μ достигает своего минимума, образует μ-остов решетки K. Более того, подрешетка L является также и μ_i-остовом решетки K для всех $i, i = 1, 2, \ldots, n$.*

Полученный остов L зависит не только от субмодулярных функций μ_i, но и от коэффициентов c_i. Обозначим остов L через $L(c_1, c_2, \ldots, c_n)$, подчеркивая его зависимость от коэффициентов $c_1, c_2, \ldots, c_n$. Поскольку для всех $\lambda > 0$ выполнено $L(\lambda\cdot c_1, \ldots, \lambda\cdot c_n) = L(c_1, \ldots, c_n)$, остов L можно рассматривать как функцию на $n-1$-мерном симплексе S^{n-1} со значениями в семействе подрешеток дистрибутивной решетки K. В случае когда μ_i являются монотонными, можно детальнее охарактеризовать структуру этого комплекса.

Теорема 2. *Пусть $\mu_1, \ldots, \mu_p$ — неубывающие, $\mu_{p+1}, \ldots, \mu_q$ — невозрастающие субмодулярные функции, а $c_i \geqslant c_i{}'$ для $i = 1, 2, \ldots, p$, $c_i \leqslant c_i'$ для $i = p+1, p+2, \ldots, q$ и $c_i = c_i'$ для $i = q+1, q+2, \ldots, n$. Тогда если $y \in L(c_1, \ldots, c_n)$ и $y' \in L(c_1', \ldots, c_n')$, то $y \wedge y' \in L(c_1, \ldots, c_n)$, $y \vee y' \in L(c_1', \ldots, c_n')$ и для всех $i, i = 1, 2, \ldots, n$, имеет место равенство $\mu_i(y) + \mu_i(y') = \mu_i(y \vee y') + \mu_i(y \wedge y')$.*

Два интервала $[x, y]$ и $[x', y']$ решетки L называются транспонированными, если $[x, y] = [b, a \vee b]$ и $[x', y'] = [a \wedge b, a]$ для некоторых a и $b \in L$. Транспонированные интервалы модулярной решетки изоморфны. Скажем, что интервалы $[x, y]$ и $[x', y']$ являются проективными (обозначение: $[x, y] \sim [x', y']$), если найдется конечная последовательность интервалов $[x, y]$, $[x_1, y_1]$, $\ldots$, $[x_N, y_N]$, $[x', y']$, в которой два любых соседних интервала транспонированы.

Теорема 3. *Максимальные цепи, соединяющие наименьший и наибольший элементы конечной модулярной решетки, имеют одинаковую длину. Если $a_0, a_1, \ldots, a_n$ и $b_0 = a_0, b_1, \ldots, b_n = a_n$ — пара таких максимальных цепей, то существует перестановка σ индексов $\{1, 2, \ldots, n\}$, такая, что $[a_{i-1}, a_i] \sim [b_{\sigma(i)-1}, b_{\sigma(i)}]$ для $i = 1, 2, \ldots, n$. Кроме того, если рассматриваемая решетка является дистрибутивной, то перестановка σ однозначно определена.*

Таким образом, $\mathcal{F} = \{[a_{i-1}, a_i], i = 1, 2, \ldots, n\}$ — семейство интервалов максимальной цепи $a_0, a_1, \ldots, a_n$ конечной дистрибутивной решетки однозначно определено с точностью до проективности. На $\mathcal{F}$ можно ввести отношение порядка $\leqslant$: $[x, y] \leqslant [z, w]$, если для каждой максимальной цепи $a_0, a_1, \ldots, a_n$ найдутся p и q, такие, что $p \leqslant q, [a_p, a_{p+1}] \sim [x, y]$ и $[a_q, a_{q+1}] \sim [z, w]$.

Пусть μ — субмодулярная функция на дистрибутивной решетке K с 0 и 1, L — подрешетка решетки K с наименьшим элементом a и наибольшим b, а $\mathcal{F} = \{[a_i, a_{i+1}], i = 1, 2, \ldots, n-1\}$, где $a = a_1, a_2, \ldots, a_n = b$ — произвольная максимальная цепь из a в b в подрешетке L. На каждом интервале $\mathcal{F}$ зададим функцию φ_i, где $i = 1, 2, \ldots, n-1$, положив $\varphi_i(x) = \mu(x) - \mu(a_i)$ для всех $x \in [a_i, a_{i+1}]$. Если $[0, a]$ и $[b, 1]$ в K не пусты, то положим $\varphi_0(x) = \mu(x)$ для всех $x \in [0, a]$ и $\varphi_n(x) = \mu(x) - \mu(b)$ для всех $x \in [b, 1]$. Аналогично определим функцию ${\varphi_i}'(x) = \mu(x) - \mu({a_i}')$ для другой максимальной цепи $a = {a_1}', {a_2}', \ldots, a'_n = b$ из a в b. Очевидно, что так определенные функции φ_i и ${\varphi_i}'$ являются субмодулярными.

В силу теоремы 3 найдется определенная перестановка σ индексов $\{1, 2, \ldots, n-1\}$, такая, что $[a_i, a_{i+1}] \sim [a'_{\sigma(i)}, a'_{\sigma(i+1)}]$ для всех $i, i = 1, 2, \ldots, n-1$. Пусть $\pi_i : [a_i, a_{i+1}] \to [a'_{\sigma(i)}, a'_{\sigma(i+1)}]$ — естественный изоморфизм, обусловленный проективностью интервалов максимальных цепей. Скажем, что определение «новой» субмодулярной функции φ_i не зависит от выбора максимальной цепи из a в b в подрешетке L дистрибутивной решетки K, если $\varphi_i(x) = \mu(x) - \mu(a_i) = \mu(\pi_i(x)) - \mu(a'_{\sigma(i)}) = \varphi'_{\sigma(i)}(\pi_i(x))$ для всех $x \in [a_i, a_{i+1}]$, где $i = 1, 2, \ldots, n-1$.

Теорема 4. *Пусть L — подрешетка конечной дистрибутивной решетки K, μ — субмодулярная функция на K. Тогда μ не зависит от выбора максимальной цепи в том и только в том случае, когда L является μ-остовом решетки K.*

Булевы интерпретации сформулированных теорем полезны при наличии эффективных алгоритмов решения задач комбинаторной оптимизации [6, 7, 8].

Список литературы

[1] *Биркгоф Г.* Теория решеток. — М.: Наука, 1984.

[2] *Гретцер Г.* Общая теория решеток. — М.: Мир, 1982.

[3] *Nakamura M., Iri M.* Fine structures om matroid intersections and their applications // Int. Symp. Circuits and Syst. Proc., Tokyo. — 1979. — P. 996–999.

[4] *Nakamura M.* Boolean sublattices connected with minimization problem on matroids // Math. Program. — 1982. — V. 22, № 1. — P. 117–120.

[5] *Fujishige S.* Submodular systems and relate topics // Math.Programm. Study. — 1984. — V. 22. — P. 113–131.

[6] Theory of matroids / Ed. N. White. — Cambridge: Cambridge Univ. Press, 2008.

[7] *Oxley J. G.* Matroid theory. — N.Y.: Oxford Univ. Press, 2006.

[8] *Revyakin A. M.* Matroids // J. Math. Sci. — 2002. — V. 108, № 1. — P. 71–130.

Синтез стратегий выбора объектов в линейной зоне обслуживания двух mobile-процессоров

М. Б. Резников, А. И. Цветков

`mikerez@mail.ru, tze@bk.ru`

Волжская государственная академия водного транспорта,
Нижний Новгород

Исследуется проблема управления обслуживанием потока объектов в рабочей зоне двух mobile-процессоров. Особенности функционирования подобных систем содержательно поясним на примере сервисных предприятий внутреннего водного транспорта, в деятельности которых всё большее распространение получает технология технического обслуживания судов «на ходу», при транзитном прохождении ими зоны ответственности сервисного предприятия. В качестве обслуживающего выступает специализированное судно, предназначенное для выполнения одного или некоторого фиксированного набора работ. Любое транзитное судно при вхождении в зону ответственности сервисного предприятия может запросить один или два различных вида обслуживания, последовательно их получить или не быть обслуженным вовсе. Основная задача диспетчера сервисного предприятия (лица, принимающего решения — ЛПР) заключается в выработке и последующем обеспечении реализации наиболее рациональной стратегии управления обслуживанием судов в условиях конкретной эксплуатационной обстановки. С позиций повышения эффективности функционирования сервисного предприятия интерес представляют суммарные доходы, интегрированные по видам обслуживания. Именно для такого обобщения математической модели [1, 2] ниже формулируется бикритериальная задача синтеза стратегий обслуживания и предлагается алгоритм синтеза оптимально-компромиссных стратегий, реализующий в рамках концепции Парето [3] идеологию динамического программирования [4, 5].

Пусть имеется детерминированный поток объектов $O(n) = \{o(1), o(2), \ldots, o(n)\}$, которые поступают в линейную рабочую зону Ξ двух независимых и невзаимозаменяемых процессоров P^1 и P^2, осуществляющих однофазное обслуживание объектов в процессе их безостановочного прохождения зоны Ξ. В дискретной идеализации рабочую зону представляем как упорядоченную последовательность элементарных участков с номерами $1, 2, \ldots, s-1, s$. Поток $O(n)$ обладает свойством бинарности, т. е. состоит из двух подпотоков O_1 и O_s, таких, что $O_1 \bigcup O_s = O(n)$ и $O_1 \bigcap O_s = \varnothing$. Объекты подпотока O_1 входят в зону через граничный участок с номером 1 и проходят её, двигаясь равномерно по направлению к участку с номером s; объекты подпотока O_s поступают в рабочую зону через участок с номером s и проходят ее аналогично в противоположном направлении. Соответственно объекты подпотока O_1 покидают зону через участок с номером s, а объекты подпотока O_s выходят из рабочей зоны через участок с номером 1.

Каждый объект $o(i)$, $i = \overline{1, n}$, характеризуется следующими целочисленными параметрами: $t(i)$ — момент поступления в зону Ξ; $d(i)$ — указатель принадлежности подпотоку O_1 или O_s ($d(i) = 1$, если $o(i) \in O_1$, и $d(i) = 0$, если $o(i) \in O_s$); $\chi(i)$ — норма времени пребывания объекта на элементарном участке; $w^1(i)$ ($w^2(i)$) — доход за обслуживание процессором P^1 (P^2) объекта $o(i)$ ($w^1(i) > 0, w^2(i) > 0$); $\tau^1(i)$ ($\tau^2(i)$) — норма длительности обслуживания объекта $o(i)$ процессором $P^1(P^2)$ ($\tau^1(i) \geqslant 0, \tau^2(i) \geqslant 0$).

В пределах зоны Ξ процессоры P^1 и P^2 могут перемещаться как автономно с постоянной скоростью в любом направлении, так и в паре с обслуживаемым объектом со скоростью последнего. Возможен также простой процессора P^1 (P^2) в ожидании подхода объекта, назначенного на обслуживание.

Движение процессоров характеризуется следующими целочисленными параметрами: z^k — номер элементарного участка, на котором расположен процессор P^k в начальный момент времени $t = 0$ ($z_k \in \{1, 2, \ldots, s\}$); T_1^k (T_s^k) — норма времени пребывания процессора P^k на элементарном участке в процессе автономного движения в направлении от участка с номером 1 к участку с номером s (от участка с номером s к участку с номером 1), $k = 1, 2$.

Стратегию обслуживания объектов потока определим в виде пары $\{\rho^1, \rho^2\}$, где ρ^k $(k = 1, 2)$ представляет собой $m(k)$-элементный ($m(k) \in [0, n]$) кортеж

$$\rho = \begin{cases} (\varphi_1^k, \psi_1^k), (\varphi_2^k, \psi_2^k), \ldots, (\varphi_{m(k)}^k, \psi_{m(k)}^k) & \text{при } m(k) \geqslant 1, \\ \varnothing & \text{при } m(k) = 0, \end{cases}$$

в записи которого использованы обозначения: φ_j^k — идентификатор объекта $o(\varphi_j^k)$, обслуживаемого процессором P^k в очередь j ($\varphi_j^k \in [1, n], j = \overline{1, m(k)}$); ψ_j^k — участок начала обслуживания объекта $o(\varphi_j^k)$ в очередь j ($\psi_j^k \in [1, l]$).

Соответственно при реализации этой стратегии суммарный доход за обслуживание объектов потока $O(n)$ процессором P^k определяется выражением $W^k(\rho) = \sum_{j=1}^{m(k)} w^k(\varphi_j^k)$, $k = 1, 2$.

Общий подход к исследованию проблемы принятия решений при наличии двух и более критериев оценки основывается на концепции Парето и в условиях рассматриваемой двухпроцессорной модели обслуживания приводит к задаче $\{\max\limits_{\rho \in \Omega}(W^1(\rho)),\ \max\limits_{\rho \in \Omega}(W^2(\rho))\}$.

Поток объектов и обслуживающие их процессоры будем рассматривать как дискретную систему, на каждом этапе j управления которой для свободного процессора формируется управление $\{\upsilon_j, \omega_j\}$, где υ_j — номер назначенного на обслуживание объекта, а ω_j — номер участка начала его обслуживания.

Состояние ξ рассматриваемой системы характеризуется значениями набора параметров $(t, l, p, r, \theta, \Lambda^1, \Lambda^2)$, где t — дискретное время; l — номер свободного процессора ($l \in [1, 2]$); p — участок зоны Ξ, на котором расположен процессор P^s ($p \in [1; l]$); r — номер объекта, обслуживание которого производит процессор $P^{(3-l)}$; θ — число тактов времени, оставшегося до завершения обслуживания процессором $P^{(3-l)}$ объекта $o(r)$; Λ^k — множество обслуженных на момент t объектов процессором P^k $(k = 1, 2)$. На начальном этапе $\xi_0 = (0, 1, z^1, z^2, 0, \varnothing, \varnothing)$.

Множество $\Phi(j)$ допустимых управлений $\{\upsilon_j, \omega_j\}$ на этапе обслуживания j может быть получено из кинематических соображений. Пусть $\mathbf{x}$ — вектор, Y — множество векторов той же размерности. Через $\mathbf{x} \oplus Y$ обозначим совокупность всех векторов $\mathbf{v}$, представимых как $\mathbf{v} = \mathbf{x} + \mathbf{y}$ ($\mathbf{y} \in Y$). Обозначим через M множество векторов-оценок, а через $\mathit{eff}\,(M)$ — максимальное по включению подмножество недоминируемых в M векторов.

Если на множестве $\Phi(j)$ выбрано некоторое управление $\{\upsilon_j, \omega_j\}$, приводящее к оценке $(a_j^1, a_j^2) = ((2-l)w^1(\upsilon_j), (l-1)w^2(\upsilon_j))$ и переводящее систему в состояние ξ_{j+1}, то, выделив из совокупности оценок эффективные, получим множество $B(\xi_{j+1})$. Выбранное управление обеспечивает любую оценку из совокупности $[(a_j^1, a_j^2) \oplus B(\xi_{j+1})]$. Тогда

$$B(\xi_j) = \mathit{eff} \left\{ \bigcup_{\{\upsilon_j, \omega_j\} \in \Phi(\xi_j)} [(a_j^1, a_j^2) \oplus B(\xi_{j+1})] \right\}. \tag{1}$$

На последнем этапе обслуживания h, переводящем систему в состояние, когда последний объект покинул зону Ξ, имеет место соотношение

$$B(\xi_h) = \{(0, 0)\}. \tag{2}$$

Алгоритмическая реализация соотношений динамического программирования (1), (2) позволяет реализовать синтез полной совокупности эффективных оценок и соответствующих им стратегий обслуживания.

Для иллюстрации рассмотрен пример при следующих значениях параметров: $n = 10$, $s = 30$, $z^1 = 18$, $z^2 = 22$, $T_1^1 = 13$, $T_1^2 = 15$, $T_s^1 = 15$, $T_s^2 = 17$; $t(i)$, $i = \overline{1, n}$: 0, 300, 800, 1000, 1400, 1800, 1900, 2300, 2600, 2900; $\tau^1(i)$, $i = \overline{1, n}$: 100, 200, 200, 200, 300, 200, 200, 100, 200, 200; $\tau^2(i)$, $i = \overline{1, n}$: 300, 200, 200, 200, 100, 200, 100, 300, 200, 200; $d(i)$, $i = \overline{1, n}$: 0, 1, 1, 0, 1, 0, 1, 0, 1, 0; $\chi(i)$, $i = \overline{1, n}$: 14, 12, 13, 15, 12, 14, 14, 17, 13, 14; $w^1(i)$, $i = \overline{1, n}$: 205, 219, 121, 297, 221, 199, 88, 148, 253, 84; $w^2(i)$, $i = \overline{1, n}$: 300, 212, 171, 208, 120, 178, 126, 97, 281, 297.

Множество эффективных оценок $(W^1(\rho),\ W^2(\rho))$ имеет следующую структуру: (1036, 1176), (815, 1296), (1339, 579), (1373, 578), (1120, 879), (1450, 417), (1255, 876). В частности, решению задачи выбора стратегии, обеспечивающей максимальный доход (суммарно по обоим процессорам) [2], соответствуют первая из приведенных оценок $(1036 + 1176 = 2212)$, $\rho^1 = \{(1,14),(3,1),(5,1),(7,6),(8,25),(9,6)\}$, $\rho^2 = \{(2,1),(4,25),(6,25),(9,1),(10,26)\}$.

Список литературы

[1] *Коган Д. И., Федосенко Ю. С., Шеянов А. В.* Проблема синтеза оптимального расписания обслуживания бинарного потока объектов mobile-процессором // Труды III Международной конференции «Дискретные модели в теории управляющих систем». — М.: Изд-во МГУ им. М. В. Ломоносова, 1998. — С. 43–46.

[2] *Резников М. Б., Федосенко Ю. С.* Задача оптимизации стратегии обслуживания бинарного потока объектов двумя mobile-процессорами в линейной рабочей зоне // Вестник Нижегородского университета. — 2007. №4. — С. 104–109.

[3] *Подиновский В. В., Ногин В. Д.* Парето-оптимальные решения многокритериальных задач. — М.: Физматлит, 2007.

[4] *Беллман Р., Дрейфус С.* Прикладные задачи динамического программирования. — М.: Наука, 1965.

[5] *Коган Д. И.* Динамическое программирование и дискретная многокритериальная оптимизация. — Н. Новгород: Изд-во ННГУ, 2005.

О гамильтоновых циклах в графах минимальных расстояний совершенных кодов

А. М. Романов

rom@math.nsc.ru

Институт математики им. С. Л. Соболева, Новосибирск

Рассмотрим векторное пространство $\mathbb{F}_q^n$ размерности n над полем Галуа GF_q. Обозначим через $d(\mathbf{x}, \mathbf{y})$ расстояние Хемминга между векторами $\mathbf{x}$ и $\mathbf{y}$. Произвольное подмножество $\mathbb{C}_q^n$ векторов из пространства $\mathbb{F}_q^n$ называется *совершенным q-ичным кодом* длины n с кодовым расстоянием 3, если для каждого вектора $\mathbf{x} \in \mathbb{F}_q^n$ существует единственный вектор $\mathbf{c}$ из множества $\mathbb{C}_q^n$, такой, что $d(\mathbf{x}, \mathbf{c}) \leqslant 1$. Известно, что совершенные q-ичные коды существуют только при $n = (q^m - 1)/(q - 1)$, где m — любое натуральное число не меньше двух. Код называется *линейным*, если он образует линейное подпространство в пространстве $\mathbb{F}_q^n$. Линейные совершенные коды с кодовым расстоянием 3 называются *кодами Хемминга*. Далее q-ичный код Хемминга длины n будем обозначать через $\mathbb{H}_q^n$. Код называется *приведённым*, если он содержит вектор $\mathbf{0}$, состоящий из одних нулей. Далее мы будем рассматривать только приведённые коды. Пусть $\mathbb{C}$ — совершенный код с параметрами кода Хемминга. *Графом минимальных расстояний* кода $\mathbb{C}$ называется граф $G(\mathbb{C})$, множество вершин которого совпадает с $\mathbb{C}$, и вершины $\mathbf{x}, \mathbf{y}$ графа $G(\mathbb{C})$ являются смежными тогда и только тогда, когда $d(\mathbf{x}, \mathbf{y}) = 3$. Совершенный код $\mathbb{C}$ называется *гамильтоновым*, если его граф $G(\mathbb{C})$ минимальных расстояний содержит гамильтонов цикл.

В настоящей работе вопрос о существовании гамильтоновых нелинейных совершенных q-ичных кодов длины $N = qn + 1$ сведён к вопросу о существовании таких кодов длины n. При $q = 3$ для всех допустимых длин $n \geqslant 13$ предложены нелинейные совершенные q-ичные коды, графы минимальных расстояний которых содержат гамильтонов цикл. Ранее в [1] было конструктивно доказано существование гамильтоновых нелинейных двоичных кодов для всех допустимых длин $n \geqslant 15$. Остаётся открытым вопрос о существовании гамильтонова цикла в графе, образованном двумя средними слоями n-мерного двоичного куба нечётной размерности. Также не доказана гипотеза Ловаша, в которой утверждается, что из вершинной транзитивности связного графа следует существование в нём гамильтонова цикла.

Проверочная матрица кода Хемминга $\mathbb{H}_q^n$ длины $n = (q^m - 1)/(q-1)$ состоит из n попарно линейно независимых столбцов h_i, где $h_i^T \in \mathbb{F}_q^m$, $i \in \{1, \ldots, n\}$. Совокупность векторов $\mathbb{F}_n^q \setminus \{\mathbf{0}\}$ порождает конечную $(m-1)$-мерную проективную геометрию $PG_{m-1}(q)$ над полем Галуа GF_q. В этой геометрии точкам соответствуют столбцы проверочной матрицы кода Хемминга $\mathbb{H}_q^n$, и три точки i, j, k лежат на одной прямой, если соответствующие им столбцы h_i, h_j, h_k являются линейно зависимыми.

Пусть $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_q^n$. Тогда *носителем* вектора $\mathbf{x}$ называется множество $S(\mathbf{x}) = \{i : x_i \neq 0\}$. В коде $\mathbb{H}_q^n$ рассмотрим вектор $\mathbf{x}$, такой, что его носитель $S(\mathbf{x})$ является $(m-2)$-мерной гиперплоскостью в $PG_{m-1}(q)$. Обозначим через $H_{\mathbf{x}}$ множество всех векторов

$\mathbf{v} \in \mathbb{H}_q^n$, таких, что $S(\mathbf{v}) \subseteq S(\mathbf{x})$. Очевидно, что $H_{\mathbf{x}}$ образует в $\mathbb{H}_q^n$ подкод предыдущей размерности.

Сумму векторов $\mathbf{x}, \mathbf{y} \in \mathbb{F}_n^q$ обозначим через $\mathbf{x} + \mathbf{y}$. Пусть $l_1, l_2, \ldots, l_n$ — прямые, проходящие через точку i ($i \in \{1, \ldots, N\}$), в проективной геометрии $PG_m(q)$ и

$$R_i^N = \mathbb{H}_q^N(l_1) + \mathbb{H}_q^N(l_2) + \cdots + \mathbb{H}_q^N(l_n),$$

где $\mathbb{H}_q^N(l_p)$ — подкод кода $\mathbb{H}_q^N$, определяемый прямой l_p, $p = 1, 2, \ldots, n$, $N = qn + 1 = (q^{m+1} - 1)/(q - 1)$. Всевозможные смежные классы $R_i^N + \mathbf{v}$ ($\mathbf{v} \in \mathbb{H}_q^N$) представляют собой совокупность *i-компонент* кода Хемминга $\mathbb{H}_q^N$. Так как размерность подпространства $\mathbb{H}_q^N(l_p)$ равна $q - 1$, то размерность R_i^N равна $(q-1)n = q^m - 1$ (см. [2, 3]). Из определения множества R_i^n следует, что минимально возможное расстояние между двумя различными векторами из R_i^n равно 3. Через $G(R_i^n)$ обозначим граф минимальных расстояний множества R_i^n.

Теорема 1. *При $m \geqslant 2$, $n = (q^m - 1)/(q - 1)$ граф $G(R_i^n)$ содержит гамильтонов цикл.*

Доказательство. Поскольку множество векторов из R_i^n образует в коде Хемминга $\mathbb{H}_q^n$ подпространство, порождённое всеми векторами веса 3 с ненулевой i-й координатой, то граф $G(R_i^n)$ минимальных расстояний содержит гамильтонов цикл. ∎

Перейдём далее к формулировке основной теоремы.

Теорема 2. *Пусть существует нелинейный совершенный q-ичный код $\mathbb{C}_q^n$ длины $n = (q^{m-1} - 1)/(q - 1)$, $m \geqslant 3$, граф минимальных расстояний которого содержит гамильтонов цикл. Тогда существует нелинейный q-ичный код $\mathbb{D}_q^N$ длины $N = qn + 1$, граф минимальных расстояний которого также содержит гамильтонов цикл.*

Доказательство. Рассмотрим конструкцию нелинейных совершенных q-ичных кодов, предложенную в [4, 5]. Эта конструкция является обобщением для q-ичных кодов конструкции из [6]. Будем считать, что столбцы проверочной матрицы кода Хемминга $\mathbb{H}_q^N$ упорядочены лексикографически. Векторы из пространства $\mathbb{F}_q^N$ также будем рассматривать как слова длины N над алфавитом $\{0, 1, \ldots, q-1\}$. Пусть

$$\mathbb{D}_q^N = \bigcup_{\mathbf{c} \in \mathbb{C}_q^n} \left(R_i^N + (\mathbf{c}|\mathbf{0})\right), \tag{1}$$

где вектор $\mathbf{0} \in \mathbb{F}_q^{(qn-n+1)}$, $i \geqslant n+1$ и вертикальная черта $(|)$ обозначает конкатенацию. Формула (1) является некоторой интерпретацией конструкции из [4, 5]. Множество $\mathbb{D}_q^N$ является совершенным q-ичным кодом длины $N = qn + 1$ с кодовым расстоянием 3. Нелинейность кода $\mathbb{D}_q^N$ следует из формулы (1) и нелинейности кода $\mathbb{C}_q^n$. Гамильтоновость кода $\mathbb{D}_q^N$ следует из теоремы 1, формулы (1) и гамильтоновости кода $\mathbb{C}_q^n$. ∎

Далее, пусть $q = 3$. При $m = 2$ все совершенные троичные коды длины $n = 4$ эквивалентны коду Хемминга $\mathbb{H}_3^4$. Граф минимальных расстояний кода Хемминга $\mathbb{H}_3^4$ представляет собой полный граф, состоящий из 9 вершин. При $m = 3$ несложно при помощи свитчингов построить нелинейные совершенные троичные коды длины $n = 13$ и проверить при помощи компьютера, что графы минимальных расстояний этих кодов содержат гамильтонов цикл.

Теорема 3. *При $q = 3$, $m \geqslant 3$ для всех допустимых длин существуют нелинейные совершенные q-ичные коды, графы минимальных расстояний которых содержат гамильтонов цикл.*

Работа выполнена при поддержке РФФИ, проект № 11-01-00997.

Список литературы

[1] *Романов А. М.* О комбинаторных кодах Грея с расстоянием 3 // Дискретная математика. — 2009. — Т. 21, № 3. — С. 73–78.

[2] *Романов А. М.* О разбиениях q-ичных кодов Хемминга на непересекающиеся компоненты // Дискрет. анализ и исслед. операций. Сер. 1. — 2004. — Т. 11, № 3. — С. 80–87.

[3] *Phelps K. T., Villanueva M.* Ranks of q-ary 1-perfect codes // Designs, Codes and Cryptogr. — 2002. — V. 27, № 1–2. — P. 139–144.

[4] *Schönheim J.* On linear and nonlinear single-error-correcting q-nary perfect codes // Inform. and Control. — 1968. — V. 12, № 1. — P. 23–26.

[5] *Lindström B.* On group and nongroup perfect codes in q symbols // Math. Scand. — 1969. — V. 25. — P. 149–158.

[6] *Васильев Ю. Л.* О негрупповых плотно упакованных кодах // Проблемы кибернетики. Вып. 8. — М.: Физматгиз, 1962.— С. 75–78.

О проверяющих тестах относительно перестановок переменных в булевых функциях

Д. С. Романов

romanov@cs.msu.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Пусть $\tilde{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ — набор значений булевых переменных $x_1, x_2, \ldots, x_n$ ($n \in \mathbb{N}$). Множество всех таких наборов $\tilde{\alpha}$ образует *n-мерный булев куб*, обозначаемый через E_2^n. *Весом* набора $\tilde{\alpha}$ называется число единиц в наборе $\tilde{\alpha}$ (обозначение: $\|\tilde{\alpha}\|$). Множество всех наборов из E_2^n, имеющих вес k, называется *k-м слоем булева куба* E_2^n и здесь обозначается через $E_2^n(k)$. Через $\rho(\tilde{\alpha}', \tilde{\alpha}'')$ ($\tilde{\alpha}', \tilde{\alpha}'' \in E_2^n$) будем обозначать *расстояние Хэмминга* между наборами (вес покоординатной суммы наборов $\tilde{\alpha}'$ и $\tilde{\alpha}''$ по модулю 2).

Пусть $f(\tilde{x}^n)$ — булева функция, формально зависящая от переменных $x_1, x_2, \ldots, x_n$. Обозначим $N_f = \{\tilde{\alpha} \mid \tilde{\alpha} \in E_2^n,\ f(\tilde{\alpha}) = 1\}$.

Пусть, далее, Γ — некоторая группа (относительно композиции) биекций на множестве E_2^n. При этом, если $\zeta \in \Gamma$ ($\zeta = \zeta(\tilde{x}^n) :\ E_2^n \to E_2^n$ — взаимно однозначное отображение на множестве E_2^n), то ζ^{-1} — биекция, обратная к ζ. (Ясно, что в группе Γ содержится тождественное отображение $e :\ E_2^n \to E_2^n$, являющееся единицей группы). Обозначим через $f_\zeta(\tilde{x}^n)$ функцию, полученную из функции $f(\tilde{x}^n)$ действием отображения ζ (т. е. $f_\zeta(\tilde{x}^n) = f(\zeta(\tilde{x}^n))$, или, что то же самое, для любого $\tilde{\alpha} \in E_2^n$ $f_\zeta(\tilde{\alpha}) = f(\zeta(\tilde{\alpha}))$). Через $\Gamma(f)$ обозначим множество всех булевых функций, полученных из f действием отображений $\zeta \in \Gamma$: $\Gamma(f) = \{f_\zeta \mid \zeta \in \Gamma\}$. Очевидно, что $f = f_e \in \Gamma(f)$.

Введем два важных частных случая для Γ. Обозначим через $\mathfrak{L}$ (соответственно, через $\mathfrak{S}$) группу Γ всех биекций на множестве E_2^n, порожденных всевозможными перестановками и отрицаниями переменных x_1, x_2, $\ldots$, x_n (соответственно, группу Γ всех биекций на множестве E_2^n, порожденных всевозможными перестановками переменных x_1, x_2, $\ldots$, x_n). Более формально, пусть $\varphi = \begin{pmatrix} 1 & 2 & \ldots & n \\ i_1 & i_2 & \ldots & i_n \end{pmatrix}$ — подстановка на множестве $\{1, 2, \ldots, n\}$ (лежащая в симметрической группе S_n), а $\tilde{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n)$ — набор из E_2^n. Для каждого набора $\tilde{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in E_2^n$ определим

$\zeta_\varphi(\tilde{\alpha}) = (\alpha_{i_1}, \alpha_{i_2}, \ldots, \alpha_{i_n})$, $\zeta_{\tilde{\varepsilon}}(\tilde{\alpha}) = \tilde{\alpha}\oplus\tilde{\varepsilon} = (\alpha_1\oplus\varepsilon_1, \alpha_2\oplus\varepsilon_2, \ldots, \alpha_n\oplus\varepsilon_n)$. Таким образом определяются биекции ζ_φ и $\zeta_{\tilde{\varepsilon}}$. Группа $\mathfrak{L}$ порождена множеством биекций $\{\zeta_\varphi, \zeta_{\tilde{\varepsilon}} \mid \varphi \in S_n,\ \tilde{\varepsilon} \in E_2^n\}$, а группа $\mathfrak{S}$ — множеством биекций $\{\zeta_\varphi \mid \varphi \in S_n\}$. Множество булевых функций $\mathfrak{L}(f)$ называется *группой Лоренца функции* f.

Множество T наборов значений переменных $x_1, x_2, \ldots, x_n$ называется *проверяющим тестом относительно действия группы* Γ *над булевой функцией* $f(x_1, x_2, \ldots, x_n)$ тогда и только тогда, когда любая не равная f функция $f_\zeta((\tilde{x}^n))$ из $\Gamma(f)$ отличается от f на множестве T. Число наборов в тесте T называется его *длиной* и обозначается $|T|$ или $l(T)$. Тест минимальной длины называется *минимальным.* Длину минимального проверяющего теста относительно действия группы Γ над булевой функцией $f(\tilde{x}^n)$ будем обозначать через $l_\Gamma^{\mathrm{check}}(f(\tilde{x}^n))$. Введем функцию Шеннона длины проверяющего теста относительно действия группы Γ над булевой функцией:

$$l_\Gamma^{\mathrm{check}}(n) = \max_{f(\tilde{x}^n)\in P_2^n} l_\Gamma^{\mathrm{check}}(f(\tilde{x}^n)),$$

Отметим, что тест относительно действия группы $\mathfrak{S}$ над булевой функцией $f(\tilde{x}^n)$ называется *тестом относительно произвольных перестановок переменных булевой функции* $f(\tilde{x}^n)$, а тест относительно действия группы $\mathfrak{L}$ над булевой функцией $f(\tilde{x}^n)$ — *тестом относительно группы Лоренца булевой функции* $f(\tilde{x}^n)$.

Обобщим методику, предложенную Г. Р. Погосяном в [1] (см. также [2]) для оценивания сверху функции Шеннона длины проверяющего теста относительно произвольных инверсий переменных в булевой функции, на получение верхних оценок функции Шеннона $l_\Gamma^{\mathrm{check}}(n)$ в случае произвольной группы Γ биекций на E_2^n.

Пусть $f(\tilde{x}^n)$ — булева функция, Γ — группа (относительно композиции) биекций на E_2^n, $A \subseteq E_2^n$ — некоторое (возможно, пустое) множество наборов, $\zeta \in \Gamma$. Обозначим через $\zeta(A)$ множество наборов $\{\zeta(\tilde{\alpha}) \mid \tilde{\alpha} \in A\}$, а через $\Phi(f, A)$ — множество всех функций из $\Gamma(f)$, неотличимых от f на множестве наборов $\zeta(A)$ (заметим, что $f = f_e \in \Phi(f, A)$). В частности, $\Phi(f, \varnothing) = \Gamma(f)$.

Лемма 1. *Пусть $f(\tilde{x}^n)$ — булева функция, Γ — группа (относительно композиции) биекций на E_2^n. Тогда для любого множества A_w ($A_w \subseteq E_2^n$) такого, что $\Phi(f, A_w) \setminus \Phi(f, E_2^n) \neq \varnothing$, существует множе-*

ство A_{w+1} $(A_{w+1} \subseteq E_2^n)$ *такое, что* $|A_{w+1}| = |A_w| + 1$ *и*

$$|\Phi(f, A_{w+1})| \leqslant 0,5 \cdot |\Phi(f, A_w)|.$$

Доказательство. Пусть $\Gamma(f) = \{f_{\zeta_i}(\tilde{x}^n) \mid i = \overline{1,t}\}$, причем булевы функции $f_{\zeta_1} = f_e = f$, $f_{\zeta_2}, \ldots, f_{\zeta_s}$ лежат в $\Phi(f, A_w)$, а булевы функции $f_{\zeta_{s+1}}, f_{\zeta_{s+2}}, \ldots, f_{\zeta_t}$ не лежат в $\Phi(f, A_w)$ $(1 \leqslant s \leqslant t)$. Так как $\Phi(f, A_w) \setminus \Phi(f, E_2^n) \neq \varnothing$, то найдутся набор $\tilde{\beta} \in E_2^n \setminus A_w$ и число $j \in \{1, 2, \ldots, s\}$ такие, что для функции $f_{\zeta_j}(\tilde{x}^n) \in \Phi(f, A_w)$ выполнено: $f_{\zeta_j}(\tilde{\beta}) \neq f(\tilde{\beta})$. Пусть среди чисел $f_{\zeta_1}(\tilde{\beta}), f_{\zeta_2}(\tilde{\beta}), \ldots, f_{\zeta_s}(\tilde{\beta})$ значение $f(\tilde{\beta})$ встречается q раз.

Случай 1. Если $q \leqslant 0,5s$, то, полагая $A_{w+1} = A_w \cup \{\tilde{\beta}\}$, получаем: $|\Phi(f, A_{w+1})| \leqslant 0,5 \cdot |\Phi(f, A_w)|$, что и требовалось.

Случай 2. Пусть теперь $0,5s < q \leqslant s$. Так как для любого $i \in \{1, 2, \ldots, t\}$ $f_{\zeta_i(\zeta_j^{-1})}(\zeta_j(\tilde{x}^n)) = f_{\zeta_i}(\tilde{x}^n)$, то булевы функции $f_{\zeta_1(\zeta_j^{-1})}$, $f_{\zeta_2(\zeta_j^{-1})}, \ldots, f_{\zeta_{j-1}(\zeta_j^{-1})}, f_{\zeta_j(\zeta_j^{-1})} = f_e = f, f_{\zeta_{j+1}(\zeta_j^{-1})}, \ldots, f_{\zeta_s(\zeta_j^{-1})}$ лежат в $\Phi(f, \zeta_j(A_w))$, а булевы функции $f_{\zeta_{s+1}(\zeta_j^{-1})}, f_{\zeta_{s+2}(\zeta_j^{-1})}, \ldots, f_{\zeta_t(\zeta_j^{-1})}$ не лежат в $\Phi(f, \zeta_j(A_w))$ (при этом, разумеется, $\Gamma(f) = \{f_{\zeta_i(\zeta_j^{-1})}(\tilde{x}^n) \mid i = \overline{1,t}\}$). Заметим, что среди чисел $f_{\zeta_1(\zeta_j^{-1})}(\zeta_j(\tilde{\beta}))$, $f_{\zeta_2(\zeta_j^{-1})}(\zeta_j(\tilde{\beta})), \ldots$, $f_{\zeta_s(\zeta_j^{-1})}(\zeta_j(\tilde{\beta}))$ значение $f(\tilde{\beta})$ встречается q раз, и что

$$f(\zeta_j(\tilde{\beta})) = f_{\zeta_j(\zeta_j^{-1})}(\zeta_j(\tilde{\beta})) = f_{\zeta_j}(\zeta_j^{-1}(\zeta_j(\tilde{\beta}))) = f_{\zeta_j}(\tilde{\beta}) \neq f(\tilde{\beta}).$$

Значит, среди чисел $f_{\zeta_1(\zeta_j^{-1})}(\zeta_j(\tilde{\beta})), f_{\zeta_2(\zeta_j^{-1})}(\zeta_j(\tilde{\beta})), \ldots, f_{\zeta_s(\zeta_j^{-1})}(\zeta_j(\tilde{\beta}))$ значение $f(\zeta_j(\tilde{\beta}))$ встречается $s-q$ раз, то есть, менее чем в половине случаев. Следовательно, полагая $A_{w+1} = \zeta_j(A_w) \cup \{\zeta_j(\tilde{\beta})\}$, получаем и в этом случае, что $|\Phi(f, A_{w+1})| \leqslant 0,5 \cdot |\Phi(f, A_w)|$.

Лемма доказана. ∎

Теорема 2. *Пусть $f(\tilde{x}^n)$ — булева функция, Γ — группа (относительно композиции) биекций на E_2^n. Тогда*

$$l_\Gamma^{\mathrm{check}}(f(\tilde{x}^n)) \leqslant \lceil \log_2 |\Gamma| \rceil.$$

Доказательство. Положим $A_0 = \varnothing$. Тогда $|\Phi(f, A_0)| = |\Gamma(f)| = |\Gamma|$. Применяя лемму 4 последовательно при $w = \overline{0, \lceil \log_2 |\Gamma| \rceil - 1}$, получим, что

$$\Phi(f, A_{\lceil \log_2 |\Gamma| \rceil}) \leqslant \frac{|\Gamma|}{2^{\lceil \log_2 |\Gamma| \rceil}} \leqslant 1,$$

откуда и следует утверждение теоремы. ∎

Из этой теоремы мгновенно вытекает

Теорема 3. *Пусть* Γ — *группа (относительно композиции) биекций на* E_2^n. *Тогда*

$$l_\Gamma^{\text{check}}(n) \leqslant \lceil \log_2 |\Gamma| \rceil.$$

Применением формулы Стирлинга получаем

Теорема 4.

$$l_{\mathfrak{L}}^{\text{check}}(n) \leqslant n \log_2 n(1+o(1)), \qquad l_{\mathfrak{S}}^{\text{check}}(n) \leqslant n \log_2 n(1+o(1)).$$

В работе [3] было показано, что нижней оценкой для функции Шеннона длины проверяющего теста относительно единичных транспозиций переменных в булевой функции, а, следовательно, и для функций Шеннона $l_{\mathfrak{L}}^{\text{check}}(n)$, $l_{\mathfrak{S}}^{\text{check}}(n)$ является величина $0,25n \log_2 n(1+o(1))$. Значит, справедлива

Теорема 5.

$$l_{\mathfrak{L}}^{\text{check}}(n) = \Theta(n \log_2 n), \qquad l_{\mathfrak{S}}^{\text{check}}(n) = \Theta(n \log_2 n).$$

Отметим также в качестве замечания, что из приведенных утверждений выводится также, что порядок функции Шеннона длины условного диагностического теста (т. е. глубины дерева решений для задачи диагностики) относительно действия симметрической группы подстановок или же группы Лоренца на переменные булевой функции равен $\Theta(n \log_2 n)$.

Работа поддержана РФФИ (проекты № 09-01-00817, № 10-01-00768).

Список литературы

[1] *Погосян Г. Р.* О проверяющих тестах для логических схем. — М.: ВЦ АН СССР, 1982.

[2] *Кудрявцев В. Б., Гасанов Э. Э., Долотова О. А., Погосян Г. Р.,* Теория тестирования логических устройств. — М.: Физматлит, 2006.

[3] *Глазунов Н. И., Горяшко А. П.* Об оценках длин обнаруживающих тестов для классов неконстантных неисправностей входов комбинационных схем // Изв. АН СССР. Сер. «Техническая кибернетика». — 1986. — № 3. — С. 197–200.

О синтезе схем, допускающих проверяющие тесты константной длины

Д. С. Романов

romanov@cs.msu.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Пусть $f(\tilde{x}^n)$ — произвольная булева функция, формально зависящая от переменных $x_1, x_2, \ldots, x_n$, а S — схема из функциональных элементов в некотором базисе B, реализующая функцию f, и на схему S действует источник неисправности U (все не введенные в работе определения можно найти в монографии [1]). Источник неисправностей, вызывающий произвольные константные неисправности на выходах функциональных элементов, обозначим через U^c, а источник неисправностей, вызывающий произвольные константные и инверсные неисправности на выходах функциональных элементов, обозначим через $U^{c,\,\mathrm{inv}}$. *Нетривиальной* будем называть такую неисправность схемы S, при которой значение на выходе какого-то элемента E схемы S на некотором входном наборе не равно значению на выходе элемента E при исправной работе схемы S на этом наборе. Множество всех нетривиальных неисправностей схемы S (вызванных источником U) обозначим через $V^U(S)$, а множество всех нетривиальных единичных неисправностей схемы S (вызванных источником U), то есть неисправностей, приводящих к поломке ровно одного элемента схемы, — через $V_1^U(S)$. *Схемной целью контроля* называется некоторое множество Z пар неисправностей, вызываемых источником неисправностей U. Схема S называется *тестопригодной (относительно схемной цели контроля Z для источника неисправностей U)* тогда и только тогда, когда любые две неисправности схемы S, образующие пару из Z, вызывают неравные функционирования схемы S. Обозначим через $W(S)$ (соответственно, через $W_1(S)$) множество всех попарно неравных функций, каждая из которых может быть реализована схемой S в результате нетривиальной неисправности из $V^U(S)$ (соответственно, из $V_1^U(S)$). Множество T наборов значений переменных $x_1, x_2, \ldots, x_n$ называется *полным* (соответственно, *единичным*) *проверяющим тестом для схемы S относительно источника неисправностей U* тогда и только тогда, когда для любой функции $g \in W(S)$ (соответственно, $g \in W_1(S)$) такой, что $g(\tilde{x}^n) \not\equiv f(\tilde{x}^n)$, найдется набор $\tilde{\alpha}$ из T, для которого выполнено неравенство $f(\tilde{\alpha}) \neq g(\tilde{\alpha})$. Количество различных наборов в тесте T называется его *длиной* и обозначается через $l(T)$ или через $|T|$. Тест минимальной длины называется *минимальным*. Сопоставим полному (соответственно, единичному) проверяющему тестированию схемную цель контроля $Z^{\mathrm{full,\,check}}$ (соответственно, $Z^{\mathrm{single,\,check}}$), представляющую собой множество всех пар неисправностей (S_0, S_i), где $S_i \in V^U(S)$ (соответственно, где $S_i \in V_1^U(S)$). Обозначим через $D_U^{\mathrm{full,\,check}}(S)$ (соответственно, через $D_U^{\mathrm{single,\,check}}(S)$) длину минимального полного (соответственно, единичного) проверяющего теста относительно источника неисправностей U в схеме S, через $D_{B,\,U}^{\mathrm{full,\,check}}(f(\tilde{x}^n))$ (соответственно, через $D_{B,\,U}^{\mathrm{single,\,check}}(f(\tilde{x}^n)))$ — минимум величины $D_U^{\mathrm{full,\,check}}(S)$ (соответственно, $D_U^{\mathrm{single,\,check}}(S)$) по всем тестопригодным относительно цели контроля $Z^{\mathrm{full,\,check}}$ (соответственно, относительно цели контроля $Z^{\mathrm{single,\,check}}$) реализующим $f(\tilde{x}^n)$ схемам S в базисе B. Через $D_{B,\,U}^{\mathrm{full,\,check}}(n)$ (соответственно, через $D_{B,\,U}^{\mathrm{single,\,check}}(n)$) обозначим *функцию Шеннона длины полного проверяющего теста относительно источника неисправностей* U, т. е. функцию $D_{B,\,U}^{\mathrm{full,\,check}}(n) = \max\limits_{f(\tilde{x}^n) \in P_2^n} D_{B,\,U}^{\mathrm{full,\,check}}(f(\tilde{x}^n))$ (соответственно, $D_{B,\,U}^{\mathrm{single,\,check}}(n) = \max\limits_{f(\tilde{x}^n) \in P_2^n} D_{B,\,U}^{\mathrm{single,\,check}}(f(\tilde{x}^n))$).

В работе S. M. Reddy [2] было доказано, что в базисе Жегалкина $B_1 = \{x\&y, x \oplus y, 1\}$ функция Шеннона длины единичного проверяющего теста не превосходит $n+3$. Н. П. Редькиным был получен следующий результат: для произвольного полного конечного базиса B была найдена верхняя оценка $D_{B,\,U^c}^{\mathrm{full,\,check}}(n) \leqslant 2(2^{\lceil n/2 \rceil} + 2^{\lfloor n/2 \rfloor} + n)$ [3, 4]. Кроме того, Н. П. Редькиным, С. В. Коваценко, Ю. В. Бородиной и П. А. Бородиным были получены константные верхние оценки функций Шеннона длин проверяющих тестов при различных однотипных неисправностях на выходах функциональных элементов [5, 6, 7, 8, 9]. С. В. Коваценко [5] установил, что в случае $B = \{x\&y, x \oplus y, 1\}$ функция Шеннона длины единичного проверяющего теста относительно инверсных неисправностей на выходах элементов равна 1 (в той же работе для того же класса схем и такого же источника неисправностей установлена верхняя оценка $n+1$ для функции Шеннона длины единичного диагностического теста и верхняя оценка

2^{n-2} для функции Шеннона длины полного диагностического теста). Н. П. Редькин [6] доказал, что в произвольном полном базисе функция Шеннона длины единичного проверяющего теста относительно инверсных неисправностей на выходах элементов не превосходит 3. Ю. В. Бородина установила [7], что в стандартном базисе B_0 функция Шеннона длины полного проверяющего теста относительно однотипных константных неисправностей на выходах элементов равна 2. Ею же доказано [8], что функция Шеннона длины единичного проверяющего теста относительно константных неисправностей типа 1 на выходах элементов в базисе Жегалкина равна 1. Ю. В. Бородина и П. А. Бородин получили [9], что функция Шеннона длины полного проверяющего теста относительно константных неисправностей типа 0 на выходах элементов в базисе Жегалкина равна 1.

Оказывается, верны следующие теоремы.

Теорема 1. *Для базиса* $B' = \{x\&y, x \oplus y, 1, \bar{x}(y \vee z) \vee x(y \sim z)\}$ *при любом* $n \in \mathbb{N}$ *справедливы неравенства:* $2 \leqslant D^{\mathrm{single,\,check}}_{B',\ U^{c,\,\mathrm{inv}}}(n) \leqslant 4$.

Теорема 2. *Существует схемный базис* $\hat{B}$, *содержащий функциональные элементы* $\{1, \bar{x}, x\&y, x \oplus y, x \sim (y \vee z \vee u)\}$ *и группу функциональных элементов с пятью и шестью входами, такой, что для него при любом* $n \in \mathbb{N}$ *имеют место неравенства* $2 \leqslant D^{\mathrm{full,\,check}}_{\hat{B},\ U^{c}}(n) \leqslant 4$.

Идеи доказательств базируются на разложении произвольной булевой функции $f(\tilde{x}^n)$ в полином Жегалкина и на использовании контролирующих блоков специального вида.

Автор выражает глубокую благодарность профессору Сергею Андреевичу Ложкину и к. ф.-м. н. Юлии Владиславовне Бородиной за обсуждение работы и ценные замечания.

Работа поддержана РФФИ (проекты №09-01-00817 и №10-01-00768).

Список литературы

[1] *Редькин Н. П.* Надежность и диагностика схем. — М: Изд-во МГУ, 1992.

[2] *Reddy S. M.* Easily testable realization for logic functions // IEEE Trans. Comput. — 1972. — V. 21, № 1. — P. 124–141.

[3] *Редькин Н. П.* О полных проверяющих тестах для схем из функциональных элементов // Вестник Моск. ун-та. Серия 1. Матем. Механика. — 1986. — № 1. — P. 72–74.

[4] *Редькин Н. П.* О полных проверяющих тестах для схем из функциональных элементов // Математические вопросы кибернетики. Вып. 2. — М.: Наука, 1989. — С. 198–222.

[5] *Коваценко С. В.* Синтез легкотестируемых схем в базисе Жегалкина для инверсных неисправностей // Вестник Моск. ун-та. Серия 15. Вычислит. матем. и киберн. — 2000. — № 2. — С. 45–47.

[6] *Редькин Н. П.* Единичные проверяющие тесты для схем при инверсных неисправностях элементов // Математические вопросы кибернетики. Вып. 12. — М.: Наука, 2003. — С. 217–230.

[7] *Бородина Ю. В.* О синтезе легкотестируемых схем в случае однотипных константных неисправностей на выходах элементов // Вестник Моск. ун-та. Серия 15. Вычислит. матем. и киберн. — 2008. — № 1. — С. 40–44.

[8] *Бородина Ю. В.* О схемах, допускающих единичные тесты длины 1 при константных неисправностях на выходах элементов // Вестник Моск. ун-та. Серия 1. Матем. Механика. — 2008. — № 5. — С. 49–52.

[9] *Бородина Ю. В., Бородин П. А.* Синтез легкотестируемых схем в базисе Жегалкина при константных неисправностях типа „0“ на выходах элементов // Дискретная математика. — 2010. — Т. 22, № 3. — С. 127–133.

NP-полнота задачи о наибольшем кратном потоке

В. С. Рублев, А. В. Смирнов

roublev@mail.ru, alexander_sm@mail.ru

Ярославский государственный университет им. П. Г. Демидова

В данной статье будет рассмотрен вопрос об *NP*-полноте задачи о наибольшем кратном потоке. Понятие кратных сетей и кратных потоков было введено в статье [1] применительно к задаче целочисленного сбалансирования трехмерной матрицы. Дадим несколько определений.

Сначала введем целое $k > 1$, которое назовем *кратностью потока*. В качестве сети рассматривается ориентированный мультиграф $G(X, U)$, между вершинами которого могут быть дуги одного из 3 видов:

1) *обычная дуга* u^o с пропускной способностью $c(u^o)$; множество обычных дуг обозначим через U^o;

2) *кратная дуга* u^k между двумя вершинами, которая состоит из k дуг одной ориентации с одинаковой пропускной способностью и одинаковым потоком по каждой из них; множество кратных дуг обозначим через U^k;
3) *связанная дуга* u между двумя вершинами, которая связана с еще $k-1$ дугой, имеющей одинаковый один из концов; все k дуг имеют одинаковую пропускную способность, и по ним идет одинаковый поток; множество связанных дуг, выходящих из одной вершины или входящих в одну вершину, будем называть *мультидугой* u^m; множество мультидуг обозначим через U^m.

Множество выходящих из вершины дуг может быть либо только кратными дугами, либо только одной мультидугой (k связанных дуг), либо только обычными дугами. Из источника x_0 сети выходят только кратные дуги, а в сток z сети входит только одна мультидуга. Если из вершины выходят связанные дуги мультидуги, то в нее обязательно входит кратная дуга. Если в вершину входит мультидуга, то из нее может выходить только кратная дуга. Определенный таким образом мультиграф $G(X, U)$ с целочисленными пропускными способностями дуг назовем *кратной* (*транспортной*) *сетью*.

Кратным потоком по сети называется целочисленная функция, определенная на множестве дуг $U = U^o \cup U^k \cup U^m$, для которой выполнены условия неотрицательности, ограниченности (пропускными способностями дуг) и неразрывности потока (в каждой вершине). *Величиной кратного потока* называется сумма φ_z входящего потока для стока z, равная сумме выходящего из источника потока. Величина φ_z должна быть кратна k. Обозначим через $c(u)$ пропускную способность дуги u, а через $f(u)$ — поток на ней.

Так же как и в случае обычной транспортной сети, для кратных сетей ставится *задача о нахождении максимального потока*. Поставим также задачу о кратном потоке величины K как задачу распознавания следующего вида: существует ли в данной сети кратности k кратный поток величины не меньшей K? Данную задачу мы в дальнейшем будем называть *задачей КПk*, где k — кратность сети. При такой постановке для положительного ответа на вопрос достаточно указать такой поток в кратной сети, величина которого в точности равна K. Однако этот поток, в точности равный K, указать не всегда возможно, так как если в кратной сети существует поток величины

kT, то в ней не всегда существует поток любой другой величины kS $(1 \leqslant S < T)$.

Теорема 1. *Задача КП2 для сети с целочисленными пропускными способностями дуг является NP-полной.*

Доказательство. Нетрудно убедиться, что задача КП2 для сети с целочисленными пропускными способностями дуг принадлежит классу *NP*.

В работе [2] была обоснована *NP*-полнота задачи целочисленного сбалансирования трехмерной матрицы. Для доказательства этого факта было построено полиномиальное сведение классической задачи о 3-сочетаниях (см. [3]) к следующему сужению класса задач целочисленного сбалансирования (названного задачей ЦС3):

$$m = t = n;\ a_{ijp} \in [0, 1)\ (i \in \overline{1,n},\ j \in \overline{1,n},\ p \in \overline{1,n});$$

$$a_{i00} \leqslant 1\ (i \in \overline{1,n});\ a_{0j0} \leqslant 1\ (j \in \overline{1,n});\ a_{00p} \leqslant 1\ (p \in \overline{1,n}).$$

Построим для произвольной индивидуальной задачи ЦС3 кратную сеть целочисленного сбалансирования (см. [1]). Установим $K = 2n$ и покажем, что задача ЦС3 полиномиально сводится к задаче КП2 с параметром $K = 2n$. При этом существенно, что величина потока в такой сети не может быть больше, чем $2n$, так как пропускная способность мультидуги с концом в z равна $2n$. Следовательно, задача КП2 для этой сети имеет решение тогда и только тогда, когда в сети возможно построить поток величины $2n$. Если существует решение задачи ЦС3, то оно индуцирует поток величины $2n$ в сети целочисленного сбалансирования. Из результатов статьи [1] следует, что любой поток величины $2n$ в кратной сети целочисленного сбалансирования, построенной по задаче ЦС3, индуцирует решение задачи ЦС3, если удовлетворяет условиям:

$$f(x_{i00}, x_{ij0}) + f(x'_{i00}, x_{ij0}) \leqslant c(x_{i00}, x_{ij0}), \quad (i \in \overline{1,n},\ j \in \overline{1,n});$$

$$f(x_{0jp}, x_{0j0}) + f(x_{0jp}, x'_{0j0}) \leqslant c(x_{0jp}, x_{0j0}), \quad (j \in \overline{1,n},\ p \in \overline{1,n});$$

$$f(x_{i0p}, x_{00p}) + f(x_{i0p}, x'_{00p}) \leqslant c(x_{i0p}, x_{00p}), \quad (i \in \overline{1,n},\ p \in \overline{1,n}).$$

Для потока величины $2n$ в кратной сети целочисленного сбалансирования, построенной по задаче ЦС3, данные условия выполняются всегда, следовательно, по этому потоку можно построить решение задачи ЦСЗ. Построение сети по матрице производится с помощью полиномиального алгоритма, все остальные операции, указанные выше, также полиномиальны, следовательно, мы произвели полиномиальное сведение задачи ЦСЗ к задаче КП2, и задача КП2 является *NP*-полной. ∎

Теорема 2. *Задача нахождения максимального кратного потока в сети кратности k с целочисленными пропускными способностями дуг является NP-полной, если $k \geqslant 2$.*

Доказательство. Для доказательства утверждения теоремы покажем сначала *NP*-полноту задачи КПk для произвольного фиксированного $k \geqslant 2$. Заметим, что задача КПk для сети с целочисленными пропускными способностями дуг принадлежит классу *NP*. Покажем полиномиальную сводимость задачи КП2, построенной по произвольной индивидуальной задаче ЦСЗ, к задаче КПk. Установим $K = kn$ и изменим сеть следующим образом.

1. Добавим вершины x_q, z_q и обычные дуги (x_q, z_q) $(q \in \overline{3, k})$. Установим пропускную способность этих дуг равной n.
2. К каждой кратной дуге добавим $k-2$ связанных дуг; пропускная способность каждой кратной дуги увеличится, соответственно, в $\frac{k}{2}$ раз.
3. Заменим мультидугу с концом в z на мультидугу $(\{z_1, \ldots, z_k\}, z)$ пропускной способности kn (каждая связанная дуга этой мультидуги имеет пропускную способность n).
4. Мультидуги вида $(x_{ijp}, \{x_{0jp}, x_{i0p}\})$ заменим на мультидуги вида $(x_{ijp}, \{x_{0jp}, x_{i0p}, x_3, \ldots, x_k\})$ $(i > 0,\ j > 0,\ p > 0)$ пропускной способности k (каждая связанная дуга этих мультидуг имеет пропускную способность 1).

Очевидно, что после выполнения указанных операций сеть будет иметь кратность k, а решение задачи КП2 с $K = 2n$ будет эквивалентно решению задачи КПk с $K = kn$. Операции шагов 1–4 выполняются за полиномиальное количество шагов. При этом существенно, что величина потока в такой сети не может быть больше, чем kn, так как пропускная способность мультидуги с концом в z равна kn. Следовательно, задача КПk для этой сети имеет решение тогда и только тогда, когда в сети возможно построить поток величины kn.

Таким образом, мы выполнили полиномиальное сведение задачи КП2, построенной по произвольной индивидуальной задаче ЦС3, к задаче КПk. В теореме 1 была доказана NP-полнота такой задачи КП2, следовательно, задача КПk также является NP-полной.

Задача КПk может быть решена при помощи следующего алгоритма:

1) нахождение максимального кратного потока;
2) проверка $F' \leqslant K$ (соответствует ответу «да» в задаче КПk), где F' — величина этого максимального потока.

Проверка шага 2 выполняется за константное время, а задача КПk NP-полна. Следовательно, задача нахождения максимального кратного потока в сети кратности k с целочисленными пропускными способностями дуг также NP-полна. ∎

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг., ГК № П161.

Список литературы

[1] *Рублев В. С., Смирнов А. В.* Задача целочисленного сбалансирования трехмерной матрицы и алгоритмы ее решения // Моделирование и анализ информационных систем. — 2010. — Т. 17, № 2. — С. 72–98.

[2] *Рублев В. С., Смирнов А. В.* NP-полнота задачи целочисленного сбалансирования трехмерной матрицы // ДАН. — 2010. — Т. 435, № 3. — С. 314–316.

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

Оптимизация вычислений объектных запросов системы управления данными DIM

В. С. Рублев, Е. А. Смирнова

roublev@mail.ru, smrn.ekaterina@gmail.com

Ярославский государственный университет им. П. Г. Демидова

Постановка задачи

Описанный в [1] язык объектно-динамических запросов ODQL для динамической информационной модели DIM, концепции которой приведены в [2], обладает полнотой — любая группа объектов DIM (вместе с любыми их свойствами) может быть выделена ODQL-запросом (см. [3]). Рассматриваются вопросы организации информации, позволяющей построить алгоритм вычисления объектного запроса с минимальным (в эвристическом смысле) *временем отклика*.

Каждый запрос согласно [1] определяет *базовый класс* объектов запроса (по первому классу фразы **from** или по классу первого свойства класса **select** при отсутствии фразы **from**), множества классов связанных с ними объектов (по фразам **from, select, links**), а также множество ограничений на выделяемые объекты (по фразам **for, where**). Для эффективной реализации такого запроса нужно прежде всего выделить схему классов объектов, отвечающую запросу (алгоритм выделения такой *схемы слоев классов* запроса из гиперграфа классов и гиперграфа объектов системы DIM приведен в [3]), а затем использовать эту схему для возможно меньшего по трудоемкости вычисления запроса. Вот здесь-то и требуется организация данных системы DIM, позволяющая это сделать.

В реляционной технологии для этого вводятся конструкции индексов таблиц и внешних связей таблиц. Мы также введем в систему DIM подобные конструкции, использование которых делает возможным оптимизацию выполнения запроса по трудоемкости.

Индекс-выборки классов и отношений классов

Подобно индексам реляционных таблиц введем индексы по любому набору свойств класса для множества его объектов, каждый из которых реализуется в виде B^*-дерева. Узел такого дерева содержит указатель на объект класса и ключ, соответствующий значениям идентификационных свойств для этого объекта. Каждый объект, класс объектов и свойство класса снабжается объектным идентификатором — уникальным натуральным ключом, характеризующим эту сущность. Главный индекс объектов класса, создающийся всегда, является индексом по объектным идентификаторам объектов класса.

Помимо B^*-дерева главный индекс сдержит указатель на двунаправленный *список выборки объектов класса* и пометку списка: 0 — список выборки пустой, 1 — список выборки содержит не все объекты класса, 2 — список выборки содержит все объекты класса. Этот главный индекс и представляет собой *индекс-выборку* класса. Каждый элемент списка выборки содержит указатель на узел этого дерева, объект которого выбран, а также указатель на предыдущий и последующий элементы списка, а каждый узел дерева содержит указатель на соответствующий элемент списка выборки.

Инициализация списка выборки объектов производится в начале выполнения запроса по фразе **for** для каждого класса схемы слоев.

Для каждого отношения классов устанавливаются индексы (B^*-деревья) отношений. Они играют роль взаимных внешних индексов множеств классов, находящихся в отношении. Помимо B^*-дерева индекс отношения содержит указатель на двунаправленный *список выборки объектов, находящихся в соответствующем отношении объектов*, и пометку списка: 0 — список выборки пустой, 1 — список выборки содержит не все объекты класса, 2 — список выборки содержит все объекты класса. Этот индекс отношения и представляет собой *индекс-выборку* отношения классов.

Каждый элемент списка выборки отношения классов содержит указатель на узел этого дерева, объекты отношения которого выбраны, а также указатель на предыдущий и последующий элементы списка, а каждый узел дерева содержит указатель на соответствующий элемент списка выборки (пустое значение, если отношение объектов этого узла не входит в выборку) и указатели на соответствующие объектам узлы индекс-выборок их классов. Для отношений *наследования, включения и истории* (см. [2]) устанавливаются по 2 индекса, определенных порядком объектных идентификаторов классов отношения. Для отношения *взаимодействия* (см. там же) устанавливается 6 индексов с учетом порядка идентификаторов объектов 4 классов отношения. Каждый узел любого из этих отношений содержит указатели на соответствующие узлы индекс-выборок классов отношения.

Алгоритм выполнения запроса

Идея алгоритма состоит в том, что для каждого класса, участвующего в запросе, создается индекс-выборка. Затем, двигаясь по схеме слоев от слоя с максимальным уровнем к нулевому (к базовому классу), мы устанавливаем индекс-выборки каждого из отношений и корректируем индекс-выборки классов отношения. В результате по индекс-выборке базового класса получаем выборку его объектов, каждый из которых удовлетворяет условиям запроса. И, наконец, двигаясь по схеме слоев от нулевого слоя к внешнему слою, определяем как результат наборы свойств запроса, удовлетворяющие его условиям.

Общий алгоритм

1. Для каждого класса фразы **from** проводится начальная установка его индекс-выборки, как это описано выше. При этом трудоемкость начальной установки оценивается как $\theta(n)$, где n — число объектов всех классов запроса.
2. По фразе **links** запроса проводится коррекция всех индекс-выборок классов, участвующих в запросе, и установка индекс-выборок отношений этих классов. Для этого выбираются по очереди отношения классов, начиная с конца фразы. Пусть A — класс внешнего слоя отношения, B — класс внутреннего слоя, C — класс связи, участвующий в отношении включения, или классы C, H — классы, участвующие в отношении взаимодействия (класс H в роли *Как*, а остальные в других ролях). При этом класс связи C может иметь в общем случае k родительских классов $P_1, \dots P_k$ $(k \geqslant 0)$. В зависимости от вида отношения выполняем *алгоритм коррекции индекс-выборок*. Его трудоемкость $\theta(n_D)$ для отношения наследования (n_D — число элементов дочернего класса), $\theta(n_A \cdot n_B \cdot n_{P_1} \cdot ... \cdot n_{P_k})$ для отношения включения и для отношения истории при $k = 0$, $\theta(n_A \cdot n_B \cdot n_C \cdot n_{P_1} \cdot ... \cdot n_{P_k})$ для отношения взаимодействия.
3. Для каждого объекта индекс-выборки базового класса применяем описанный ниже *алгоритм обхода с возвратом схемы слоев* для получения всех деревьев связанных объектов запроса и для

каждого такого дерева выбираем по объектам, входящим в него, значения свойств фразы **select**.

Трудоемкость алгоритма обхода с возвратом схемы слоев является в общем случае экспоненциальной, так как алгоритм обхода дерева слоев является переборным. Например, если схема слоев содержит m вершин-классов, а индекс-выборка каждого класса после использования алгоритма коррекции индекс-выборок содержит не более p объектов, то трудоемкость алгоритма обхода схемы слоев можно оценить как $\theta(m^p)$. Однако так как сложность выбора каждого кортежа значений запроса в данном алгоритме оценивается как $\theta(m)$, то при выборе запросом k кортежей значений и предположении, что ограничение **where** отсекает число кортежей порядка $\theta(k)$, трудоемкость такого запроса оценивается как $\theta(m \cdot k)$.

Проблемы реализации

Поскольку в организации вычислений запроса основную роль играют индекс-выборки классов и отношений классов, а узлы индекс-выборок содержат не данные, а указатели для них, то данные могут размещаться в динамической памяти (типа *кучи*). Для определения лучшей платформы реализации в настоящее время разработан тестовый пример и выбраны для проведения статистических испытаний 3 возможные платформы: Oracle, C++, Java.

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России», ГК № 02.740.11.0207.

Список литературы

[1] *Рублев В. С.* Язык объектных запросов динамической информационной модели DIM // Моделирование и анализ информационных систем. — 2010. — Т. 17, № 3. — С. 144–161.

[2] *Писаренко Д. С., Рублев В. С.* Объектная СУБД «Динамическая информационная модель DIM и ее основные концепции» // Моделирование и анализ информационных систем. — 2009. — Т. 16, № 1. — С. 62–91.

[3] *Рублев В. С.* Запросная полнота языка ODQL динамической информационной модели DIM // Ярославский педагогический вестник. Серия Физико-математические и естественные науки. — 2011. — Вып. 5. — С. 54–62.

Уточненные оценки функции Шеннона в некоторых базисах схем из функциональных элементов, вложенных в единичный куб

О. А. Садовников

oleg.a.sadovnikov@gmail.com

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Рассматривается задача реализации произвольной функции алгебры логики (ФАЛ) в классе схем из функциональных элементов (СФЭ), вложенных в n-мерный единичный куб B^n. Ранее данная задача была рассмотрена, в частности, О. Б. Седелевым в работах [2, 3].

В данной работе, как и в указанных выше, рассматривается обычный класс СФЭ со специальным функционалом сложности, связанным с «геометрической» реализацией СФЭ. Заметим, что подобный «геометрический» подход к сложности ранее уже применялся в традиционных классах схем. В частности, известным примером подобного применения являются клеточные схемы, где СФЭ размещается в плоской прямоугольной решётке, а под сложностью схемы понимается площадь указанной решётки.

Выберем в качестве структуры геометрической реализации n-мерный единичный куб B^n. Под «геометрической реализацией» СФЭ Σ в кубе B^n будем понимать её квазигомеоморфное вложение в B^n, то есть отображение, сопоставляющее каждой вершине v СФЭ Σ некоторую вершину w куба B^n (образ вершины v), а максимальному по включению пучку $E(v)$ дуг, исходящих из вершины v, — соответствующее ориентированное («транзитное») дерево в кубе B^n с корнем в вершине w куба B^n, такое, что множество его листьев совпадает с множеством вершин куба, являющихся образами концов дуг из пучка $E(v)$. При этом «транзитные» деревья не могут иметь общих внутренних вершин. Отметим, что такое вложение является обобщением известного гомеоморфного вложения.

Пусть Σ — некоторая СФЭ. Тогда под её сложностью $R(\Sigma)$ будем понимать минимальную размерность n-мерного единичного куба B^n, допускающего квазигомеоморфное вложение Σ.

Традиционно, через $P_2(n)$ обозначим множество всех ФАЛ, зависящих от n переменных $x_1, \ldots, x_n$. Как обычно, введём функционал

$R_Б(f)$ сложности ФАЛ $f \in P_2(n)$ в конечном полном базисе Б и положим его равным $R(\Sigma_f^Б)$, где $\Sigma_f^Б$ — СФЭ минимальной сложности $R(\Sigma_f^Б)$, реализующая ФАЛ f в базисе Б.

Наконец, обычным образом введём функцию Шеннона $R_Б(n)$, равную сложности реализации «самой сложной» ФАЛ из $P_2(n)$:

$$R_Б(n) = \max_{f \in P_2(n)} R_Б(f).$$

Для указанной функции Шеннона О. Б. Седелевым в работе [2] была получена верхняя оценка для $R_{Б_0}(n)$ в стандартном базисе $Б_0 = \{\&, \vee, \neg\}$:

$$R_{Б_0}(n) \leqslant n - \lfloor \log \log n \rfloor + 8. \tag{1}$$

В работе [3] была получена такая же верхняя оценка для так называемого мультиплексорного базиса $Б_\mu = \{\mu(x, y_0, y_1), 0, 1\}$, состоящего из мультиплексорной ФАЛ $\mu(x, y_0, y_1) = \bar{x}y_0 \vee xy_1$ и двух константных ФАЛ 0 и 1:

$$R_{Б_\mu}(n) \leqslant n - \lfloor \log \log n \rfloor + 8. \tag{2}$$

Основным результатом данной работы являются улучшенные по сравнению с (1) и (2) верхние оценки для функции Шеннона в базисах $Б_0$ и $Б_\mu$.

При построении требуемых схем для получения улучшенных верхних оценок функции Шеннона применяется специальный метод [4] гомеоморфного (а следовательно — и квазигомеоморфного) вложения полного двоичного дерева $\mathcal{D}_n$ глубины n в единичный куб размерности $(n+2)$, сохраняющий соседство поддеревьев. При таком вложении выполняется следующее условие: два любых «соседних» полных поддерева одинаковой глубины в дереве $\mathcal{D}_n$ вкладываются в соседние подкубы одинаковой размерности в кубе B^{n+2} (под «соседними» здесь понимаются поддеревья, корни которых имеют общую вершину-предка в $\mathcal{D}_n$).

Данный метод вложения не является оптимальным по размерности (известно [1], что $\mathcal{D}_n$ можно гомеоморфно вложить в B^{n+1}), однако в ряде случаев он упрощает вложение схемы. В частности, с его помощью легко доказать следующий факт:

Теорема 1. *Пусть* $\mathbf{Q}_n(x_1, \ldots, x_n)$ *— система всех ФАЛ от переменных* $x_1, \ldots, x_n$ *вида* $x_1^{\sigma_1}, \ldots, x_n^{\sigma_n}$*, где* $\sigma_i \in \{0, 1\}, i = \overline{1, n}$*, то есть система всех элементарных конъюнкций ранга* n *от* n *переменных. Тогда* $R_{Б_0}(\mathbf{Q}_n) \leqslant n + 4$.

Доказательство. Построим подходящую схему Σ в базисе $Б_0 = \{\&, \vee, \neg\}$ с n входами $(x_1, \ldots, x_n)$ и 2^n выходами, реализующую систему из 2^n ФАЛ $\mathbf{Q}_n$. Подключим к каждому входу схемы элемент отрицания. Обозначим через $a_i, i = \overline{1, n}$, элемент $\{\&\}$, один вход которого подсоединен к i-му входу схемы Σ, а через $b_i, i = \overline{1, n}$, — элемент $\{\&\}$, один вход которого подсоединен к соответствующему элементу отрицания.

Пусть $\mathcal{D}_{n+1}$ — полное двоичное дерево глубины $(n + 1)$. Разместим в его вершинах элементы схемы Σ типа a_i и b_i, $i = \overline{1, n}$, следующим образом. В каждом i-м ярусе $(i = \overline{1, n})$ расположим элементы a_i и b_i так, чтобы корни любых двух соседних поддеревьев в дереве $\mathcal{D}_{n+1}$ содержали элементы различного типа. Для корректности удалим элементы a_1 и b_1 из первого яруса и отождествим вершины первого яруса с входом x_1 и соответствующим ему элементом отрицания.

Легко видеть, что на элементах n-го яруса дерева реализуются все 2^n ФАЛ из системы $\mathbf{Q}_n$, следовательно, требуемая схема Σ построена. Отметим, что дерево $\mathcal{D}_{n+1}$ указанным выше методом можно вложить в единичный куб размерности $(n + 3)$.

Рассмотрим единичный куб B^{n+4}. Выделим в нем 2 подкуба K_1 и K_2 размерности $(n + 3)$. Вложим в подкуб K_1 дерево $\mathcal{D}_{n+1}$. Отметим, что из условия соседства поддеревьев следует, что в подкубе K_1 можно выделить два подкуба L_1 и L_2 размерности $(n+2)$, такие, что все элементы типа a_i попадут в подкуб L_1, а все элементы типа b_i — в подкуб L_2 $(i = \overline{1, n})$.

Выделим в подкубе K_2 подкубы M_1 и M_2, параллельные подкубам L_1 и L_2 в подкубе K_1 соответственно. Из условия соседства поддеревьев, для любого $i = \overline{1, n}$ вершины в кубе M_1, соответствующие вершинам типа a_i в кубе L_1, можно соединить цепью C_i. Проведем эту цепь, разместим в произвольной ее вершине вход x_i схемы Σ и соединим все вершины цепи C_i с соответствующими им вершинами типа a_i в кубе L_1, обеспечив тем самым подвод переменной x_i ко всем элементам типа a_i. В кубе M_2 проведем аналогичную цепь

C_i', в произвольной ее вершине разместим элемент отрицания, соответствующий входу x_i, и соединим эту вершину с соответствующей вершиной цепи C_i — это возможно из того же условия соседства поддеревьев.

Проделаем аналогичные построения для всех $i = \overline{2, n}$. В результате будет получено требуемое вложение схемы Σ в куб B^{n+4}. ∎

Применение описанной здесь техники позволяет получить следующие улучшенные (по сравнению с (1) и (2)) верхние оценки для функции Шеннона в классе СФЭ, вложенных в единичный куб:

Теорема 2. $R_{Б_0}(n) \leqslant n - \lfloor \log\log n \rfloor + 7$.

Теорема 3. $R_{\mu}(n) \leqslant n - \lfloor \log\log n \rfloor + 4$.

Работа выполнена при поддержке РФФИ, проект 09-01-00817-а.

Список литературы

[1] *Ложкин С. А., Седелев О. Б.* О реализации функций алгебры логики BDD, вложенными в единичный куб // Вестник Московского университета. Сер. 15. Вычисл. матем. и киберн. — 2006. — № 4. — С. 29–35.

[2] *Седелев О. Б.* О реализации функций алгебры логики схемами из функциональных элементов, вложенными в единичный куб // Вестник Московского университета. Сер. 15. Вычисл. матем. и киберн. — 2008. — № 1. — С. 44–50.

[3] *Седелев О. Б.* О сложности реализации функций алгебры логики схемами из функциональных элементов в мультиплексорном базисе, вложенными в единичный куб // Тезисы XV Межд. школы-семинара «Синтез и сложность управляющих систем». — Новосибирск: Изд-во Ин-та матем. им. С. Л. Соболева, 2004.

[4] *Садовников О. А.* О реализации функций алгебры логики схемами с подведением переменных, вложенными в единичный куб // Материалы XVIII Межд. школы-семинара «Синтез и сложность управляющих систем». — М.: Изд-во механико-математического факультета МГУ, 2009.

К вопросу о числе совершенных кодов

А. А. Сапоженко

sapozhenko@mail.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

В статье Ю. Л. Васильева [1] получена нижняя оценка числа C_n двоичных *плотно упакованных кодов* длины n с расстоянием 3, называемых также *совершенными*. Ряд авторов конструктивными средствами улучшали эти оценки. По-видимому, наилучшая к настоящему времени нижняя оценка получена С. А. Малюгиным в [2]. Она имеет вид

$$C_n \geqslant 2^{2^{(n+1)/2-\log_2(n+1)}} \cdot 2^{2^{(n-2)/4}}.$$

С. В. Августинович в [3] получил верхнюю оценку вида

$$C_n < 2^{\left(1+O(1/n)\right)2^{n-\frac{3}{2}\log n+\log\log(en)+d}}, \tag{1}$$

где $d = \log_2 \sqrt{\frac{8}{\pi}}$. В статье [3] слагаемое d отсутствует, хотя из хода доказательства следует, что оно должно быть.

Цель данной публикации состоит в некотором улучшении этой верхней оценки на основе использования оценок числа независимых множеств, полученных в [4, 5]. Через B^n будем обозначать единичный n-мерный куб. В [3] дано следующее определение. Подмножество $M \subseteq B^n$ называется *базовым*, если для любых двух совершенных кодов K_1 и K_2 найдется вершина α, такая, что $\alpha \in K_1$ и $\alpha \notin K_2$. Ясно, что по множеству $M \cap K$ однозначно определяется совершенный код K. Обозначим через $\widetilde{M}$ множество $B^n_{(n-1)/2} \cup B^n_{(n+1)/2}$. В [3] доказана следующая

Теорема 1. *Множество $\widetilde{M}$ является базовым.*

Приведенная выше верхняя оценка из [3] получается в качестве следствия теоремы 1 и следующего равенства из известной статьи Г. С. Шапиро и Д. Л. Злотника [6]

$$|K \cap \widetilde{M}| = \frac{1}{(n+1)}\left(\binom{n+1}{(n+1)/2} + n\binom{(n+1)/2}{(n+1)/4}\right). \tag{2}$$

Из (2) при достаточно больших n следует, что

$$|K \cap \widetilde{M}| \leqslant |\widetilde{M}|/n. \tag{3}$$

Из (2) и (3) вытекает (1).

Идея понижения верхней оценки из [3] состоит в использовании того факта, что $K \cap \widetilde{M}$ является максимальным независимым множеством в графе $G_n = (\widetilde{M}, E)$, с множеством ребер $E = \{\{\alpha, \beta\} : \{\alpha, \beta\} \subseteq \widetilde{M}, 1 \leqslant d(\alpha, \beta) \leqslant 2\}$. То есть вместо произвольных множеств из $\widetilde{M}$, которые подсчитывались в [3], мы будем учитывать только те, что являются независимыми. Будут использоваться утверждения из статей [4, 5]. В [4] доказана следующая

Теорема 2. *Пусть G — регулярный степени k граф на n вершинах с числом независимости α. Тогда число $i(G)$ его независимых множеств удовлетворяет неравенству*

$$i(G) \leqslant (1 + n/2\alpha)^{\alpha} \cdot 2^{n\sqrt{\log k}/k}. \tag{4}$$

Замечание. Для всякого кода K в B^n множество $K \cap \widetilde{M}$ является максимальным независимым множеством в графе G_n. В свою очередь, G_n является регулярным степени $\binom{(n+1)}{2}$ графом на $|\widetilde{M}| = 2\binom{n}{(n-1)/2}$ вершинах с числом независимости, не превосходящим $2|\widetilde{M}|/(n+3)$. В статье В. Е. Алексеева [5] доказана следующая

Теорема 3. *Для всякого n-вершинного графа G с числом независимости α справедливо неравенство*

$$i(G) \leqslant \left(\frac{n}{\alpha} + 1\right)^{\alpha}. \tag{5}$$

Из теорем 1 и 3 вытекает

Теорема 4. *При достаточно больших n выполняется неравенство*

$$C_n \leqslant 2^{\left(1+O(1/n)\right)2^{n-\frac{3}{2}\log n+\log\log(n+1)+d}}. \tag{6}$$

Доказательство. С учетом сказанного выше и применением неравенства (5) имеем

$$C_n \leqslant (1+n)^{|\widetilde{M}|/n} \leqslant (n+1)^{\binom{n+1}{(n+1)/2}/n} =$$

$$= 2^{\left(\sqrt{\frac{2}{\pi}}+O\left(\frac{1}{n}\right)\right)\frac{2^{n+1}}{\sqrt{(n+1)n}}\log(n+1)} \leqslant 2^{\left(1+O(1/n)\right)2^{n-\frac{3}{2}\log n+\log\log(n+1)+d}}.$$

∎

Оценка (6) лучше оценки Августиновича, но вытекающая из (6) оценка для величины $\log\log C_n$ асимптотически отличается от той, что следует из (1), лишь слагаемым порядка $1/\log\log n$. Применение (4) вместо (5) не приводит к улучшению оценки (6). Хотя главный член асимптотики для $\log i(G)$ в (4) лучше оценки Алексеева (5), остаточный член в (4) оказывается существенным.

Другая идея улучшения верхней оценки C_n состоит в использовании оценки числа $t(G)$ максимальных независимых множеств графа G из [5]. Ребро графа называется *доминирующим*, если любая вершина графа смежна хотя бы с одним из концов этого ребра. Пусть m_1 — число доминирующих ребер, m_0 — число ребер, не являющихся доминирующими, а $p = p(G)$ — максимальный размер порожденного паросочетания в графе G. В [5] доказана следующая

Теорема 5. *Для всякого графа G*

$$t(G) \leqslant \left(\frac{m_0}{p} + 1\right)^p + m_1.$$

В определенном выше графе G_n доминирующие ребра отсутствуют при $n \geqslant 3$, т. е. $m_1 = 0$. При условии, что $p = p(G_n) = \Omega(|M|/n)$, понижения верхней оценки не произойдет. Если окажется, что $p = p(G_n) = \Omega(|M|/n^\alpha)$, где $\alpha > 1$, то удастся увеличить константу при отрицательном втором члене для $\log C_n$. По-видимому, дальнейшее понижение верхней оценки для C_n с использованием понятия базового множества следует искать на пути понижения мощности последнего.

Поддержано РФФИ, грант 10-01-00768-а.

Список литературы

[1] *Васильев Ю. Л.* О негрупповых плотно упакованных кодах // Проблемы кибернетики. — 1962. — № 8. — С. 337–339.

[2] *Малюгин С. А.* О нижней оценке числа совершенных двоичных кодов // Дискрет. анализ и исслед. опер. Сер. 1. — 1999. — Т. 6, № 4. — С. 44–48.

[3] *Августинович С. В.* Об одном свойстве совершенных двоичных кодов // Дискрет. анализ и исслед. опер. Сер. 1. — 1995. — Т. 2, № 1. — С. 4–6.

[4] *Сапоженко А. А.* Верхняя оценка числа независимых множеств в графах // ДАН. — 2007. — Т. 414, № 6. — С. 1–3.

[5] *Алексеев В. Е.* Верхняя оценка числа максимальных независимых множеств графа // Дискретная математика. — 2007. — Т. 19, № 2. — С. 84–89.

[6] *Shapiro H. S., Slotnik D. L.* On the mathematical theory error correcting codes // IBM J. of Res. and Devel. — 1959. — V. 3, № 1. — P. 84–89.

О взаимодействии мобильных агентов с топологической средой

С. В. Сапунов

sapunov sv@iamm.ac.donetsk.ua

Институт прикладной математики и механики НАН Украины, Донецк

В работе рассматривается задача определения мобильным агентом (МА) своего положения в среде моделируемой графом с помеченными вершинами. Эта задача относится к проблематике взаимодействия управляющей и управляемой систем, являющейся классической для теоретической кибернетики [1, 2]. В настоящее время эта проблема актуальна в связи с задачами навигации автономных мобильных роботов [3].

Графы с помеченными вершинами

Конечным графом с помеченными вершинами (помеченным графом) назовем четверку $G = (V, E, M, \mu)$, где V, E, M — конечные множества вершин, ребер и меток соответственно, $\mu : G \to M$ — сюръективная функция разметки. Помеченный граф G назовем детерминированным (Д-графом), если в множестве преемников любой его вершины все вершины помечены различно. Д-граф G назовем сильно детерминированным (СД-графом), если в замкнутой окрестности любой его вершины все вершины помечены различно.

Неотличимость вершин

Последовательность меток вершин $w = \mu(g_1)\ldots\mu(g_k)$, соответствующую некоторому пути $g_1 \ldots g_k$ в графе G, назовем словом длины k, порожденным вершиной g_1. Языком L_g вершины g назовем множество всех слов, порожденных этой вершиной. Будем говорить, что вершины $g, h \in V$ ε-неотличимы, если $L_g = L_h$. Лингвистическим идентификатором (ЛИ) вершины $g \in V$ назовем конечное множество слов $W_g \subseteq M^+$ т.ч. для любой вершины $h \in V$ равенство $W_g \cap L_g = W_g \cap L_h$ выполняется тогда и только тогда, когда $g = h$.

Через S_g обозначим подграф графа G, порожденный всеми вершинами, достижимыми из вершины $g \in V$. Будем говорить, что вершины $g, h \in V$ σ-неотличимы, если $S_g \cong S_h$. Пусть G_g и H_h являются инициально-связными помеченными графами с выделенными вершинами g и h соответственно. Обозначим через $G_g \cap H_h$ операцию нахождения наибольшего связного подграфа $G'_g \subseteq G_g$, содержащего выделенную вершину g, такого, что G'_g изоморфно вложим в H_h, причем вершина g отображается в вершину h. Топологическим идентификатором (ТИ) вершины $g \in V$ назовем помеченный граф D_g т.ч. для любой вершины $h \in V$ изоморфизм $D_g \cap S_g \cong D_g \cap S_h$ существует тогда и только тогда, когда $g = h$.

Показано, что $\sigma \subseteq \varepsilon$, причем обратное включение не выполняется. Предложены полиномиальные методы построения ЛИ и ТИ вершин помеченных графов. Показано, что гомоморфный образ растущего помеченного дерева, соответствующего ЛИ W_g вершины $g \in V$ является ТИ этой вершины. Показано, что обратное утверждение в общем случае неверно.

Эксперименты с графами

Экспериментом с графом G относительно априорной информации I, цели C и средств S назовем процесс, состоящий из трех этапов: 1) построение теста P на основе I и C; 2) получение МА экспериментальных данных W на основе P и S; 3) вывод заключений о свойствах графа на основе W и I. Априорная информация — это класс графов, к которому принадлежит G. В качестве S выступают возможности МА перемещаться по графу, воспринимать локальную информацию о вершинах и ставить в них дополнительные стираемые/нестираемые метки. Эксперимент назовем диагностическим, если априори полностью известен граф G и МА установлен в произвольную начальную вершину этого графа, а целью является определение этой вершины, т. е. отличение этой вершины от всех других вершин. Из определения эксперимента следует, что он осуществляется посредством прохождения МА некоторого связанного с тестом P множества путей по графу из начальной вершины.

В работах [4, 5] автором были предложены методы построения и реализации диагностических экспериментов с помеченными графами основывающиеся на проверке ε-эквивалентности вершин при помощи их ЛИ. Целью данной работы является создание аналогичных методов, основывающихся на ТИ вершин.

Диагностический эксперимент

Прямой суммой помеченных графов G и H назовем помеченный граф $G + H$, полученный объединением множеств вершин и ребер этих графов (с предварительным переобозначением вершин так, чтобы исходные графы не имели общих вершин). Соединением инициально-связных помеченных графов G_g и H_h назовем инициально-связный помеченный граф $\langle G_g + H_h \rangle$, полученный из графов G_g и H_h отождествлением их начальных вершин и последующей детерминизацией, т. е. многократным и исчерпывающим применением следующей операции: если в множестве преемников некоторой вершины оказываются одинаково отмеченные вершины, то такие вершины отождествляются с заменой возникающих кратных дуг одной дугой.

Следующая теорема дает метод построения диагностических тестов для СД-графов.

Теорема 1. *Граф $\left\langle \sum_{g \in V} D_g \right\rangle$ является диагностическим тестом для графа G для любого множества ТИ $\{D_g\}_{g \in V}$.*

МА может перемещаться по ребрам графа от вершины к вершине, оставлять маркер в текущей вершине, а также обнаруживать и подбирать маркер в случае его нахождения в текущей вершине. Находясь в вершине, МА считывает ее метку и метки смежных с ней вершин.

1-й этап эксперимента состоит в построении тестового графа P по множеству ТИ всех вершин графа G. Получение экспериментальных данных заключается в том, что МА, стартуя из неизвестной ему вершины h графа G, проверяет наличие/отсутствие в G путей, совпадающих по разметке с путями обхода в ширину графа P из его инициальной вершины. В зависимости от исхода каждой из этих проверок сокращается множество гипотетически возможных начальных вершин. По окончании работы алгоритма остается ровно одна такая вершина. Показано, что временная сложность предложенного алгоритма проведения диагностического эксперимента полиномиальна от числа вершин исследуемого графа.

Список литературы

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

[2] *Капитонова Ю. В., Летичевский А. А.* Математическая теория проектирования вычислительных систем.— М.: Наука, 1988.

[3] *Dudek G., Jenkin M.* Computational Principles of Mobile Robotics.— Cambridge: Cambridge University Press, 2000.

[4] *Сапунов С. В.* Определение положения робота в топологической среде // Искусственный интеллект. — 2008. — Т. 4. — С. 558–565.

[5] *Грунский И. С., Сапунов С. В.* Идентификация вершин помеченных графов // Труды ИПММ НАН Украины. — 2010. — Т. 21. — С. 86–97.

Максимальная мощность (k, l)-множества свободного от сумм в циклической группе

В. Г. Саргсян

vahe_sargsyan@yahoo.com

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Подмножество $A \subseteq Z_n$ называется (k, l)-*множеством, свободным от сумм* (МСС), если уравнение $x_1+x_2+\cdots+x_k = y_1+y_2+\cdots+y_l$ не имеет решения в A. Положим $kA = \{x_1+x_2+\cdots+x_k \mid x_1, \ldots, x_k \in A\}$, а $c \star A = \{c \cdot a \mid a \in A\}$ для любого $c \in Z_n$. Определение (k, l)-множества, свободного от сумм, эквивалентно тому, что $kA \cap lA = \varnothing$, которое, в свою очередь, эквивалентно тому, что $0 \notin kA - lA = \{x_1 + x_2 + \cdots + x_k - y_1 - y_2 - \cdots - y_l \mid x_1, \ldots, x_k, y_1, \ldots, y_l \in A\}$. Через $\lambda_{k,l}(Z_n)$ обозначим максимальную мощность (k, l)-множества, свободного от сумм в Z_n, а через $\alpha_{k,l}(Z_n)$ — максимальную мощность (k,l)-арифметической прогрессии, свободной от сумм в циклической группе Z_n. Предположим, что $k \neq l \,(\mathrm{mod}\; n)$ и $k > l$.

Теорема 1 [1]. *Для любого n*

$$\lambda_{k,l}(Z_n) = \max_{d|n} \left\{ \frac{\alpha_{k,l}(Z_d) \cdot n}{d} \right\}.$$

Лемма 2 [1]. *Для любого n*

1. *Если $n|(k-l)$, то $\alpha_{k,l}(Z_n) = 0$;*
2. *Если НОД$(n, k-l) = 1$, то*

$$\alpha_{k,l}(Z_n) = \max \left\{ \frac{n}{p}, \left\lfloor \frac{n-2}{k+l} \right\rfloor + 1 \right\},$$

где p — наименьший простой делитель n;

3. *Если $1 <$ НОД$(n, k-l) < n$, то*

$$\max \left\{ \frac{n}{\rho_1}, \left\lfloor \frac{n-1-\delta(n)}{k+l} \right\rfloor + 1 \right\} \leqslant \\ \leqslant \alpha_{k,l}(Z_n) \leqslant \max \left\{ \frac{n}{\rho_1}, \frac{n}{2\rho_2}, \left\lfloor \frac{n-2}{k+l} \right\rfloor + 1 \right\},$$

где $\delta(n) =$ НОД$(n, k-l)$, ρ_1 — наименьший делитель n, такой, что $(k-l)$ не делится на ρ_1, а ρ_2 — наименьший делитель n, такой, что $(k-l)$ делится на ρ_2.

Определение. Пусть S — непустое подмножество группы G. Стабилизатором подмножества S называется множество $H(S) = \{g \in G \mid g + S = S\}$, имеющее максимальную мощность. Ясно, что $H(S)$ представляет собой подгруппу группы G.

Лемма 3 [1]. *Пусть A — (k,l)-максимальное по включению множество, свободное от сумм в Z_n. Тогда*

1. *$k(A + H(kA)) = kA$;*
2. *$(A + H(kA))$ — (k,l)-множество, свободное от сумм в Z_n;*
3. *$A + H(kA) = A$;*
4. *A — объединение смежных классов $H(kA)$.*

Положим $\delta(n) = \text{НОД}(n, k-l)$, $n = \delta(n) \cdot n_1$, $k - l = \delta(n) \cdot m_1$, $R = \{0, n_1, 2 \cdot n_1, \ldots, (\delta(n)-1) \cdot n_1\}$, $|R| = \delta(n)$.

Теорема 4. *Пусть A — (k,l)-множество, свободное от сумм в Z_n. Тогда*

1. $A \cap R = \varnothing$*;*
2. $c \star A$ *является (k,l)-множеством, свободным от сумм в Z_n, тогда и только тогда, когда* $0 \notin c \star A$*;*
3. $(kA - lA) \cap R = \varnothing$*;*
4. $H(kA - lA) \cap (kA - lA) = \varnothing$.

Доказательство.

1. Предположим противное, и пусть $i \cdot n_1 \in A$ ($i \in \{0, \ldots, \delta(n)-1\}$), тогда $(k-l) \cdot i \cdot n_1 \in kA - lA$, а $(k-l) \cdot i \cdot n_1 = \delta(n) \cdot m_1 \cdot i \cdot n_1 = n \cdot m_1 \cdot i = 0$, т. е. $0 \in kA - lA$, что противоречит тому, что A (k,l)-свободно от сумм.

2. Если $0 \in c \star A$, то, очевидно, $c \star A$ не является (k,l)-множеством, свободным от сумм. Теперь предположим, что $0 \notin c \star A$, но $c \star A$ не является (k,l)-множеством, свободным от сумм, т. е. существуют $x_1, x_2, \ldots, x_k$, $y_1, y_2, \ldots, y_l \in A$, такие, что $c \cdot x_1 + c \cdot x_2 + \cdots + c \cdot x_k = c \cdot y_1 + c \cdot y_2 + \cdots + c \cdot y_l$, т. е. $x_1 + x_2 + \cdots + x_k = y_1 + y_2 + \cdots + y_l$, что противоречит тому, что A (k,l)-свободно от сумм.

Как следствие получим, что $\delta(n) \star A$ (k,l)-свободно от сумм.

3. Предположим противное, и пусть $i \cdot n_1 \in (kA - lA)$ ($i \in \{0, \ldots, \delta(n)-1\}$), тогда $\delta(n) \cdot i \cdot n_1 \in \delta(n) \star (kA - lA) = k(\delta(n) \star A) - l(\delta(n) \star A)$, а $\delta(n) \cdot i \cdot n_1 = 0$, т. е. $0 \in k(\delta(n) \star A) - l(\delta(n) \star A)$, что противоречит тому, что $\delta(n) \star A$ (k,l)-свободно от сумм.

4. Предположим противное, и пусть $h \in H(kA - lA)$ и $h \in (kA - lA)$, тогда $-h(\in H(kA - lA)) + h(\in kA - lA) = 0(\in kA - lA)$, что противоречит тому, что A (k,l)-свободно от сумм. ∎

Лемма 5 [1]. *Для любого n*

$$\alpha_{k,l}(Z_n) \geqslant \left\lfloor \frac{n - 1 - \delta(n)}{k + l} \right\rfloor + 1,$$

где $\delta(n) = \text{НОД}(n, k-l)$.

Теорема 6. *Для любого n*

$$\max_{d|n}\left\{\left(\left\lfloor\frac{d-1-\delta(d)}{k+l}\right\rfloor+1\right)\frac{n}{d}\right\}\leqslant$$
$$\leqslant\lambda_{k,l}(Z_n)\leqslant\max\left\{\frac{n}{2\rho},\max_{d|n}\left\{\left(\left\lfloor\frac{d-1-\delta(d)}{k+l}\right\rfloor+1\right)\frac{n}{d}\right\}\right\},\quad(1)$$

где $\delta(d)$ = НОД$(d,k-l)$, ρ — наименьший делитель n, такой, что $(k-l)$ делится на ρ.

Доказательство. Пусть A — (k,l)-максимальная по мощности арифметическая прогрессия с разницей t, свободная от сумм в Z_d, и НОД$(t,d)=1$. Тогда из теоремы 4 получим, что

$$(k+l)|A|-(k+l-1)=|kA-lA|\leqslant d-\delta(d),$$

т. е.

$$|A|\leqslant\left\lfloor\frac{d-1-\delta(d)}{k+l}\right\rfloor+1,\quad(2)$$

где $\delta(d)$ = НОД$(d,k-l)$.

В [1] доказано, что если НОД$(t,d)\neq 1$, то

$$\frac{d}{\rho_1}\leqslant|A|\leqslant\max\left\{\frac{d}{\rho_1},\frac{d}{2\rho_2}\right\}\quad(3)$$

и

$$\max_{d|n}\left\{\left(\left\lfloor\frac{d-1-\delta(d)}{k+l}\right\rfloor+1\right)\cdot\frac{n}{d}\right\}\geqslant\frac{n}{\rho_1},\quad(4)$$

где A — (k,l)-максимальная по мощности арифметическая прогрессия с разницей t, свободная от сумм в Z_d, $\delta(d)$ = НОД$(d,k-l)$, ρ_1 — наименьший делитель d, такой, что $(k-l)$ не делится на ρ_1, а ρ_2 — наименьший делитель d, такой, что $(k-l)$ делится на ρ_2.

Из леммы 5, (2) и (3) получим, что

$$\max\left\{\frac{d}{\rho_1},\left\lfloor\frac{d-1-\delta(d)}{k+l}\right\rfloor+1\right\}\leqslant$$
$$\leqslant\alpha_{k,l}(Z_d)\leqslant\max\left\{\frac{d}{\rho_1},\frac{d}{2\rho_2},\left\lfloor\frac{d-1-\delta(d)}{k+l}\right\rfloor+1\right\},$$

а из теоремы 1 и (4) получим (1). ∎

Поддержано грантом РФФИ № 10-01-00768-а.

Список литературы

[1] *Bajnok B.* On the maximum size of a (k, l)-sum-free subset of an abelian group // International Journal of Number Theory. — 2009. — V. 5, № 6. — P. 953–971.

[2] *Hamidoune Y. O., Plagne A.* A new critical pair theorem applied to sum-free sets in Abelian groups // Commentarii Mathematici Helvetici. — 2004. — V. 79 — P. 183–207.

[3] *Wallis W. D., Street A. P., Wallis J. S.* Combinatorics: Room squares, sum-free sets, Hadamard matrices. — Berlin, Heidelberg, New York: Springer-Verlag, 1972.

[4] *Green B., Ruzsa I.* Sum-free sets in abelian groups // Israel Journal of Mathematics.— 2005. — V. 147— P. 157–188.

[5] *Nathanson M. B.* Additive number theory: Inverse problems and the geometry of sumsets. — Berlin, Heidelberg, New York: Springer-Verlag, 1996.

[6] *Lev V. F.* Large sum-free sets in Z/pZ // Israel Journal of Mathematics. — 2006. — V. 154 — P. 221–233.

[7] *Bier T., Chin A. Y. M.* On (k, l)-sets in cyclic groups of odd prime order // Bull. Austral. Math. Soc. — 2001. — V. 63, № 1. — P. 115–121.

[8] *Diananda P. H., Yap H. P.* Maximal sum-free sets of elements of finite groups // Proc. Japan Acad. — 1969. — V. 45. — P. 1–5.

Использование графовых моделей при распараллеливании метода Холецкого для решения разреженных симметричных СЛАУ

Я. Ю. Сафонова

safonova.yana@gmail.com

Нижегородский государственный университет им. Н. И. Лобачевского

Введение

Системы линейных уравнений $Ax = b$ с симметричной положительно определенной матрицей A возникают при численном решении дифференциальных уравнений. Для решения систем такого рода применяют разложение Холецкого, находящее верхнетреугольную матрицу U, такую, что $U^T U = A$. Тогда решение исходной системы эквивалентно решению двух систем:

$$U^T y = b, Ux = y.$$

Алгоритм нахождения разложения, предложенный в [1], последовательно ищет строки искомой матрицы и не обладает параллелизмом, поскольку значения в строке i зависят от значений в предыдущих $i-1$ строках. Однако если матрица A является разреженной, т. е. содержит $O(n^2)$ нулей, где n — порядок матрицы, то строка i может не зависеть от значений предыдущих строк и выделение «независимых» друг от друга групп строк позволяет организовать параллельный расчет. В [2] дано определение дерева исключения матрицы как отражения ее структуры разреженности и сформулированы теоремы, позволяющие сконструировать параллельный алгоритм нахождения разложения по известному дереву исключения. Построение дерева исключения, являющееся, по сути, нахождением портрета матрицы U, весьма трудоемкая задача [3, 4]. При этом существуют алгоритмы переупорядочивания матрицы [1], в ходе выполнения которых определяется вид дерева исключения. В данной статье рассматривается взаимосвязь между переупорядочиванием матрицы и эффективностью применения параллельного алгоритма разложения.

Дерево исключения разреженной матрицы и параллельный алгоритм Холецкого

Пусть $G(A) = (V, E)$ — граф матрицы A. Деревом исключения называется дерево, множество вершин которого совпадает с V, а ребро (v_i, v_j) существует тогда и только тогда, когда $i = \min\{k : u_{kj} \neq 0 \; \& \; k > j\}$. Таким образом, v_j является родителем v_i, если первый наддиагональный элемент в столбце j располагается в строке i. Для дерева исключения справедлив ряд свойств [2], общий смысл которых заключается в том, что строки, соответствующие вершинам в непересекающихся поддеревьях, таких, что ни одна вершина из одного поддерева не является предком вершин другого поддерева, можно обрабатывать независимо. Обработкой одной вершины назовем нахождение значений в соответствующей строке путем последовательного исключения всех строк, являющихся ее потомками.

Параллельный алгоритм нахождения разложения матрицы использует помеченное дерево исключения, где листом считается вершина, не имеющая непомеченных потомков. На каждой итерации алгоритма формируется список листьев и организуется их параллельная обработка, по окончании которой обработанные вершины помечаются. Такой подход имеет существенный недостаток — большое количество синхронизирующих операций, время выполнения которых будет снижать производительность. Поэтому на практике используют разбиение дерева исключения на слои большей, чем 1, высоты, при этом внутри каждого слоя осуществляется параллельная обработка независимых поддеревьев. Время работы такого параллельного алгоритма зависит от высоты дерева исключения: чем меньше высота дерева при заданном количестве вершин, тем больше возможное количество параллельных операций внутри слоев.

Переупорядочивание матрицы и связь с видом дерева исключения

Для уменьшения заполнения фактора используется переупорядочивание исходной матрицы, т. е. решение системы сводится к решению эквивалентной системы $PAP^T(Px) = Py$, где P — матрица перестановок. Одним из способов определения матрицы P является метод вложенных сечений. Известны [1] оценки заполнения метода, говорящие в пользу его применения. Другой важной особенностью является формирование структуры дерева исключения в процессе переупорядочивания. На каждой итерации после выбора разделителя образуются два подграфа, не имеющие общих вершин и ребер. При вычислении строк, соответствующих вершинам в первом подграфе, заполнение будет возникать только в столбцах, соответствующих другим вершинам подграфа и вершинам разделителя. Аналогичные рассуждения применимы ко второму подграфу. Следовательно, вершины подграфов являются потомками вершин разделителя в дереве исключения. Полученное дерево исключения будет бинарным, а выбор в качестве разделителя среднего уровня смежности гарантирует хорошую сбалансированность. Предварительное применение метода вложенных сечений будет давать выигрыш при параллельной факторизации.

Вычислительный эксперимент

Полученные теоретические выводы проверены для матриц, возникающих в различных прикладных областях. Каждая матрица была переупорядочена методом вложенных сечений и алгоритмом Катхилла–Макки. Для всех представлений, включая исходное, построены деревья исключения и разложение Холецкого. Для конкретной матрицы сравнивались такие показатели, как высота дерева исключения и степень заполнения фактора. Оба способа переупорядочивания, алгоритм Катхилла–Макки и метод вложенных сечений, дают хорошие результаты с точки зрения заполнения.

Название матрицы	Степень заполнения		
	Без переупорядочивания	После CM	После ND
bcsstk14	0,116	0,128	0,094
plat1919	0,636	0,06	0,046
nasa2146	0,072	0,087	0,067
ex9	0,028	0,051	0,043
nasa2910	0,11	0,306	0,116
ex13	0,039	0,072	0,06
bcsstk24	0,32	0,109	0,07
nasa4704	0,081	0,096	0,045
s1rmq4m1	0,066	0,087	0,042

Однако значения высоты, соответствующие переупорядочиванию методом вложенных сечений, значительно ниже значений высоты исходного дерева исключения и полученного после работы алгоритма Катхилла–Макки.

Название матрицы	Высота дерева исключения		
	Без переупорядочивания	После CM	После ND
bcsstk14	1752	1764	350
plat1919	1902	1918	198
nasa2146	2145	2145	369
ex9	2556	2567	412
nasa2910	2581	2886	856
ex13	3361	3361	401
bcsstk24	3560	3561	636
nasa4704	4630	4703	785
s1rmq4m1	5488	5488	646

Полученные результаты свидетельствуют о том, что использование параллельного алгоритма разложения матрицы, упорядоченной методом вложенных сечений, будет давать наибольшую эффективность и умеренное заполнение.

Список литературы

[1] *Писсанецки С.* Технология разреженных матриц. — М.: Мир, 1988.

[2] *Heggerness P.* Minimizing fill-in size and elimination tree height in parallel Cholesky factorization. — Bergen: Department of Informatics University of Bergen, 1992. — P. 15–17.

[3] *van Grondelle J.* Symbolic sparse Cholesky factorization using elimination trees. — Utrecht: Department of Mathematics Utrecht University, 1999. — P. 3–6.

[4] *Dereniowski D., Kubake M.* Cholesky factorization of matrices in parallel and ranking of graphs // 5-th International Conference on Parallel Processing and Applied Mathematics. — Czestochowa: Springer, 2004. — P. 985–992.

О сложности k-значных функций в одном классе полиномов

С. Н. Селезнева

selezn@cs.msu.su

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Одной из нормальных форм задания функций являются полиномы. Каждая k-значная функция может быть задана однозначным обычным полиномом тогда и только тогда, когда k — простое число [1]. Обобщения полиномов (поляризованные, обобщенно-поляризованные, обобщенные полиномы) вводятся для уменьшения сложности задания функций. К настоящему времени точная оценка функции Шеннона длины получена только для булевых функций в классе поляризованных полиномов [2]. Для k-значных функций (при простых $k \geqslant 3$) в классах поляризованных и обобщенно-поляризованных полиномов нижняя и верхняя оценки функции Шеннона длины совпадают только по порядку [3, 4]. А в классе обобщенных полиномов даже для булевых функций (и для k-значных при простых $k \geqslant 3$) нижняя и верхняя оценки функции Шеннона длины различаются по порядку [5–7].

В настоящей работе вводится один класс полиномов. Эти полиномы являются «почти» обычными, поляризация возможна только по одной выделенной переменной. Но поляризация понимается в более широком смысле. Однако этого достаточно, чтобы уменьшить сложность функций в этом классе полиномов почти так же, как в классе поляризованных полиномов. При том, что в поляризованных полиномах поляризация допускается по всем переменным. В работе найдены верхняя и нижняя оценки функции Шеннона длины k-значных функций (при простых $k \geqslant 3$) в этом классе полиномов, а также ее точная оценка для булевых функций.

Основные понятия

Пусть $k \geqslant 2$, $E_k = \{0, 1, \ldots, k-1\}$. Функцию $f(x_1, \ldots, x_n)$, $n = 0, 1, \ldots$, назовем k-значной, если $f : E_k^n \to E_k$. Если $k = 2$, то k-значная функция называется *булевой*. Множество всех k-значных функций обозначим как P_k, множество всех k-значных функций, зависящих от n переменных, как P_k^n.

Пусть k — простое число. Будем рассматривать сложение и умножение по $\mathrm{mod}\ k$.

Произведение вида
$$x_{i_1}^{m_1} \cdot \ldots \cdot x_{i_r}^{m_r},$$
в котором все переменные x_{i_j} попарно различны и $1 \leqslant m_1, \ldots, m_r \leqslant k-1$, называется (*обычным*) *мономом*.

Степенью монома X называется число $d(X) = \sum\limits_{j=1}^{r} m_j$.

Будем считать константу 1 вырожденным мономом степени 0.

Сумма вида
$$\sum_{i=1}^{l} c_i \cdot X_i,$$
где $c_i \in E_k \setminus \{0\}$ — коэффициенты, X_i — различные мономы, $i = 1, \ldots, l$, называется (*обычным*) *полиномом*.

Число различных слагаемых полинома P называется его *длиной* $l(P)$, максимальная степень его слагаемых — его степенью $d(P)$.

Будем полагать константу 0 вырожденным полиномом с длиной и степенью, равными 0.

Если и только если k — простое число, каждую k-значную функцию можно однозначно задать (обычным) полиномом по mod k [1].

Введем понятие полинома, *поляризованного по выделенной переменной* x_i, или *квазиполинома*. Не ограничивая общности, рассмотрим случай переменной x_1.

Множество $\delta = \{s_1(x_1), \ldots, s_k(x_1)\} \subseteq P_k^1$ назовем *поляризующим*, если

1) в нем ровно k различных функций ($|\delta| = k$);
2) для каждого m, $0 \leqslant m \leqslant k-1$, найдутся такие коэффициенты $c_{1m}, \ldots, c_{km} \in E_k$, что

$$x_1^m = \sum_{i=1}^{k} c_{im} s_i(x_1).$$

Другими словами, множество δ является *полиномиальным базисом* в пространстве полиномов степени не выше $(k-1)$ над полем $(E_k; +, \cdot)$ (k — простое число).

Квазимономом (*по множеству* δ) назовем произведение некоторой функции $s(x_1) \in \delta$ и обычного монома, в котором нет переменной x_1.

Степенью квазимонома $X = s(x_1) \cdot X'$, где $s(x_1) \in \delta$ и обычный моном X' не содержит переменную x_1, назовем $d(X) = d(s_1) + d(X')$.

Квазиполиномом (*по множеству* δ) назовем сумму различных квазимономов по тому же множеству с ненулевыми коэффициентами из E_k.

Число различных слагаемых квазиполинома P назовем его *длиной* $l(P)$, максимальную степень слагаемых — его *степенью* $d(P)$.

О задании k-значных функций квазиполиномами

Теорема 1. *Пусть k — простое число. Для каждого поляризующего множества δ, для каждой k-значной функции $f(x_1, \ldots, x_n)$ существует однозначно определенный задающий ее квазиполином полином по этому множеству.*

О сложности квазиполиномов

Введем сложностные характеристики k-значных функций в классе квазиполиномов. Для функции $f(x_1, \ldots, x_n) \in P_k^n$ ее *длиной*

$l^{\text{к.п.}}(f)$ (*степенью* $d^{\text{к.п.}}(f)$) в классе квазиполиномов назовем минимальную длину (минимальную степень) среди всех квазиполиномов, задающих f.

Введем функции Шеннона длины и степени k-значных функций в классе квазиполиномов

$$L_k^{\text{к.п.}}(n) = \max l^{\text{к.п.}}(f), \qquad D_k^{\text{к.п.}}(n) = \max d^{\text{к.п.}}(f),$$

где максимум берется по всем функциям, зависящим от n переменных.

Для функции Шеннона $L_k^{\text{п.п.}}(n)$ длины поляризованных полиномов по векторам поляризации $(d_1, \ldots, d_n) \in E_k^n$ известны следующие оценки:

1) $L_2^{\text{п.п.}}(n) = \lfloor \frac{2}{3} 2^n \rfloor$ [2];
2) $L_k^{\text{п.п.}}(n) \leqslant \dfrac{k(k-1)}{k(k-1)+1} k^n$, $k \geqslant 3$ [3];
3) $\dfrac{k-1}{k} k^n \lesssim L_k^{\text{п.п.}}(n)$ при $n \to \infty$ [4];
4) $L_k(1) = k - 1$ [8].

Мы доказываем следующие оценки функций Шеннона степени и длины квазиполиномов k-значных функций при $k \geqslant 2$.

Теорема 2. *Пусть k — простое число, $k \geqslant 2$. Тогда*

$$D_k^{\text{к.п.}}(n) = (k-1)n.$$

Теорема 3. $L_2^{\text{к.п.}}(n) = \lfloor \frac{2}{3} 2^n \rfloor$.

Заметим, что точная оценка длины квазиполиномов для булевых функций совпадает с точной оценкой длины поляризованных полиномов булевых функций [2].

Теорема 4. *Пусть k — простое число, $k \geqslant 3$. Тогда*

$$\frac{k-1}{k} k^n \lesssim L_k^{\text{к.п.}}(n) \quad \text{при} \quad n \to \infty.$$

Теорема 5. *Пусть k — простое число, $k \geqslant 3$. Тогда*

$$L_k^{\text{к.п.}}(n) \leqslant \frac{k}{k+1} k^n.$$

Работа выполнена при поддержке РФФИ, гранты 09-01-00701-а и 10-01-00768-а.

Список литературы

[1] *Яблонский С.В.* Введение в дискретную математику. М.: Наука, 2001.

[2] *Перязев Н.А.* Сложность булевых функций в классе полиномиальных поляризованных форм // Алгебра и логика. — 1995. — Т. 34, № 3. — С. 323–326.

[3] *Селезнева С.Н.* О сложности представления функций многозначных логик поляризованными полиномами // Дискретная математика. — 2002. — Т. 14, № 2. — С. 48–53.

[4] *Алексеев В.Б.*, *Вороненко А.А.*, *Селезнева С.Н.* О сложности реализации функций k-значной логики поляризованными полиномами // Тр. V Межд. конф. «Дискретные модели в теории управляющих систем» (Ратмино, 26–29 мая 2003 г.). — М.: МГУ, 2003. — С. 8–9.

[5] *Even S.*, *Kohavi I.*, *Paz A.* On minimal modulo 2 sums of products for switching functions // IEEE Trans. Elect. Comput. — 1967. — P. 671–674.

[6] *Кириченко К.Д.* Верхняя оценка сложности полиномиальных нормальных форм булевых функций // Дискретная математика. — 2005. — Т. 17, № 3. — С. 80–88.

[7] *Селезнева С.Н.*, *Дайняк А.Б.* О сложности обобщенных полиномов k-значных функций // Вестник Моск. унив. Серия 15. Вычисл. матем. и кибернетика. — 2008. — С. 34–39.

[8] *Селезнева С.Н.* О сложности поляризованных полиномов функций многозначных логик, зависящих от одной переменной // Дискретная математика. — 2004. — Т. 16, № 2. — С. 117–120.

О подобии некоторых верхнетреугольных матриц над кольцом целых чисел

С. В. Сидоров

sesidorov@yandex.ru

Нижегородский государственный университет им. Н. И. Лобачевского

Пусть A и B — две целочисленные матрицы порядка n. Будем говорить, что матрица A подобна матрице B над кольцом целых чисел $\mathbf{Z}$, если существует такая унимодулярная матрица $X \in \mathbf{Z}^{n\times n}$, что

$B = X^{-1}AX$. Матрица называется унимодулярной, если ее определитель равен 1 или -1.

Нормальной диагональной формой Смита называется целочисленная матрица $\left(\begin{array}{c|c} D & 0 \\ \hline 0 & 0 \end{array}\right)$ с неотрицательными элементами, где $D = \operatorname{diag}\{d_1, \ldots, d_r\}$ — диагональная матрица, d_{k+1} делится на d_k $(k = 1, \ldots, r-1)$.

Известно, что для любой матрицы $A \in \mathbf{Z}^{m\times n}$ существуют такие унимодулярные матрицы $U \in \mathbf{Z}^{m\times m}$ и $V \in \mathbf{Z}^{n\times n}$, что $UAV = S_A$ — нормальная диагональная форма Смита.

Через E_n будем обозначать единичную матрицу порядка n.

Лемма 1. *Матрица* $\left(\begin{array}{c|c} \alpha E_m & A \\ \hline 0 & \beta E_n \end{array}\right)$ *подобна над кольцом целых чисел матрице* $\left(\begin{array}{c|c} \alpha E_m & S_A \\ \hline 0 & \beta E_n \end{array}\right)$, *где S_A — нормальная диагональная форма Смита для матрицы A.*

Доказательство. Пусть $U \in \mathbf{Z}^{m\times m}$ и $V \in \mathbf{Z}^{n\times n}$ – унимодулярные матрицы, удовлетворяющие условию $UAV = S_A$. Рассмотрим блочную унимодулярную матрицу $\left(\begin{array}{c|c} U^{-1} & 0 \\ \hline 0 & V \end{array}\right)$. Тогда

$$\left(\begin{array}{c|c} U^{-1} & 0 \\ \hline 0 & V \end{array}\right)^{-1} \left(\begin{array}{c|c} \alpha E_m & A \\ \hline 0 & \beta E_n \end{array}\right) \left(\begin{array}{c|c} U^{-1} & 0 \\ \hline 0 & V \end{array}\right) =$$

$$= \left(\begin{array}{c|c} U & 0 \\ \hline 0 & V^{-1} \end{array}\right) \left(\begin{array}{c|c} \alpha E_m & A \\ \hline 0 & \beta E_n \end{array}\right) \left(\begin{array}{c|c} U^{-1} & 0 \\ \hline 0 & V \end{array}\right) =$$

$$= \left(\begin{array}{c|c} \alpha U & UA \\ \hline 0 & \beta V^{-1} \end{array}\right) \left(\begin{array}{c|c} U^{-1} & 0 \\ \hline 0 & V \end{array}\right) =$$

$$= \left(\begin{array}{c|c} \alpha E_m & UAV \\ \hline 0 & \beta E_n \end{array}\right) = \left(\begin{array}{c|c} \alpha E_m & S_A \\ \hline 0 & \beta E_n \end{array}\right).$$

∎

Лемма 2 [1]. *Пусть $A, B \in \mathbf{Z}^{n\times n}$. Если A и B подобны над кольцом целых чисел, то $\Delta_k(A) = \Delta_k(B)$ $(k = 1, \ldots, n)$, где $\Delta_k(A)$ и $\Delta_k(B)$ — наибольшие общие делители миноров k-го порядка матриц A и B соответственно.*

Теорема. *Матрицы* $A = \left(\begin{array}{c|c} \alpha E_m & A_1 \\ \hline 0 & \alpha E_n \end{array}\right)$ *и* $B = \left(\begin{array}{c|c} \alpha E_m & B_1 \\ \hline 0 & \alpha E_n \end{array}\right)$ *подобны над кольцом целых чисел тогда и только тогда, когда* $S_{A_1} = S_{B_1}$, *где* S_{A_1} *и* S_{B_1} *— нормальные диагональные формы Смита для матриц* A_1 *и* B_1 *соответственно.*

Доказательство. Согласно лемме 1 матрицы A и B подобны матрицам $A' = \left(\begin{array}{c|c} \alpha E_m & S_{A_1} \\ \hline 0 & \alpha E_n \end{array}\right)$ и $B' = \left(\begin{array}{c|c} \alpha E_m & S_{B_1} \\ \hline 0 & \alpha E_n \end{array}\right)$ соответственно.

Достаточность. Поскольку $S_{A_1} = S_{B_1}$, то $A' = B'$ и матрицы A и B подобны в силу транзитивности отношения подобия.

Необходимость. Матрицы A' и B' подобны, так как подобны матрицы A и B. Ясно, что тогда подобны матрицы $A'' = A' - \alpha E_{m+n} = \left(\begin{array}{c|c} 0 & S_{A_1} \\ \hline 0 & 0 \end{array}\right)$ и $B'' = B' - \alpha E_{m+n} = \left(\begin{array}{c|c} 0 & S_{B_1} \\ \hline 0 & 0 \end{array}\right)$. Согласно лемме 2 $\Delta_k(A'') = \Delta_k(B'')$ $(k = 1, \ldots, m+n)$. Если $S_{A_1} = \left(\begin{array}{c|c} D_1 & 0 \\ \hline 0 & 0 \end{array}\right)$, $D_1 = \operatorname{diag}\{d_1, \ldots, d_r\}$, $S_{B_1} = \left(\begin{array}{c|c} D_2 & 0 \\ \hline 0 & 0 \end{array}\right)$, $D_2 = \operatorname{diag}\{d'_1, \ldots, d'_{r'}\}$, то наибольший общий делитель миноров k-го порядка в матрице A'' равен $\Delta_k(A'') = d_1 \cdot \ldots \cdot d_k$ $(k = 1, \ldots, r)$. Аналогично $\Delta_k(B'') = d'_1 \cdot \ldots \cdot d'_k$ $(k = 1, \ldots, r')$. Поскольку $\operatorname{rank} A'' = r$, $\operatorname{rank} B'' = r'$, а ранги подобных матриц равны, то $r = r'$. При $k = 1$ получаем $d_1 = \Delta_1(A'') = \Delta_1(B'') = d'_1$. Далее по индукции легко доказывается, что $d_k = d'_k$ для всех $k = 1, \ldots, r$. Следовательно, $S_{A_1} = S_{B_1}$. ∎

Работа выполнена при поддержке РФФИ, проект 09-01-00545-а.

Список литературы

[1] *Шевченко В. Н., Сидоров С. В.* О подобии матриц второго порядка над кольцом целых чисел // Известия вузов. Математика. — 2006. — № 4. — С. 57–64.

Моделирование работы ациклического алгоритма на многопроцессорной вычислительной системе

В. В. Слободской

slvital@gmail.com

Нижегородский государственный университет им. Н. И. Лобачевского

Введение

С появлением многоядерных компьютеров всё чаще ставится задача распараллеливания алгоритмов, сложность которой увеличивается, когда алгоритм должен наилучшим образом использовать вычислительные ресурсы различных вычислительных систем с различной конфигурацией и количеством процессоров. Возникает задача моделирования работы алгоритма на различных вычислительных системах.

Как и в [1], будем рассматривать алгоритмы, не содержащие ветвлений и циклов, представимые в виде канонической сети взаимозависимых операций. Предполагается, что любой алгоритм при известном наборе исходных данных представим в виде ациклического алгоритма без ветвлений. Таким образом, в общем случае под категорию рассматриваемых алгоритмов подходит любой алгоритм с набором всех необходимых для его работы исходных данных.

Вычислительная система — набор из n процессоров, обладающих различными скоростями выполнения операций. Считается, что время выполнения любой операции на процессоре обратно пропорционально скорости вычисления процессора. Задана матрица скоростей передачи данных от одного процессора другому. В отличие от [2] в этой работе рассматривается общая модель задачи.

Алгоритм представляет собой каноническую сеть взаимозависимых операций, где каждая операция (кроме начальных) имеет набор предшествующих операций и каждая операция (кроме завершающих) имеет набор последующих операций. Операция не может начинаться, пока не завершены все предшествующие ей операции. Для каждой операции известно время ее выполнения на самом медленном процессоре. Для взаимозависимых операций задано количество данных, которые необходимо передать для выполнения последующей операции. Операция не может начинаться, пока не переданы данные со всех операций, предшествующих этой операции. Передача данных может начинаться только после завершения операции.

Необходимо таким образом назначить операции на процессоры, чтобы общее время завершения обработки всех операций было минимальным.

В качестве эксперимента рассматривается инструмент, позволяющий для любой программы, реализующей некоторый алгоритм на некотором наборе входных данных, получить каноническую сеть взаимозависимых операций, которая может быть использована для решения задачи распараллеливания данного алгоритма, моделируя тем самым его выполнение на любой вычислительной системе. Для решения подобной задачи предлагается использовать алгоритмы, рассмотренные в [1, 3, 4].

Общая математическая модель

Исходные параметры математической модели

J — множество всех операций, I — множество процессоров, $K(j)$ — множество операций, непосредственно предшествующих операции с номером j, $K(j) \subset J$, $j \in J$; t_j — время выполнения операции j на процессоре со скоростью 1, $t_j > 0$, $j \in J$; p_i — скорость процессора i, $p_i > 0$, $i \in I$.

Пусть $\|M\|$ — матрица размерности $|J| \times |J|$, элемент которой m_{kl} — количество данных, которые должны быть переданы после выполнения операции k для операции l, $k \in K(l)$, $k, l \in J$; $\|S\|$ — матрица размерности $|I| \times |I|$, элемент которой s_{uv} определяет скорость передачи данных от процессора u к процессору v, $s_{uv} > 0$, $u, v \in I$; $Q = \{(j,k) \mid k, j \in J, k \notin K(j), j \notin K(k)\}$; $J^0 = \{j \mid j \in J, K(j) = \varnothing\}$; $J^D = \{j \mid j \in J, \nexists k \in J, j \in K(k)\}$.

Варьируемые параметры математической модели

$|J|$-мерный вектор $\mathbf{x}$, компонента которого x_j определяет время начала выполнения операции j, $j \in J$, и булеву матрицу $\|Y\|$ размерности $|J| \times |J|$, элемент которой $y_{ji} = 1$, если операция j выполняется на i-м процессоре, $j \in J$, $i \in I$; 0 — в противном случае.

Ограничения математической модели

Технологическое ограничение

$$x_k \geqslant \max_{j \in K(k)} \left(x_j + t_j \sum_{l \in I} \frac{y_{jl}}{p_l} + m_{jk} \sum_{i \in I} \sum_{l \in I} \frac{y_{ji} y_{kl}}{s_{il}} \right), \; k \in J. \qquad (1)$$

Невозможность выполнения двух операций на одном процессоре одновременно

$$x_k \geqslant x_j + t_j \sum_{l \in I} \frac{y_{jl}}{p_l} \quad \text{или} \quad x_j \geqslant x_k + t_k \sum_{l \in I} \frac{y_{kl}}{p_l},$$
$$\text{если } (j,k) \in Q,\ y_{jl} = y_{kl},\ \text{для всех } l \in I;\ j,k \in J. \tag{2}$$

Каждая операция должна выполняться на одном процессоре

$$\sum_{l \in I} y_{jl} = 1,\ j \in J, \tag{3}$$

$$x_j \geqslant 0,\ j \in J^0, \tag{4}$$

$$y_{ji} \in \{0,1\},\ j \in J,\ i \in I,\ x_j \geqslant 0,\ j \in J. \tag{5}$$

Задача минимизации общего времени выполнения операций

Требуется найти такое решение системы ограничений (1)–(5), на котором принимает минимальное значение критерий, определяющий общее время выполнения операций:

$$F(\mathbf{x}, Y) = \max_{j \in J^D} \left(x_j + t_j \sum_{l \in I} \frac{y_{jl}}{p_l} \right). \tag{6}$$

Моделирование работы алгоритма

Будем считать, что алгоритм реализован на языке программирования и представляет собой скомпилированное приложение. Для того чтобы получить детальную информацию о выполнении этого приложения, воспользуемся динамической инструментацией кода [5].

В качестве операции выделим блок кода, не содержащий инструкций-переходов (в дальнейшем – *простой блок*). Путем динамической инструментации кода найдём все простые блоки и добавим в код начала каждого блока запись информации об адресе блока, его размере и времени начала его работы. Таким образом мы сможем получить список операций приложения, а также их длительности.

Для определения зависимостей между операциями добавим инструментирование операций чтения и записи в память, а также инструкций, осуществляющих изменение содержимого регистров процессора. Будем сохранять информацию о том, какие участки памяти, а также регистры были считаны текущим простым блоком без предшествующей им записи. Определяя, какой блок осуществлял запись в эти ячейки памяти или регистры, мы получим зависимость между найденным и текущим блоком, тем самым будем иметь отношение взаимозависимости между соответствующими блокам операциями.

После запуска приложения на однопроцессорной системе, проведя анализ полученной информации, мы получим каноническую сеть взаимозависимых операций, соответствующих простым блокам приложения.

В качестве примера рассмотрим простую программу, вычисляющую произведение двух квадратных матриц. Моделировать выполнение будем на машинах архитектуры SMP с доступом к общей памяти, т. е. будем считать, что время передачи данных между процессорами равно нулю. Скорость вычислительных элементов системы будем считать равной скорости процессора, на котором осуществлялся сбор данных.

В результате анализа работы этого приложения была получена каноническая сеть взаимозависимых операций, состоящая из 29808 операций, время работы – 186,070 мс.

Для решения задачи распараллеливания использовался эвристический метод, описанный в [4]. Результаты моделирования на вычислительных системах с различным числом процессоров (мс): 96,139 (2П); 53,016 (4П); 33,022 (8П); 24,301 (16П).

В работе применен новый способ распараллеливания алгоритмов, основанный на решении оптимизационной задачи, поставленной в рамках общей математической модели. Предложен метод получения канонической сети взаимозависимых операций любого алгоритма с известными входными данными, что позволяет моделировать любой алгоритм (приложение) на любой вычислительной системе.

Данный подход можно использовать для оценки уровня параллелизма задачи. Варьируя вычислительные ресурсы, можно исследовать поведение алгоритма на различных вычислительных системах.

Список литературы

[1] *Слободской В. В.* Задача распараллеливания ациклического алгоритма // Вестник ННГУ. — 2008. — № 5. — С. 113–119.

[2] *Прилуцкий М. Х., Слободской В. В.* Распределение ресурсов в задачах балансирования с постоянными длительностями // Вестник ВГАВТ. Межвузовская серия «Моделирование и оптимизация сложных систем». Вып. 1. — 2008.

[3] *Слободской В. В.* Использование метода ветвей и границ для решения задачи распараллеливания ациклического алгоритма // Тезисы докладов конференции «Технологии Microsoft в теории и практике программирования». — Н.Новгород: Издательство ННГУ, 2009.

[4] *Слободской В. В.* Эвристический метод для решения задачи распараллеливания ациклического алгоритма на многопроцессорной вычислительной системе // Вестник ННГУ. — 2010. — № 4. — С. 182–186.

[5] *Chi-Keung Luk, Robert Cohn, Robert Muth, Harish Patil, Artur Klauser, Geoff Lowney, Steven Wallace, Vijay Janapa Reddi, Kim Hazelwood.* Pin: Building Customized Program Analysis Tools with Dynamic Instrumentation. Programming Language Design and Implementation (PLDI) Chicago: 2005. — P. 190–200.

Потоки в кратных сетях специального вида

А. В. Смирнов

alexander_sm@mail.ru

Ярославский государственный университет им. П. Г. Демидова

Понятие кратных сетей и кратных потоков было введено в статье [1] применительно к задаче целочисленного сбалансирования трехмерной матрицы. Вводится целое $k > 1$ — *кратность потока.* В качестве сети рассматривается ориентированный мультиграф $G(X, U)$, между вершинами которого могут быть дуги одного из трех видов:

1) *обычная дуга* u^o с пропускной способностью $c(u^o)$; множество обычных дуг обозначим через U^o;
2) *кратная дуга* u^k между двумя вершинами, которая состоит из k дуг одной ориентации с одинаковой пропускной способностью и одинаковым потоком по каждой из них; множество кратных дуг обозначим через U^k;

3) *связанная дуга* u между двумя вершинами, которая связана с еще $k-1$ дугами, имеющих одинаковый один из концов; все k дуг имеют одинаковую пропускную способность, и по ним идет одинаковый поток; множество связанных дуг, выходящих из одной вершины или входящих в одну вершину, будем называть *мультидугой* u^m; множество мультидуг обозначим через U^m.

Множество выходящих из вершины дуг может быть либо только кратными дугами, либо только одной мультидугой (k связанных дуг), либо только обычными дугами. Из источника x_0 сети выходят только кратные дуги, а в сток z сети входит только одна мультидуга. Если из вершины выходят связанные дуги мультидуги, то в нее обязательно входит кратная дуга. Если в вершину входит мультидуга, то из нее может выходить только кратная дуга. Определенный таким образом мультиграф $G(X,U)$ с целочисленными пропускными способностями дуг назовем *кратной* (*транспортной*) *сетью*.

Кратным потоком по сети называется целочисленная функция, определенная на множестве дуг $U = U^o \cup U^k \cup U^m$, для которой выполнены условия неотрицательности, ограниченности (пропускными способностями дуг) и неразрывности потока (в каждой вершине). *Величиной кратного потока* называется сумма φ_z входящего потока для стока z, равная сумме выходящего из источника потока. Величина φ_z должна быть кратна k. Обозначим через $c(u)$ пропускную способность дуги u, а через $f(u)$ — поток на ней. Так же как и в случае обычной транспортной сети, для кратных сетей ставится *задача о нахождении максимального потока*. В работе [2] рассмотрен вопрос об NP-полноте этой задачи для сети кратности $k \geqslant 2$.

Пусть имеется кратная сеть произвольной кратности k. Пусть при удалении всех мультидуг сеть распадается на $k+2$ компоненты связности (связность понимается в слабом смысле, то есть без учета направления дуг), при этом одна компонента состоит только из вершины z, компонента, содержащая вершину x_0, содержит только кратные дуги, а остальные k компонент содержат только обычные дуги. Если при этом каждая мультидуга имеет ровно один конец в каждой из k компонент, содержащих обычные дуги, то такую сеть мы будем называть *делимой*. Примером делимой сети является кратная сеть целочисленного сбалансирования трехмерной матрицы, рассмотренная в [1].

Обозначим через P_0 компоненту делимой сети, содержащую кратные дуги, через $P_1, \dots, P_k$ обозначим компоненты, содержащие обычные дуги. Тогда *частью* G_i делимой сети ($i \in \overline{1,k}$) назовем объединение соответствующей компоненты P_i с инцидентными ей связанными дугами всех мультидуг, кроме мультидуги с концом в z, а также с i-й связанной дугой каждой кратной дуги компоненты P_0. Заметим, что возможность выделения частей G_i является особенностью делимых сетей. Обозначим начальные вершины мультидуги с концом в z через $z_1, \dots, z_k$. Вершины, являющиеся началом остальных мультидуг, будем обозначать через y_j. Объединение мультидуги u_z^m, идущей из вершин $z_1, \dots, z_k$ в z, и k путей μ_r ($r \in \overline{1,k}$), каждый из которых является ориентированным путем в соответствующей части G_r из вершины x_0 в вершину z_r, назовем *обобщенным путем*, если каждый из путей μ_r проходит через одну и ту же вершину y_j. Кратный поток назовем *полным*, если любой обобщенный путь из x_0 в z имеет дугу u (обычную, кратную или мультидугу), поток по которой равен ее пропускной способности $f(u) = c(u)$. *Проекция* C_i ($i \in \overline{1,k}$) подграфа C на часть сети G_i — это часть подграфа C, образованная его вершинами и дугами, принадлежащими G_i. Так как части G_i ($i \in \overline{1,k}$) сети G представляют собой обычные транспортные сети с источником x_0 и стоком z_i, то будем называть некоторый путь из x_0 в z_i *путем прорыва в части* G_i, если $f(u) < c(u)$ на прямых дугах и $f(u) > 0$ на обратных дугах этого пути.

Пусть в делимой сети определен некоторый поток φ величины $\varphi_z > 0$. *Кратным циклом* в делимой сети $G(X, U)$, где X – множество вершин, U – множество дуг (обычных, кратных или мультидуг), назовем такой подграф $C(X', U')$, $X' \subset X$, $U' \subset U$, для которого:

1) проекции $C_1, \dots, C_k$ на части $G_1, \dots, G_k$ соответственно есть объединение некоторых циклов, причем дуги, поток по которым ненулевой, могут проходиться в обратном направлении;
2) проекции $C_1, \dots, C_k$ *согласованы* (одинаковы) на общей части подграфов $G_1, \dots, G_k$;
3) C_1 представим в виде $C_1 = \cup\{C_j^1\}$, где C_j^1 – некоторые циклы и $C_j^1 \not\subset C_k^1$ $\forall j \neq k$; при этом для любой дуги u из G_1 выполняется неравенство $0 \leqslant f(u) + a^+(u) - a^-(u) \leqslant c(u)$, где $a^+(u)$ – это число циклов C_j^1, в которых дуга u проходится в прямом направлении, $a^-(u)$ — это число циклов C_j^1, в которых дуга u проходится в

обратном направлении. Такое же условие должно выполняться и для $C_2, \ldots, C_k$.

Обобщенным путем прорыва в делимой сети $G(X, U)$ для некоторого кратного потока φ назовем такой подграф $S(X', U')$, $X' \subset X$, $U' \subset U$, для которого:

1) каждая из проекций $S_1, \ldots, S_k$ на части $G_1, \ldots, G_k$ соответственно есть объединение ровно одного пути прорыва из x_0 в z_i и некоторых циклов, причем дуги, поток по которым ненулевой, могут проходиться в обратном направлении;
2) проекции $S_1, \ldots, S_k$ *согласованы* на общей части подграфов $G_1, \ldots, G_k$;
3) S_1 представим в виде $S_1 = \mu_1 \cup \{C_j^1\}$, где μ_1 – путь прорыва, C_j^1 – некоторые циклы и $C_j^1 \neq C_k^1\ \forall j \neq k$; при этом для любой дуги u из G_1 выполняется неравенство $0 \leqslant f(u) + a^+(u) - a^-(u) \leqslant c(u)$, где $a^+(u)$ – это число элементов множества $\mu_1 \cup \{C_j^1\}$, в которых дуга u проходится в прямом направлении, $a^-(u)$ – это число элементов множества $\mu_1 \cup \{C_j^1\}$, в которых дуга u проходится в обратном направлении. Такое же условие должно выполняться и для $S_2, \ldots, S_k$;
4) дуга $(\{z_1, \ldots, z_k\}, z) \in U'$ и $f(\{z_1, \ldots, z_k\}, z) < c(\{z_1, \ldots, z_k\}, z)$;
5) S не содержит кратного цикла.

Обозначим $G_\varphi = G(X, U_\varphi)$; $U_\varphi = \{u | f(u) \neq 0\}$.

Теорема 1. *Пусть $\varphi^1(U)$, $\varphi^2(U)$ – два полных потока в делимой сети $G(X, U)$, причем $\varphi_z^1 \leqslant \varphi_z^2$. Пусть $\varphi^0(U) = \varphi^2(U) - \varphi^1(U)$. Тогда граф $G_{\varphi^0} = \{S_i\} \cup \{C_j\}$, где $\{S_i\}$ — множество всех обобщенных путей прорыва из x_0 в z, $\{C_j\}$ — множество всех кратных циклов. В случае когда $\varphi_z^1 = \varphi_z^2 = \varphi_z^{\max}$, $\{S_i\} = \varnothing$.*

Алгоритм нахождения максимального потока в делимой сети является обобщением алгоритма нахождения максимального кратного потока для решения задачи целочисленного сбалансирования трехмерной матрицы, рассмотренного в работе [1]. Работа алгоритма происходит в два этапа: *этап построения полного потока*; *этап увеличения потока* (если полученный полный поток не является максимальным, то увеличиваем поток при помощи *обобщенного алгоритма пометок* (см. ниже) до тех пор, пока поток не станет максимальным).

Алгоритм нахождения полного потока прост. На данном этапе увеличения потока можно добиваться, находя все возможные обобщенные пути прорыва из x_0 в z через вершины y_j без обратных дуг и увеличивая поток по ним. В итоге будет получен полный поток.

Рассмотрим обобщенный алгоритм пометок. Идея алгоритма состоит в следующем: в проекциях $G_1, \ldots, G_k$ поочередно строятся пути прорыва $\mu_1, \ldots, \mu_k$ (возможно, в объединении с некоторыми циклами) до тех пор, пока пути в обеих проекциях не станут согласованными либо же не останется вариантов для продолжения построения пути. В первом случае объединение $\mu_1, \ldots, \mu_k$ с мультидугой с концом в z даст обобщенный путь прорыва, во втором случае производится откат до *«точки ветвления»*, после чего выполнение алгоритма возобновляется.

Теорема 2. *Максимальный поток в делимой сети может быть найден при помощи указанного выше алгоритма.*

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг., ГК № П161.

Список литературы

[1] *Рублев В. С., Смирнов А. В.* Задача целочисленного сбалансирования трехмерной матрицы и алгоритмы ее решения // Моделирование и анализ информационных систем. — 2010. — Т. 17, № 2. — С. 72–98.

[2] *Смирнов А. В.* NP-полнота задачи нахождения максимального потока в кратной сети // Заметки по информатике и математике. Вып. 2 — Ярославль: ЯрГУ им. П. Г. Демидова, 2010. — С. 122–131.

О структуре матриц оптимального локально-префиксного кодирования языков

Т. Г. Смирнова

smirnova-tg@mail.ru

Нижегородский государственный университет им. Н. И. Лобачевского

В работе исследуются вопросы сжатия информации в классе алфавитных отображений с использованием двоичных локально-префиксных кодов: всякому начальному отрезку сообщения соответствует своя группа букв, которые могут за ним непосредственно последовать в сообщении, и если для любой такой группы букв применяется префиксный код, то весь код называется локально-префиксным.

Пусть $A = \{0, 1\}$ — алфавит канала связи, $B = \{b_1, \ldots, b_m\}$ — алфавит языка сообщений $L \subseteq B^*$, где B^* — свободный моноид над алфавитом B. Алфавитное кодирование $f = f_V$ задается схемой: $b_i \to v_i$, где $v_i \in A^+$ $(i = 1, \ldots, m)$. Вектор $V = \langle v_1, \ldots, v_m \rangle$ называется кодом, определяющим f_V, а вектор $d\,(V) = \langle d_1, \ldots, d_m \rangle$, где $d_i = |v_i|$ — длина элементарного кода v_i, называется спектром длин кода V.

Будем говорить, что слова α и β находятся в отношении антипрефиксности $\overline{\pi}$ и обозначать $\alpha\,\overline{\pi}\,\beta$, если никакое из них не является префиксом другого. Префиксные коды, т. е. коды, у которых $v_i\,\overline{\pi}\,v_j$ для любых $i \neq j$, составляют один из основных классов однозначно декодируемых кодов переменной длины.

В настоящей работе рассматриваются локально-префиксные коды [1, 2], соответствующие алфавитному кодированию локальных моделей языков глубины 1.

Пусть $L \subseteq B^*$. Окрестностью глубины 1 слова $\alpha \in B^*$ назовем множество

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

а множество $\mathfrak{M} = \mathfrak{M}_1\,(L) = \{\varepsilon\,(\alpha) :\ \varepsilon\,(\alpha) \neq \oslash,\ \alpha \in B^*\}$ — локальной моделью языка L глубины 1.

Построение локальной модели глубины 1 существенно зависит от исходного способа задания языка. Известно [3], что для контекстно-свободных языков эта проблема алгоритмически неразрешима. В классе регулярных языков локальная модель глубины 1 может быть описана достаточно просто.

Пусть $L = L(\Gamma)$ — язык, порожденный автоматным источником Γ с n состояниями над алфавитом B, тогда

$$\varepsilon\,(\alpha) = \{b \in B :\ \varphi_\Gamma\,(\alpha b, q_0) \neq \oslash\}\,,$$

где q_0 — начальное состояние источника. Пусть $\varphi_\Gamma\,(\alpha, q_0) = \varphi_\Gamma\,(\beta, q_0) = q_i$, тогда $\varphi_\Gamma\,(\alpha b, q_0) = \varphi_\Gamma\,(\beta b, q_0) = \varphi_\Gamma\,(b, q_i)$ и $\varepsilon\,(\alpha) = \varepsilon\,(\beta) = \varepsilon\,(q_i)$. Таким образом, локальная модель $\mathfrak{M}$ регулярного языка $L = L(\Gamma)$ представляет собой множество непустых алфавитных окрестностей состояний $q_0, \ldots, q_{n-1}$ источника Γ:

$$\mathfrak{M}(L) = \{\varepsilon_{i+1} = \varepsilon(q_i) = \{b \in B : \varphi_\Gamma(b, q_i) \neq \oslash\} : i = 0, \ldots, n-1\}.$$

Всякой локальной модели $\mathfrak{M}(L) = \{\varepsilon_1, \ldots, \varepsilon_n\}$ языка L, $\varepsilon_i \subseteq B$ $(i = 1, \ldots, n)$, соответствует граф антипрефиксности

$$G = K(\varepsilon_1) \cup \ldots \cup K(\varepsilon_n),$$

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

Код $V = \{v_1, \ldots, v_m\}$ называется локально-префиксным относительно графа G, если кодовые слова v_i, v_j, соответствующие смежным вершинам b_i, b_j графа G, находятся в отношении антипрефиксности, т. е. $v_i \,\overline{\pi}\, v_j$. Заметим, что для полных m-вершинных графов K_m задача локально-префиксного кодирования совпадает с задачей префиксного кодирования m-буквенного алфавита в обычном смысле.

Множество всех спектров, реализуемых q-ичными локально-префиксными относительно G кодами, монотонно относительно частичного порядка, определенного покомпонентной сравнимостью векторов. Минимальные элементы этого множества образуют матрицу $M(G)$ оптимального локально-префиксного кодирования локальных моделей, представимых графом G. Это множество конечно для любого графа по теореме Диксона и для любого графа может быть расшифровано.

Необходимое условие реализуемости спектра $d(V) = \langle d_1, \ldots, d_m \rangle$ двоичным локально-префиксным относительно G кодом записывается в виде системы неравенств Мак-Миллана для каждой из клик $K(\varepsilon_1), \ldots, K(\varepsilon_n)$ графа G:

$$\sum_{b_i \in K(\varepsilon_j)} 2^{-d_i} \leqslant 1 \quad (j = 1, \ldots, n). \tag{1}$$

При $n = 1, 2$ система неравенств (1) определяет также и достаточное условие реализуемости спектра, однако при $n \geqslant 3$ подобное утверждение не имеет места.

Пусть $\mathfrak{G}_m$ — множество всех локальных моделей над алфавитом B и $\Sigma_m = \{M(G) : G = \mathfrak{M}(L), L \subseteq B^*\}$ — множество различных матриц оптимального двоичного локально-префиксного кодирования.

На $\mathfrak{G}_m$ определяется отношение эквивалентности ε: будем говорить, что графы G_1 и G_2 находятся в отношении ε, если матрицы $M(G_1)$ и $M(G_2)$ — перестановочно-эквивалентны, т.е. матрицу $M(G_2)$ можно получить из матрицы $M(G_1)$ элементарными перестановками строк и столбцов и наоборот. Тогда фактор-множество $\mathfrak{G}_m/\varepsilon$ изоморфно Σ_m и $\varphi_\varepsilon : G \to M(G)$ есть гомоморфизм $\mathfrak{G}_m$ на Σ_m.

Теорема 1. *$|\Sigma_m| \leqslant c_m$, где c_m — число связных графов с m вершинами.*

На множестве Σ_m определим отношение частичного порядка: будем говорить, что $M \leqslant M'$, если для любой частотной характеристики P и для любого спектра $\tilde{d}_1$ из M' найдется спектр $\tilde{d} \in M$, такой, что $C(\tilde{d}, P) \leqslant C(\tilde{d}_1, P)$.

Теорема 2. *Множество Σ_m содержит наибольший элемент $M_{\max} = M(K_m)$. Строками матрицы $M_{\max}$ являются все целочисленные векторы $\langle d_1, \ldots, d_m \rangle$, для которых $\sum\limits_{i=1}^{m} 2^{-d_i} = 1$, и только они.*

Теорема 3. *Множество Σ_m содержит наименьший элемент $M_{\min} = (\underbrace{1, \ldots, 1}_{m})$. $\varphi_\varepsilon^{-1}(M_{\min}) = \mathfrak{B}_m$, где $\mathfrak{B}_m$ — класс двудольных графов с m вершинами.*

Теорема 4. *$|\Sigma_m| \leqslant c_m - b_m + 1$, где c_m и b_m — соответственно числа связных и связных двудольных графов с m вершинами.*

В таблице ниже приведены значения $|\Sigma_m|$ при $m \leqslant 6$. Обратим внимание, что при $m \leqslant 4$ множество $[\Sigma_m, \leqslant]$ является линейно упорядоченным и может быть легко представлено диаграммой Хассе, причем при $m = 2$ имеем $M_{\max} = M_{\min} = M(K_2)$.

m	$\|\Sigma_m\|$	c_m	b_m	$c_m - b_m + 1$
2	1	1	1	1
3	2	2	1	2
4	4	6	3	4
5	13	21	5	17
6	58	112	17	96

Множество Σ_3 содержит два элемента $M_{\max} = M(K_3)$ и $M_{\min} = M(P_3)$, где P_3 — простой путь с тремя вершинами.

Множество Σ_4, кроме элементов $M_{\max} = M(K_4)$ и $M_{\min} = M(\mathfrak{B}_4)$, содержит элементы $M' = M(G')$, где граф $G' = K_4 - e$ получен из K_4 удалением ребра, и $M'' = M(G'')$, где граф $G'' = G' - e'$ получен из G' удалением ребра, инцидентного вершине степени 2. Для элементов множества Σ_4 имеем $M_{\min} \leqslant M(G'') \leqslant M(G') \leqslant M_{\max}$.

Для частично упорядоченного множества $[\Sigma_5, \leqslant]$, не являющегося цепью и содержащего 13 элементов, построение диаграммы усложняется. С использованием результатов [4] при $m \geqslant 6$ получено описание верхних четырех ярусов диаграммы Хассе для $[\Sigma_m, \leqslant]$.

Список литературы

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

[2] *Марков Ал. А., Смирнова Т. Г.* О словарных раскрасках и некоторых совершенных графах // Дискретная математика. — 1990. — Т. 2, № 2. — С. 16–32.

[3] *Жильцова Л. П.* Об алфавитном кодировании контекстно-свободных языков // Комбинаторно-алгебраические методы в прикладной математике: Межвуз. тематич. сб. научн. тр. — Горький, 1983. — С. 106–123.

[4] *Смирнова Т. Г.* Алфавитное кодирование в классе локально-префиксных кодов // Восьмая Межд. конф. «Дискретные модели в теории управляющих систем»: Москва, 6–9 апреля, 2009. Электронный сборник материалов конференции. — М., 2009. — С. 170–172.

Релаксационный субградиентный метод минимизации овражных выпуклых функций

П. И. Стецюк

stetsyukp@gmail.com

Институт кибернетики им. В. М. Глушкова НАН Украины, Киев

Рассмотрим задачу:

$$\text{найти} \quad x^\star = \arg\min_{x \in R^n} f(x), \tag{1}$$

где $f(x)$ — выпуклая функция и ее субградиент $\partial f(x)$ для $\forall x \in R^n$ и $\forall x^\star \in X^\star$ удовлетворяет следующему неравенству:

$$(x - x^\star, \partial f(x)) \geqslant \gamma(f(x) - f^\star), \quad \text{где } \gamma \geqslant 1. \tag{2}$$

Здесь $X^\star$ — ограниченное множество точек минимума функции $f(x)$, $f^\star$ — минимальное значение функции $f(x)$: $f^\star = f(x^\star)$, $x^\star \in X^\star$. Параметр γ определяет величину максимального сдвига по выпуклости функции $f(x)$ в неравенстве (2) и может быть использован для специальных классов выпуклых функций, для которых $\gamma > 1$. Так, например, градиент выпуклой квадратичной функции удовлетворяет неравенству (2) с параметром $\gamma = 2$.

Если $f^\star$ известно, то для решения задачи (1), (2) можно использовать метод amsg2p [1]. В его основу положен второй из субградиентных методов с преобразованием пространства и регулировкой шага Агмона–Моцкина–Шонберга [2]. Преобразование пространства в методе amsg2p реализуется с помощью одноранговогo эллипсоидального оператора [3] и только на тех итерациях метода, когда тупым является хотя бы один из углов — угол между двумя последовательными субградиентами либо угол между последним субградиентом и агрегатным вектором, который является выпуклой комбинацией вычисленных на предыдущих итерациях субградиентов. Отсюда и название метода, где «ams» указывает на способ регулировки шага в направлении нормированного антисубградиента, а «g2p» указывает, что ams-шаг используется в пространстве переменных, преобразованном с помощью двух последних субградиентов (g2) и агрегатного вектора (p).

Если в задаче (1) структура поверхностей уровня функции $f(x)$ такова, что преобразования пространства не нужны, то метод amsg2p равносилен известному субградиентному методу Б.Т. Поляка [4]

$$x_{k+1} = x_k - h_k \frac{\partial f(x_k)}{\|\partial f(x_k)\|}, \quad h_k = \frac{\gamma(f(x_k) - f^\star)}{\|\partial f(x_k)\|}, \ k = 0, 1, 2, \ldots, \quad (3)$$

с тем лишь отличием, что неравенство (2) для специальных классов функций реализует более сильные ams-шаги h_k, чем это делается для выпуклой функции при $\gamma = 1$. Отметим, что ams-шаг h_k еще называют шагом Поляка или шагом Агмона–Моцкина.

В настоящей заметке дополним метод amsg2p возможностью либо находить такую точку, где выпуклая функция $f(x)$ принимает значение, равное f_{min}, либо гарантировать достаточное условие, что такой точки не существует.

Метод amsg2p находит точку $x^\star_\varepsilon \in \{x : f(x) - f_{min} \leqslant \varepsilon\}$ и соответствующий ей неотрицательный номер итерации $k^\star_\varepsilon$. Отрицательный номер $k^\star_\varepsilon$ означает, что на итерации $|k^\star_\varepsilon|$ получено достаточное условие отсутствия точки x^*_ε.

Алгоритм-функция amsg2p: $(x^\star_\varepsilon, k^\star_\varepsilon) = \mathsf{amsg2p}(x_0, \varepsilon, r_0, f_{min}, \gamma)$

На итерации $k = 0$ заданы: начальное приближение $x_0 \in R^n$; начальный радиус r_0 такой, что $\|x_0 - x^\star\| \leqslant r_0$ для $\forall x^\star \in X^\star$; достаточно малое $\varepsilon > 0$. Вычислим $f(x_0)$ и $\partial f(x_0)$. Если $f(x_0) - f_{min} \leqslant \varepsilon$, то $x^\star_\varepsilon = x_0$, $k^\star_\varepsilon = 0$ и окончание алгоритма. Иначе положим $h_0 = \dfrac{\gamma(f(x_0) - f_{min})}{\|\partial f(x_0)\|}$, $\xi_0 = \dfrac{\partial f(x_0)}{\|\partial f(x_0)\|} \in R^n$, $p_0 = 0 \in R^n$, $B_0 = I$ — единичная $(n\times n)$-матрица. Перейдем к следующей итерации.

Пусть на k-й итерации получены $x_k \in R^n$, h_k, r_k, $\xi_k \in R^n$, $p_k \in R^n$, B_k — матрица $n\times n$. Для $(k+1)$-й итерации выполним пп. 1–5.

1. Вычислим $t_k = h_k/r_k$. Если $t_k > 1$, то $k^\star_\varepsilon = -(k+1)$ и окончание алгоритма. Иначе положим $r_{k+1} = r_k\sqrt{1-t_k^2}$ и вычислим очередное приближение
$$x_{k+1} = x_k - h_k B_k \xi_k.$$
2. Вычислим $f(x_{k+1})$ и $\partial f(x_{k+1})$. Если $f(x_{k+1}) - f_{min} \leqslant \varepsilon$, то $x^\star_\varepsilon = x_{k+1}$, $k^\star_\varepsilon = k+1$ и окончание алгоритма. Иначе положим
$$\xi_{k+1} = \frac{B_k^T \partial f(x_{k+1})}{\left\|B_k^T \partial f(x_{k+1})\right\|}, \quad h_{k+1} = \frac{\gamma(f(x_{k+1}) - f_{min})}{\left\|B_k^T \partial f(x_{k+1})\right\|}.$$
3. Вычислим $\lambda_1 = -p_k^T \xi_{k+1}$ и $\lambda_2 = -\xi_k^T \xi_{k+1}$. Положим
$$p_{k+1} = \begin{cases} \dfrac{\lambda_1}{\sqrt{\lambda_1^2+\lambda_2^2}}\, p_k + \dfrac{\lambda_2}{\sqrt{\lambda_1^2+\lambda_2^2}}\, \xi_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 > 0,\\ p_k, & \text{если } \lambda_1 > 0 \text{ и } \lambda_2 \leqslant 0,\\ \xi_k, & \text{если } \lambda_1 \leqslant 0 \text{ и } \lambda_2 > 0,\\ 0, & \text{если } \lambda_1 \leqslant 0 \text{ и } \lambda_2 \leqslant 0. \end{cases}$$
4. Вычислим $\mu_k = p_{k+1}^T \xi_{k+1}$. Если $-1 < \mu_k < 0$, то вычислим
$$B_{k+1} = B_k + (B_k\eta)\,\xi_{k+1}^T,\ \eta = \left(\frac{1}{\sqrt{1-\mu_k^2}} - 1\right)\xi_{k+1} - \frac{\mu_k}{\sqrt{1-\mu_k^2}} p_{k+1}$$

и пересчитаем

$$h_{k+1} = \frac{h_{k+1}}{\sqrt{1-\mu_k^2}}, \qquad p_{k+1} = \frac{1}{\sqrt{1-\mu_k^2}}(p_{k+1} - \mu_k \xi_{k+1}).$$

Иначе положим $B_{k+1} = B_k$ и $p_{k+1} = 0$.

5. Перейдем к следующей итерации с x_{k+1}, h_{k+1}, r_{k+1}, ξ_{k+1}, p_{k+1}, B_{k+1}.

Теорема. *Если $f_{min} \geqslant f^\star$, то для $\forall x^\star \in X^\star$ справедливы неравенства*

$$\|A_{k+1}(x_{k+1} - x^\star)\|^2 \leqslant \|A_k(x_k - x^\star)\|^2 - \left(\frac{\gamma(f(x_k) - f_{min})}{\|B_k^T \partial f(x_k)\|}\right)^2, \ k = 0, 1, ...,$$

Здесь $A_k = B_k^{-1}, A_{k+1} = B_{k+1}^{-1}$.

Теорема означает, что в методе amsg2p преобразование пространства таково, что в каждом очередном преобразованном пространстве переменных гарантируется уменьшение расстояния до множества точек минимума. Благодаря этому для каждой итерации $k > 1$ имеем неравенство

$$\|A_k(x_k - x^\star)\|^2 \leqslant \|x_0 - x^\star\|^2 - \sum_{i=0}^{k-1} \left(\frac{\gamma(f(x_i) - f_{min})}{\|B_i^T \partial f(x_i)\|}\right)^2 = r_0^2 - \sum_{i=0}^{k-1} h_i^2 = r_k^2,$$

с помощью которого обеспечивается достаточное условие отсутствия точки $x_\varepsilon^\star$ при $f_{min} < f^\star - \varepsilon$ (реализовано в п. 1 алгоритм-функции amsg2p).

Антиовражная техника в методе amsg2p направлена на уменьшение степени овражности поверхностей уровня выпуклых функций подобно тому, как это сделано в r-алгоритмах [5]. Детерминант матрицы B_k стремится к нулю, так как если на k-м шаге реализуется преобразование пространства, то

$$\det(B_{k+1}) = \det(B_k) \det(I + \eta \xi_{k+1}^T) = \det(B_k)\sqrt{1 - \mu_k^2}.$$

Для овражных функций это обеспечивает ускоренную сходимость метода amsg2p при произвольной начальной стартовой точке x_0 и достаточно малых значениях параметра ε ($\varepsilon \sim 10^{-10} - 10^{-14}$).

Последовательное уточнение границ на минимальное значение функции $f^\star$, например, по методу дихотомии позволяет использовать метод amsg2p для нахождения достаточно точного приближения к единственной точке минимума овражных функций. В докладе проиллюстрируем это на примере известной тестовой задачи maxquad [6], которая связана с минимизацией существенно овражной выпуклой кусочно-квадратичной функции $f(x) = \max_{1\leqslant k\leqslant 5} \varphi_k(x)$, $x{\in}R^{10}$. Здесь $\varphi_k(x) = x^T H_k x - b_k^T x$, H_k — симметричные $(10{\times}10)$-матрицы, такие, что $H_{kij} = e^{i/j}\cos(ij)\sin k$, если $i < j$, и $H_{kii} = i\,|\sin k|\,/10 + \sum\limits_{j\neq i} |H_{kij}|$, а компоненты векторов b_k определяются $b_{ki} = e^{i/k}\sin(ik)$.

Список литературы

[1] *Стецюк П. И.* Метод amsg2p для овражных выпуклых функций // Информационный бюллетень Ассоциации математического программирования. № 12. — Екатеринбург: УрО РАН, 2011. — С. 57–58.

[2] *Стецюк П. И.* Субградиентные методы переменной метрики, использующие шаг Агмона-Моцкина и одноранговый эллипсоидальный оператор // Труды АТИК. — 2007–2008. Кишинэу: Эврика, 2009. — Т. I (XII). — С. 16–25.

[3] *Стецюк П. И.* Ортогонализующие линейные операторы в выпуклом программировании // Кибернетика и системный анализ. — 1997. — № 3. — С. 97–119.

[4] *Поляк Б. Т.* Введение в оптимизацию. — М.: Наука, 1983.

[5] *Шор Н. З.* Методы минимизации недифференцируемых функций и их приложения — Киев: Наукова думка, 1979.

[6] *Lemarechal C.* Numerical experiments in nonsmooth optimization // Progress in nondifferentiable optimization / Ed. E. A. Nurminski. CP-82-58. — Laxenburg: International Institute for Applied System Analysis, 1982. — P. 61–68.

Граница смены триплетной периодичности в гене как область потенциальной склейки

Ю. М. Суворова, Е. В. Коротков

suvorovay@gmail.com, genekorotkov@gmail.com

Центр «Биоинженерия» РАН, Москва

В силу особенностей генетического кода гены, кодирующие белки, обладают характерным свойством — триплетной периодичностью (ТП), причем периодичность в них чаще всего не является явной. Анализ типов ТП показал, что ТП большинства генов принадлежит к одному из 2500 классов, это небольшое число по сравнению с общим количеством генов. В то же время существуют гены, содержащие участки с разными типами ТП [1]. Смена типа периодичности может быть свидетельством того, что данный ген был «склеен» из нескольких в процессе эволюции. Такие гены принято называть склеенными (*fused genes*) [2].

Поиск склеенных генов

Задача настоящей работы — создать алгоритм, который позволил бы определить границы перехода между участками с различной ТП в гене. Такие переходы по аналогии с генами мы называем *склейкой ТП*.

Различие ТП смежных участков. Далее мы рассматриваем генетическую последовательность как символьную $S = \{s(k), k = 1, 2, \dots L\}$ из алфавита $A = a, t, c, g$. Для некоторой координаты x будем рассматривать смежные участки длиной l (l кратно трем) $S_{10} = S(x-l+1, x)$ и $S_{20} = S(x+1, x+l)$. Также необходимо учесть возможность сдвига фазы ТП в точке x (так называемый «сдвиг рамки считывания» [3]). Для этого помимо S_{20} слева от x добавим к рассмотрению еще два участка: $S_{21} = S(x+2, x+l)$; $S_{22} = S(x+3, x+l)$ — участки длиной l со сдвигом на один и два символа соответственно.

Для определения размытой ТП в генетических текстах используются матрицы ТП — частотные матрицы $M(S_i)$ размера 3×4. Ячейка такой матрицы $m(i, j)$ — число символов типа i, находящихся в позиции j ($i = 1, \dots 4$; $i = 1$ для a, $i = 2$ для t, $i = 3$ для c, $i = 4$ для g) ($j = 1 \dots 3$) в рассматриваемой последовательности S_k. В нашем случае имеем четыре таких матрицы $M(S_{10})$, $M(S_{20})$, $M(S_{21})$

и $M(S_{22})$. Затем, чтобы избежать влияния отдельных символов, мы привели каждую матрицу к нормальному виду, используя

$$n(i,j) = \frac{m(i,j) - lp(i,j)}{\sqrt{lp(i,j)(1-p(i,j))}}, \quad p(i,j) = \frac{\left(\sum\limits_{j=1}^{3} m(i,j)\right)\left(\sum\limits_{i=1}^{4} m(i,j)\right)}{l^2}.$$

Обозначим приведенные матрицы $N(S_{10})$, $N(S_{20})$, $N(S_{21})$ и $N(S_{22})$. Условием смены ТП в точке x будет отличие матрицы левого участка от всех матриц справа от x:

$$D(N(S_{10}), N(S_{2k})) > D_0;\ \ \forall k\ \ (k = 0, 1, 2). \tag{1}$$

Здесь $D(N(S_{10}), N(S_{2k}))$ — мера расхождения между двумя матрицами, D_0 — пороговое значение. В качестве меры расхождения мы использовали поэлементное сравнение матриц:

$$D(N(S_{10}), N(S_{2k})) = \sum_{i=1}^{4}\sum_{j=1}^{3}\left(\frac{(n_{s_{10}}(i,j) - n_{s_{2j}}(i,k))}{\sqrt{2}}\right)^2.$$

Величина $D(N(S_{10}), N(S_{2k}))$ имеет асимптотическое распределение $\chi^2(6)$. В соответствии с условием (1) мы выбираем величину $D_{min}(x)$ — минимум из трех величин $D(N(S_{10}), N(S_{2k}))$ $(k = 0, 1, 2)$ в точке x. Тогда итоговая мера определяется как $Z = -\log(D_{min} > x) = -3\log(P(\chi^2(6)) > D_0))$.

Поиск точки перехода. Такие вычисления проводились на протяжении всей последовательность гена. Шаг координаты x равен трем символам. На каждом шаге варьируются границы рассматриваемых участков $l = 60...600$, l кратно трем. Координата, соответствующая максимуму итоговой величины Z в последовательности, может рассматриваться как место потенциальной склейки, если ее значение превышает некоторый порог Z_0.

Определение порогового значения. Для определения значимого уровня был использован метод статистических испытаний Монте-Карло. Описанным выше алгоритмом были обработаны все последовательности генов банка данных KEGG/Genes-48 (`http://www.genome.jp/kegg/genes.html`) [4], общий объем около 4 млн. последовательностей. Для создания контрольной выборки все гены KEGG-48 были перемешаны с сохранением триплетной структуры последовательностей. Т. е. отдельно перемешивались символы, стоящие на первых, вторых и третьих позициях последовательности. При таком перемешивании сглаживаются все существующие в гене неоднородности ТП. Из сравнения результатов обработки «случайных» и реальных последовательностей было определено пороговое значение величины Z на уровне 5%.

Результаты

На уровне 5% в банке данных KEGG-48 было найдено более 300 тысяч генов, содержащих минимум одну смену типа периодичности.

В 131323 случаях, используя программу поиска подобий BLAST [5], мы обнаружили, что участки с разным типом ТП могли изначально принадлежать разным генам, которые в процессе эволюции были «склеены» в одну последовательность. Возможно, что для остальных случаев такие предковые гены тоже существовали, но могли быть утеряны в процессе эволюции или еще не помещены в базу данных.

Список литературы

[1] *Frenkel F. E., Korotkov E. V.* Classification analysis of triplet periodicity in protein-coding regions of genes // Gene. — 2008. — V. 421, № 12. — P. 52–60.

[2] *Pasek S., Risler J. L., Brŭzellec P.* Gene fusion/fission is a major contributor to evolution of multi-domain bacterial proteins // Bioinformatics. — 2006. — V. 22, № 12. — P. 1418–1423.

[3] *Korotkov E. V., Korotkova M. A.* Study of the triplet periodicity phase shifts in genes // J. Integrative Bioinformatics. — 2010. — V. 7, № 3. — P. 131–141.

[4] *Kanehisa M., Goto S.* KEGG: Kyoto Encyclopedia of Genes and Genomes // Nucleic Acids Res. — 2000. — V. 28, № 1. — P. 27–30.

[5] *Altschul S. F., Gish W., Myers E. W., Lipman D. J.* Basic Local Alignment Search Tool // J. Molecular Biology. — 1990. — V. 215, № 3. — P. 403–410.

Метод вольтерровых функционально-операторных уравнений в теории оптимального управления распределенными системами

В. И. Сумин

v_sumin@mail.ru

Нижегородский государственный университет им. Н. И. Лобачевского

В теории оптимального управления некоторый компромисс между стремлением к общности построений (призванной выявить новые связи и закономерности), с одной стороны, и желанием получить те или иные результаты в удобной для приложений форме, с другой, достигается, по-видимому, при переходе к описанию управляемых систем на языке функционально-операторных (функциональных) уравнений (см., например, [1]). Указанным требованиям унификации построений в теории оптимального управления распределенными системами во многом удовлетворяет предложенная в [2] (см. также [3, 4]) форма описания *управляемых начально-краевых задач* (УНКЗ) с помощью *вольтерровых функционально-операторных уравнений* (ВФОУ) вида

$$z(t) = f\left(t, A[z](t), v(t)\right), \quad t \in \Pi, \quad z \in L_p^m \equiv L_p^m\left(\Pi\right), \qquad (1)$$

где $\Pi \subset \mathbf{R}^n$ – множество изменения независимых переменных $t = \{t^1, ..., t^n\}$; $f(.,.,.) : \Pi \times \mathbf{R}^l \times \mathbf{R}^s \to \mathbf{R}^m$ – заданная функция; $v(.) \in \mathcal{D} \subset L_k^s$ – управление; $A : L_p^m \to L_q^l$ – линейный ограниченный оператор, вольтерров на некоторой системе T подмножеств основного множества Π в том смысле, что для любого $H \in T$ сужение $A\left[z\right]|_H$ не зависит от значений $z|_{\Pi \setminus H}$; $p, q, k \in [1, +\infty]$. Приведенное определение «оператора, вольтеррова на системе множеств» из [2] — это многомерное обобщение известного определения А.Н. Тихонова функционального оператора типа Вольтерра [5]. К ВФОУ (1) с богатыми системами T естественным образом (обращением главной части) приводятся разнообразные УНКЗ для эволюционных уравнений (параболических, гиперболических, интегродифференциальных, с запаздываниями и др.) (см., например, [3, 6–10, 15–17, 20, 21]).

Переход от УНКЗ к эквивалентным ВФОУ (1) адекватен многим проблемам распределенной оптимизации. Он позволяет: получить конструктивные общие условия сохранения глобальной разрешимости УНКЗ при возмущении распределенных, граничных, начальных управлений и управлений, входящих в старшие коэффициенты уравнений (в этом случае управлением в (1) служит и оператор A), см., например, [2–4, 6–11] и обзор в [7]; показать, что для распределенных задач оптимизации характерно сильное вырождение особых управлений, когда вместе с *необходимыми условиями оптимальности* (НУО) первого порядка (например, поточечным принципом максимума) вырождаются и НУО второго порядка, и предложить новый общий способ вывода НУО особых управлений, использующий теорию тензорных произведений лебеговых пространств, см., например, [12, 13]; показать, что при выводе НУО сингулярность (в смысле Ж.-Л. Лионса [14]) часто можно «преодолевать» классическим способом (не прибегая к предложенному в [14] весьма, как правило, трудоемкому методу адаптированного штрафа), и решить ряд поставленных в [14] задач получения «сингулярных НУО» (см. [6]). Переход к ВФОУ оказывается естественным при обосновании принципа максимума (см., например, [15, 16]), численных методов оптимизации (см., например, [17, 18]) и при рассмотрении других вопросов теории оптимального управления распределенными системами (расширение оптимизационных задач, поточечные оценки решений УНКЗ, достаточные условия глобальной разрешимости нелинейных УНКЗ при всех допустимых управлениях и др.; см., например, [19–21].

Потребности теории оптимизации заставляют развивать собственно теорию вольтерровых операторов и уравнений в функциональных и абстрактных пространствах (см., например, [3, 4, 8, 11, 22]). Весьма полезным в теории оптимального управления оказалось введенное в [4] (см. также [11]) понятие равностепенно квазинильпотентного семейства операторов (см., например, [7, 10, 15, 16]).

В докладе предполагается дать обзор результатов, полученных в теории оптимального управления распределенными системами методом ВФОУ в последнее время.

Работа выполнена при финансовой поддержке ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (проект НК-13П-13) и АЦВП «Развитие потенциала высшей школы (2009–2011 гг.)» Минобрнауки РФ (регистрационный номер проекта 2.1.1/3927).

Список литературы

[1] *Варга Дж.* Оптимальное управление дифференциальными и функциональными уравнениями. — М.: Наука, 1977. — 622 с.

[2] *Сумин В. И.* Функционально-операторные вольтерровы уравнения в теории оптимального управления распределенными системами // ДАН СССР. — 1989. — Т. 305, № 5. — С. 1056–1059.

[3] *Сумин В. И.* Функциональные вольтерровы уравнения в теории оптимального управления распределенными системами. — Н. Новгород: Изд-во ННГУ, 1992. — 110 с.

[4] *Сумин В. И.* Управляемые функциональные вольтерровы уравнения в лебеговых пространствах // Вестник ННГУ. Математическое моделирование и оптимальное управление. — 1998. — № 2. — С. 138–151.

[5] *Тихонов А. Н.* О функциональных уравнениях типа Вольтерра и их применениях к некоторым задачам математической физики // Бюллетень Московского университета. Секция А. — Т. 1, № 8. — С. 1–25.

[6] *Сумин В. И.* К проблеме сингулярности распределенных управляемых систем. I; II; III; IV // Вестник ННГУ. Математическое моделирование и оптимальное управление. — 1999. — № 2(21). — С. 145–155; — 2001. — № 1(23). — С. 198–204; — 2002. — № 1(25). — С. 164–174; — 2004. — № 1(27). — С. 175–183.

[7] *Сумин В. И.* Проблема устойчивости существования глобальных решений управляемых краевых задач и вольтерровы функциональные уравнения // Вестник ННГУ. Математика. — 2003. № 1. — С. 91–108.

[8] *Сумин В. И., Чернов А. В.* О достаточных условиях устойчивости существования глобальных решений вольтерровых операторных уравнений // Вестник ННГУ. Математическое моделирование и оптимальное управление. — 2003. № 1. — С. 39–49.

[9] *Беляева О. А., Сумин В. И.* О задаче Коши для полулинейного волнового уравнения с управляемым старшим коэффициентом // Вестник Тамбовского университета. Естеств. и технич. науки. — 2007. — Т. 12, № 4. — С. 410–412.

[10] *Лисаченко И. В., Сумин В. И.* Об условиях устойчивости существования глобальных решений управляемой задачи Гурса–Дарбу // Вестник ННГУ. Математическое моделирование и оптимальное управление. — 2006. — № 2. — С. 64–81.

[11] *Сумин В. И.* Равностепенная квазинильпотентность: определения, признаки, примеры применения // Вестник Тамбовского университета. Естеств. и технич. науки. — 2010. — Т. 15, № 1. — С. 453–466.

[12] *Сумин В. И.* Сильное вырождение особых управлений в распределенных задачах оптимизации // ДАН СССР. — 1991. — Т. 320, № 2. — С. 295–299.

[13] *Сумин В. И.* Об особых управлениях поточечного принципа максимума в распределенных задачах оптимизации // Вестник Удмуртского государственного университета. Математика. Механика. Компьютерные науки. — 2010. — № 3. — С. 70–80.

[14] *Лионс Ж.-Л.* Управление сингулярными распределенными системами. — М.: Наука, 1987. — 368 с.

[15] *Сумин В. И.* Вольтерровы функциональные уравнения и принцип максимума для распределенных оптимизационных задач // Вестник ННГУ. Математическое моделирование и оптимальное управление. — 2004. — № 1. — С. 178–191.

[16] *Лисаченко И. В., Сумин В. И.* Принцип максимума для терминальной задачи оптимизации системы Гурса–Дарбу в классе функций с суммируемой смешанной производной // Вестник Удмуртского государственного университета. Математика. Механика. Компьютерные науки. — 2011. — № 2.

[17] *Сумин В. И.* Об обосновании градиентных методов для распределенных задач оптимального управления // Ж. вычисл. матем. и мат. физики. — 1990. — Т. 30, № 1. — С. 3–21.

[18] *Чернов А. В.* К вопросу о сходимости метода условного градиента в распределенных задачах оптимизации // Вестник ННГУ. — 2010. — № 2. — С. 124–130.

[19] *Сумин В. И.* О расширении оптимизационных задач, связанных с функциональными уравнениями в пространствах существенно ограниченных функций // Вестник ННГУ. Математическое моделирование и оптимальное управление. — 1998. — № 1. — С. 126–133.

[20] *Чернов А. В.* О поточечной оценке разности решений управляемого функционально-операторного уравнения в лебеговом пространстве // Матем. заметки. — 2010. — Т. 88, № 2. — С. 288–302.

[21] *Чернов А. В.* Об одном мажорантном признаке тотального сохранения глобальной разрешимости управляемого функционально-операторного уравнения // Изв. вузов. Математика. — 2011. — № 3. — С. 95–107.

[22] *Сумин В. И., Чернов А. В.* Операторы в пространствах измеримых функций: вольтерровость и квазинильпотентность // Дифференц. уравнения. — 1998. — Т. 34, № 10. — С. 1402–1411.

О регуляризованной теореме Куна–Таккера и ее применении в оптимальном управлении и некорректных задачах

М. И. Сумин

msumin@sinn.ru

Нижегородский государственный университет им. Н. И. Лобачевского

Обсуждаются так называемая регуляризованная теорема Куна–Таккера в недифференциальной форме для параметрической задачи выпуклого программирования в гильбертовом пространстве в случае сильно выпуклого функционала цели и возможности ее приложения при решении задач оптимизации, оптимального управления и некорректных задач. Эта теорема, доказательство которой основывается на методе двойственной регуляризации (см., например, [1, 2]), представляет собой утверждение в терминах минимизирующих последовательностей о возможности аппроксимации решения задачи выпуклого программирования точками минимума ее регулярной функции Лагранжа без каких-либо предположений регулярности самой оптимизационной задачи. Подчеркнем при этом, что аппроксимирующие решение задачи точки конструктивно указываются, поскольку они генерируются с помощью алгоритма двойственной регуляризации.

1. Регуляризованная теорема Куна–Таккера

Рассматриваемая параметрическая задача выпуклого программирования имеет вид:

$$(P_{p,r}) \qquad f(z) \to \min,\ Az = h + p,\ g(z) \leqslant r,\quad z \in \mathcal{D} \subset Z,$$

где $p \in H$, $r = (r_1, \ldots, r_m) \in R^m$ — параметры, $f : D \to R^1$ — непрерывный сильно выпуклый функционал, $A : Z \to H$ — линейный ограниченный оператор, $g(z) \equiv (g_1(z), \ldots, g_m(z))$, $g_i : D \to R^1$, $i = 1, \ldots, m$, — непрерывные выпуклые функционалы, $\mathcal{D}$ — выпуклое замкнутое множество, Z, H — гильбертовы пространства. Обозначим через $z^0_{p,r}$ решение задачи $(P_{p,r})$ в случае его существования. Введем необходимые обозначения

$$L_{p,r}(z, \lambda, \mu) \equiv f(z) + \langle \lambda, Az - h - p\rangle + \langle \mu, g(z) - r\rangle,$$

$$\mathcal{D}^{\varepsilon}_{p,r} \equiv \{z \in \mathcal{D} :\ \|Az - h - p\| \leqslant \varepsilon,\ g_i(z) \leqslant r_i + \varepsilon,\ i = 1, \ldots, m\},\ \ \varepsilon \geqslant 0$$

и определим минимизирующее приближенное решение (МПР) в смысле Дж. Варги задачи $(P_{p,r})$ как последовательность $z^k \in \mathcal{D}$, $k = 1, 2, \dots$, такую, что $f(z^k) \to \beta(p,r)$, $z^k \in \mathcal{D}^{\varepsilon^k}_{p,r}$, $\varepsilon^k \to 0$, $k \to \infty$. Здесь $\beta(p,r) \equiv \{f(z^0_{p,r})$, если $z^0_{p,r}$ существует; $+\infty$ в противном случае$\}$ — S-функция задачи $(P_{p,r})$. Определим также двойственную задачу

$$V_{p,r}(\lambda,\mu) \to \sup, \ \ (\lambda,\mu) \in H \times R^m_+, \quad V_{p,r}(\lambda,\mu) \equiv \inf_{z \in \mathcal{D}} L_{p,r}(z,\lambda,\mu).$$

Теорема 1. *Пусть $f : \mathcal{D} \to R^1$ — непрерывный сильно выпуклый субдифференцируемый функционал. Для существования ограниченного МПР в задаче $(P_{p,r})$ (и, значит, одновременно сходящегося сильно к $z^0_{p,r}$ при $k \to \infty$) необходимо и достаточно, чтобы существовала последовательность $(\lambda^k, \mu^k) \in H \times R^m_+$, $k = 1, 2, \dots$, такая, что*

$$L_{p,r}(z[\lambda^k,\mu^k],\lambda^k,\mu^k) = V_{p,r}(\lambda^k,\mu^k) \to \sup_{(\lambda,\mu) \in H \times R^m_+} V_{p,r}(\lambda,\mu),$$

$$z[\lambda^k,\mu^k] \equiv \operatorname{argmin}\{L_{p,r}(z,\lambda^k,\mu^k),\, z \in \mathcal{D}\} \in \mathcal{D}^{\varepsilon^k}_{p,r}, \ \varepsilon^k \to 0, \ k \to \infty,$$

$$\langle(\lambda^k,\mu^k),(Az[\lambda^k,\mu^k] - h - p, g(z[\lambda^k,\mu^k]) - r)\rangle \geqslant 0,$$

а последовательность $z[\lambda^k,\mu^k]$, $k = 1, 2, \dots$, была ограничена. При этом она является искомым МПР и $z[\lambda^k,\mu^k] \to z^0_{p,r}$, $k \to \infty$. В качестве последовательности $(\lambda^k,\mu^k) \in H \times R^m_+$, $k = 1, 2, \dots$, может быть взята последовательность, генерируемая алгоритмом двойственной регуляризации [1, 2], *в соответствии с которым $(\lambda^k,\mu^k) = \operatorname{argmax}\{V_{p,r}(\lambda,\mu) - \alpha_k\|(\lambda,\mu)\|^2,\, (\lambda,\mu) \in H \times R^m_+\}$, α_k, $k = 1, 2, \dots$, — произвольная последовательность сходящихся к нулю положительных чисел.*

Сформулированная теорема содержит в себе свой классический аналог и отличается от него двумя важными обстоятельствами:

1) она справедлива без каких-либо предположений регулярности (существования вектора Куна–Таккера) задачи $(P_{p,r})$;
2) она «устойчива» по отношению к ошибкам исходных данных и может использоваться, в частности, для решения некорректных задач.

Свойства сходимости в теореме 1 теснейшим образом связаны со свойствами субдифференцируемости выпуклой функции значений β

задачи $(P_{p,r})$. Она обобщается на случай задачи $(P_{p,r})$ с выпуклым функционалом цели, а также и на случай нелинейных задач математического программирования, когда роль классических функций Лагранжа берут на себя их модифицированные аналоги, конструкции которых являются следствиями дифференциальных свойств S-функций параметрических задач.

2. Приложения регуляризованной теоремы Куна–Таккера

Теорема Куна–Таккера в регуляризованной форме становится средством практического решения оптимизационных и некорректных задач, для которых характерным является отсутствие информации о разрешимости двойственной задачи.

Приложения в математическом программировании. Теорема 1 может быть непосредственно использована для приближенного решения задач математического программирования $(P_{p,r})$. При этом в случае существования седловой точки в задаче $(P_{p,r})$ конструктивно указываемая последовательность $(\lambda^k, \mu^k) \in H \times R^m_+$, $k = 1, 2, \ldots$, будет сходиться к нормальному решению двойственной задачи, а соответствующая последовательность $z[\lambda^k, \mu^k]$, $k = 1, 2, \ldots$, — к решению $z^0_{p,r}$ задачи $(P_{p,r})$. Если же двойственная задача не имеет решения, то последняя сходимость остается в силе, тогда как $\|(\lambda^k, \mu^k)\| \to +\infty,\ k \to \infty$.

Приложения в оптимальном управлении. Теорема 1 может быть непосредственно использована также для приближенного решения задач оптимального управления как сосредоточенными, так и распределенными системами, сводящихся к задаче математического программирования $(P_{p,r})$. В частности, одной из таких задач является параметрическая задача оптимального управления с операторным ограничением типа равенства и конечным числом функциональных ограничений-неравенств, имеющая вид

$$g_0(u) \equiv \int_0^T (\langle F(t)x[u](t), x[u](t)\rangle + \langle G(t)u(t), u(t)\rangle)dt \to \min,$$

$$\langle \varphi_1(t), x[u](t)\rangle = h(t) + p(t) \text{ при п.в. } t \in X,$$

$$\varphi_{2,i}(x[u](T)) \leqslant r_i \ \ i = 1, \ldots, m,$$

$$\dot{x} = A(t)x + B(t)u(t), \ \ x(0) = x_0 \in R^n, \ \ t \in [0, T], \ u \in \mathcal{D},$$

где $p \in H$, $r = (r_1, \ldots, r_m)^*$ — параметры, $g_0 : L_2(0,T) \to R^1$ — сильно выпуклый функционал, F, $A : [0,T] \to R^{n\times n}$, $B : [0,T] \to R^{n\times m}$, $G : [0,T] \to R^{m\times m}$ — непрерывные матрицы, φ_1, $h \in C[0,T]$ — заданные функции, $\varphi_{2,i} : R^n \to R^1$ — выпуклая, непрерывная вместе с градиентом $\nabla_x \varphi_{2,i}$ функция, $\mathcal{D} \equiv \{u \in L_2(0,T) : u(t) \in U$ при п.в. $t \in (0,T)\}$, $U \subset R^m$ — выпуклый компакт, $X \subset [0,T]$, $X = \mathrm{cl}\,\overset{\circ}{X}$, $Z \equiv L_2(0,T)$, $H \equiv L_2(X)$.

Приложения в некорректных задачах. В отличие от своего классического аналога теорема 1 может применяться для решения широкого класса некорректных задач, и в первую очередь для нахождения нормальных решений операторных уравнений первого рода $Az = h + p$ с линейным ограниченным оператором $A : Z \to H$, Z, H — гильбертовы пространства. Одним из наиболее известных представителей таких уравнений является, например, популярное в различных физических приложениях интегральное уравнение Фредгольма первого рода с интегральным оператором $A(z)(x) \equiv \int_a^b K(x,s)z(s)\,ds$, $c \leqslant x \leqslant d$, с $z \in Z \equiv L_2(a,b)$, с правой частью $h \in L_2(c,d)$, с параметром $p \in L_2(c,d)$, с непрерывным на прямоугольнике $\Pi \equiv [a,b]\times[a,b]$ ядром K, поиск нормального решения для которого сводится к задаче минимизации $\|z\|^2 \to \min$, $Az = h + p$, $z \in L_2(a,b)$.

Работа выполнена при поддержке АВЦП «Развитие научного потенциала высшей школы (2009–2011 гг.)» Минобрнауки РФ (проекты 2.1.1/3927, 2.1.1/13303) и ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (проект НК-13П-13).

Список литературы

[1] *Сумин М. И.* Регуляризация в линейно выпуклой задаче математического программирования на основе теории двойственности // Журн. вычисл. матем. и матем. физ. — 2007. — Т. 47, № 4. — С. 602–625.

[2] *Сумин М. И.* Некорректные задачи и методы их решения. Материалы к лекциям для студентов старших курсов: Учебное пособие. — Нижний Новгород: Издательство Нижегородского госуниверситета, 2009.

Оптимальный поиск экстремальной области функции

В. П. Тарасова

Институт машиноведения им. А. А. Благонравова РАН, Москва

Возникающие в исследовании прикладных оптимизационных задач функции являются, как правило, многоэкстремальными. В то же время общие эффективные методы решения таких задач отсутствуют. Поэтому представляется естественным рассматривать классы многоэкстремальных функций с ограничениями, позволяющими выделять каким-либо способом область наибольшего или наименьшего значения функции, и затем решать задачу оптимизации [1] в этой области.

В работе методом моделирования стратегии противника [2] решается задача оптимального поиска области наибольших значений для некоторого класса T вещественных функций. Функции из класса T определены в заданной области Ω m-мерного евклидова пространства, и удовлетворяют следующим условиям:

— для каждой функции f из класса T существует такая область наибольших значений Δ, $\Delta \subset \Omega$, в точках которой значения функции f превышают ее значения в любых других точках области Ω;
— области Δ, Ω, есть m-мерные кубы с ребрами, параллельными соответствующим осям координат;
— длина ребра гиперкуба Δ равна некоторому фиксированному вещественному δ, одному и тому же для всех функций класса T, длина ребра гиперкуба Ω равна $k\delta$, $k = 1, 2, \dots$.

Задача поиска. Требуется для любой функции f из T по информации о значениях этой функции в p точках гиперкуба Ω, $p > (k-1)^m$, приближенно найти область наибольших значений Δ функции f.

Поиск будет осуществляться p-ходовыми стратегиями вычислителя, принадлежащими некоторому множеству и определяющими последовательный выбор точек из области Ω. То есть, выбор каждой очередной точки делается с учетом всей предыдущей информации.

Предполагается, что точное значение любой функции из класса T в выбранной точке становится сразу известным вычислителю.

Областью неопределенности назовем наименьшую по объему область из Ω, содержащую область наибольших значений Δ. До начала поиска областью неопределенности является вся область Ω. Область неопределенности определяется после каждого хода вычислителя по правилам, реализующим свойства функций из класса T, которые назовем правилами подсчета выигрыша.

Приведем пример двух таких правил:

1. если две точки, взятые в гиперкубе Ω, удалены друг от друга на расстояние, большее δ, то точка с меньшим значением функции не принадлежит области наибольших значений Δ;
2. если обе точки, находящиеся на расстоянии меньшем или равном δ, не принадлежат области Δ, то весь отрезок с концами в этих точках также не принадлежит области Δ.

Замечание. Приведенные правила всегда справедливы для любых двух точек, лежащих на одной прямой, при условии, что эта прямая параллельна одной из осей координат. Это утверждение касается и всех других правил подсчета выигрыша.

В работе используется теоретико-игровой язык, удобный для исследования такого рода задач, и к тому же позволяющий формализовать некоторые интуитивные понятия. Задача поиска области наибольших значений для любой функции из класса T представляется многоходовой (позиционной) антагонистической игрой $J\langle T\rangle$ вычислителя (первого игрока) с «природой» (вторым игроком). В качестве «природы» выступает класс T.

Стратегии вычислителя (первого игрока) принадлежат некоторому множеству $\mathfrak{R}$ p-ходовых детерминированных стратегий, и определяют выбор точек из области Ω. Стратегиями «природы» являются функции класса T.

Ходы игроков в игре делаются поочередно: ход первого игрока (выбор точки из области неопределенности) — ответный ход второго игрока (значение функции в этой точке). Такая пара называется ходом в игре.

Результатом стратегии поиска, или выигрышем первого игрока является объем области неопределенности, полученной после p ходов в игре с применением правила подсчета выигрыша. Эффективность любой стратегии A из множества $\mathfrak{R}$ в игре $J\langle T\rangle$ (в классе T) оценивается ее гарантированным выигрышем $\delta_p(A,T)$, равным $\delta_p(A,T) = \max\limits_{f\in T}\delta_p(A,f)$.

Задача оптимального поиска. Под задачей оптимального поиска области наибольших значений для функций класса T понимается задача построения оптимальной стратегии первого игрока (вычислителя) в игре $J\langle T\rangle$. Стратегия A_0 первого игрока является оптимальной (минимаксной) в игре $J\langle T\rangle$, если она имеет минимальный гарантированный выигрыш, то есть $\delta_p(A_0,T) = \min\limits_{A\in\mathfrak{R}}\delta_p(A,T)$.

Основная идея метода моделирования стратегии противника заключается в сведении задачи поиска оптимальной стратегии первого игрока в игре $J\langle T\rangle$ к такой же задаче в некоторой игре $J\langle S\rangle$, которая отличается от игры $J\langle T\rangle$ только множеством стратегий второго игрока, состоящим из одной специальной стратегии S, обладающей следующими свойствами:

1. В точках (x_i), выбираемых по любой стратегии первого игрока, стратегия S выдает значения $s(x_i)$, и при этом существует функция f из класса T такая, что $f(x_1) = s(x_1), \ldots, f(x_p) = s(x_p)$.
2. К значениям, выдаваемым стратегией S, применяются те же правила подсчета выигрыша, что и в игре $J\langle T\rangle$.
3. Стратегия первого игрока из множества $\mathfrak{R}$ стратегий оптимальна в игре $J\langle T\rangle$ тогда и только тогда, когда она оптимальна в игре $J\langle S\rangle$, и ее гарантированный выигрыш в игре $J\langle T\rangle$ равен ее выигрышу в игре $J\langle S\rangle$.

Стратегии S с такими свойствами мы называем синкретическими (синкретическим противником) для игры $J\langle T\rangle$.

Конструктивное определение синкретической для игры $J\langle T\rangle$ стратегии S является главной трудностью при решении задачи. Понятие синкретической стратегии представляет собой формализацию интуитивного понятия «худшей» для вычислителя функции в классе.

В соответствии с методом моделирования стратегии противника, сначала строится предполагаемая синкретическая стратегия S^0, которая должна удовлетворять условиям 1, 2, и быть походово-оптимальной стратегией второго игрока в игре $J\langle S^0\rangle$.

Затем определяется оптимальная против S^0 стратегия вычислителя, и проверяется выполнение равенства в условии 3. Если это равенство выполняется, то стратегия S^0 будет синкретической стратегией S для игры $J\langle T\rangle$, а найденная оптимальная против нее стратегия первого игрока — оптимальной стратегией в игре $J\langle T\rangle$.

Теорема. *Оптимальными стратегиями первого игрока в игре $J\langle S\rangle$ являются такие стратегии, принадлежащие множеству $\mathfrak{R}$, которые находят одну точку из области наибольших значений Δ, за наименьшее число ходов, а затем осуществляют оптимальный поиск отрезков наибольших значений функции из T* [3] *на каждой из m линий, проходящих через найденную точку, и параллельных осям координат. Эти же стратегии будут оптимальными стратегиями вычислителя в игре $J\langle T\rangle$.*

Численный результат. Если A_0 — любая из описанных оптимальных p-ходовых стратегий, то она находит первую точку из области Δ за $(k-1)^m$ ходов, а оставшиеся $p-(k-1)^m$ ходов используются для оптимального одномерного поиска. В случае, если выполняется условие $n = \lfloor p-(k-1)^m \rfloor / m$ — целое и большее единицы, то по окончании поиска будет найден m-мерный куб Γ, принадлежащий области Ω и содержащий область Δ с ребром, длина которого равна $(\delta + \delta/F_n)$, где F_n — n-е число Фибоначчи: $F_n = F_{n-1} + F_{n-2}$, $F_0 = F_1 = 1$.

Гарантированный выигрыш $\delta_p(A_0, T)$ стратегии вычислителя A_0, оптимальной в игре $J\langle T\rangle$, равен выигрышу этой стратегии в игре $J\langle S\rangle$ — объему m-мерного куба Γ: $\delta_p(A_0, T) = (\delta + \delta/F_n)^m$.

В случае, если величина $p-(k-1)^m$ не кратна m, но для поиска по каждой из m линий, проходящих через найденную из области Δ точку, остается больше одного хода, то получается m-мерный параллелепипед, содержащий область Δ.

Список литературы

[1] *Тарасова В. П.* О классах многоэкстремальных функций, допускающих поиск глобального экстремума методом Фибоначчи // ЖВМ и МФ. — 2004. — Т. 44, № 1. — С. 87–92.

[2] *Тарасова В. П.* Метод стратегии противника в задачах оптимального поиска. — М.: Издательство МГУ, 1988.

[3] *Тарасова В. П.* Оптимальные алгоритмы поиска отрезка наибольших значений для некоторого класса функций // ЖВМ и МФ. — 1981. — Т. 21, № 5. — С. 1108–1115.

Восстановление графа агентом с ограниченными ресурсами

Е. А. Татаринов

MDgerelo@yandex.ru

Институт прикладной математики и механики НАН Украины, Донецк

Введение

Рассматривается задача распознавания конечного, неориентированного графа без петель и кратных ребер при помощи агента, который перемещается по нему и изменяет метки на вершинах и ребрах графа [1]. Целью данной работы было построение алгоритма восстановления графа, для которого $T^*(n) = O(n)$. В настоящей работе предложен новый алгоритм восстановления неизвестного графа, в котором оптимизируется количество шагов и различных камней, которые используются при проведении эксперимента по восстановлению графа. Уменьшение количества шагов происходит за счет использования агентом дополнительных ресурсов (камней и счетчиков). Данный алгоритм основан на методе восстановления графа при помощи построения на его вершинах неявной нумерации [2].

Основные обозначения

Рассматриваются конечные, неориентированные графы без петель и кратных ребер. Все неопределяемые понятия общеизвестны, и их можно найти, например, в [3–5]. Пусть $G = (V, E)$ — граф, у которого V — множество вершин, E — множество ребер, $E \subseteq V \times V$, мощности n и m соответственно. Ясно что $m \leqslant \frac{n(n-1)}{2}$. Тройку $((u, v), v)$ будем называть инцидентором («точкой прикосновения») ребра (u, v) и вершины v. Множество таких троек обозначим I. Множество $L = V \cup E \cup I$ назовем множеством элементов графа G. Краской будем называть ресурс, который имеется у агента в неограниченном количестве, а камнем — ресурс, который имеется у агента в единичном экземпляре. Краски и камни образуют множество меток M, которые могут быть использованы при раскраске элементов графа G. Если элемент графа метится некоторой краской, то предыдущая краска стирается и вместо нее наносится новая. Если элемент графа метят камнем, то существующая на нем краска не уничтожается, а становится «невидимой» до тех пор, пока камень будет находиться на элементе. Обозначим h_t максимальную высоту дерева при поиске в глубину.

Окрестностью $O(v)$ вершины v будем называть множество элементов графа, состоящее из вершины v, всех вершин u смежных с v, всех ребер (u, v) и всех инциденторов $((v, u), v), ((v, u), u)$. Изоморфизмом графа G и графа H назовем такую биекцию $\varphi : V_G \to V_H$, что $(u, v) \in E_G$ точно тогда, когда $(\varphi(u), \varphi(v)) \in E_H$. Изоморфные графы равны с точностью до обозначения вершин и раскраски их элементов.

Постановка задачи

Пусть задан класс K графов: конечных, неориентированных, без петель и кратных ребер, все элементы которых окрашены краской w. Задан мобильный агент, который передвигается по графу из вершины v в вершину u по ребру (u, v). При этом он может изменить окраску вершин v, ребра (u, v), инциденторов $((v, u), v), ((v, u), u)$ метками из множества M. Находясь в вершине v, агент A воспринимает («видит») метки всех элементов окрестности $O(v)$ и на этом основании определяет, по какому ребру (u, v) он будет перемещаться и как будет перекрашивать элементы из $O(v)$. Агент A обладает конечной, неограниченно растущей внутренней памятью, в которой фиксируется результат функционирования агента на каждом шаге и, кроме того, постепенно выстраивается представление графа G, вначале неизвестного агенту, списками ребер и вершин.

Надо разработать такой алгоритм функционирования (т. е. передвижения по графу и раскраски его элементов) агента A, что он, будучи помещен в произвольную вершину произвольного, неизвестного ему графа $G \in K$, через конечное число шагов построит граф H, изоморфный G (т. е. восстановит G).

Стратегия алгоритма

Алгоритм основан на методе обхода графа в глубину. Стратегия обхода графа в глубину хорошо известна [4, 5]. Она изменена с учетом того, что агенту граф изначально неизвестен, агент может воспринимать локальную информацию о графе и строить на вершинах графа нумерацию в порядке посещения вершин (M-нумерация) [3].

Реализовывать нумерацию агент будет неявно, при помощи набора камней. При этом для каждого камня i существует счетчик количества непомеченных ребер, которые есть в окрестности текущей вершины, перед тем как агент установит в ней камень i. Счетчик уменьшается на единицу, когда агент «видит» в окрестности текущей вершины вершину, помеченную камнем i. Когда счетчик обнулится, камень станет свободным и агент перейдет в вершину, снимет с нее камень i, вершину пометит краской b и вернется обратно. Агент использует для пометки свободный на данный момент камень. Камень считается свободным, если его счетчик равен нулю. Свободный на данный момент камень — камень с наименьшим номером. Камень i ассоциируется с M-номером, который неявно был присвоен вершине в момент установки в ней камня i. Такая ассоциация сохраняется до тех пор, пока камень не станет свободным.

В процессе обхода графа в глубину агент строит неявное дерево поиска в глубину, относительно которого ребра графа разбиваются на два множества: древесные и обратные. Древесные ребра проходятся два раза — в прямом и обратном направлениях. При первом прохождении по древесному ребру оно восстанавливается в графе G и один из его инциденторов метится краской r. После перехода по ребру агент устанавливает в новой вершине свободный камень и восстанавливает все обратные ребра текущей вершины. Для восстановления обратных ребер агенту не требуется переходить по ним, поскольку, находясь в текущей вершине, помеченной камнем i, он увидит камень j, установленный в вершине, с которой смежно восстанавливаемое обратное ребро. Зная i и j, агент может однозначно определить ассоциируемые с ними неявные M-номера и восстановить обратное ребро. При установке в вершине камня она добавляется в помеченный путь, при пометке ее краской b она удаляется из помеченного пути.

Алгоритм останавливается, когда помеченный путь будет пустым, а все вершины будут помечены краской b.

Теорема 1. *Выполняя алгоритм восстановления графа, агент восстановит любой граф G, с точностью до изоморфизма.*

Теорема 2. *Агент восстановит любой граф G, с точностью до изоморфизма, за время $O(n)$, при этом будет использовано $O(m)$ ячеек памяти, две краски и h_t камней.*

Выводы

Предложен новый полиномиальный алгоритм восстановления графа, который использует $O(n)$ шагов, $O(n)$ ячеек памяти, две краски r и b, а так же h_t камней, $h_t \leqslant n$. Предложенный алгоритм, также как и алгоритм [6], имеет линейную сложность, но использует не h_t красок, а h_t камней. Кроме того, предложенный алгоритм использует две различные краски, как и алгоритмы [2, 7, 8], но в отличие от последних, имеющих нелинейную сложность, предложенный алгоритм, за счет использования дополнительных камней, имеет линейную сложность. Модификации предложенного алгоритма позволяют уменьшить количество используемых камней для некоторых видов графов. Найдены виды графов, для восстановления которых агент будет использовать постоянное число.

Список литературы

[1] *Dudek G., Jenkin M.* Computational principles of mobile robotic. — Cambridge Univ., 2000.

[2] *Грунский И. С., Тихончев М. Ю.* О распознавании графов конечным автоматом // Вестник Донецкого университета. Серия А. Естественные науки. — 2001. — Вып. 2. — С. 351–356.

[3] *Грунский И. С., Татаринов Е. А.* Распознавание конечного графа блуждающим по нему агентом // Вестник Донецкого университета. Серия А. Естественные науки. — 2009. — Вып. 1. — С. 492–497.

[4] *Татаринов Е. А.* M-нумерация как метод распознавания графов // Збірник наукових праць «Питання прикладної математики математичного моделювання». — 2010. — С. 260–272.

[5] *Татаринов Е. А.* Базовый алгоритм восстановления конечного графа // Труды ИПММ НАН Украины. — 2010. — Т. 21.

[6] *Касьянов В. Н., Евстигнеев В. А.* Графы в программировании: обработка, визуализация и применение. — С. 216–227. СПб.: БХВ — Петербург, 2003.

[7] *Ахо А., Хопкрофт Дж., Ульман Дж.* Построение и анализ вычислительных алгоритмов. — М.: Мир, 1979.

[8] *Кормен Т., Лейзерсон Ч., Ривест Р.* Алгоритмы: построение и анализ. — М.: МЦНМО, 2001.

Диагностика управляющих автоматов со счетно-бесконечными множествами состояний

В. А. Твердохлебов

tverdokhlebovva@list.ru

Институт проблем точной механики и управления РАН, Саратов

Рассматривается диагностирование дискретных детерминированных автоматов со счетно-бесконечными множествами состояний. Используется новый способ задания законов функционирования автоматов символьными и числовыми графиками, точки которых представляют пары символьных и числовых автоматных отображений. Такой новый способ задания автоматов со счетно-бесконечным множеством состояний позволяет доопределять частичные автоматные отображения классическими методами интерполяции. Методы распознавания автоматов при таком способе задания законов их функционирования базируются на методах анализа, синтеза и распознавания геометрических кривых линий. Основными являются теоремы о связях автоматных отображений и произвольных геометрических кривых.

Введение

Рассматривается распознавание автомата средствами эксперимента с конечным семейством дискретных детерминированных автоматов со счетно-бесконечными множествами состояний. В связи с этим исключаются задания автоматов таблицами, графами, матрицами и т. п. Используется разработанное и изложенное в работах [1–3] задание автоматных отображений точками на аналитически заданных геометрических кривых. Для этого линейно упорядочиваются последовательности входных и последовательности выходных сигналов, на основании чего линейно упорядочивается автоматное отображение. Полученный символьный график преобразуется в график точек с целочисленными координатами (номерами последовательностей по введенным линейным порядкам). Числовая форма автоматного отображения позволяет доопределять частично заданные законы функционирования автомата классическими методами интерполяции графиков. Размещение точек геометрического образа законов функционирования автомата на аналитически заданных кривых позволяет на основе решений равенств и неравенств, построенных для уравнений кривых, определять области эффективного диагностирования.

Геометрические образы законов функционирования автоматов

Дискретный детерминированный автомат $A = (S, X, Y, \delta, \lambda)$, где S — счетно-бесконечное множество состояний, X и Y — конечные множества входных и выходных сигналов, $\delta : S \times X \to S$ и $\lambda : S \times X \to Y$ — функции переходов и выходов, функционирует в соответствии с уравнениями динамики $s(t+1) = \delta(s(t), x(t)), y(t) = \lambda(s(t), x(t))$. Функции δ и λ распространяются до функций вида $\delta : S \times X^* \to S$ и $\lambda : S \times X^* \to Y$, где X^* — множество всех конечных последовательностей в алфавите X, по правилам: $(\forall s \in S)(\forall x \in X)(\forall p \in X^*)\{\delta(s, xp) = \delta(\delta(s, x), p)\}$ и $(\forall s \in S)(\forall x \in X)(\forall p \in X^*)\{\lambda(s, px) = \lambda(\delta(s, p), x)\}$. Если ввести линейные порядки ω_1 на X^* и ω_2 на Y, то автоматное отображение $\rho_s = \bigcup\limits_{p \in X^*} \{(p, \lambda(s, p))\}$ может быть преобразовано в два графика: символьный график в системе координат с осью абсцисс (X^*, ω_1) и осью ординат (Y, ω_2) и числовой график в главном квадранте прямоугольной декартовой системы координат на плоскости на основе замены символьных координат точек их номерами по порядкам ω_1 и ω_2. Предполагается, что автоматное отображение ρ_s представляет автомат со счетно-бесконечным множеством состояний. Это означает существенное снятие ограничений в задании автоматного отображения точками графика.

Интерполяция частично заданных законов функционирования автоматов

Числовая форма автоматного отображения, представленная графиком, позволяет доопределять частично заданные законы функционирования автоматов на глубокой и разработанной основе — доопределять графики классическими методами интерполяции (Ньютона, Лагранжа, Гаусса, Стирлинга, Бесселя и т. д.). Для этого все или некоторые известные точки автоматного отображения полагаются узлами интерполяции. Метод интерполяции имеет вид принимаемой гипотезы. Аналитическая формула геометрической кривой, на которой расположены точки автоматного отображения, позволяет любое счетное множество точек кривой рассматривать как точки автоматного отображения.

Распознавание автоматов, заданных геометрическими образами

Пусть $A = (S, X, Y, \delta, \lambda, s_0)$ — инициальный дискретный детерминированный автомат с конечным или счетно-бесконечным множеством состояний S, ρ_s — соответствующее ему автоматное отображение, G — числовой график автоматного отображения. Предположим далее, что множество дефектов I реальной системы R задано семейством автоматов $\alpha = \{A_i\}_{i \in I}$, где $A_i = (S_i, X, Y, \delta_i, \lambda_i, s_{0i})$, а $\beta = \{G_i\}_{i \in I}$ — семейство числовых графиков автоматных отображений. Пусть, кроме этого, числовые графики семейства β размещены на геометрических кривых линиях вида $y = f_i(x)$, где $i \in I$. Неравенства вида $f_i(x) \neq f_j(x)$, где $i \neq j$, определяют области, в которых содержащиеся входные последовательности $p \in X^*$ являются диагностическими тестами для пары дефектов (i, j). Равенства вида $f_i(x) = f_j(x)$ определяют области входных последовательностей, неэффективных для диагностирования.

Принципиально важным является установление взаимосвязей между автоматными отображениями автоматов со счетно-бесконечным множеством состояний и геометрическими кривыми линиями.

Теорема 1. *Пусть* $A = (S,\ X,\ Y,\ \delta,\ \lambda,\ s_0)$ *— инициальный дискретный детерминированный автомат с конечным или счетно-бесконечным множеством состояний* S, ω_1 *— линейный порядок на* X^* *и* $(\alpha_0, \alpha_l]$ *— полуинтервал на оси ординат, где* $l = |Y|$. *Тогда*

— *для любого взаимно-однозначного отображения «в»* $\varphi : N^+ \to R$, *где для любых* $n, n' \in N^+$ *из* $n < n'$ *следует* $\varphi(n) < \varphi(n')$;

— *для любого разбиения полуинтервала* $(\alpha_0, \alpha_l]$ *на* l *полуинтервалов* $(\alpha_0, \alpha_1]$, $(\alpha_1, \alpha_2]$, $\ldots$, $(\alpha_{l-1}, \alpha_l]$ *и взаимно-однозначного отображения* $\nu : Y \to \{(\alpha_{i-1}, \alpha_i], 1 \leqslant i \leqslant l\}$,

пара чисел (j, β), *где* $j \in Pr_2\varphi$ *и* $\beta \in (\alpha_0, \alpha_l]$, *однозначно определяет пару* (p, y_i), *для которой* j *— номер* $p \in X^*$ *по порядку* ω_1 *и* $\beta \in (\alpha_{i-1}, \alpha_i]$.

Теорема 2. *Любые:*

— *геометрическая кривая* $y = f(x)$*;*
— *последовательность* h *точек*

$$(x_{i_1}, f(x_{i_1})), (x_{i_2}, f(x_{i_2})), ..., (x_{i_j}, f(x_{i_j})), \ldots,$$

где $x_{i_1} < x_{i_2} < \cdots < x_{i_j} < \ldots$*;*
— *число* $m \in N^+$ *и разбиение последовательности* h *на подпоследовательности из* m *элементов каждая;*
— *полуинтервал* $\Delta = (\alpha, \beta]$ *на оси ординат, где* $\min\limits_{x \in \Delta} f(x) < \alpha < \beta \leqslant \max\limits_{x \in \Delta} f(x)$*;*
— *разбиение полуинтервала* Δ *на конечное множество полуинтервалов вида* $(\alpha, \alpha_1]$, $(\alpha_1, \alpha_2]$, $\ldots$, $(\alpha_{l-1}, \beta]$, *где* $l \in N^+$,

однозначно определяют геометрический образ законов функционирования дискретного детерминированного автомата с конечным или счетно-бесконечным множеством состояний, с m *входными и* l *выходными сигналами.*

Выводы. Новый способ задания управляющих автоматов — представление автоматных отображений числовыми графиками, размещенными на аналитически заданных геометрических кривых линиях, — снимает ограничения на множество состояний автомата (быть конечным множеством). После представления законов функционирования дискретных детерминированных автоматов числовыми графиками, размещенными на геометрических кривых линиях, поиск решения задач диагностирования автоматов может производиться с использованием методов анализа геометрических кривых линий.

Список литературы

[1] *Твердохлебов В. А.* Техническое диагностирование в геометрической интерпретации задач, моделей и методов // Автоматизация проектирования дискретных систем. — Минск: Белорус. гос. ун-т, 1995. — С. 97.

[2] *Твердохлебов В. А.* Геометрические образы конечных детерминированных автоматов // Изв. Сарат. ун-та. Новая серия. Сер. Математика, механика, информатика. — 2005. — Т. 5, № 1. — С. 141–153.

[3] *Твердохлебов В. А.* Геометрические образы законов функционирования автоматов. — Саратов: Научная книга, 2008.

О построении восстановлений баз данных для некоторых классов формул-ограничений

Е. Е. Трифонова

etrifonova@yandex.ru

Институт прикладной математики им. М. В. Келдыша РАН, Москва

В работе рассматривается использование формальных логических моделей для устранения противоречий в содержимом баз данных. *Схемой базы данных* будем называть некое конечное множество предикатов $\mathbb{P}$ и множество ограничений, записанных в виде формулы Φ. *Базой данных* D будем называть совокупность таблиц $\mathbb{T}$ и формулы Φ: $D = \langle \mathbb{T}, \Phi \rangle$. При этом каждом предикату P из множества предикатов $\mathbb{P}$ ставится в соответствие таблица T из $\mathbb{T}$, которая определяет значение истинности предиката. Элементом таблицы является *кортеж*.

Будем обозначать множество кортежей, которые присутствуют в таблице T, как X_T, а множество кортежей, которые присутствуют в базе D, как $X = X(D)$, $X = \bigcup\limits_{T \in \mathbb{T}} X_T$. Определим множество функций $F : X \to \mathbb{N}$, где $\mathbb{N}$ — множество натуральных чисел. *Вспомогательным предикатом* от двух переменных $g(x_1, x_2)$ будем называть выражение вида $(f_i(x_1) \mathcal{R} f_j(x_2))$, где $f_i, f_j \in F$, а $\mathcal{R}$ — одно из отношений $<$, $>$, $\leqslant$, $\geqslant$, $\neq$, $=$. Пусть G — множество всевозможных вспомогательных предикатов.

В качестве формулы-ограничения Φ будем рассматривать замкнутые формулы на языке первого порядка, записанные с использованием связок $\vee$, $\&$, $\neg$, $\to$, предикатов из $\mathbb{P}$ и вспомогательных предикатов из G. Переменные принимают значения из множества кортежей X. Если формула Φ истинна на множестве X, то базу данных D будем называть *непротиворечивой*. Если все таблицы базы данных пусты, то считаем, что любая формула Φ, удовлетворяющая вышеуказанным условиям, истинна.

Если формула Φ ложна на множестве X, то будем говорить, что в базе данных D содержатся противоречия. Устранять противоречия будем посредством удаления кортежей из базы данных. *Восстановлением* Q для базы данных D будем называть базу данных, для которой выполняется следующее:

1. Схемы баз данных Q и D совпадают.
2. Каждый кортеж из таблицы Q содержится в соответствующей таблице D.
3. База данных Q не содержит противоречий.
4. Добавление к Q любого кортежа из D, который в Q не содержится, в соответствующую таблицу приводит к тому, что в Q возникают противоречия.

Будем обозначать Z_Q множество кортежей, удалённых из D для получения восстановления Q: $Z_Q = X(D) \setminus X(Q)$. Для каждой базы, содержащей противоречия, существует некоторое множество восстановлений. Те из них, которые содержат наибольшее число кортежей, будем называть *наилучшими восстановлениями*. Множество наилучших восстановлений для базы D будем обозначать $\mathbb{Q}_{max} = \mathbb{Q}_{max}(D)$.

Пусть задана формула Φ и требуется построить наилучшее восстановление $Q_{max} \in \mathbb{Q}_{max}$. Рассмотрим его построение для некоторых классов формул.

Утверждение 1. *Пусть $P \in \mathbb{P}$, T — таблица предиката P, $\Phi = \forall x P(x)$. Тогда $X(Q_{max}) = X_T$.*

Данная формула Φ фактически означает, что любой кортеж из базы данных содержится в таблице T предиката P.

Утверждение 2. *Пусть $P \in \mathbb{P}$, T — таблица предиката P, $\Phi = \exists x P(x)$. Тогда $X(Q_{max}) = \varnothing$.*

Построение восстановления в этом случае подразумевает, что если в P есть кортежи, то формула истинна, а если нет, то построить восстановление можно только путём удаления всех кортежей из базы данных.

Для сокращения записи будем называть *элементарным вспомогательным условием* и обозначать $h(x_1, \dots, x_n)$ конъюнкцию вспомогательных предикатов, построенную следующим образом:

$$h(x_1, \dots, x_n) = g_1(x_{i_1}, x_{j_1}) \,\&\, g_2(x_{i_2}, x_{j_2}) \,\&\, \dots \,\&\, g_k(x_{i_k}, x_{j_k}),$$

где $i_1, \dots, i_k, j_1, \dots, j_k \in \{1, \dots, n\}$, $g_1, g_2, \dots, g_k \in G$. Элементарное вспомогательное условие может быть тождественно равно «истине». Далее в утверждениях будем считать, что везде под $h(x_1, \dots, x_n)$ подразумевается элементарное вспомогательное условие, если не оговорено иное.

Утверждение 3. *Пусть* $D = \langle \mathbb{T}, \Phi \rangle$, $T \in \mathbb{T}$ — *таблица предиката* P,

$$\Phi = \forall x P(x) \,\&\, h(x) \,\&\, \Phi_1.$$

Тогда $X(Q_{max}(D)) = X_T \cap X(Q_{max}(D_1))$, *где* $D_1 = \langle \mathbb{T}, \Phi_1 \rangle$.

В данном случае необходимо сначала удалить все кортежи, которые не содержатся в таблице T, а потом устранять противоречия, относящиеся ко второй части формулы.

Следствие 1. *Пусть* $P_1, P_2, \ldots, P_n \in \mathbb{P}$ *и* $T_1, T_2, \ldots, T_n$ — *соответствующие им таблицы,*

$$\Phi = \forall x_1 P_1(x_1) h_1(x_1) \,\&\, \forall x_2 P_2(x_2) \,\&\, h_2(x_2) \,\&\, \ldots \,\&\, \forall x_n P_n(x_n) \,\&\, h_n(x_n).$$

Тогда $X(Q_{max}) = X_{T_1} \cap X_{T_2} \cap \ldots \cap X_{T_n}$.

Следствие 2. *Пусть* $P_1, P_2 \in \mathbb{P}$, T_1, T_2 — *таблицы, соответствующие* P_1, P_2,

$$\Phi = \forall x_1 P_1(x_1) \,\&\, h_1(x_1) \,\&\, \exists x_2 P_2(x_2) \,\&\, h_2(x_2).$$

Тогда $X(Q_{max}) = \{x | (x \in X_{T_1} \cap X_{T_2}) \,\&(h_1(x) \,\&\, h_2(x) = 1)\}$.

Утверждение 4. *Пусть* $P_1, \ldots, P_n \in \mathbb{P}$ *и* $T_1, \ldots, T_n$ — *соответствующие им таблицы,* $\Phi = \exists x_1 P_1(x_1) \& h_1(x_1) \vee \ldots \vee \exists x_n P_n(x_n) \& h_n(x_n)$. *Тогда* $X(Q_{max}) = \varnothing$.

Эта формула будет истинной только в том случае, когда в каждой из таблиц предиката будет содержаться хотя бы один кортеж, удовлетворяющий условию. Если же это не так, то возникает противоречие, которое можно устранить только удалением всех кортежей из всех таблиц предикатов.

Утверждение 5. *Пусть* $P_1, \ldots, P_n \in \mathbb{P}$ *и* $T_1, \ldots, T_n$ — *соответствующие им таблицы,* $\Phi = \forall x_1 (P_1(x_1) \rightarrow h_1(x_1)) \& \ldots \& \forall x_n (P_n(x_1) \rightarrow h_n(x_n))$.

Тогда $X(Q_{max}) = \bigcup\limits_{i=1}^{n} \{x | (x \in T_i) \& (h_i(x) = 1)\}$.

В данном случае формула Φ означает, что в таблице каждого из предикатов P должны содержаться данные, которые удовлетворяют записанному условию. Если какой-то кортеж ему не удовлетворяет, то его надо удалить. При этом дополнительных противоречий из-за производимых удалений возникать не будет.

Утверждение 6. *Пусть* $D = \langle \mathbb{T}, \Phi \rangle$ *и* $T_1, T_2 \in \mathbb{T}$ *— таблицы, соответствующие предикатам* P_1, P_2. *Пусть* $\Phi = \Phi_1 \vee \Phi_2$, *где* $\Phi_1 = \forall x_1(P_1(x_1) \rightarrow h_1(x_1))$, $\Phi_2 = \forall x_2(P_2(x_2) \rightarrow h_2(x_2))$. *Обозначим* $D_1 = \langle \mathbb{T}, \Phi_1 \rangle$, $D_2 = \langle \mathbb{T}, \Phi_2 \rangle$. *Тогда* $X(Q_{max}(D))$ *будет равно тому из множеств* $X(Q_{max}(D_1)), X(Q_{max}(D_2))$, *которое будет содержать больше кортежей.*

Утверждение 7. *Пусть* $P_1, P_2 \in \mathbb{P}$ *и* T_1, T_2 *— соответствующие им таблицы,*

$$\Phi = \forall x_1 \exists x_2(P_1(x_1) \rightarrow (P_2(x_2) \& h(x_1, x_2))).$$

Тогда $Z_{Q_{max}} = \{x | (x \in T_1)$ *и для всех* $y \in T_2$ $h(x, y) = 0\}$, *если таблица* T_2 *не пустая, и* $Z_{Q_{max}} = X_{T_1}$, *если* T_2 *пустая.*

Возникновение противоречий для данной формулы может быть из-за того, что для какого-либо кортежа из P_1 нет такого кортежа из P_2, чтобы условие выполнялось, соответственно, надо удалить все такие кортежи из P_1. Если же P_2 — пустая, то тогда необходимо удалить все кортежи из P_1.

Утверждение 8. *Пусть* $D = \langle \mathbb{T}, \Phi \rangle$ *и* $T_1, T_2 \in \mathbb{T}$ *— таблицы, соответствующие предикатам* P_1, P_2. *Пусть* $\Phi = \Phi_1 \vee \Phi_2$, *где*

$$\Phi_1 = \forall x_1 \exists x_2(P_1(x_1) \rightarrow (P_2(x_2) \& h_1(x_1, x_2))),$$

$$\Phi_2 = \forall x_3 \exists x_4(P_1(x_3) \rightarrow (P_2(x_4) \& h_2(x_3, x_4))).$$

Обозначим $D_1 = \langle \mathbb{T}, \Phi_1 \rangle$, $D_2 = \langle \mathbb{T}, \Phi_2 \rangle$. *Тогда* $Z_{Q_{max}(D)}$ *будет равно либо* $Z_{Q_{max}(D_1)}$, *либо* $Z_{Q_{max}(D_2)}$, *в зависимости от того, в каком из них содержится меньше кортежей, а также* $Z_{Q_{max}(D)} \subseteq X_{T_1}$.

В соответствии с утверждением 7 удаление кортежей для построения восстановления следует производить из P_1. Кроме того, в соответствии с утверждением 6 достаточно сделать истинной одну из частей дизъюнкции, выбор которой необходимо осуществлять по числу удаляемых кортежей.

Утверждение 9. *Пусть* $P_1, P_2 \in \mathbb{P}$ *и* T_1, T_2 *— соответствующие им таблицы,*

$$\Phi = \forall x_1 \exists x_2(\neg P_1(x_1) \vee \neg P_2(x_2) \vee h(x_1, x_2)).$$

Обозначим $Z_1 = \{x | (x \in T_1)$ *и для всех* $y \in T_2$ $h(x, y) = 0\}$, *если* T_2 *— не пустая, и* $Z_1 = X_{T_1}$, *если* T_2 *— пустая. Тогда* $Z_{Q_{max}}$ *будет* *равно либо Z_1, либо X_{T_2}, в зависимости от того, в каком из множеств кортежей меньше.*

Работа выполнена при финансовой поддержке РФФИ и программы фундаментальных исследований ОМН РАН «Алгебраические и комбинаторные методы математической кибернетики и информационные системы нового поколения».

Об одной задаче теории расписаний

М. А. Трушников

ctgy@ya.ru

Московский государственный университет им. М. В. Ломоносова

Рассматривается задача составления однопроцессорного расписания для n работ на отрезке времени $[0, 2m]$. Работы — это открытые отрезки, нужно расположить их внутри отрезка $[0, 2m]$ так, чтобы никакие два из них не пересекались. Входными данными задачи являются целые числа m и n и действительные числа $l_1, l_2, \ldots, l_n$, l_i — длина i-й работы, $0 < l_i \leqslant 1$. Стоимость выполнения работ на отрезках $[1, 2], [3, 4], \ldots, [2m-1, 2m]$ ровно в k раз больше, чем на отрезках $[0, 1], [2, 3], \ldots, [2m-2, 2m-1]$, где k — константа. Далее отрезки из первой и второй групп будут называться дорогими и дешевыми соответственно. Для оценки качества расписания используется следующая целевая функция. Пусть $l_i^{'}$ — время, в течение которого i-я работа выполнялась в одном из дешевых отрезков, $l_i^{''} = l_i - l_i^{'}$. Требуется минимизировать функцию

$$\sum_{i=1}^{n} \left(l_i^{'} + k \cdot l_i^{''} \right).$$

Такую постановку можно интерпретировать следующим образом. Требуется составить расписание на m суток вперед, притом дорогие отрезки соответствуют дневному времени, дешевые — ночному, и предполагается, что ночью потребление электроэнергии стоит дешевле. Различные задачи, связанные с энергосберегающими расписаниями, рассмотрены, например, в [1].

Задача является NP-трудной, так как к ней можно свести задачу о разбиении набора чисел на два множества, суммы элементов в которых совпадают.

Теорема. *Для любого* $\varepsilon > 0$ *существует полиномиальный от* n *и* m *алгоритм* A_ε, *строящий решение стоимости, не превосходящей* $(1 + k\varepsilon)\, OPT$, *где* OPT *— точное решение.*

Доказательство. Для удобства переформулируем задачу. Есть m бесконечных контейнеров. Стоимость помещения множества работ суммарного размера L в любой контейнер есть L, если $L \leqslant 1$, и $1 + (L-1)k$ — иначе. Требуется расположить исходные n работ в m таких контейнерах, минимизировав суммарную стоимость. Ясно, что из решения переформулированной задачи можно получить решение исходной с тем же значением целевой функции.

Алгоритм. На вход подается действительное число $\varepsilon > 0$.

— Разобьем все работы на две группы:

$$B = \{i : t_i \geqslant \varepsilon\} \quad \text{и} \quad S = \{i : t_i < \varepsilon\}.$$

Пусть $n^{'} = |B|$. Ясно, что $OPT \geqslant \varepsilon \cdot n^{'}$.

— Введем $p = 2/\varepsilon^2$. Разобьем все работы из B на $\lceil p \rceil$ групп: $\{B_1, B_2, \ldots, B_{\lceil p \rceil}\}$. В группу с индексом $\lceil p \rceil$ поместим $\lceil n'/p \rceil$ самых коротких работ. В следующую группу — следующие по величине $\lceil n'/p \rceil$ работ. И так далее. В B_1 будут находиться самые длинные работы. Ясно, что $|B_1| \leqslant \lfloor n'/p \rfloor$.

— Теперь для $i = 2, \ldots, n$ длину каждой работы из B_i округлим вверх до длины самой короткой работы из B_{i-1}. Работы из множеств $B_2, \ldots, B_{\lceil p \rceil}$ с округленными длинами обозначим $B^{'}$.

— Заметим, что в $B^{'}$ длины работ принимают не более $\lceil p \rceil$ различных значений. Найдем точное решение для работ $B^{'}$ с помощью динамического программирования за время $O(m \cdot n^{2p})$. Алгоритм будет описан ниже. Полученное распределение работ $B^{'}$ по m контейнерам применим к исходным (неокругленным) работам из множеств $B_2, \ldots, B_{\lceil p \rceil}$.

— Работы из B_1 разместим в бесконечном дорогом пространстве любого из контейнеров. Заметим, что

$$|B_1| \leqslant \lfloor n^{'}/p \rfloor \leqslant \frac{n^{'}\varepsilon^2}{2} \leqslant \frac{\varepsilon \cdot OPT}{2}.$$

Следовательно, стоимость размещения всех работ из B не превосходит
$$(1+\varepsilon/2)\, OPT\,. \tag{1}$$

— Работы из множества S будем добавлять к уже полученному распределению работ из B. Переберем работы из S в произвольном порядке и каждую будем помещать в самый свободный контейнер. После этого либо все маленькие работы будут выполнены в дешевое время, либо каждый контейнер заполнен хотя бы на $1-\varepsilon$. В первом случае упаковка работ из S не добавляет ничего к оценке (1), во втором — вклад в погрешность решения при упаковке S можно оценить сверху как $\varepsilon \cdot k \cdot (1+2\varepsilon)\cdot OPT$ при условии, что $\varepsilon \leqslant 1/2$.

Грубая оценка стоимости упаковки всех n работ в m контейнеров по данному алгоритму имеет вид
$$OPT\left(1+\frac{\varepsilon\cdot k}{2}\right)+3\varepsilon\cdot k\cdot OPT = OPT\left(1+\frac{7\varepsilon\cdot k}{2}\right).$$

Для завершения доказательства осталось описать, как используется динамическое программирование.

Алгоритм динамического программирования. На вход подается набор работ $l_1,\dots,l_n$, $l_i \geqslant \varepsilon$. Длины работ могут принимать не более $\lceil p\rceil$ различных значений. За n_i обозначим число работ i-го типа, $i=1,\dots,\lceil p\rceil$. За $M(a_1,a_2,\dots,a_{\lceil p\rceil},s)$ обозначим минимальную стоимость расположения a_1 работ первого типа, …, $a_{\lceil p\rceil}$ работ типа $\lceil p\rceil$ в s контейнерах. Для $s=1$ эти значения вычисляются очевидным образом. Остальные значения будем вычислять, используя формулу
$$\begin{aligned}M(a_1,a_2,\dots,a_{\lceil p\rceil},s) &=\\ &=\min_{\overline{v}}(M(a_1-v_1,\dots,a_{\lceil p\rceil}-v_{\lceil p\rceil},s-1)+M(v_1,\dots,v_{\lceil p\rceil},1)),\end{aligned}$$
где $\overline{v}=(v_1,\dots,v_{\lceil p\rceil})$, $v_i\leqslant a_i$.

Для вычисления каждого из $n^{\lceil p\rceil}$ значений для $s=2,\dots,m$ нужно перебрать $n^{\lceil p\rceil}$ векторов $\overline{v}$. Итого, сложность алгоритма — $O(m\cdot n^{2\lceil p\rceil})$. ∎

Список литературы

[1] *Albers S.* Energy-efficient algorithms // Communications of the ACM. — 2010. — V. 53, № 5. — P. 86–96.

[2] *Fernandez de la Vega W., Lueker G. S.* Bin packing can be solved within $1+\varepsilon$ in linear time // Combinatorica. — 1981. — V. 1, № 4. — P. 349–355.

Об оценках глубины α-пополнений систем функций трехзначной логики

Д. В. Трущин

dimkatr@yandex.ru

Московский государственный университет им. М. В. Ломоносова, Механико-математический факультет

Рассматривается задача о реализации функций трехзначной логики α-формулами, т. е. такими формулами, в которых каждая подформула содержит не более одной нетривиальной главной подформулы. В качестве меры сложности формул рассматривается глубина. В работе приведена последовательность функций, для которой справедливы экспоненциальные нижние оценки глубины над системой из всех бинарных операций с правым сокращением. Кроме того, получены экспоненциальные верхние оценки глубины произвольной функции над рассматриваемой системой.

Через P_k обозначим множество всех функций k-значной логики, $k \geqslant 2$, а через $H(n)$ — множество всех функций, принадлежащих множеству H, $H \subseteq P_k$, и зависящих только от переменных $x_1, \ldots, x_n$. Пусть $\mathfrak{A}$ — конечная система функций из P_k. Замыкание системы $\mathfrak{A}$ (относительно операций суперпозиции и введения фиктивной переменной) обозначим через $[\mathfrak{A}]$. Необходимые определения можно найти в [1].

Пусть Φ — некоторая формула над $\mathfrak{A}$. Сложностью $L(\Phi)$ этой формулы называется число символов переменных, входящих в нее. Глубину $D(\Phi)$ формулы Φ определим индуктивно. Если Φ состоит только из символа переменной, то $D(\Phi) = 0$. Если Φ имеет вид $f(\Phi_1, \ldots, \Phi_m)$, где $f \in \mathfrak{A}$, а $\Phi_1, \ldots, \Phi_m$ — некоторые формулы над $\mathfrak{A}$, то $D(\Phi) = 1 + \max D(\Phi_i)$, где максимум берется по всем $i = 1, \ldots, m$. Для любой функции $f \in [\mathfrak{A}]$ положим $L_{\mathfrak{A}}(f) = \min L(\Phi), D_{\mathfrak{A}}(f) = \min D(\Phi)$, где минимум берется по всем формулам Φ над $\mathfrak{A}$, реализующим f.

Известно [2], что для любой полной конечной системы булевых функций $\mathfrak{A}$ и любой булевой функции $f(x_1,\ldots,x_n)$ выполнено соотношение
$$L_{\mathfrak{A}}(f) \lesssim \frac{2^n}{\log_2(n)}.$$

В работе [3] показано, что для произвольной конечной системы булевых функций $\mathfrak{A}$ и любой функции $f(x_1,\ldots,x_n) \in [\mathfrak{A}]$ справедливы неравенства
$$L_{\mathfrak{A}}(f) \leqslant c^n, \quad D_{\mathfrak{A}}(f) \leqslant dn,$$
где c и d — некоторые константы, зависящие от $\mathfrak{A}$.

Следуя [4], определим индуктивно понятие α-формулы Φ над конечной системой функций алгебры логики $\mathfrak{A}$. Символ переменной является тривиальной α-формулой. Выражение вида $f(\varphi, x_{i_2},\ldots,x_{i_m})$, где φ — α-формула над $\mathfrak{A}$, f — символ m-местной функции из $\mathfrak{A}$, $m \geqslant 1$, а $x_{i_2},\ldots,x_{i_m}$ — символы переменных, также является α-формулой. Отметим, что каждая α-формула является формулой над $\mathfrak{A}$. Множество всех функций, реализуемых α-формулами над $\mathfrak{A}$, будем называть α-пополнением системы $\mathfrak{A}$ и обозначать через $[\mathfrak{A}]_\alpha$. Система $\mathfrak{A} \subset P_k$ называется α-порождающей для некоторого класса функций $H \subset P_k$, если $[\mathfrak{A}]_\alpha = H$. Система $\mathfrak{A} \subset P_k$ называется α-полной, если она является α-порождающей для P_k.

Пусть $\mathfrak{A}$ — конечная система булевых функций, $f \in [\mathfrak{A}]_\alpha$. Положим $D^\alpha_{\mathfrak{A}}(f) = \min D(\Phi)$, $L^\alpha_{\mathfrak{A}}(f) = \min L(\Phi)$, где минимум берется по всем α-формулам Φ над $\mathfrak{A}$, реализующим f. Формулу Φ над $\mathfrak{A}$ назовем минимальной (для функции f), если Φ реализует f и $D(\Phi) = D^\alpha_{\mathfrak{A}}(f)$. Определим функцию Шеннона $D^\alpha_{\mathfrak{A}}(n) = \max D^\alpha_{\mathfrak{A}}(f)$, где максимум берется по всем функциям $f \in H(n)$, $H = [\mathfrak{A}]_\alpha$. Отметим, что для введенных мер сложности справедливы неравенства $c_1 D^\alpha_{\mathfrak{A}}(f) \leqslant L^\alpha_{\mathfrak{A}}(f) \leqslant c_2 D^\alpha_{\mathfrak{A}}(f)$, где c_1 и c_2 — положительные константы, зависящие от $\mathfrak{A}$.

В работе [5] показано, что для любой конечной системы $\mathfrak{A}$ булевых функций существует многочлен $P(n)$, такой, что $D^\alpha_{\mathfrak{A}}(n) \leqslant P(n)$. Известно также [6, 5], что в P_2 не существует конечных α-полных систем. При этом в P_k при $k \geqslant 3$ конечные α-полные системы существуют [4, 6, 7].

В данной работе рассматриваются функции трехзначной логики. Положим $E_3 = \{0, 1, 2\}$.

Двухместную функцию $f(x_1, x_2) \in P_3$ будем называть бинарной операцией с правым сокращением, если для любых $b, c \in E_3$ существует, и притом ровно один, элемент $a \in E_3$, такой, что $f(a, b) = c$. Множество всех бинарных операций с правым сокращением обозначим через $\mathfrak{B}$.

Одноместную функцию $s(x) \in P_3$ будем называть подстановкой, если для любых различных $a_1, a_2 \in E_3$ справедливо неравенство $s(a_1) \neq s(a_2)$. Множество всех подстановок обозначим через S. Пусть $s \in S$. Легко видеть, что существует бинарная операция с правым сокращением f, такая, что для любых $a, b \in E_3$ имеет место равенство $f(a, b) = s(a)$, т.е. $f(x, y) = s(x)$. Таким образом, $[\mathfrak{B}]_\alpha = [\mathfrak{B} \cup S]_\alpha$. В работе [7] показано, что система, состоящая из всех бинарных операций с правым сокращением и всех подстановок, содержит α-полную подсистему. Поэтому система $\mathfrak{B}$ также α-полна.

Пусть $n \geqslant 1$, $a_1, a_2, \ldots, a_n \in E_3$. Следуя [4], положим

$$\psi_{a_1,\ldots,a_n}(x_1, \ldots, x_n) = \begin{cases} 1, & \text{если } x_i = a_i, \text{ для всех } i = 1, \ldots n, \\ 0, & \text{иначе.} \end{cases}$$

Теорема. *Для любого $n \geqslant 1$ и любых $a_1, a_2, \ldots, a_n \in E_3$ имеют место неравенства*

$$\frac{2^n - 5}{4} \leqslant D^\alpha_{\mathfrak{B}}(\psi_{a_1,\ldots,a_n}) \leqslant 3 \cdot 2^{n-1} - 2.$$

Следствие 1. *При $n \longrightarrow \infty$ выполнено соотношение*

$$D^\alpha_{\mathfrak{B}}(\psi_{a_1,\ldots,a_n}) \asymp 2^n.$$

Следствие 2. *Пусть $n \geqslant 1$ и $f(x_1, \ldots, x_n) \in P_3$. Тогда имеет место неравенство*

$$D^\alpha_{\mathfrak{B}}(f) \leqslant (3 \cdot 2^{n-1} - 2) \cdot 3^n.$$

В заключение автор выражает искреннюю признательность А. Б. Угольникову за постановку задачи и обсуждение результатов работы.

Работа выполнена при финансовой поддержке РФФИ (проект 11-01-00508) и программы фундаментальных исследований Отделения математических наук РАН «Алгебраические и комбинаторные методы математической кибернетики и информационные системы нового поколения», проект «Задачи оптимального синтеза управляющих систем».

Список литературы

[1] *Угольников А.Б.* Классы Поста. — М.: Изд-во ЦПИ при механико-математическом факультете МГУ им. М. В. Ломоносова, 2008.

[2] *Лупанов О.Б.* О сложности реализации функций алгебры логики формулами // Проблемы кибернетики. Вып. 3. — М.: Физматгиз, 1960. — С. 61–80.

[3] *Угольников А.Б.* О глубине формул в неполных базисах // Математические вопросы кибернетики. Вып. 1. — 1988. — С. 242–245.

[4] *Глухов М.М.* Об α-замкнутых классах и α-полных системах функций k-значной логики // Дискретная математика. — 1989. — Т. 1, вып. 1. С. 16–21.

[5] *Трущин Д.В.* О глубине α-пополнений систем булевых функций // Вестн. Моск. ун-та. Сер. 1. Математика. Механика. — 2009. — Вып. 2. — С. 72–75.

[6] *Чернышов А.Л.* Условия α-полноты систем функций многозначной логики // Дискретная математика. — 1992. — Т. 4, вып. 4. — С. 117–130.

[7] *Шабунин А.Л.* Примеры α-полных систем k-значной логики при $k = 3, 4$ // Дискретная математика. — 2006. — Т. 18, вып. 4. — С. 45–55.

О сложности односторонних клеточных схем фиксированной высоты с кратными входами

А. Ю. Улесова

ulesova@gmail.com

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

В данной работе рассматривается модель односторонних клеточных схем с кратными входами фиксированной высоты, то есть клеточные схемы ограниченной высоты, в которых входные переменные могут подаваться только на верхнюю границу. В такой модели схем для функции Шеннона, характеризующей минимальную площадь клеточной схемы высоты h над произвольным полным базисом из клеточных элементов, реализующей функции алгебры логики (ФАЛ) от n переменных из инвариантного класса Q с характеристикой σ (см., например, [4]), установлена асимптотика вида $\sigma\frac{h2^n}{\log n}$ при $h \geqslant 4$. Для функции Шеннона, соответствующей классу симметрических относительно всех своих переменных функций, установлена асимптотика вида $\frac{hn}{\log n}$ при $h \geqslant 4$.

Схема из клеточных элементов (см. [1, 2]) представляет собой плоскую прямоугольную решетку, в каждой клетке которой расположен один из элементов схемы. Размеры всех элементов одинаковы и принимаются за единицу. Элементы могут быть как функциональными, то есть реализующими какую-то функцию от своих входов, так и коммутационными, которые служат для передачи сигнала к следующим элементам с возможным изменением направления.

Рассмотрим плоские прямоугольные схемы над базисом $B = \{f_1, \ldots, f_r\}$, состоящим из r функциональных элементов, реализующих некоторые ФАЛ, и из трех коммутационных элементов: разветвление, пересечение, изолятор. Базис B выбирается таким образом, что множество функций $\{f_1, \ldots, f_r\}$ является полным в классе всех ФАЛ. Построенный базис из клеточных элементов является при этом полным в классе клеточных схем в том смысле, что произвольную ФАЛ можно реализовать клеточной схемой высоты 2 над этим базисом (см. [3]).

Каждый из элементов базиса может быть повернут в плоскости на угол, кратный 90 градусам. В качестве входов и выходов схемы выбираем входы и выходы элементов, расположенных на границе схемы. Функциональные элементы схемы соединяются с помощью коммутационных таким образом, чтобы полученная схема представляла собой схему из функциональных элементов (СФЭ). Считаем, что клеточная схема реализует такие же ФАЛ, что и соответствующая ей СФЭ.

Ограничим класс рассматриваемых схем. Односторонней клеточной схемой назовем клеточную схему, входные переменные которой могут подаваться только на одну ее границу. Клеточной схемой с кратными входами назовем клеточную схему, каждая входная переменная которой может подаваться в схему многократно. Исследуем односторонние клеточные схемы с кратными входами (ОКСКВ) фиксированной высоты. Дадим практическое обоснование этой модели. Пусть имеется n параллельных проводников, по которым передаются значения переменных $x_1, \ldots, x_n$. Ниже, под этими проводниками, располагается клеточная схема, на верхнюю границу которой могут подаваться переменные, идущие выше. Выход схемы находится в правом нижнем углу схемы. Интерес к такого рода схемам обусловлен, прежде всего, тем фактом, что в реальных схемах часто функциональная часть разбивается на блоки, соединенные между собой несколькими линиями передачи данных. При этом коммутация функциональных блоков может занимать достаточно большую часть схемы. Рассматриваемые схемы дают возможность проводить вычисления по ходу передачи сигналов, занимая лишь небольшую часть площади всей схемы. В общем виде ОКСКВ используется для реализации некоторых вспомогательных вычислений. В реальных схемах данная конструкция может встретиться в следующем виде: по некоторой шине передаются несколько сигналов, и например, в процессе передачи требуется выполнить проверку целостности передаваемых сигналов, для чего надо будет построить ОКСКВ. Вполне очевидным требованием является ограничение на высоту ОКСКВ.

Обозначим высоту ОКСКВ Σ через $h(\Sigma)$, длину — через $\lambda(\Sigma)$, площадь $A(\Sigma) = h(\Sigma)\lambda(\Sigma)$ и будем считать, что $h(\Sigma) \leqslant \lambda(\Sigma)$. Пусть f — произвольная ФАЛ, тогда $A_B(f)$ — наименьшая из площадей $A(\Sigma)$, где минимум берется по всем ОКСКВ Σ над базисом B, реализующим f.

Введем две функции Шеннона для площади схем в рассматриваемой модели. Пусть P_2 — множество всех ФАЛ. Для некоторого класса функций $K, K \subseteq P_2$, обозначим через $K(n)$ множество функций из K, зависящих от n переменных. Положим функцию $A_B(K)$ равной максимальной из площадей $A_B(f)$, где максимум берется по всем ФАЛ f из класса K. Рассмотрим случай, когда высота схем фиксирована. Напомним, что для любой f существует ОКСКВ Σ высоты $h(\Sigma) = 2$, реализующая эту функцию f [3]. Для любого $h \geqslant 2$ определены функция $A_B^h(f)$, равная наименьшей из площадей $A(\Sigma)$, где минимум берется по всем ОКСКВ Σ над базисом B высоты h, реализующим f, и функция $A_B^h(K)$, равная максимальной из площадей $A_B^h(f)$, где максимум берется по всем ФАЛ f из класса K.

А. А. Тиунчик в работе [3] доказал, что в произвольном полном базисе B функция $A_B^3(P_2(n))$ имеет порядок роста $2^n / \log n$.

Напомним, что класс Q, $Q \subseteq P_2$, называется *инвариантным классом* тогда и только тогда, когда Q замкнут относительно операций добавления и изъятия фиктивных переменных, переименования (без отождествления) переменных и подстановки констант вместо переменных (см. [4]).

Теорема 1. *Для инвариантного класса Q верна оценка для функции Шеннона $A_B^h(Q(n)) \sim \sigma h 2^n / \log n$ при $h \geqslant 4$.*

Теорема 2. *Для класса S симметрических функций — класса, замкнутого относительно операции перестановки переменных, — верна оценка для функции Шеннона при $h \geqslant 4$*

$$A_B^h(Q(n)) \sim \sigma \frac{h 2^n}{\log n}, \qquad A_B^h(S(n)) \sim \frac{hn}{\log n}.$$

Если рассматривать функцию Шеннона в классе клеточных схем с кратными входами, где входные переменные могут подаваться по всей границе схемы, то асимптотика функции Шеннона уменьшается вдвое. При этом она справедлива для схем высоты $h \geqslant 7$. В стандартном базисе $\{\&, \vee, \neg\}$ данные оценки достигаются при значениях $h \geqslant 3$ и $h \geqslant 5$ соответственно.

Доказательство верхних оценок в теореме основывается на асимптотически оптимальном методе построения формул, предложенном Лупановым в [5]. Модель ОКСКВ и ее оценки имеют практическую ценность при построении программируемых (настраиваемых) процессоров. Также интересно применение данной модели для экономии пространства интегральной схемы при больших затратах на трассировку схемы.

Работа выполнена при поддержке РФФИ, проект 09-01-00817-а.

Список литературы

[1] *Кравцов С. С.* О реализации функций алгебры логики в одном из классов схем из функциональных и коммутационных элементов // Проблемы кибернетики. — 1967. — Т. 19. — С. 285–292.

[2] *Альбрехт А.* О схемах из клеточных элементов // Проблемы кибернетики. — 1975. — Т. 33. — С. 209–214.

[3] *Тиунчик А. А.* О реализации функций алгебры логики клеточными схемами ограниченной ширины // Методы дискретного анализа в решении экстремальных задач. — Новосибирск: Институт математики СО АН СССР, 1990. — С. 73–83.

[4] *Яблонский С. В.* О невозможности элиминации перебора всех функций из P_2 при решении некоторых задач теории схем // ДАН СССР. — 1959. — Т. 124, № 1. — С. 54–62.

[5] *Лупанов О. Б.* О реализации функций алгебры логики формулами из конечных классов (формулами ограниченной глубины) в базисе $\&, \vee, \neg$ // Проблемы кибернетики. — 1967. — Т. 19. — С. 5–14.

Разнообразие шаров в графах с фиксированными числом вершин и диаметром

Т. И. Федоряева

tatiana.fedoryaeva@gmail.com

Институт математики им. С. Л. Соболева СО РАН, Новосибирск

Пусть $\mathcal{X} = (X, \rho)$ — конечное метрическое пространство, $B_i(x) = \{y \in X \mid \rho(x, y) \leqslant i\}$ — шар радиуса i с центром в точке x. Рассмотрим покрытия пространства $\mathcal{X}$ перекрывающимися шарами фиксированного радиуса. Традиционно плотность таких покрытий определяется как среднее число шаров, содержащих точку пространства. Математической формализацией этого понятия может служить функция $\Theta : \mathcal{M}_i \to \mathbf{Q}^+$ [1], определяемая следующим образом:

$$\Theta(M) = \frac{1}{|X|} \sum_{x \in X} \theta_M(x),$$

где $\mathcal{M}_i$ — совокупность всех покрытий пространства $\mathcal{X}$ шарами радиуса i, $M \in \mathcal{M}_i$, $\theta_M : X \to \mathbf{N}$, $\theta_M(x)$ — число элементов покрытия M, содержащих $x \in X$. Ввиду конечности пространства $\mathcal{X}$ существует покрытие с наибольшей плотностью. Такие покрытия естественным образом возникают, когда, например, требуется максимизировать плотность покрытия системы связи, износоустойчивость различного рода промышленных покрытий, защищенность для схем элементов, степень контроля и т. п. Поскольку функция плотности Θ строго возрастает на $\mathcal{M}_i$ относительно порядка по включению (см. утверждение 1 из [1]), то покрытие пространства $\mathcal{X}$ шарами фиксированного радиуса i с наибольшей плотностью представляет собой систему всех различных шаров радиуса i, а число элементов такого покрытия есть число $\tau_i(\mathcal{X})$ всех различных шаров радиуса i пространства $\mathcal{X}$.

На величину $\tau_i(\mathcal{X})$ можно также взглянуть с другой стороны. В случае дискретных метрических пространств различные шары одного радиуса могут иметь разные «форму» и «объём». Например, для метрического пространства обыкновенного графа G с $V(G) = \{a, b, x, y\}$ и $E(G) = \{ax, xb, by, ya, xy\}$ имеем $B_1(a) = \{a, x, y\}$, $B_1(x) = B_1(y) = \{a, b, x, y\}$, $B_1(b) = \{b, x, y\}$. В этой связи величина $\tau_i(\mathcal{X})$ может рассматриваться как число «способов размещения» шара радиуса i в n-элементном метрическом пространстве $\mathcal{X}$.

Далее возникает вопрос: каково наибольшее число шаров заданного радиуса в метрических пространствах фиксированного «объёма»? Формализацией этого вопроса для метрических пространств обыкновенных связных графов с обычным расстоянием между вершинами (т. е. длиной кратчайшей цепи, соединяющей эти вершины) является задача нахождения числа

$$\overline{\tau}_i(\Omega) = \max_{G \in \Omega} \tau_i(G),\ i \geqslant 0,$$

где Ω — класс всех обыкновенных связных графов наперёд фиксированного «объёма». В качестве Ω будем рассматривать класс Γ_n всех n-вершинных обыкновенных связных графов и класс $\Gamma_{n,\,d}$ всех n-вершинных обыкновенных связных графов диаметра d. Отметим, что для класса $\Gamma^{\,d}$ всех графов диаметра d максимум значений $\tau_i(G)$, $G \in \Gamma^{\,d}$, не достигается при любых $d > 0$ и $i < d$ (это следует, например, из существования для каждого $n \geqslant 2d > 0$ графа из класса $\Gamma_{n,\,d}$ с полным разнообразием шаров [1]).

В общем виде для произвольного класса Ω обыкновенных связных графов задача нахождения точных верхних и точных нижних оценок числа различных шаров заданного радиуса в графах из класса Ω, т. е. соответственно $\overline{\tau}_i(\Omega)$ и $\underline{\tau}_i(\Omega) = \min\limits_{G \in \Omega} \tau_i(G)$, была сформулирована в [1, 2] и решена для класса Γ_n, класса T_n всех n-вершинных деревьев и класса $\mathrm{T}_{n,\,d}$ всех n-вершинных деревьев диаметра d, т. е. установлены точные значения $\overline{\tau}_i(\Gamma_n)$, $\underline{\tau}_i(\Gamma_n)$, $\overline{\tau}_i(\mathrm{T}_n)$, $\underline{\tau}_i(\mathrm{T}_n)$, $\overline{\tau}_i(\mathrm{T}_{n,\,d})$, $\underline{\tau}_i(\mathrm{T}_{n,\,d})$ для любого $i \geqslant 0$. В следующей теореме 1 получены точные верхние оценки $\overline{\tau}_i(\Gamma_{n,\,d})$ для класса $\Gamma_{n,\,d}$ (точные нижние оценки $\underline{\tau}_i(\Gamma_{n,\,d})$ найдены в [1, 2]). Здесь считаем, что рассматриваемый класс графов $\Gamma_{n,\,d}$ не пуст, т. е. $n \geqslant d+1 \geqslant 2$ или $n = d+1 = 1$.

Теорема 1.

$$\overline{\tau}_i(\Gamma_{n,\,d}) = \begin{cases} n, & \textit{если } 0 \leqslant i < d \textit{ и } i \leqslant \max\{\left\lfloor \frac{d}{2} \right\rfloor,\, s\}, \\ 3(d-i)+1, & \textit{если } \left\lfloor \frac{d}{2} \right\rfloor < s < i < d, \\ n+d+\left\lfloor \frac{d}{2} \right\rfloor - 3i, & \textit{если } s \leqslant \left\lfloor \frac{d}{2} \right\rfloor < i \leqslant \left\lfloor \frac{d}{2} \right\rfloor + s,\ i < d, \\ 2(d-i)+1, & \textit{если } \left\lfloor \frac{d}{2} \right\rfloor + s < i < d, \\ 1, & \textit{если } i \geqslant d, \end{cases}$$

где $s = n - d - 1$.

Далее, для произвольного графа G числа $\tau_i(G)$, $i \geqslant 0$, образуют вектор $\tau(G) = (\tau_0(G), \tau_1(G), \ldots, \tau_i(G), \ldots, \tau_d(G))$, называемый *вектором разнообразия шаров графа* G [3], причём для них выполняется система неравенств $\tau_0(G) = |V(G)| \geqslant \ldots \geqslant \tau_i(G) \geqslant \tau_{i+1}(G) \geqslant \ldots \geqslant \tau_d(G) = 1$ (здесь $V(G)$ — множество вершин графа G и $d = d(G)$ — диаметр графа G). Впервые векторы такого вида были рассмотрены в [4], где было предложено изучать строение графов как дискретных метрических пространств через разнообразие и пересекаемость метрических шаров, содержащихся в графе. При таком подходе естественно возникает класс графов, обладающих локальным t-разнообразием шаров. Пусть $0 \leqslant t < d(G)$.

Определение 1 [4]. Граф G обладает локальным t-разнообразием шаров, если $|V(G)| = \tau_0(G) = \tau_1(G) = \ldots = \tau_t(G)$. Граф G с локальным t-разнообразием шаров при $t = d(G) - 1$ называется графом полного разнообразия шаров.

Отметим, что в [5] графы с локальным и полным разнообразием шаров использовались для построения графа G с вектором разнообразия шаров $\tau(G)$, совпадающим с наперёд заданным целочисленным вектором $\overline{\tau}$ с убывающими компонентами при дополнительных ограничениях на исходный вектор $\overline{\tau}$. В связи с этим возникает вопрос: всегда ли существуют n-вершинные графы диаметра d с локальным t-разнообразием шаров (или даже полным разнообразием шаров), как они устроены и какой вид имеет их вектор разнообразия шаров? В [1] автором исследован вопрос существования графа с локальным t-разнообразием шаров (полным разнообразием шаров) в классе $\Gamma_{n,\,d}$, и описаны все такие возможные значения параметров n, d и t.

Теорема 2 [1]. *Класс $\Gamma_{n,\,d}$ содержит граф с локальным t-разнообразием шаров тогда и только тогда, когда выполняется одно из следующих условий:*

(i) $0 \leqslant t \leqslant \left\lfloor \frac{d}{2} \right\rfloor$, $n \geqslant d+1 \geqslant 2$;
(ii) $\left\lfloor \frac{d}{2} \right\rfloor < t < d$, $n \geqslant d+1+t$.

Следствие 1. *В классе $\Gamma_{n,\,d}$ существует граф с полным разнообразием шаров тогда и только тогда, когда $n \geqslant 2d > 0$ или $n = d+1 = 3$.*

В теореме 2 был найден наименьший порядок графов диаметра d с локальным t-разнообразием шаров (полным разнообразием шаров), а в следующей теореме 3 (следствии 2) с точностью до изоморфизма описываются графы наименьшего порядка диаметра d с локальным t-разнообразием шаров (полным разнообразием шаров) и вычисляются их векторы разнообразия шаров. В работе строятся n-вершинные графы $H^i_{n,\,d,\,t}$ диаметра d и доказывается

Теорема 3.

(i) *Пусть $0 \leqslant t \leqslant \lfloor d/2 \rfloor$. Тогда простая цепь P длины d — единственный с точностью до изоморфизма граф диаметра d с локальным t-разнообразием шаров наименьшего возможного порядка и $\tau(P) = (\Delta^d_0, \Delta^d_1, \ldots, \Delta^d_d)$.*
(ii) *Пусть $\lfloor d/2 \rfloor < t < d$. Тогда n-вершинные графы $H^i_{n,\,d,\,t}$, где $n = d+1+t$ и $i = 0, 1, \ldots, \lfloor (d-t-1)/2 \rfloor$, — все с точностью до изоморфизма графы диаметра d с локальным t-разнообразием шаров наименьшего возможного порядка, причем $\tau(H^i_{n,\,d,\,t}) = (n, \ldots, n, \Delta^d_{t+1}, \Delta^d_{t+2}, \ldots, \Delta^d_d)$.*

Здесь

$$\Delta^d_j = \begin{cases} d+1, & \text{если } 0 \leqslant j \leqslant \lfloor d/2 \rfloor, \\ 2(d-j)+1, & \text{если } \lfloor d/2 \rfloor < j < d, \\ 1, & \text{если } j \geqslant d. \end{cases}$$

Следствие 2. *Для любого $d > 0$ существует единственный с точностью до изоморфизма граф диаметра d с полным разнообразием шаров наименьшего возможного порядка, а именно $2d$-вершинный цикл при $d > 2$ и цепь длины d при $d \leqslant 2$.*

Список литературы

[1] *Федоряева Т. И.* Векторы разнообразия шаров для графов и оценки их компонент // Дискрет. анализ и исслед. операций. Сер. 1. — 2007. — Т. 14, № 2. — С. 47–67.

[2] *Федоряева Т. И.* Векторы разнообразия шаров и свойства их компонент // Труды VII Международной конференции «Дискретные модели в теории управляющих систем» (Москва, 4–6 марта 2006 г.). — М.: Изд-во МГУ, 2006. — С. 374–378.

[3] *Федоряева Т. И.* Разнообразие шаров в метрических пространствах деревьев // Дискрет. анализ и исслед. операций. Сер. 1. — 2005. — Т. 12, № 3. — С. 74–84.

[4] *Евдокимов А. А.* Локально изометрические вложения графов и свойство продолжения метрики // Сибирский журнал исследования операций. — 1994. — Т. 1, № 1. — С. 5–12.

[5] *Рычков К. Л.* О достаточных условиях существования графа с заданным разнообразием шаров // Дискрет. анализ и исслед. операций. — Сер. 1. — 2006. — Т. 13, № 1. — С. 99–108.

Задача синтеза стратегий обслуживания потока объектов в системе с накопительным компонентом

Ю. С. Федосенко, А. С. Куимова, Д. В. Минаев

fds@aqua.sci-nnov.ru, anastasia.kuimova@gmail.com, minaev@aqua.sci-nnov.ru

Волжская государственная академия водного транспорта, Нижний Новгород

Рассматривается модель однопроцессорного обслуживания конечного детерминированного потока объектов в системе с накопительным компонентом. Модель описывает схему массового завоза нефтепродуктов в специфических навигационных условиях приполярного региона Западной Сибири [1]. Для случая однокритериальной оценки качества управления обслуживанием соответствующая оптимизационная задача синтеза стратегий обслуживания исследовалась в работе [2]. Необходимость её бикритериального обобщения обусловлена требованием адекватности математического описания целому ряду практически значимых эксплуатационных ситуаций.

Ниже изучается задача синтеза стратегий обслуживания при наличии двух оценочных критериев.

Рассмотрим n-элементный поток объектов $O_n = \{o_1, o_2, \ldots, o_n\}$, подлежащих однократному однофазному обслуживанию стационарным процессором P. Поток O_n обладает свойством бинарности, т.е. состоит из двух подпотоков — входящего O^+ и исходящего O^-. Принадлежность объекта o_i, $i = \overline{1,n}$, тому или иному подпотоку O^+ (O^-) определяется значением булева параметра w_i: $w_i = +1$, если $o_i \in O^+$, и $w_i = -1$, если $o_i \in O^-$.

Считаем, что подпоток O^+ либо представляет собой пустое множество $\oslash$ объектов, либо состоит из совокупности объектов $\{o_{q(1)}, o_{q(2)}, \ldots, o_{q(r)}\}$, $q(j) \in [1, 2, \ldots, n]$, $j = \overline{1,r}$, $1 \leqslant r \leqslant n$. Аналогично определяем подпоток O^- как состоящий из пустого множества или из совокупности $\{o_{q(1)}, o_{q(2)}, \ldots, o_{q(u)}\}$, $q(z) \in [1, 2, \ldots, n]$, $z = \overline{1,u}$, $1 \leqslant u \leqslant n$. При этом $r + u = n$, $q(z) \neq q(j)$ для любых пар допустимых значений z и j. Для каждого объекта o_i, $i = \overline{1,n}$, определены следующие целочисленные параметры: t_i — момент поступления в очередь на обслуживание $(0 \leqslant t_1 \leqslant t_2 \leqslant \ldots \leqslant t_n)$, τ_i — норма длительности обслуживания, a_i — штраф за единицу времени простоя в ожидании обслуживания, d_i — мягкий директивный срок завершения обслуживания $(d_i \geqslant t_i + \tau_i)$, v_i — объемная характеристика. Если обслуживание объекта o_i начинается в момент времени t_i^* $(t_i^* \geqslant t_i)$, то величина индивидуального штрафа по этому объекту определяется значением линейной функции вида $a_i(t_i^* - t_i)$.

Процессор P, обслуживающий поток O_n, снабжен накопительным элементом с объемной характеристикой V, которая в начальный момент времени $t = 0$ имеет значение V_0 и в любой момент времени не может превосходить известной величины V^*. В результате обслуживания объекта o_i из подпотока $O^+(O^-)$ значение характеристики V увеличивается (уменьшается) на величину v_i, $i = \overline{1,n}$. Обслуживание любого объекта из подпотока O^+ считается возможным, если в результате его реализации значение характеристики V не превысит величины V^*. Аналогично обслуживание любого объекта o_i, $i = \overline{1,n}$, из подпотока O^- считается возможным, если к его началу значение характеристики V не меньше объемной характеристики v_i этого объекта. Процессор P готов к обслуживанию потока объектов O_n в момент времени $t = 0$. Обслуживание объекта o_i, $i = \overline{1,n}$, может быть начато свободным процессором в любой момент времени t на полуинтервале $t \geqslant t_i$ и осуществляется без прерываний; необслуженный объект не может покинуть очередь к процессору; одновременное обслуживание процессором двух и более объектов и его непроизводительные простои запрещены.

Стратегия обслуживания объектов S 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. Множество всех допустимых стратегий обслуживания обозначим через Ω и считаем верным всегда выполняемое на практике условие $2v_i \leqslant V^*$, $i = \overline{1, n}$. Тогда необходимым и достаточным условием непустоты множества допустимых стратегий Ω является выполнение неравенств

$$0 \leqslant V_0 + \sum_{i: o_i \in O^+} v_i - \sum_{i: o_i \in O^-} v_i \leqslant V^*.$$

Обозначим через $t^*(i_k, S)$ и $\overline{t}(i_k, S)$ моменты начала и завершения обслуживания объекта с индексом i_k при реализации стратегии S. Считаем, что реализация стратегии обслуживания компактна и между $t^*(i_k, S)$ и $\overline{t}(i_k, S)$ имеют место соотношения: $t^*(i_1, S) = t_{i_1}$; $t^*(i_k, S) = \max(\overline{t}(i_{k-1}, S), t_{i_k})$, $k = \overline{2, n}$; $\overline{t}(i_k, S) = t^*(i_k, S) + \tau_{i_k}$, $k = \overline{1, n}$.

Качество стратегии S оценивается по значениям двух минимизируемых критериев

$$K_1(S) = \sum_{k=1}^{n} a_{i_k}(t^*(i_k, S) - t_{i_k}), \ K_2(S) = \max_{1 \leqslant k \leqslant n} (\overline{t}(i_k, S) - d_{i_k}, 0).$$

Общий подход к исследованию проблемы принятия решений при наличии нескольких критериев оценки базируется на концепции Парето [3] и для рассматриваемой модели обслуживания приводит к следующей бикритериальной задаче

$$\left\{ \min_{S \in \Omega}(K_1(S)), \ \min_{S \in \Omega}(K_2(S)) \right\} \tag{1}$$

выделения в плоскости $(K_1(S), K_2(S))$ полной совокупности эффективных оценок и последующего построения соответствующих им оптимально-компромиссных стратегий.

Задача относится к числу NP-трудных [4]. Для её решения построены рекуррентные соотношения динамического программирования [5–7], процесс вычислительной реализации которых состоит из трёх этапов. На первом этапе выполняется разметка для определения достижимых состояний системы. Фиксируются финальные состояния, соответствующие завершению процесса обслуживания всех объектов потока O_n. На втором этапе выполняется построение множеств эффективных оценок для произвольных достижимых состояний. Эти состояния характеризуются тем, что множество эффективных оценок для них неизвестно, но для всех непосредственно следующих за ними состояний множество эффективных оценок уже известно. Последней в процессе решения задачи (1) определяется полная совокупность эффективных оценок. На третьем, последнем этапе работы последовательно синтезируется стратегия обслуживания, соответствующая выбранной лицом, принимающим решения, эффективной оценке.

В качестве иллюстрации рассмотрен пример при следующих значениях параметров модели: $V^* = 300$, $V_0 = 261$; $t_1 = 0$, $t_2 = 3$, $t_3 = 5$, $t_4 = 7$, $t_5 = 11$, $t_6 = 11$, $t_7 = 15$, $t_8 = 19$, $t_9 = 23$, $t_{10} = 25$; $\tau_1 = 10$, $\tau_2 = 10$, $\tau_3 = 4$, $\tau_4 = 2$, $\tau_5 = 2$, $\tau_6 = 5$, $\tau_7 = 7$, $\tau_8 = 8$, $\tau_9 = 4$, $\tau_{10} = 5$; $a_1 = 1$, $a_2 = 6$, $a_3 = 2$, $a_4 = 2$, $a_5 = 3$, $a_6 = 4$, $a_7 = 3$, $a_8 = 5$, $a_9 = 6$, $a_{10} = 3$; $d_1 = 10$, $d_2 = 14$, $d_3 = 9$, $d_4 = 10$, $d_5 = 14$, $d_6 = 19$, $d_7 = 22$, $d_8 = 28$, $d_9 = 30$, $d_{10} = 33$; $v_1 = 35$, $v_2 = 6$, $v_3 = 128$, $v_4 = 68$, $v_5 = 107$, $v_6 = 35$, $v_7 = 50$, $v_8 = 84$, $v_9 = 11$, $v_{10} = 51$; $w_1 = -1$, $w_2 = -1$, $w_3 = -1$, $w_4 = -1$, $w_5 = 1$, $w_6 = -1$, $w_7 = 1$, $w_8 = -1$, $w_9 = 1$, $w_{10} = 1$.

Путем реализации описанных выше этапов синтеза получаем полную совокупность эффективных оценок и им соответствующих оптимально-компромиссных стратегий обслуживания:

$$
\begin{aligned}
&(349, 44) - \{2, 4, 5, 6, 9, 8, 10, 3, 1, 7\},\ (326, 51) - \{2, 4, 5, 6, 9, 8, 10, 3, 7, 1\},\\
&(435, 30) - \{2, 4, 5, 6, 3, 9, 1, 7, 8, 10\},\ (424, 33) - \{2, 4, 5, 6, 3, 9, 1, 8, 7, 10\},\\
&(430, 32) - \{2, 4, 5, 6, 3, 9, 1, 10, 7, 8\},\ (376, 38) - \{2, 4, 5, 6, 3, 9, 8, 1, 10, 7\},\\
&(351, 43) - \{2, 4, 5, 6, 3, 9, 8, 10, 1, 7\},\ (328, 50) - \{2, 4, 5, 6, 3, 9, 8, 10, 7, 1\},\\
&(405, 35) - \{2, 4, 5, 6, 3, 9, 10, 1, 7, 8\},\ (465, 24) - \{1, 4, 5, 6, 3, 9, 2, 7, 8, 10\}.
\end{aligned}
$$

Список литературы

[1] Северный завоз. Википедия `http://ru.wikipedia.org/wiki/Северный_завоз`.

[2] *Коган Д. И., Федосенко Ю. С., Шеянов А. В.* Моделирование и оптимизация управления потоком объектов в однопроцессорной системе с изодромным элементом // Межвузовский сб. науч. тр. Вып. 273. Ч. 1. — Н. Новгород: Изд-во ВГАВТ, 1996. — С. 44–54.

[3] *Подиновский В. В., Ногин В. Д.* Парето-оптимальные решения многокритериальных задач. — М.: Физматлит, 2007.

[4] *Коган Д. И., Федосенко Ю. С.* Задача диспетчеризации: анализ вычислительной сложности и полиномиально разрешимые подклассы // Дискретная математика. — 1996. — Т. 8, № 3. — С. 135–147.

[5] *Беллман Р., Дрейфус С.* Прикладные задачи динамического программирования. — М.: Наука, 1965.

[6] *Klamroth K., Wiecek M.* Dynamic Programming Approaches to the Multiple Criteria Knapsack Problem. Technical Report #666. Dept. of Math. Sc., Clemson University. — Clemson: SC, 1998.

[7] *Коган Д. И.* Динамическое программирование и дискретная многокритериальная оптимизация. — Нижний Новгород: Изд-во ННГУ им. Н. И. Лобачевского, 2005.

Управляющие конфликтные системы и аппроксимация потока Гнеденко–Коваленко

А. М. Федоткин

fandr@vmk.unn.ru

Нижегородский государственный университет им. Н. И. Лобачевского

В известной монографии [1] впервые рассматривается неординарный пуассоновский поток, когда в каждый вызывающий момент поступает не более двух заявок. Однако не приводятся примеры реальных потоков такой вероятностной структуры. Более того, в указанной монографии нет обоснования, когда и при каких условиях реально возникают такого рода потоки. В данной работе, применяя кибернетический подход для управляющих систем обслуживания, получаем условия, при которых транспортные потоки машин на магистралях могут быть аппроксимированы потоками Гнеденко–Коваленко.

Введение

Обозначим через $(\Omega,\ \mathcal{F}, \mathbf{P}(\cdot))$ вероятностную модель для случайного потока требований (заявок), которые поступают в некоторую реальную систему массового обслуживания. Элемент ω из множества Ω есть описание элементарного исхода потока требований. При этом $\mathcal{F}$ есть σ-алгебра всех наблюдаемых исходов потока заявок и $\mathbf{P}(\cdot)$ является вероятностной функцией на $\mathcal{F}$. При каждом $i = 1, 2, \ldots$ случайная величина τ_i' определяет i-й момент поступления требования в систему. Тогда случайная последовательность $\{\tau_i';\ i \geqslant 1\}$ является математической моделью реального потока заявок. Потоку $\{\tau_i'; i \geqslant 1\}$ взаимно-однозначно соответствует так называемый считающий случайный процесс $\{\eta(t);\ t \geqslant 0\}$. Здесь $\eta(t)$ при $t > 0$ определяет число поступивших в систему заявок за промежуток времени $[0,\ t)$ и $\eta(t) = \eta(t-0)$, $\eta(0) = 0$. Как правило, случайные величины $\tau_{i+1}' - \tau_i'$, $i \geqslant 1$, являются зависимыми и имеют различные функции распределения. В этом случае практически не удается найти распределения вероятностей для процесса $\{\eta(t);\ t \geqslant 0\}$. В работе рассматривается именно такая сложная ситуация.

Пусть теперь $\tau_0 = \tau_1'$ и $\{\tau_i = \tau_{k_i}';\ i \geqslant 0\}$ при $i \geqslant 0$, $k_i \in \{1, 2, \ldots\}$, $k_0 = 1$, есть подпоследовательность случайной последовательности $\{\tau_i'; i \geqslant 1\}$. Требования и моменты τ_{k_i}', $i \geqslant 0$, в которые эти требования поступают в систему, будем называть стробирующими. Обозначим через η_i число всех заявок на промежутке времени $[\tau_i, \tau_{i+1})$, $i \geqslant 0$. В дальнейшем вместо потока требований $\{\tau_i';\ i \geqslant 1\}$ предлагается рассматривать поток $\{(\tau_i, \eta_i);\ i \geqslant 0\}$, где $\eta_i = \eta(\tau_{i+1}) - \eta(\tau_i) = k_{i+1} - k_i$, $i \geqslant 0$. Случайная величина η_i при каждом $i \geqslant 0$ измеряет размер так называемой i-й группы потока $\{(\tau_i, \eta_i);\ i \geqslant 0\}$.

К сожалению, почти всегда распределения потока заявок $\{\tau_i';\ i \geqslant 1\}$ неизвестны или, в лучшем случае, имеют сложный вид. Поэтому возникает проблема выбора функциональной зависимости элементов τ_i, $i \geqslant 0$, от моментов τ_i', $i \geqslant 1$, которая позволяет по наблюдениям за конечным отрезком последовательности $\{\tau_i';\ i \geqslant 1\}$ идентифицировать конечномерные распределения стробирующего потока $\{(\tau_i, \eta_i);\ i \geqslant 0\}$.

Определение стробирующего потока $\{(\tau_i, \eta_i);\ i \geqslant 0\}$

В работе предлагается такой алгоритм построения последовательности $\{(\tau_i, \eta_i);\ i \geqslant 0\}$, для которого указанная последовательность будет составлена из независимых и одинаково распределенных векторных случайных величин. В этом случае легко определить конечномерные распределения стробирующего потока $\{(\tau_i, \eta_i);\ i \geqslant 0\}$. Перейдем к описанию алгоритма.

Пусть при $c = 0, 1, \ldots$ моменты $\tau_i^{(c)} < \tau_{i+1}^{(c)}$, $i \geqslant 0$, совпадают с некоторыми элементами последовательности $\{\tau_i';\ i \geqslant 1\}$, т. е. $\tau_i^{(c)} = \tau'_{k_{c,i}}$, $k_{c,i} \in \{1, 2, \ldots\}$. Тогда величина $\eta_i^{(c)} = k_{c,i+1} - k_{c,i}$ задает число всех типов заявок на промежутке $[\tau_i^{(c)}, \tau_{i+1}^{(c)})$. При новом описании исходного потока $\{\tau_i';\ i \geqslant 1\}$ в виде последовательности $\{(\tau_i^{(c)}, \eta_i^{(c)});\ i \geqslant 0\}$ величину $\eta_i^{(c)}$ условно назовем i-й группой, а величину $\delta_i^{(c)} = \tau'_{k_{c,i+1}} - \tau'_{k_{c,i+1}-1}$ — интервалом между последовательными группами $\eta_i^{(c)}$, $\eta_{i+1}^{(c)}$. Моменты $\tau_i^{(c)}$, $c \geqslant 0$, $i \geqslant 0$, будем строить с помощью рекуррентных соотношений: $k_{0,i+1} = \inf\{j :\ j > k_{0,i},\ \tau_j' - \tau_{j-1}' \geqslant h_0\}$, $s_c = \inf\{j :\ j \geqslant 0,\ \eta_j^{(c)} \leqslant d,\ \eta_{j+1}^{(c)} \leqslant d,\ \delta_j^{(c)} < h_1, \eta_j^{(c)} = \eta_{j-1}^{(c)}\}$, $\tau_i^{(c+1)} = \tau_i^{(c)}$ при $i \leqslant s_c$ и $\tau_i^{(c+1)} = \tau_{i+1}^{(c)}$ при $i > s_c$. В этих формулах при каждом $c = 0, 1, \ldots$ величина $\eta_{-1}^{(c)} = 1$, $k_{0,0} = 1$, d — некоторое натуральное число, а постоянные величины h_0, h_1 удовлетворяют условию $h_0 < h_1$. Этот алгоритм выбора последовательностей $\{(\tau_i^{(c)}, \eta_i^{(c)});\ i \geqslant 0\}$, $c = 0, 1, \ldots$, используя величину h_0, сначала разбивает исходный процесс $\{\tau_i';\ i \geqslant 1\}$ на группы с целью получения маркированного точечного процесса $\{(\tau_i^{(0)}, \eta_i^{(0)});\ i \geqslant 0\}$ нулевого уровня. Далее, последовательно, начиная с нулевой группы $\eta_0^{(0)}$, алгоритм объединяет первые две соседние группы $\eta_j^{(0)}$ и $\eta_{j+1}^{(0)}$, если каждая из них содержит не более d заявок, интервал между такими группами строго меньше величины h_1 и, наконец, выполняется равенство $\eta_j^{(0)} = \eta_{j-1}^{(0)}$. Это позволяет найти процесс $\{(\tau_i^{(1)}, \eta_i^{(1)});\ i \geqslant 0\}$ первого уровня, к которому применяем ту же самую процедуру, что и к процессу $\{(\tau_i^{(0)}, \eta_i^{(0)});\ i \geqslant 0\}$. В результате получаем маркированный точечный процесс $\{(\tau_i^{(2)}, \eta_i^{(2)});\ i \geqslant 0\}$ второго уровня и т. д.

Теорема 1. *Для любой реализации $\omega = \{t'_i;\ i \geqslant 1\}$ случайной последовательности $\{\tau'_i;\ i \geqslant 1\}$ и для любого фиксированного $i \geqslant 0$ существуют пределы $\lim\limits_{c\to\infty} k_{c,i}(\omega)$, $\lim\limits_{c\to\infty} \tau_i^{(c)}(\omega)$.*

Данная теорема позволяет для любого $i \geqslant 0$ определить случайные величины $k_i = \lim\limits_{c\to\infty} k_{c,i}$, $\tau_i = \lim\limits_{c\to\infty} \tau_i^{(c)}$. При таком алгоритме выбора потока $\{(\tau_i, \eta_i);\ i \geqslant 0\}$ имеем $\tau_i = \tau'_{k_i}$, $\eta_i = k_{i+1} - k_i$ для всех $i \geqslant 0$ и, значит, таким способом определяем число всех требований на промежутке времени $[\tau_i, \tau_{i+1})$. Для определения приемлемого тестового распределения случайной величины η_i для всех $i \geqslant 0$ предположим, что процесс формирования i-й группы осуществляется некоторой управляющей конфликтной системой массового обслуживания с переменной структурой. В такой системе стробирующая заявка из i-й группы является обслуживающим устройством для другого типа (нестробирующих) требований из этой группы. Под обслуживанием можно, например, понимать пересылку требования из i-й группы в $(i+1)$-ю группу или её выход из системы. Обозначим через $\eta_0(\omega; t, \Delta t)$ случайное число нестробирующих заявок, поступающих за промежуток времени $[0, t)$ в i-ю группу по закону Пуассона с параметром $\lambda_0 > 0$. Далее через $\xi(\omega; t, \Delta t)$ обозначим случайное число обслуженных нестробирующих заявок из i-й группы за промежуток времени $[t, t+\Delta t)$. Наконец, величина $\varkappa(\omega; t)$ определяет число всех типов заявок из i-й группы в момент $t \geqslant 0$. Пусть $\varkappa(\omega; t) = 1, 2$. Это возможно, если при малых $\Delta t > 0$ имеет место:

$$\mathbf{P}(\{\omega\colon \xi(\omega; t, \Delta t) = 1\} | \{\omega\colon \varkappa(\omega; t) = 2, \eta_0(\omega; t, \Delta t) = 0\}) = \mu_1 \Delta t - o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \xi(\omega; t, \Delta t) = 1\} | \{\omega\colon \varkappa(\omega; t) = 2, \eta_0(\omega; t, \Delta t) = 1\}) = 1,$$

где параметр μ_1^{-1} определяет среднее время обслуживания в случае, когда i-я группа состоит из двух заявок, и символ $o(\Delta t)$ обозначает относительно Δt бесконечно малую величину более высокого порядка. С использованием результатов из [2] доказывается, что $\lim\limits_{t\to\infty} \mathbf{P}(\{\omega\colon \varkappa(\omega; t) = 1\}) = \mu_1(\lambda_0 + \mu_1)^{-1} = p$ и $\lim\limits_{t\to\infty} \mathbf{P}(\{\omega\colon \varkappa(\omega; t) = 2\}) = \lambda_0(\lambda_0 + \mu_1)^{-1} = q$. Теперь можно выдвинуть гипотезу H_0 о том, что стационарный режим функционирования групп требований задается последовательностью $\{\eta_i;\ i \geqslant 0\}$ из независимых случайных величин, каждая из которых имеет распределение вида: $\mathbf{P}(\{\omega\colon \eta_i(\omega) = 1\}) = p$, $\mathbf{P}(\{\omega\colon \eta_i(\omega) = 2\}) = q$. Будем рассматривать поток из неоднородных требований как неординарный пуассоновский поток в случае, если в любой вызывающий момент поступает одна заявка с вероятностью p и две заявки — с вероятностью q. При пуассоновском потоке стробирующих заявок с интенсивностью $\lambda > 0$ и обозначениях $P_k(t) = \mathbf{P}(\{\omega\colon \eta(\omega;t) = k\})$, $k = 0, 1, \ldots$, получены формулы: $P_0(t) = e^{-\lambda t}$, $P_1(t) = \lambda t p e^{-\lambda t}$, $P_k(t) = e^{-\lambda t}\sum_{i=0}^{[\frac{k}{2}]} C_{k-i}^i p^{k-2i} q^i \frac{(\lambda t)^{k-i}}{(k-i)!}$, $k = 2, 3, \ldots$, $\mathbf{M}\eta(t) = \lambda t(1+q)$, $\mathbf{D}\eta(t) = \lambda t(1+3q)$, $\mathrm{Ка}\eta(t) = (1+7q)(\lambda t)^{-1/2}(1+3q)^{-3/2}$, $\text{Э}\eta(t) = (1+15q)(\lambda t)^{-1}(1+3q)^{-2}$. Здесь через символы $\mathbf{M}(\cdot)$, $\mathbf{D}(\cdot)$, $\mathrm{Ка}(\cdot)$ и $\text{Э}(\cdot)$ обозначены математическое ожидание, дисперсия, коэффициент асимметрии и эксцесс случайной величины. Изучены экстремальные свойства приведенных числовых характеристик, и получены оценки параметров законов распределения для неординарного потока $\{\eta(t); t > 0\}$ неоднородных требований. Приводится методика проверки гипотезы H_0 и гипотезы о движении автомобилей на магистралях по закону Гнеденко–Коваленко.

Список литературы

[1] *Гнеденко Б. В., Коваленко И.* Введение в теорию массового обслуживания. — М.: Ком Книга, 2005.

[2] *Федоткин М. А., Федоткин А. А.* Дискретные модели в теории транспортных потоков // Сб. научн. статей VIII Международной научн. конф. «Дискретные модели в теории управляющих систем». — М.: МАКС Пресс, МГУ, 2009. — С. 305–311.

Управляющие системы и механизм образования транспортных пачек на магистралях с интенсивным движением

М. А. Федоткин, Е. В. Кудрявцев

fma5@rambler.ru

Нижегородский государственный университет им. Н. И. Лобачевского

Решена проблема построения и изучения математической модели пространственной и временной характеристик неоднородного транспортного потока на магистрали при большой плотности быстрых машин и неинтенсивном движении медленных машин. При построении вероятностной модели транспортного трафика существенно использовался кибернетический подход, предложенный Ляпуновым–Яблонским.

Введение

Математической теории транспортных потоков в случае независимых интервалов между моментами пересечения автомобилями некоторой поперечной линии магистрали посвящено большое число монографий и статей [1]. При этом предполагается, что транспортный поток сотоит из однородных или однотипных машин. Для реального транспортного потока каждый автомобиль осуществляет движение в экстремальных погодных и дорожных условиях. При этом скорости автомобилей являются непрерывными случайными величинами с различными интегральными функциями распределения. В силу этого транспортный поток машин на магистралях существенно отличается от потока случайных событий классической теории массового обслуживания. Рассмотрен нетрадиционный способ описания потока неоднородных машин, который основан на изучении распределения величины транспортной пачки и распределения потока так называемых медленных или головных в автоколонне машин. При этом предполагается интенсивное движение другого типа машин — быстрых. Пусть $(\Omega, \mathcal{F}, \mathbf{P}(\cdot))$ — вероятностная модель функционирования транспортной пачки и ω — произвольный элемент достоверного события Ω. Тогда $\mathcal{F}$ есть σ-алгебра всех наблюдаемых исходов процесса фукционирования транспортной пачки и $\mathbf{P}(\cdot)$ является вероятностной функцией на $\mathcal{F}$. В соответствии с этой задачей обозначим через $\xi(\omega; t, \Delta t)$ число быстрых машин, которые поступают в транспортную пачку за промежуток времени $[t, t+\Delta t)$ по закону Пуассона с интенсивностью $\lambda > 0$. Пусть случайная величина $\chi(\omega; t)$ измеряет число всех типов машин в транспортной пачке в момент времени $t \geqslant 0$. Обозначим теперь через $\eta(\omega; t, \Delta t)$ случайное число быстрых машин, которые могут обогнать медленную за промежуток времени $[t, t+\Delta t)$. Вполне естественно предположить, что при малых значениях $\Delta t > 0$ условные вероятности событий, которые порождаются дискретной случайной величиной $\eta(\omega; t, \Delta t)$, определяются следующими соотношениями:

$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)=0\}\mid\{\omega\colon \chi(\omega;t)=2,\xi(\omega;t,\Delta t)=0\})=1-\mu_1\Delta t+o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)=1\}\mid\{\omega\colon \chi(\omega;t)=2,\xi(\omega;t,\Delta t)=0\})=\mu_1\Delta t-o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)=0\}\mid\{\omega\colon \chi(\omega;t)=3,\xi(\omega;t,\Delta t)=0\})=1-\mu_2\Delta t+o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)=1\}\mid\{\omega\colon \chi(\omega;t)=3,\xi(\omega;t,\Delta t)=0\})=\mu_2\Delta t-o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)=0\}\mid\{\omega\colon \chi(\omega;t)=m,\xi(\omega;t,\Delta t)=0\})=1-\mu_3\Delta t+o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)=1\}\mid\{\omega\colon \chi(\omega;t)=m,\xi(\omega;t,\Delta t)=0\})=\mu_3\Delta t-o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)=0\}\mid\{\omega\colon \chi(\omega;t)=m-3,\xi(\omega;t,\Delta t)=1\})=1-O(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)=1\}\mid\{\omega\colon \chi(\omega;t)=m-3,\xi(\omega;t,\Delta t)=1\})=O(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \eta(\omega;t,\Delta t)\geqslant 2\}\mid\{\omega\colon \chi(\omega;t)=m-2,\xi(\omega;t,\Delta t)=1\})=o(\Delta t),$$
$$m=4,5,\ldots$$

В этих равенствах параметры μ_1^{-1} и μ_2^{-1} задают среднее время обгона в случае, когда транспортная пачка состоит из двух и трех машин соответственно. Аналогично параметр μ_3^{-1} определяет среднее время обгона, если транспортная пачка состоит из четырех и более машин. Параметры μ_1,μ_2,μ_3 будем называть интенсивностями обгона. Таким способом моделируется зависимость среднего времени обгона от числа машин в транспортной пачке. Итак, при заданном размере транспортной пачки условная вероятность того, что за (где угодно расположенный) промежуток Δt по меньшей мере две машины обгонят медленную, есть величина бесконечно малая по сравнению с Δt. Обозначим через $Q(t,m)$ вероятность $\mathbf{P}(\{\omega\colon \chi(\omega;t)=m\})$ при фиксированных $t>0$ и $m=1,\ 2,\ \ldots$ В работе [2] при $\lambda<\mu_3$ было отмечено, что существует единственное стационарное распределение $\lim\limits_{t\to\infty}Q(t,m)=Q(m)>0,\sum_{m=1}^{\infty}Q(m)=1$ для числа $\chi(\omega)$ всех типов машин транспортной пачки в стационарном режиме движения автоколонн по магистрали. Для данного потока обозначим теперь через $\varkappa(\omega;t)$ случайное число всех типов машин, которые пересекают фиксированную поперечную линию автомагистрали за промежуток времени $[0,t)$. При $\alpha=\lambda\mu_1^{-1},\beta=\lambda\mu_2^{-1},\gamma=\lambda\mu_3^{-1}$, $p=(1+\alpha+\alpha\beta/(1-\gamma))^{-1}$ и пуассоновском движении медленных машин с интенсивностью $\mu > 0$ получены формулы:

$$Q(1) = (1 + \alpha + \alpha\beta/(1-\gamma))^{-1}, \quad Q(2) = \alpha(1 + \alpha + \alpha\beta/(1-\gamma))^{-1},$$
$$Q(m) = \alpha\beta\gamma^{m-3}(1 + \alpha + \alpha\beta/(1-\gamma))^{-1}, \quad m \geqslant 3;$$
$$\mathbf{P}(\{\omega\colon \varkappa(\omega;t) = 0\}) = e^{-\mu t},$$

$$\mathbf{P}(\{\omega\colon \varkappa(\omega;t) = k\}) = e^{-\mu t}\sum_{n=0}^{[\frac{k}{2}]}\alpha^n\Big(\frac{(\mu tp)^{k-n}}{n!(k-2n)!} + \\ + \sum_{m=1}^{\min\{k-2n,n\}}\beta^m\sum_{l=0}^{k-2n-m}\gamma^l\frac{(\mu tp)^{k-n-m-l}C_{m+l-1}^l}{(n-m)!m!(k-2n-m-l)!}\Big);$$

$$\mathbf{M}\varkappa(\omega;t) = \mu tp\left(1 + 2\alpha + \alpha\beta\left[\frac{2}{1-\gamma} + \frac{1}{(1-\gamma)^2}\right]\right),$$

$$\mathbf{D}\varkappa(\omega;t) = \mu tp\left(1 + 4\alpha + \alpha\beta\left[\frac{4}{1-\gamma} + \frac{3}{(1-\gamma)^2} + \frac{2}{(1-\gamma)^3}\right]\right),$$

$$K_a\varkappa = \left(1 + 8\alpha + \alpha\beta\left[\frac{8}{1-\gamma} + \frac{7}{(1-\gamma)^2} + \frac{6}{(1-\gamma)^3} + \frac{6}{(1-\gamma)^4}\right]\right)\times \\ \times(\mu tp)^{-1/2}\left(1 + 4\alpha + \alpha\beta\left[\frac{4}{1-\gamma} + \frac{3}{(1-\gamma)^2} + \frac{2}{(1-\gamma)^3}\right]\right)^{-\frac{3}{2}},$$

$$Э\varkappa = \Big(1 + 16\alpha + \alpha\beta\Big[\frac{16}{1-\gamma} + \frac{15}{(1-\gamma)^2} + \frac{14}{(1-\gamma)^3} + \frac{12}{(1-\gamma)^4} + \\ + \frac{24}{(1-\gamma)^5}\Big]\Big)(\mu tp)^{-1}\left(1 + 4\alpha + \alpha\beta\left[\frac{4}{1-\gamma} + \frac{3}{(1-\gamma)^2} + \frac{2}{(1-\gamma)^3}\right]\right)^{-2}.$$

Транспортная пачка и управляющие системы

В данной работе произвольная транспортная пачка представляется как некоторая управляющая кибернетическая система [3], для которой имеет место принцип непрерывности актов её функционирования во времени $t > 0$. Для такой системы выделены схема, информация, координаты и функция. Для схемы определены её структурные блоки: 1) входной полюс – пуассоновский входной поток быстрых машин с интенсивностью $\lambda > 0$, который поступает в транспортную пачку; 2) внешняя память – транспортная пачка из быстрых машин и медленной машины; 3) устройство по переработке внешней памяти – правило отбора быстрых машин из транспортной пачки для обгона медленной машины; 4) внутренняя память – механизм обгона для быстрых машин медленной машины; 5) устройство по переработке внутренней памяти – изменение механизма обгона для быстрых машин в зависимости от числа машин в транспортной пачке; 6) выходной полюс – поток быстрых машин, которые обогнали медленную. Для такого типа управляющей системы решены следующие задачи: 1) проведено кодирование информации или нелокальное описание структурных блоков; 2) выявлены функциональные и статистические связи между блоками схемы; 3) определены свойства исходных и искомых характеристик системы. Это позволяет изучить экстремальные свойства основных числовых характеристик транспортного потока и получить оценки параметров его закона распределения.

Работа выполнена в ННГУ по теме № 0120.0602598 «Анализ дискретных управляющих систем обслуживания и систем вычисления булевых функций».

Список литературы

[1] *Хейт Ф. А.* Математическая теория транспортных потоков. — М.: Мир, 1966.

[2] *Федоткин М. А., Кудрявцев Е. В.* Построение и исследование математической модели неоднородного дорожного трафика // Массовое обслуживание: потоки, системы, сети. — Минск: БГУ–РИВШ, 2011. — № 21. — С. 76–81.

[3] *Федоткин М. А.* Нелокальный способ задания управляемых случайных процессов // Математические вопросы кибернетики. — М.: Наука, 1998. — № 7. — С. 332–344.

Исследование математической модели трафика автомобилей на основе подхода Ляпунова–Яблонского

М. А. Федоткин, М. А. Рачинская

fma5@rambler.ru

Нижегородский государственный университет им. Н. И. Лобачевского

С использованием общих свойств управляющих кибернетических систем построена и изучена вероятностная модель движения неоднородного транспорта на автомагистрали в плохих дорожных и погодных условиях. Приводится метод получения оценок параметров закона распределения размера транспортной пачки при неинтенсивном движении быстрых машин.

Введение

В классической теории транспортных потоков рассматривается ситуация [1], когда последовательные моменты пересечения машинами виртуальной поперечной линии магистрали на оси времени образуют неординарный процесс Пуассона. Однако на практике действие некоторых факторов (плохие погодные и дорожные условия, разнородность машин) затрудняет беспрепятственное движение машин по автомагистрали. При таких условиях автомобили не могут свободно обгонять друг друга и мы можем наблюдать образование транспортных пачек или автоколонн. В силу этого построение и изучение как вероятностной модели пространственного расположения машин на магистрали, так и вероятностной модели последовательности зависимых временных интервалов между ближайшими машинами представляет значительные трудности. В работе [2] предполагается, что каждая автоколонна состоит из одной медленной машины во главе и очереди быстрых машин, ожидающих возможности обгона. Поэтому в работе [2] процесс формирования транспортной пачки рассматривается как функционирование системы массового обслуживания с переменной структурой и с ограниченной очередью. При этом каждая медленная машина может быть интерпретирована как обслуживающее устройство для быстрых машин. Здесь под обслуживанием понимается обгон быстрой машиной медленной. В отсутствие медленных машин быстрые двигаются относительно беспрепятственно.

Естественно предположить, что быстрые машины поступают в пачку по закону Пуассона с относительно малой интенсивностью. Пусть $(\Omega, \mathcal{F}, \mathbf{P}(\cdot))$ — вероятностная модель функционирования транспортной пачки и ω — произвольный элемент достоверного события Ω. Тогда $\mathcal{F}$ есть σ-алгебра всех наблюдаемых исходов процесса функционирования транспортной пачки и $\mathbf{P}(\cdot)$ является вероятностной функцией на $\mathcal{F}$. Обозначим теперь через $\eta_0(\omega; t)$ случайное число быстрых машин, поступающих за промежуток времени $[0, t)$ в автоколонну по закону Пуассона с параметром $\lambda_0 > 0$. Далее через $\xi(\omega; t, \Delta t)$ обозначим случайное число быстрых машин, обогнавших медленную за промежуток времени $[t, t + \Delta t)$, и введем случайную величину $\varkappa(\omega; t)$, считающую число всех типов машин в транспортной пачке в момент времени $t \geqslant 0$. Пусть $\varkappa(\omega; t)$ принимает значения из множества $\{1, 2, \ldots, N\}$. Это возможно, если интенсивность обгона медленной машины быстрыми значительно превышает интенсивность поступления быстрых машин в автоколонну. В этом случае образуются транспортные пачки относительно небольшого размера. Поскольку среднее время обгона быстрыми машинами медленной зависит от числа машин в транспортной пачке, то необходимо различать следующие ситуации. Пусть μ_1^{-1} и μ_2^{-1} — среднее время обгона в случае, когда пачка состоит из двух и трех машин соответственно. Предположим также, что среднее время обгона не меняется, если в пачке находится более трех машин. Обозначим его через μ_3^{-1}. Параметры μ_1, μ_2 и μ_3 будем называть интенсивностями обгона в указанных случаях. На параметры системы следует наложить ограничение $\lambda_0 < \mu_3$, которое содержательно описывает условие, что интенсивность обгона превышает интенсивность поступления машин в пачку. Учитывая, что машины в транспортном потоке не могут быть потеряны и что пачка не может состоять более чем из N машин, предположим выполнение следующего ограничения. Если быстрая машина догоняет полную автоколонну из N автомобилей, то она все же присоединяется к ней. При этом одновременно с этим быстрая машина, движущаяся вслед за головной, обязательно совершает обгон. Теперь можем следующими соотношениями определить при малых $\Delta t > 0$ условные вероятности событий, порожденных величиной $\xi(\omega; t, \Delta t)$:

$$\mathbf{P}(\{\omega\colon \xi(\omega; t, \Delta t) = 0\} \mid \{\omega\colon \varkappa(\omega; t) = 1, \eta_0(\omega; t, \Delta t) = 1\}) = 1 - O(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \xi(\omega; t, \Delta t) = 0\} \mid \{\omega\colon \varkappa(\omega; t) = 2, \eta_0(\omega; t, \Delta t) = 0\}) = 1 - \mu_1 \Delta t + o(\Delta t),$$

$$\mathbf{P}(\{\omega\colon \xi(\omega;t,\Delta t)=1\}\mid\{\omega\colon \varkappa(\omega;t)=2,\eta_0(\omega;t,\Delta t)=0\})=\mu_1\Delta t-o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \xi(\omega;t,\Delta t)=0\}\mid\{\omega\colon \varkappa(\omega;t)=3,\eta_0(\omega;t,\Delta t)=0\})=1-\mu_2\Delta t+o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \xi(\omega;t,\Delta t)=1\}\mid\{\omega\colon \varkappa(\omega;t)=3,\eta_0(\omega;t,\Delta t)=0\})=\mu_2\Delta t-o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \xi(\omega;t,\Delta t)=0\}\mid\{\omega\colon \varkappa(\omega;t)=k,\eta_0(\omega;t,\Delta t)=0\})=1-\mu_3\Delta t+o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \xi(\omega;t,\Delta t)=1\}\mid\{\omega\colon \varkappa(\omega;t)=k,\eta_0(\omega;t,\Delta t)=0\})=\mu_3\Delta t-o(\Delta t),$$
$$\mathbf{P}(\{\omega\colon \xi(\omega;t,\Delta t)=1\}\mid\{\omega\colon \varkappa(\omega;t)=N,\eta_0(\omega;t,\Delta t)=1\})=1,$$

где $4\leqslant k\leqslant N$. Из этих соотношений следует, что при заданном размере транспортной пачки условная вероятность того, что за (где угодно расположенный) промежуток Δt по меньшей мере две машины обгонят медленную, есть величина бесконечно малая по сравнению с Δt. Обозначим через $Q(t,k)$ вероятность $\mathbf{P}(\{\omega\colon \varkappa(\omega;t)=k\})$, которая определена при $k=1,2,\ldots,N$ и $t\geqslant 0$. В работе [2] с использованием теоремы Маркова при $\lambda<\mu_3$ доказано, что существует единственное стационарное распределение $\lim\limits_{t\to\infty}Q(t,m)= Q(m)>0,\sum_{m=1}^{\infty}Q(m)=1$ для числа $\varkappa(\omega)$ всех типов машин транспортной пачки в стационарном режиме движения автоколонн по магистрали. Пусть $\nu_1=\frac{\lambda_0}{\mu_1}$, $\nu_2=\frac{\lambda_0}{\mu_2}$, $\nu_3=\frac{\lambda_0}{\mu_3}$, $p=(1+\nu_1+\nu_1\nu_2)^{-1}$, $q=\nu_1(1+\nu_1+\nu_1\nu_2)^{-1}$, $s=\nu_1\nu_2(1+\nu_1+\nu_1\nu_2)^{-1}$, а величина $\eta(\omega;t)$ подсчитывает число всех типов автомобилей, пересекающих поперечную линию магистрали за время $[0,t)$, и $P_k(t)=\mathbf{P}(\{\omega\colon \eta(\omega;t)=k\})$, $k\geqslant 0$. Будем рассматривать транспортный поток как неординарный пуассоновский поток в случае, если в любой вызывающий момент поступает одна заявка с вероятностью p, две заявки — с вероятностью q и три — с вероятностью s. При пуассоновском движении медленных машин с интенсивностью $\lambda>0$ и $N=3$ получены формулы:

$$Q(1)=p,\quad Q(2)=q,\quad Q(3)=s,\quad P_0(t)=e^{-\lambda t},\quad P_1(t)=\lambda tpe^{-\lambda t},$$
$$P_k(t)=e^{-\lambda t}\sum_{i=0}^{[\frac{k}{2}]}\sum_{j=0}^{[\frac{k-2i}{3}]}\binom{k-i-2j}{i,j,k-2i-3j}p^{k-2i-3j}q^is^j\frac{(\lambda t)^{k-i-2j}}{(k-i-2j)!},$$
$$k=2,3,\ldots$$
$$\mathbf{M}\eta(t)=\lambda t(1+q+2s),\qquad \mathbf{D}\eta(t)=\lambda t(1+3q+8s),$$
$$K_a\eta(t)=(1+7q+26s)(\lambda t)^{-1/2}(1+3q+8s)^{-3/2},$$
$$\text{Э}\eta(t)=(1+15q+80s)(\lambda t)^{-1}(1+3q+8s)^{-2},$$

где через символы $K_a(\cdot)$ и $Э(\cdot)$ обозначены коэффициент асимметрии и эксцесс случайной величины соответственно.

Представление модели транспортной автоколонны на магистрали в виде эволюционной управляющей системы

В этой работе произвольная транспортная пачка представляется как управляющая кибернетическая система [3], для которой имеет место принцип непрерывности актов её функционирования во времени $t \geqslant 0$. Для системы выделены схема, информация, координаты и функция. Для схемы определены структурные блоки: 1) входной полюс — пуассоновский поток быстрых машин, поступающих в автоколонну; 2) внешняя память — транспортная пачка из медленной и быстрых машин; 3) устройство по переработке внешней памяти — правило отбора быстрых машин для обгона медленной машины; 4) внутренняя память — механизм обгона для быстрых машин; 5) устройство по переработке внутренней памяти — изменение механизма обгона в зависимости от размера автоколонны; 6) выходной полюс — поток быстрых машин, которые обогнали медленную. Для такой управляющей системы решены задачи: 1) проведено кодирование информации или нелокальное описание блоков; 2) выявлены функциональные и статистические связи между блоками; 3) определены свойства исходных и искомых характеристик системы. Это позволяет изучить экстремальные свойства числовых характеристик транспортного потока и получить оценки параметров его закона распределения.

Работа выполнена в ННГУ по теме № 0120.0602598 «Анализ дискретных управляющих систем обслуживания и систем вычисления булевых функций».

Список литературы

[1] *Хейт Ф. А.* Математическая теория транспортных потоков. — М.: Мир, 1966.

[2] *Fedotkin M. A., Rachinskaya M. A.* Investigation of Traffic Flows Characteristics in Case of the Small Density // Queues: Flows, Systems, Networks. Proc. Int. Conf. «Modern Probabilistic Methods for Analysis and Optimization of Information and Telecommunication Networks». — Minsk: BSU-RIVH, 2011. — № 21. — С. 82–87.

[3] *Федоткин М. А.* Процессы обслуживания и управляющие системы // Математические вопросы кибернетики. Вып. 6. — М.: Наука, 1996. — С. 51–70.

Кибернетический подход к изучению выходных процессов управления потоками Бартлетта

М. А. Федоткин, А. А. Федоткин

fma5@rambler.ru

Нижегородский государственный университет им. Н. И. Лобачевского

Введение

В работе [1] впервые дано обоснование, когда и при каких условиях возникают потоки Бартлетта. Подробно изучены вероятностные и интегральные свойства таких потоков. Показана эффективность применения потоков Бартлетта при построении и изучении дискретных моделей движения неоднородных автомобилей на магистралях. С использованием общего понятия кибернетической управляющей системы обслуживания [2] в данной работе построена и исследована математическая модель выходных потоков, возникающих в системе обслуживания и управления m конфликтными потоками Бартлетта в классе циклических алгоритмов. Получены рекуррентные соотношения как для векторной марковской последовательности, которая является математической моделью конфликтной системы массового обслуживания с переменной структурой, так и для одномерных распределений этой последовательности. Доказано, что нелокальное описание выходных потоков в таких неклассических системах обслуживания можно выполнить с помощью маркированного точечного процесса с выделенной дискретной компонентой. Следует отметить, что проблема выходных потоков была решена только для простейших классических систем с ожиданием при пуассоновсом входном потоке и показательном законе обслуживания. Проведена классификация по Колмогорову состояний рассматриваемой системы обслуживания и управления m конфликтными потоками Бартлетта. Найдены необходимое и достаточное условия существования стационарного режима системы. С использованием метода имитационного моделирования осуществляется статистический анализ основных характеристик управляющей системы обслуживания. Этот анализ позволяет определить квазиоптимальное управление потоками по условию минимума средних задержек требований в системе. Аналитические и численные результаты интерпретируются и применяются при управлении транспортными конфликтными потоками машин на локальном перекрестке.

Построение модели управляющей системы обслуживания и её анализ

При кибернетическом подходе [2] для любой управляющей системы обслуживания необходимо выделить схему, информацию, координаты и функцию. Схема управляющей системы обслуживания отражает ее скелетное строение и дает возможность графического изображения системы с помощью фиксированного числа ее структурных блоков и заданных связей между блоками. Функциональная схема включает следующие структурные блоки: *a*) входные потоки Π_1, Π_2, ..., Π_m неоднородных требований — первый тип входных полюсов; *b*) потоки насыщения $\Pi_1^{(\text{н})}$, $\Pi_2^{(\text{н})}$, ..., $\Pi_m^{(\text{н})}$ (выходные потоки при максимальной загрузке и эффективном функционировании системы) — второй тип входных полюсов; *c*) неограниченные накопители $N^{(1)}$, $N^{(2)}$, ..., $N^{(m)}$ очередей соответственно по входным потокам Π_1, Π_2, ..., Π_m — внешняя память; *d*) устройства $\delta_1, \delta_2, \ldots, \delta_m$ по организации дисциплины очередей в накопителях или стратегии механизма обслуживания — блок по переработке информации внешней памяти; *e*) обслуживающее устройство с $2m$ состояниями $\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(2m)}$ — внутренняя память; *f*) циклический граф переключений этих состояний, когда после состояния $\Gamma^{(r)}$ осуществляется мгновенный переход в состояние $\Gamma^{(r+1)}$ при $r < 2m$ и в состояние $\Gamma^{(1)}$ при $r = 2m$, — блок по переработке информации внутренней памяти; *g*) выходные потоки $\overline{\Pi}_1, \overline{\Pi}_2, \ldots, \overline{\Pi}_m$ — выходные полюсы. Общая схема таких управляющих систем обслуживания представлена на рис. 1 в работе [2]. Набор состояний очередей в накопителях, множество состояний обслуживающего устройства, входных потоков, потоков насыщения и потоков обслуженных требований полностью определяют информацию управляющей системы обслуживания. Номера всех входных потоков, потоков насыщения, выходных потоков, накопителей, механизмов формирования очередей и номера состояний обслуживающего устройства задают координаты управляющей системы обслуживания, которые определяют расположение блоков на схеме. Функция системы — это, прежде всего, циклическое управление потоками (разрешение или запрещение начала обслуживания каждого из них) и непосредственно обслуживание неоднородных требований. При каждом фиксированном $j \in \{1, 2, \ldots, m\}$ в состоянии $\Gamma^{(2j-1)}$ в течение времени T_{2j-1} согласно экстремальной стратегии δ_j [2] обслуживаются только требования потока Π_j в количестве не более величины $[\mu_j T_{2j-1}]$, а в состоянии $\Gamma^{(2j)}$ в течение времени T_{2j} запрещается обслуживание потоков. Здесь параметр μ_j^{-1} определяет среднее время обслуживания требования потока Π_j или μ_j есть интенсивность потока насыщения $\Pi_j^{(\text{н})}$.

На оси времени выберем начальный момент $\tau_0 > 0$, совпадающий с некоторым моментом смены состояния обслуживающего устройства. В дальнейшем будем отслеживать значения всех интересующих нас величин в дискретные моменты $\tau_i, i = 0, 1, \ldots$, переключений состояний обслуживающего устройства или на промежутках $[\tau_i, \tau_{i+1}), i = 0, 1, \ldots$ При $j = 1, 2, \ldots, m$ и $i \geqslant 0$ введем случайные элементы: 1) $\eta_{j,i}$ — число заявок потока Π_j, поступивших в систему за промежуток $[\tau_i, \tau_{i+1})$; 2) $\xi_{j,i}$ — максимально возможное число заявок потока Π_j, которые система виртуально может обслужить на промежутке времени $[\tau_i, \tau_{i+1})$; 3) Γ_i — состояние обслуживающего устройства на промежутке $[\tau_i, \tau_{i+1})$; 4) $\varkappa_{j,i}$ — число требований потока Π_j, находящихся в системе в момент τ_i; 5) $\xi'_{j,i}$ — число заявок потока Π_j, которые в действительности покидают систему на промежутке времени $[\tau_i, \tau_{i+1})$; 6) $\xi'_{j,-1}$ — число заявок потока Π_j, которые в действительности покидают систему на промежутке $[0, \tau_0)$.

Теорема 1. *Если значение функции $u(\Gamma^{(r)})$ равно $\Gamma^{(1)}$ при $r = 2m$ и равно $\Gamma^{(r+1)}$ при $r = 1, 2, \ldots, 2m-1$, то имеет место равенство* $(\Gamma_{i+1}, \varkappa_{j,i+1}, \xi'_{j,i}) = (u(\Gamma_i), \max\{0, \varkappa_{j,i}+\eta_{j,i}-\xi_{j,i}\}, \min\{\varkappa_{j,i}+\eta_{j,i}, \xi_{j,i}\})$.

Обозначим через $T = T_1 + T_2 + \cdots + T_{2m}$ период циклического управления потоками и через $\mathbf{M}\eta_j$ математическое ожидание числа требований потока Π_j, поступивших в систему за каждый промежуток времени $[\tau_i, \tau_i + T)$. Формула для вычисления математического ожидания $\mathbf{M}\eta_j$ числа требований потока Бартлетта на каждом промежутке $[\tau_i, \tau_i + T)$ приведена в работе [1].

Теорема 2. *Последовательность $\{(\Gamma_i, \varkappa_{j,i}, \xi'_{j,i-1}); i = 0, 1, \ldots\}$ при начальном распределении вектора $(\Gamma_0, \varkappa_{j,0}, \xi'_{j,-1})$ является марков-*

ской и неравенство $\mathbf{M}\eta_j - [\mu_j T_{2j-1}] < 0$ *является необходимым и достаточным условием для существования единственного стационарного режима в системе по потоку* Π_j.

Для рассматриваемых систем в общем случае аналитическим путем не удается получить такие ее характеристики, как законы распределения времени переходного процесса, длин очередей, времени ожидания начала обслуживания требований по потокам и законы распределения выходных потоков. Поэтому важно найти численные оценки указанных законов распределения и их интегральных характеристик, например численные оценки загрузки системы. Принцип блочного строения схемы управляемых систем обслуживания [2] позволяет построить имитационную модель таких систем. Программная реализация имитационной модели выполнена средствами разработок CodeGear RAD Studio 2009 на языке Object Pascal. Полный объём программы составляет 4,36 Мб на жестком диске, а без режима визуализации объём равен 1,37 Мб. Для моделирования использовался компьютер на базе процессора Intel Core 2 Duo. Результаты исследований на имитационной модели интерпретированы на задаче управления конфликтными транспортными потоками на пересечении магистралей. При этом за счет укрупнения некоторых состояний $\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(2m)}$ светофора задача управления m потоками легко сводится к задаче управления двумя наиболее интенсивными конфликтными потоками. Итак, в имитационной модели, ради простоты, полагаем, что $m = 2$ и интенсивными потоками являются Π_1 и Π_2. С использованием теоретических результатов и имитационного моделирования предложен метод решения проблемы определения квазиоптимальных параметров циклического управления транспортными потоками на изолированных перекрестках по условию минимума оценки среднего взвешенного времени ожидания начала обслуживания произвольного требования. Более того, квазиоптимальные параметры обеспечивают сравнительно небольшое значение дисперсии каждого из выходных потоков, и, значит, можно значительно уменьшить задержки транспорта на соседних перекрестках.

Работа выполнена в ННГУ по теме № 0120.0602598 «Анализ дискретных управляющих систем обслуживания и систем вычисления булевых функций».

Список литературы

[1] *Федоткин М. А., Федоткин А. А.* Дискретные модели в теории транспортных потоков // Сб. научн. статей VIII Международной научной конференции «Дискретные модели в теории управляющих систем». — М.: МАКС Пресс, МГУ, 2009. — С. 305–311.

[2] *Федоткин М. А.* Процессы обслуживания и управляющие системы // Математические вопросы кибернетики. Вып. 6. — М.: Наука, 1996. — С. 51–70.

Построение глубоких отсечений в булевом программировании

О. В. Хамисов

khamisov@isem.sei.irk.ru

Институт систем энергетики им. Л. А. Мелентьева СО РАН, Иркутск

Введение

В докладе описывается методика, впервые примененная в [2] для построения глубоких отсечений в вогнутом программировании и затем адаптированная в [3] для задач булева программирования. Дано теоретическое обоснование построения глубоких отсечений при помощи так называемого вогнутого продолжения.

Вогнутое продолжение

Пусть $X \subset \mathbb{R}^n$ — выпуклое замкнутое множество, $f : X \to \mathbb{R}$ — вогнутая, непрерывная на X функция.

Определение 1. Функция $W : \mathbb{R}^n \to \mathbb{R}$ называется вогнутым продолжением функции f на множестве X, если выполняются следующие условия:

1) W — непрерывная вогнутая функция;
2) $W(x) = f(x) \ \ \forall x \in X$;
3) $W(x) \geqslant f(x) \ \ \forall x \in \mathbb{R}^n$.

Множество всех вогнутых продолжений обозначим $EXT(f, X)$. Очевидно, что $f \in EXT(f, X)$.

Определение 2. Функция F называется максимальным вогнутым продолжением функции f на множестве X, если

1) $F \in EXT(f, X)$;
2) $F(x) \geqslant W(x)\ \forall x \in \mathbb{R}^n,\ \ \forall W \in EXT(f, X)$.

Везде далее будем предполагать, что f — дифференцируемая функция. Нетрудно показать, что максимальное вогнутое продолжение F вогнутой дифференцируемой функции f на X определяется следующим образом

$$F(x) = \min_{y \in X}\{f(y) + \nabla f(y)^T(x - y)\}. \tag{1}$$

Введем в рассмотрение множество

$$D = \{d \in \mathbb{R}^n : d = \nabla f(x),\ x \in X\},$$

которое называется образом градиентного отображения множества X, и $co(D)$ — выпуклую оболочку D.

Теорема. *Предположим, что*

$$0 \notin int(co(D)).$$

Тогда максимальное вогнутое продолжение F имеет рецессивное направление на $\mathbb{R}^n$.

Главное отличие максимального вогнутого продолжения F от исходной функции f состоит в том, что F может иметь рецессивные направления, даже если f рецессивных направлений не имеет. Именно существование рецессивных направлений у F и позволяет строить более глубокие отсечения.

Очевидно, что максимальное вогнутое продолжение имеет смысл в том случае, когда задача (1) практически разрешима.

Пример. Функция f — вогнутая квадратичная функция

$$f(x) = x^T Q x + c^T x,$$

Q — отрицательно полуопределенная симметричная матрица. Тогда наилучшее вогнутое продолжение F определяется следующим образом:

$$F(x) = \min_{y \in X}\{2y^T Q x - y^T Q y\} + c^T x. \tag{2}$$

Следовательно, вычисление значения функции F в точке эквивалентно решению задачи выпуклого программирования (2).

Идейная основа построения глубоких отсечений

В данном разделе исследуется возможность построения глубоких отсечений в булевом программировании, основанных на использовании вогнутых продолжений.

Пусть заданы многогранное множество

$$X = \{x : Ax \leqslant b, 0 \leqslant x_j \leqslant 1, j = \overline{1,n}\} \subset \mathbb{R}^n,$$

где A — $(m \times n)$-матрица, $b \in \mathbb{R}^m$, и невырожденная вершина x^0 множества X, не все компоненты которой целые числа. Требуется построить правильное отсечение, т. е. построить такую плоскость $L = \{x \in \mathbb{R}^n : \alpha^T x = \beta\}$, что открытое полупространство $H^{>} = \{x \in \mathbb{R}^n : \alpha^T x > \beta\}$ содержит точку x^0, а полупространство $H^{\leqslant} = \{x \in \mathbb{R}^n : \alpha^T x \leqslant \beta\}$ содержит все целочисленные точки множества X.

Как указано в [1], многие правильные отсечения можно построить, используя точки пересечения ребер многогранного конуса с вершиной x^0, образованного активными в точке x^0 ограничениями, с границей выпуклого множества $D \supset X$, такого, что целочисленные точки множества X принадлежат границе D. В работах Гомори, Балаша и Гловера в качестве D предлагается использовать полосу, двойственный гиперкуб и шар

$$S = \left\{x : \sum_{i=1}^{n} x_i(1 - x_i) \geqslant 0\right\}. \tag{3}$$

Заметим, что при построении этих отсечений используется не вся информация о множестве X: на построение отсечений никак не влияют *неактивные* в точке x^0 ограничения. В докладе предлагается метод построения правильных отсечений, использующий информацию о всех ограничениях множества X.

Обозначим функцию, с помощью которой определяется множество S в (3), через $s(x)$

$$s(x) = \sum_{i=1}^{n} x_i(1 - x_i).$$

Нетрудно видеть, что $s(x)$ — вогнутая функция. Определим вогнутое продолжение

$$F_s(x) = \min_{y \in X} \{ f(y) + \nabla f(y)^T (x - y) \},$$

$$F_s(x) = s(x), \quad x \in X,$$
$$F_s(x) \geqslant s(x), \quad x \notin X,$$

и множество

$$G = \{ x \in \mathbb{R}^n : F_s(x) \geqslant 0 \}.$$

Нетрудно показать, что целочисленные точки множества X являются граничными точками множества G. Следовательно, отсечения, построенные с помощью множества G, будут правильными.

В данном случае

$$F_s(x) = \sum_{i=1}^{n} x_i + \min_{y \in X} \left\{ \sum_{i=1}^{n} (y_i^2 - 2y_i x_i) \right\}. \tag{4}$$

Задача (4) есть задача минимизации выпуклой сепарабельной квадратичной функции на выпуклом многогранном множестве. В силу (1) отсечения, построенные с помощью множества Z (будем называть их отсечениями по вогнутому продолжению), будут не хуже (на практике, как правило, лучше) отсечений, построенных с помощью шара S. Плата за улучшение глубины отсечений состоит в решении задач (4). Что касается отсечений по полосе и двойственному гиперкубу, то сравнение с ними отсечений по вогнутому продолжению зависит от конкретной геометрической ситуации. В докладе приводятся примеры, иллюстрирующие эффективность различных отсечений, и приводится предварительный численный эксперимент.

Работа выполнена при поддержке РФФИ, проект 09-01-00307-а.

Список литературы

[1] *Булатов В. П.* Методы погружения в задачах оптимизации. — Новосибирск: Наука, 1977.

[2] *Bulatov V. P., Khamisov O. V.* The Cutting Method in E^{n+1} Through Concave Extension for Solving Global Extremum Problem // 21 JAHRESTAGUNG «Mathematische Optimerung». — Berlin: Humboldt University, 1989. — P. 16–19.

[3] *Хамисов О. В.* О построении новых отсечений в целочисленном программировании // Тр. XIII Байкальской международной школы-семинара «Методы оптимизации и их приложения». Т. 1. — Иркутск: ИСЭМ СО РАН, 2005. — С. 607–612.

Решение проблемы обобщенной минимизации для многоленточных автоматов с одной существенной лентой

В. Е. Хачатрян, Я. Г. Великая

khachatryan@bsu.edu.ru, velikaya@bsu.edu.ru

Белгородский государственный университет

Определяется множество многоленточных автоматов с одной существенной лентой. Приводится процедура, использующая полную систему фрагментных эквивалентных преобразований, получения по любому автомату с одной существенной лентой всех минимальных, по числу состояний ему эквивалентных.

Многоленточные детерминированные автоматы [1] являются обобщением обычных конечных детерминированных автоматов. Не решена задача нахождения по заданному автомату ему эквивалентного, содержащего минимальное число состояний. Способ нахождения минимального автомата, используемый для конечных детерминированных автоматов, не приемлем, поскольку для конечных детерминированных автоматов минимальный единственен и совпадает с тупиковым, т. е. автоматом, не содержащим эквивалентных состояний. Для многоленточных автоматов существуют классы эквивалентности, содержащие бесконечное число тупиковых [2]. Более того, минимальный автомат может быть не единственным [3], поэтому проблема отыскания по заданному автомату всех ему эквивалентных минимальных называется обобщенной проблемой минимизации. В [4] обсуждается поиск одного минимального автомата и в случае, когда рассматриваемое множество автоматов отлично от множества, исследуемого в данной работе.

В [5] рассматривается случай двухленточных автоматов. В данной работе дается обобщение полученных в [5] результатов в случае, когда число лент $n > 2$.

Рассматриваются n-ленточные автоматы над алфавитом $P = \{p, q_1, q_2, \ldots, q_{n-1}\}$ и $Q = \{0, 1\}$, в виде диаграммы, где метками алфавита P помечены вершины диаграмм, а метками Q — дуги, из них выходящие. Путь, ведущий из входа в выход диаграммы, описывается его историей, т. е. последовательностью меток вершин (ленты) и меток дуг (символы на лентах), из них выходящих. История пути разбивается на подыстории, каждая из которых состоит из одних и тех же меткок вершин, с последующей меткой дуг. Автоматы эквивалентны, когда для любого пути одного из них в другом существует путь с тем же множеством подысторий и наоборот.

Автомат назовем автоматом с одной существенной лентой, если пути, ведущие в нем из входа в выход, могут содержать подыстории, для которых вершины с меткой q_i, $i = 1, \ldots, n-1$, имеют выходящие дуги, помеченные одним и тем же символом алфавита Q.

Процедура получения по данному автомату с одной существенной лентой всех минимальных ему эквивалентных основана на использовании полной системы эквивалентных фрагментарных преобразований.

Преобразование П1 [6] позволяет склеивать или расклеивать вершины автомата, а преобразование П2 [6] позволяет «переносить через фрагмент» множество q_i, $i = 1, \ldots, n-1$, вершин, если такой меткой помечены все входы или выходы фрагмента. Обозначим T систему преобразований, состоящую из преобразований П1 и П2.

Теорема 1. *T является полной системой фрагментных эквивалентных преобразований для n-ленточных автоматов, $n \geqslant 2$, с одной существенной лентой.*

Рассмотрим процедуру преобразования автомата с одной существенной лентой, состоящую из 3 этапов.

Этап 1. Пусть D — исходный автомат. С помощью преобразования П2 все $q_i, i = 1, \ldots, n-1$, состояния в автомате D переносим как можно «ближе» ко входу. На линейных участках добьемся условия: если $i < j$, то состояния q_i должны предшествовать состоянию q_j. С помощью преобразования П1 склеим все эквивалентные состояния. Полученный автомат обозначим через D'.

Этап 2. Применяя преобразования П2, уменьшим число состояний q_i, $i = 1, \ldots, n$, в автомате D' до тех пор, пока это возможно. Полученный автомат обозначим через D''.

Этап 3. В автомате D'' осуществим расклейку тех p-состояний, которые позволяют за счёт последующего применения преобразования П2 уменьшить число q_i-состояний таким образом, что уменьшится общее число всех состояний. Полученный автомат обозначим через D^*.

Теорема 2. *Автомат D^* является минимальным в классе эквивалентности D.*

Доказательство. Доказательство теоремы в общих чертах повторяет доказательство, предложенное в [5] для случая $n = 2$. ∎

Обозначим через K преобразования типа П2, не меняющие общее число состояний многоленточного автомата.

Теорема 3. *Если D^* — минимальный автомат с одной существенной лентой, то, применяя к нему преобразования вида K, получим все минимальные в классе эквивалентности D^*.*

Список литературы

[1] *Rabin M. O., Scott D. P.* Finite Automata and Their Decision Problems // IBM J. of Research and Development. — 1959. — V. 3, № 2. — P. 114–125.

[2] *Хачатрян В. Е.* Трансформационный метод в моделях вычислений // Вестник компьютерных и информационных технологий. — 2008. — № 4. — С. 52–55.

[3] *Подловченко Р. И., Хачатрян В. Е.* Минимальность и тупиковость многоленточных автоматов // Вестник компьютерных и информационных технологий. — 2008. — № 2. — С. 92–120.

[4] *Tamm H.* On Minimality and Size Reduction of One-Tape and Multitape Finite Automata. Helsinki: University of Helsinki, 2004.

[5] *Подловченко Р. И., Хачатрян В. Е.* Полное решение проблемы минимизации для одного множества бинарных двухленточных автоматов // Дискретная математика. — 2010. — Т. 22, № 3. — С. 146–159.

О сведении задачи распознавания выполнимости формул логики линейного времени к распознаванию выполнимости формул логики высказываний

Р. В. Хелемендик

romash@keldysh.ru

Институт прикладной математики им. М. В. Келдыша РАН, Москва

Логика линейного времени (ллв) является расширением логики высказываний (лв), в которой наряду с классическими связками добавлены следующие временны́е: $\bigcirc$ (в следующий момент), $\Box$ (всегда), $\Diamond$ (в некоторый момент), U (до тех пор, пока). Основанный на методе семантических таблиц для модальных логик (см. [1]) алгоритм распознавания выполнимости формул ллв изложен в работе [2]. Следуя работе [3], мы будем называть моделью для формулы конечную или бесконечную последовательность вершин с выделенной начальной вершиной, в которой эта формула истинна. Формула называется выполнимой, если она истинна в некоторой модели.

В настоящей работе проведено сведéние задачи распознавания выполнимости формул ллв к распознаванию выполнимости формул лв. Для произвольной формулы ллв Θ с учетом её особенностей строится формула лв Θ^{pq}, которая выполнима тогда и только тогда, когда выполнима Θ. В случае выполнимости Θ^{pq} по этой формуле получается модель для исходной формулы Θ. Данное сведéние применимо, в частности, и для ллв с обычной семантикой (см. [2, 4]), в которой моделями для формул считаются только бесконечные последовательности вершин, и оно даёт возможность использовать многочисленные SAT-решатели в лв для распознавания выполнимости формул ллв.

1. Основные определения

Каждая пропозициональная переменная есть *формула*. Если φ и ψ *формулы*, то θ, являющаяся одним из 9 выражений: $\top$, $\neg\varphi$, $(\varphi \vee \psi)$, $(\varphi \wedge \psi)$, $(\varphi \to \psi)$, $\bigcirc\varphi$, $\Box\varphi$, $\Diamond\varphi$, $(\varphi\ U\ \psi)$, тоже называется *формулой*. Других *формул* нет.

Моделью будем называть пару $M = \langle N, L \rangle$, где N – связный ориентированный граф с выделенной начальной вершиной u_0, каждая вершина которого имеет не более одного сына[1] , а L – функция означивания, сопоставляющая каждой вершине множество пропозициональных переменных. *Полным путём* в графе называется бесконечный путь или цепь, последняя вершина которой не имеет сыновей.

Истинность формулы θ в вершине u_i модели M (обозначим это $M, u_i \models \theta$) определяется индуктивно.

— Если $\theta = \top$, то $M, u_i \models \theta$.
— Если $\theta = p$, то $M, u_i \models \theta \rightleftharpoons p \in L(u_i)$.
— Если $\theta = \neg\varphi$, то $M, u_i \models \theta \rightleftharpoons M, u_i \not\models \varphi$ (неверно $M, u_i \models \varphi$).
— Если $\theta = (\varphi \wedge \psi)$, то $M, u_i \models \theta \rightleftharpoons (M, u_i \models \varphi$ и $M, u_i \models \psi)$.
— Если $\theta = (\varphi \vee \psi)$, то $M, u_i \models \theta \rightleftharpoons (M, u_i \models \varphi$ или $M, u_i \models \psi)$.
— Если $\theta = (\varphi \to \psi)$, то $M, u_i \models \theta \rightleftharpoons (M, u_i \not\models \varphi$ или $M, u_i \models \psi)$.
— Если $\theta = \bigcirc\varphi$, то $M, u_i \models \theta \rightleftharpoons M, u_j \models \varphi$ для сына u_j вершины u_i либо вершина u_i не имеет сына.
— Если $\theta = \Box\varphi$, то $M, u_i \models \theta \rightleftharpoons$ для полного пути с началом в вершине u_i в каждой его вершине u_j верно $M, u_j \models \varphi$.
— Если $\theta = \Diamond\varphi$, то $M, u_i \models \theta \rightleftharpoons$ для полного пути с началом в вершине u_i существует вершина u_j, для которой верно $M, u_j \models \varphi$.
— Если $\theta = (\varphi\ U\ \psi)$, то $M, u_i \models \theta \rightleftharpoons$ для полного пути с началом в вершине u_i существует вершина u_j, для которой верно $M, u_j \models \psi$, а в каждой вершине u_k этого пути, предшествующей u_j, верно $M, u_j \models \varphi$.

Формула θ *истинна в модели* M, если она истинна в выделенной вершине u_0 этой модели. Формула θ *выполнима*, если она истинна в некоторой модели. Формула θ *общезначима*, если она истинна в каждой модели. В дальнейшем положим $\bot \rightleftharpoons \neg\top$.

2. Общее описание сведе́ния

Построение сведе́ния задачи распознавания выполнимости формул логики ветвящегося времени (лвв) к распознаванию выполнимости формул лв описано в работе [5]. Однако уже на синтаксическом уровне ллв (соответствующий ей англоязычный термин – Linear Time Logic, *Ltl*) не является частным случаем лвв (соответствующий ей англоязычный термин – Computational Tree Logic, Ctl) [4]. Тем не менее, построение по формуле ллв Θ формулы лв Θ^{pq}, выполнимой

[1] Т. е. N — конечная или бесконечная последовательность вершин.

тогда и только тогда, когда выполнима исходная формула Θ, возможно, и оно проводится по общей схеме, изложенной в работе [5], с учётом синтаксических и семантических отличий ллв и лвв.

В случае выполнимости формулы Θ^{pq} модель $M = \langle N, L\rangle$ для формулы Θ строится по модели для формулы Θ^{pq} следующим образом. Последовательность N вершин в модели определяется интерпретацией вспомогательных пропозициональных переменных q_j, используемых, в частности, для кодирования специального вида подформул φ_j исходной формулы Θ. А функция означивания L определяется по специальной интерпретации элементов множеств $\{p_i\}$ – пропозициональных переменных формулы Θ^{pq}, соответствующих конкретным пропозициональным переменным p_i исходной формулы Θ.

3. Корректность и полнота сведе́ния

Имеют место следующие теоремы.

Теорема 1. *Если формула Θ^{pq} выполнима, то выполнима и формула Θ. Формула Θ истинна в модели M, построенной по модели для формулы Θ^{pq}.*

Доказательство. Проводится по индукции проверкой истинности формулы Θ в модели M, построенной по модели для формулы Θ^{pq}. ∎

Теорема 2. *Если формула Θ выполнима, то выполнима и формула Θ^{pq}.*

Доказательство. Утверждение теоремы равносильно тому, что из невыполнимости формулы Θ^{pq} следует невыполнимость формулы Θ, т. е. общезначимость формулы $\neg\Theta$. Для доказательства последнего утверждения используется модификация алгоритма построения выводов общезначимых формул из аксиом (см. [3, 6, 7]). ∎

В работе [3] введена формула $\Psi = \Box\neg\bigcirc\bot$ и установлена равносильность выполнимости формулы Θ ллв с рассматриваемой нами обобщённой семантикой выполнимости формулы $(\Theta\wedge\Psi)$ ллв с обычной семантикой. Таким образом, настоящее сведе́ние задачи распознавания выполнимости формул ллв к распознаванию выполнимости формул лв полностью применимо и для ллв с обычной семантикой.

Работа выполнена при финансовой поддержке РФФИ и программы фундаментальных исследований ОМН РАН «Алгебраические и комбинаторные методы математической кибернетики и информационные системы нового поколения».

Список литературы

[1] *Chagrov A. V., Zakharyaschev M. V.* Modal Logic. — Oxford: Clarendon Press, 1997. — V. 35 of Oxford Logic Guides.

[2] *Wolper P.* The Tableau Method for Temporal Logic: An Overview // Logique et Analyse. — 1985. — V. 28, June–Sept. 85. — P. 119–136.

[3] *Хелемендик Р. В.* Об одном обобщении логики линейного времени // Материалы X Международного семинара «Дискретная математика и ее приложения» (Москва, МГУ, 1–6 февраля 2010 г.). — М.: Изд-во механико-математического факультета МГУ, 2010. — С. 209–212.

[4] *Emerson E. A., Halpern J. Y.* ‘Sometimes’ and ‘Not Never’ Revisited: On Branching versus Linear Time Temporal Logic // JACM. — 1986. — V. 33, № 1. — P. 151–178.

[5] *Хелемендик Р. В.* О сведении задачи распознавания выполнимости формул логики ветвящегося времени к распознаванию выполнимости формул логики высказываний // Сб. тр. XVII Международной школы-семинара «Синтез и сложность управляющих систем» им. акад. О. Б. Лупанова (Новосибирск, 27 октября – 1 ноября 2008 г.). — М.: Изд-во механико-математического факультета МГУ, 2008. — С. 177–180.

[6] *Хелемендик Р. В.* Алгоритм распознавания формул логики ветвящегося времени и эффективный алгоритм построения выводов общезначимых формул из аксиом // Математические вопросы кибернетики. Вып. 15. — М.: Физматлит, 2006. — С. 217–266.

[7] *Хелемендик Р. В.* Об эффективном алгоритме построения выводов формул логики ветвящегося времени из аксиом с использованием логики высказываний // Дискретные модели в теории управляющих систем: VIII Международная конференция, Москва, 6–9 апреля, 2009 г.: Труды. — М.: Издательский отдел факультета ВМиК им. М. В. Ломоносова; МАКС Пресс, 2009. — С. 325–329.

Свойства непрерывных детерминированных функций с задержкой

А. Н. Черепов

ANCherepov@mail.ru

Смоленский филиал Московского энергетического института (технический университет)

Задача об оценке сложности реализации непрерывных функций и классов таких функций дискретными функциями, близкими к автоматным, была поставлена А. Н. Колмогоровым [1]. В работе [2] в качестве множества приближающих функций рассматривался класс дискретных детерминированных функций с задержкой и соответствующих им непрерывных функций с задержкой. Этот класс появляется при естественном обобщении понятия детерминированной функции. Если с помощью классических детерминированных функций нельзя приблизить с любой точностью произвольную непрерывную функцию, то функции с задержкой позволяют это сделать. В предлагаемой работе приводятся некоторые свойства детерминированных функций с задержкой.

Рассмотрим множество всех бесконечных двоичных последовательностей E. Множество всех функций вида $f : E^n \to E$ обозначим P. Предположим, что $a_1, a_2, \ldots, a_n$ — последовательности из E, а $\tilde{a} = (a_1, a_2, \ldots, a_n)$ — набор таких последовательностей. Пусть $a_1|k, a_2|k, \ldots, a_n|k$ — первые k членов последовательностей $a_1, a_2, \ldots, a_n$ соответственно, тогда $\tilde{a}|k = (a_1|k, a_2|k, \ldots, a_n|k)$.

Назовем функцию f детерминированной, если для всех $k = 1, 2, \ldots$ выполнено:

$$\forall \tilde{a}, \tilde{b} : \tilde{a}|k = \tilde{b}|k \Rightarrow f(\tilde{a})|k = f(\tilde{b})|k.$$

Класс всех детерминированных функций обозначим P_d, все остальные функции из P будем называть недетерминированными. Дальнейшие определения взяты из работы [2].

Определение 1. Говорим, что функция f является детерминированной функцией с задержкой τ, где τ — произвольное неотрицательное целое число, если для любого $k = 1, 2, 3, \ldots$ и любых $\tilde{a}, \tilde{b}$ выполнено:

$$\tilde{a}|k + \tau = \tilde{b}|k + \tau \ \Rightarrow\ f(\tilde{a})|k = f(\tilde{b})|k.$$

Множество всех функций с задержкой τ обозначим P_d^τ. Заметим, что $P_d^0 = P_d$. В класс P_d^∞ включим все функции множеств P_d^τ при $\tau = 0, 1, \ldots$

Утверждение 1 [2]. *Класс P_d^∞ замкнут.*

Из определения следует, что функцию с задержкой τ от n аргументов можно трактовать так: берется дискретное детерминированное устройство с n входами, преобразующее бесконечные входные последовательности из нулей и единиц в такую же выходную последовательность, и рассматривается выход этого устройства не с первого момента времени, а с момента времени $\tau + 1$. Осуществляемое преобразование и считается детерминированной функцией с задержкой τ. Таким образом, понятие функции с задержкой является естественным обобщением детерминированных функций. Аналогичным образом можно ввести понятие конечного автомата с задержкой.

Перейдем к вопросу о возможности реализации непрерывных функций, осуществляющих отображение точек n-мерного единичного куба на отрезок $[0, 1]$. Сопоставим каждой двоичной последовательности $a(1)a(2)\ldots a(i)\ldots$ некоторое число отрезка $[0, 1]$, равное $0{,}a(1)a(2)\ldots a(i)\ldots$. Из двух возможных представлений $0{,}a(1)a(2)\ldots a(i)100\ldots = 0{,}a(1)a(2)\ldots a(i)0111\ldots$ выберем первое. Для числа 1 берется представление $1 = 0{,}1111\ldots$.

Определение 2. Функцию $\mu : [0, 1] \to E$ определим следующим образом: для любого числа $a \in [0, 1]$, где $a = 0{,}a_1a_2a_3\ldots$, значение функции равно $\mu(a) = a_1a_2a_3\ldots$.

Определение 3. Функцию $\mu^{-1} : E \to [0, 1]$ определим так, что для любой бесконечной двоичной последовательности $a = a_1a_2a_3\ldots$

1) если существует натуральное k, такое, что $a = a_1a_2\ldots a_k01111\ldots$, т. е. начиная с $(k+2)$-го члена a представляет собой бесконечную последовательность из единиц, то $\mu^{-1}(a) = 0{,}a_1a_2\ldots a_k1000\ldots$;
2) для остальных последовательностей $\mu^{-1}(a) = 0{,}a_1a_2a_3\ldots$.

Определение 4. Любой дискретной функции $f : E^n \to E$ однозначно сопоставим действительную функцию $g : [0, 1]^n \to [0, 1]$ так, что для любой n-мерной точки

$$(b_1, b_2, \ldots, b_n) \in [0, 1]^n : g(b_1, b_2, \ldots, b_n) = \mu^{-1} f(\mu(b_1), \mu(b_2), \ldots, \mu(b_n)).$$

Действительную функцию g будем называть соответствующей дискретной функции f.

Множества действительных функций, соответствующих множествам дискретных функций $P, P_d, P_d^\tau, P_d^\infty$, будем обозначать соответственно $D, \mu^{-1}(P_d)$, $\mu^{-1}(P_d^\tau)$, $\mu^{-1}(P_d^\infty)$. Заметим, что при таком сопоставлении теряется информация о преобразовании последовательностей вида $a_1 a_2 \ldots a_i 0111\ldots$. Кроме того, может возникать некоторое несоответствие выхода дискретной функции значению функции действительных аргументов, в том случае когда выходной последовательности соответствует двоично-рациональное число.

Утверждение 2 [2]. *Класс действительных функций D_d^∞ замкнут относительно суперпозиции.*

Теорема 3 [2]. *Для любого $\varepsilon > 0$ и любой функции $f(\tilde{x}) \in C[0,1]^n$ существует число $\tau \geqslant 0$ и функция $d(\tilde{x}) \in D_d^\tau$, такие, что $d(\tilde{x})$ ε-равна $f(\tilde{x})$.*

Теорема 4 [2]. *Функции одного аргумента классов D_d^τ непрерывны справа во всех точках и непрерывны слева во всех точках, кроме, может быть, двоично-рациональных точек отрезка* $[0,1]$.

Разобьем отрезок $[0,1]$ на полуинтервалы $[k/2^i, k+1/2^i)$ и обозначим это разбиение I_h, где $h = 1/2^i$.

Теорема 5. *Функция f принадлежит множеству D_d^τ тогда и только тогда, когда при любом $i = \tau+1,\ldots$ на каждом интервале $I_h(\tilde{a})$ $(h = 2^{-i})$ выполнено неравенство $f(\tilde{a}) \leqslant f(\tilde{x}) < f(\tilde{a}) + 2^{-(i-\tau)}$.*

Можно указать дополнительные свойства непрерывных функций множеств D_d^τ.

Теорема 6. *Непрерывная функция f принадлежит множеству D_d^τ тогда и только тогда, когда при любом $i = \tau+1,\ldots$, если $|x-y| < 2^{-i}$, то $|f(x)-f(y)| < 2^{-(i-\tau)}$.*

Следствие. *Если непрерывные функции f, g принадлежат множеству D_d^∞, то их сумма и разность тоже принадлежат D_d^∞.*

Так как функции, удовлетворяющие условию Липшица, имеют конечное изменение [3], то справедливо следующее утверждение.

Теорема 7. *Непрерывная функция f, принадлежащая множеству D_d^τ, может быть представлена в виде разности двух возрастающих функций.*

Рассмотрим связь модуля непрерывности произвольной функции с величиной задержки приближающей функции.

Теорема 8. *Пусть* $f(x)$ *при* $\delta = 1/2^k$, $k > 1$, *имеет модуль непрерывности* $\omega(\delta) \leqslant \varepsilon$, *тогда существует функция* $g(x) \in D_d^\tau$, *где* $\tau = \lceil k + \log_2 \varepsilon \rceil + 1$, ε-*приближающая* $f(x)$.

В заключение автор выражает свою благодарность В. А. Буевичу за постановку задачи и полезные советы.

Список литературы

[1] *Офман Ю.* О приближенной реализации непрерывных функций на автоматах // ДАН СССР. — 1963. — Т. 152, № 4. — С. 823–825.

[2] *Черепов А. Н.* Оценки сложности приближения непрерывных функций некоторых классов детерминированными функциями с задержкой // Дискретная математика. — 2010. — Т. 22, № 1. — С. 83–103.

[3] *Натансон И. П.* Теория функций вещественной переменной. —М.: Наука, 1974. — С. 480.

О равномерной поточечной оценке приращения решения управляемого функционально-операторного уравнения

А. В. Чернов

chavnn@mail.ru

Нижегородский государственный университет им. Н. И. Лобачевского

В статье развивается предложенная в [1] методика получения поточечных (по всем независимым переменным) оценок приращения глобального решения управляемых *начально-краевых задач* (НКЗ).

Об оценках решений дифференциальных уравнений

В настоящее время имеется обширная литература, посвященная теории нелинейных дифференциальных уравнений с частными производными (см., например, [2], там же библиографию). Что касается известных автору оценок решений, их естественно разбить на следующие три категории:

1) Оценки *приращения решения* управляемых НКЗ (при изменении входящих в постановку задачи управляющих функций) *по норме* подходящего банахова пространства (см., например, [3–7], там же

дальнейшую библиографию). Оценки указанного типа оказываются востребованными при обосновании сходимости численных методов оптимизации (см., например, [3]), при выводе необходимых условий оптимальности (см., например, [8, 9]), а также при исследовании особых управлений (см., например, [10]).

2) Поточечные (по всем независимым переменным) оценки самих решений как таковых снизу и (или) сверху. Ввиду обширности литературы по этой тематике приведем лишь имена некоторых авторов: *гиперболические уравнения*: I. Kubiaczyk, M. Cichon, M. Dawidowski, A. Belarbi, M. Benchohra, Н. И. Погодаев; *параболические и обобщенно-параболические*: А. К. Гущин, А. В. Лежнев, В. И. Ушаков, Ф. Х. Мукминов, Л. М. Кожевникова, Г. А. Рудых, А. В. Синицын, Л. И. Камынин, В. И. Налимов; *эллиптические уравнения*: М. А. Красносельский, Е. Б. Дынкин, С. Е. Кузнецов, J.-F. Le Gall, M. Marcus, L. Véron. В работах перечисленных авторов соответствующие оценки носили зачастую вспомогательный характер и использовались при исследовании таких вопросов, как асимптотическое поведение решений, устойчивость (в том или ином смысле) и т. п.

3) Прочие оценки: поточечные по части переменных; оценки решений или их следов по норме; оценки различных интегральных выражений, содержащих решения, и т. д. и т. п. Поскольку оценки такого типа не имеют непосредственного отношения к данной статье, приведем лишь (разумеется, далеко не полный) список авторов: В. И. Юдович, Ю. А. Дубинский, С. И. Похожаев, И. В. Скрыпник, М. И. Вишик, А. Е. Мамонтов, Ф. Бенилан, В. Л. Камынин, N. Trudinger, J. G. Morris, R. Adams, A. Kufner, G. Talenti и т. д.

В данной статье речь идет об оценках, сочетающих в себе свойства 1) и 2), и более тонких по сравнению с 1). Основная идея состоит в сведении управляемой НКЗ к функционально-операторному уравнению вида

$$y(t) = w(t) + A\left[g\Big(\cdot, y(\cdot), u(\cdot)\Big)\right](t), \quad t \in \Pi, \quad y(\cdot) \in L_q^l(\Pi), \qquad (1)$$

и последующем применении соответствующих абстрактных результатов (см. далее) для этого уравнения. Здесь $\Pi \subset \mathbb{R}^n$ — измеримое по Лебегу ограниченное множество, $w(\cdot) \in L_q^l(\Pi)$; $u(\cdot) \in \mathcal{D}$ — управляющая функция (управление), $\mathcal{D} \subset L_r^s(\Pi)$ — ограниченное множество, $A : L_p^m(\Pi) \to L_q^l(\Pi)$ — ЛОО, числа $n, m, l, s \in \mathbb{N}$, $1 \leqslant p < q < +\infty$ заданы; функция $g(\cdot,\cdot,\cdot) :\ \Pi \times \mathbb{R}^l \times \mathbb{R}^s \to \mathbb{R}^m$ — заданная функция, удовлетворяющая условиям типа Каратеодори:

а) $g(t, x, w)$ непрерывно дифференцируема по $x \in \mathbb{R}^l$ для почти всех (п.в.) $t \in \Pi$, при каждом $w \in \mathbb{R}^s$ и вместе с производной $g'_x(t, x, w) : \Pi \times \mathbb{R}^l \times \mathbb{R}^s \to \mathbb{R}^{m\times l}$ измерима по $t \in \Pi$ и непрерывна по $\{x, w\} \in \mathbb{R}^l \times \mathbb{R}^s$ для п.в. $t \in \Pi$;

б) для каждого $u(\cdot) \in L_r^s(\Pi)$ формула $\mathbf{G}_u[y] = g\Big(\cdot, y(\cdot), u(\cdot)\Big)$ определяет оператор $\mathbf{G}_u : L_q^l(\Pi) \to L_p^m(\Pi)$;

в) для каждого $u \in L_r^s(\Pi)$ формула $\mathcal{G}_u[y] = g'_x\Big(\cdot, y(\cdot), u(\cdot)\Big)$ определяет оператор $\mathcal{G}_u : L_q^l(\Pi) \to L_\sigma^{m\times l}(\Pi)$, $\frac{1}{q} + \frac{1}{\sigma} = \frac{1}{p}$.

Что касается различных примеров сведения управляемых НКЗ к уравнению (1), см., например, [1, 6, 11]. Далее относительно ЛОО A будем предполагать, что он обладает положительной мажорантой $B : L_p(\Pi) \to L_q(\Pi)$, у которой для любого $\delta > 0$ существует вольтеррова δ-цепочка [3, 4, 6, 12].

Формулировка основного результата

Пусть $V(\cdot) \in L_r(\Pi)$, $v(\cdot) \in L_r^s(\Pi)$ — некоторые фиксированные функции, причем для $u = v$ существует решение $y = y_v$ уравнения (1); $\mathcal{V}$ — заданное подсемейство семейства функций $\Big\{u(\cdot) \in L_r^s(\Pi) : |u(t)| \leqslant V(t) \text{ для п.в. } t \in \Pi\Big\}$. Пользуясь результатами статьи [1], можно показать, что существуют $a(\cdot) \in L_\sigma^+(\Pi)$ и $b \geqslant 0$, такие, что для любых $\Delta y \in L_q^l(\Pi)$, $u \in \mathcal{V}$ имеем

$$\Big|g'_x(\cdot, y_v + \Delta y, u)\Big| \leqslant a(\cdot) + b\left(|\Delta y|^{q/\sigma} + |u|^{r/\sigma}\right) \leqslant a_0(\cdot) + b\cdot|\Delta y|^{q/\sigma},$$

где $a_0 = a + b|V|^{r/\sigma} \in L_\sigma(\Pi)$. Впрочем, во многих случаях параметры $a_0(\cdot)$ и b можно указать и непосредственно. Положим для краткости

$$\Delta_u g(y_v) = g\Big(\cdot, y_v(\cdot), u(\cdot)\Big) - g\Big(\cdot, y_v(\cdot), v(\cdot)\Big), \quad u(\cdot) \in L_r^s(\Pi),$$

и выберем функцию $\overline{\varphi} \in L_q^+(\Pi)$ так, чтобы $\sup\limits_{u\in\mathcal{V}} B\Big[\big|\Delta_u g(y_v)\big|\Big] \leqslant \overline{\varphi}$. В качестве таковой годится, например,

$$\overline{\varphi}(t) = B\Big[\max_{w\in\mathbb{R}^s:|w|\leqslant V(\cdot)}\Big|g\big(\cdot, y_v(\cdot), w\big) - g\big(\cdot, y_v(\cdot), v(\cdot)\big)\Big|\Big](t).$$

Следующее уравнение естественно назвать *мажорантным* для (1):

$$x = bB\Big[x^{q/p}\Big] + B\Big[a_0(\cdot)x\Big] + \overline{\varphi}, \quad x \in L_q^+. \tag{2}$$

Формулируемое далее утверждение является существенно более сильным по сравнению с теоремой 1.1 из [1].

Теорема 1. *Пусть мажорантное уравнение* (2) *имеет решение* $\overline{x}$. *Тогда для любого решения уравнения* (1) $y = y_u$, *отвечающего некоторому* $u \in \mathcal{V}$, *имеет место поточечная оценка:* $|\Delta y(t)| = |y_u(t) - y_v(t)| \leqslant \overline{x}(t)$ *для п.в.* $t \in \Pi$.

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (проект НК-13П(9)) и АЦВП «Развитие научного потенциала высшей школы (2009–2010 гг.)» Минобрнауки РФ (№ 2.1.1/3927).

Список литературы

[1] *Чернов А. В.* О поточечной оценке разности решений управляемого функционально-операторного уравнения в лебеговых пространствах // Матем. заметки. — 2010. — Т. 88, № 2. — С. 288–302.

[2] *Свешников А. Г., Альшин А. Б., Корпусов М. О.* Нелинейный функциональный анализ и его приложения к уравнениям в частных производных. — М.: Научный мир, 2008.

[3] *Сумин В. И.* Об обосновании градиентных методов для распределенных задач оптимального управления // Ж. вычисл. матем. и матем. физ. — 1990. — Т. 30, № 1. — С. 3–21.

[4] *Сумин В. И.* Функциональные вольтерровы уравнения в теории оптимального управления распределенными системами. Часть I. — Н. Новгород: ННГУ, 1992.

[5] *Аргучинцев А. В., Крутикова О. А.* Оптимизация полулинейных гиперболических систем с гладким граничным управлением // Изв. вузов. Математика. — 2001. — № 2. — С. 3–12.

[6] *Сумин В. И., Чернов А. В.* О достаточных условиях устойчивости существования глобальных решений вольтерровых операторных уравнений // Вестник Нижегородского университета им. Н.И. Лобачевского. Сер. Математическое моделирование и оптимальное управление. — 2003. — Т. 26, № 1. — С. 39–49.

[7] *Злотник А. А.* Слабые решения уравнений движения вязкой сжимаемой реагирующей бинарной смеси: единственность и непрерывная по Липшицу зависимость от данных // Матем. заметки. — 2004. — Т. 75, № 2. — С. 307–311.

[8] *Морозов С. Ф., Сумин В. И.* Оптимизация нелинейных процессов переноса // Докл. АН СССР. — 1979. — Т. 247, № 4. — С. 794–798.

[9] *Плотников В. И., Сумин В. И.* Оптимизация распределенных систем в лебеговом пространстве // Сиб. матем. журн. — 1981. — Т. 22, № 6. — С. 142–161.

[10] *Сумин В. И.* Сильное вырождение особых управлений в распределенных задачах оптимизации // Докл. АН СССР. — 1991. — Т. 320, № 2. — С. 295–299.

[11] *Чернов А. В.* Об одном мажорантном признаке тотального сохранения глобальной разрешимости управляемого функционально-операторного уравнения // Изв. вузов. Математика. — 2011. — № 3. — С. 95–107.

[12] *Сумин В. И., Чернов А. В.* Операторы в пространствах измеримых функций: вольтерровость и квазинильпотентность // Дифференц. уравнения. — 1998. — Т. 34, № 10. — С. 1402–1411.

О сложности распознавания предикатов в трехзначной логике

Б. В. Чокаев

g110@yandex.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Пусть $E_3 = \{0, 1, 2\}$. Отображение $R(x, y, z) : E_3^3 \to \{$истина, ложь$\}$ называется 3-местным отношением (предикатом) на E_3. Определим $U(R)$ как класс всех функций $f(x_1, \ldots, x_n)$ от любого числа n переменных, отображающих E_3^n в E_3, и для любых наборов

$\widetilde{\alpha} = (\alpha_1, \ldots, \alpha_n)$, $\widetilde{\beta} = (\beta_1, \ldots, \beta_n)$, $\widetilde{\gamma} = (\gamma_1, \ldots, \gamma_n)$ из E_3^n, удовлетворяющих импликации

$$R^n(\widetilde{\alpha}, \widetilde{\beta}, \widetilde{\gamma}) \Rightarrow R(f(\widetilde{\alpha}), f(\widetilde{\beta}), f(\widetilde{\gamma})),$$

где отношение R^n определяется формулой

$$R^n(\widetilde{\alpha}, \widetilde{\beta}, \widetilde{\gamma}) \Leftrightarrow \forall j R(\alpha_j, \beta_j, \gamma_j).$$

Рассмотрим следующую задачу. Пусть для заданной функции $f(x_1, \ldots, x_n)$, отображающей E_3^n в E_3, требуется выяснить, принадлежит ли f классу $U(R)$. Нас будет интересовать сложность алгоритмов для этой задачи для произвольного 3-местного предиката R. Будем считать, что функция задана вектором её значений при стандартном (лексикографическом) упорядочении наборов переменных, т. е. длина входа равна $N = 3^n$. В качестве алгоритмов будем рассматривать неветвящиеся битовые вычисления.

Естественный алгоритм для решения указанной задачи, опирающийся просто на определение класса $U(R)$, требует просмотра всех выборок по 3 из N наборов, т. е. имеет сложность по порядку не менее N^3. Различные варианты метода полилинейных форм [1] позволяют существенно понизить эту оценку. Пусть Q_0 — множество неотрицательных рациональных чисел. Пусть F — поле чисел вида $p + q\lambda + r\lambda^2$, где $p, q, r \in Q$, λ — корень неприводимого над полем рациональных чисел кубического многочлена с рациональными коэффициентами. Для произвольного 3-местного предиката R на E_3 рассмотрим 3-линейную форму

$$\psi = \sum_{i=0}^{2} \sum_{j=0}^{2} \sum_{k=0}^{2} t(i, j, k) x_i y_j z_k$$

с коэффициентами из Q_0, удовлетворяющую следующему условию:

$$t(i, j, k) > 0 \Leftrightarrow R(i, j, k).$$

Теорема 1 [1]. *Пусть 3-линейную форму ψ можно представить в виде*

$$\psi = \sum_{r=1}^{d} (a_{10}^r x_0 + a_{11}^r x_1 + a_{12}^r x_2)(a_{20}^r y_0 + a_{21}^r y_1 + a_{22}^r y_2)(a_{30}^r z_0 + a_{31}^r z_1 + a_{32}^r z_2),$$

где a^r_{ij} — некоторые константы из F. Тогда существует алгоритм, который для любой функции $f(x_1,\dots,x_n)$, отображающей E_3^n в E_3 и заданной вектором её значений, отвечает на вопрос "$f \in U(R)$" с числом битовых операций

$$L_{б}(n) \leqslant L_{ар}(n)\cdot O(n\log n\log\log n),$$

$$L_{ар} = \begin{cases} O(d^n), & \text{если } d>3,\\ O(3^n), & \text{если } d<3,\\ O(3^n n), & \text{если } d=3.\end{cases}$$

Произвольной 3-линейной форме ψ с коэффициентами из Q_0 поставим в соответствие систему из трех матриц K_0, K_1, K_2 размера 3×3, с элементами $a_{ij0}=t(i,j,0)$, $a_{ij1}=t(i,j,1)$, $a_{ij2}=t(i,j,2)$, $0\leqslant i,j\leqslant 2$ соответственно. Рангом этой системы матриц назовем минимальное число матриц ранга 1, через которые линейно выражается (над полем F) каждая из матриц K_0, K_1, K_2. Нетрудно доказать следующую лемму о связи ранга системы K_0, K_1, K_2 и разложения 3-линейной формы ψ.

Лемма 2. *Ранг системы матриц K_0, K_1, K_2 меньше либо равен d тогда и только тогда, когда 3-линейную форму ψ можно разложить в сумму d слагаемых.*

Из теоремы 1 и леммы 2 следует, что каждому 3-местному предикату $R(x,y,z)$ можно взаимно-однозначно сопоставить три матрицы размера 3×3 с элементами из $\{0,1\}$. При этом, для уменьшения ранга этой системы матриц, мы можем её модифицировать, заменяя единицы на произвольные положительные рациональные числа.

Теорема 3. *Для любого 3-местного предиката на E_3 существует алгоритм, который для любой функции $f(x_1,\dots,x_n)$ определяет, принадлежит ли f классу $U(R)$ с числом битовых операций*

$$L_b \leqslant O(N^{\log_3 6}\log^2 N).$$

Доказательство. Пусть A, B, C — произвольная система матриц с элементами из $\{0,1\}$. Для доказательства теоремы достаточно построить 6 матриц ранга 1 с элементами из поля F, в линейной оболочке которых лежит система матриц, модифицированная из системы A, B, C. Если у всех трех матриц A, B, C определитель равен нулю, то мы можем построить две матрицы, через которые выражается матрица A, две матрицы, через которые выражается B, и две матрицы, через которые выражается матрица C. Пусть среди матриц A, B, C существует невырожденная матрица. Разобьем доказательство на несколько шагов.

Первый шаг. Пусть $|A| \neq 0$. Рассмотрим этот определитель:

$$|A| = a_{11}a_{22}a_{33}+a_{12}a_{23}a_{31}+a_{13}a_{21}a_{32}-a_{13}a_{22}a_{31}-a_{11}a_{23}a_{32}-a_{12}a_{21}a_{33}.$$

Если в этом разложении присутствует хотя бы одно положительное слагаемое и хотя бы одно отрицательное слагаемое, то, заменив один единичный элемент матрицы A на двойку, сделаем матрицу A вырожденной. Так же поступим с остальными невырожденными матрицами системы.

Второй шаг. Если после первого шага среди матриц A, B, C осталось не более одной невырожденной матрицы, то приступаем к третьему шагу. Пусть $|A| \neq 0$ и $|B| \neq 0$. Рассмотрим определитель матрицы $A + \lambda B$:

$$|A + \lambda B| = |B| \cdot \lambda^3 + \tau(B, A) \cdot \lambda^2 + \tau(A, B) \cdot \lambda + |A|, \tag{1}$$

$$\begin{aligned} &\tau(A, B) = \\ &b_{11}(a_{22}a_{33} - a_{23}a_{32}) + b_{12}(a_{23}a_{31} - a_{21}a_{33}) + b_{13}(a_{21}a_{32} - a_{22}a_{31}) + \\ &+ b_{21}(a_{13}a_{32} - a_{12}a_{33}) + b_{22}(a_{11}a_{33} - a_{13}a_{31}) + b_{23}(a_{12}a_{31} - a_{11}a_{32}) + \\ &+ b_{31}(a_{12}a_{23} - a_{13}a_{22}) + b_{32}(a_{13}a_{21} - a_{11}a_{23}) + b_{33}(a_{11}a_{22} - a_{12}a_{21}). \end{aligned}$$

Нам нужно подобрать рациональное число λ и некоторый элемент a_{ij} матрицы A таким образом, чтобы этот определитель был равен нулю. Пусть A_{ij} — матрица, полученная из A вычёркиванием i-й строки и j-го столбца, $\widehat{A_{ij}}$ — матрица, полученная из A заменой единичного элемента a_{ij} на 0. Тогда определитель $|A + \lambda B|$ можно записать в виде

$$|A + \lambda B| = a_{ij} \cdot |A_{ij} + \lambda B_{ij}| + |\widehat{A_{ij}} + \lambda B|.$$

Из этой формулы видно, что если числа $|A_{ij} + \lambda B_{ij}|$ и $|\widehat{A_{ij}} + \lambda B|$ разных знаков, то выбором элемента a_{ij} можно обнулить определитель $|A + \lambda B|$. Заметим, что $|\widehat{A_{ij}} + \lambda B|$ — кубический многочлен от

λ, а $|A_{ij} + \lambda B_{ij}|$ — квадратный, при условии, что $|B_{ij}| \neq 0$. В этом случае можно найти такое λ, что числа $|A_{ij} + \lambda B_{ij}|$ и $|\widehat{A_{ij}} + \lambda B|$ будут разных знаков. Если для некоторого ненулевого элемента a_{ij} матрицы A матрица B_{ij} невырождена, то сделаем матрицу $A + \lambda B$ вырожденной. Иначе, если для некоторого ненулевого элемента b_{ij} матрицы B матрица A_{ij} невырождена, то сделаем матрицу $B + \lambda A$ вырожденной. Иначе, определитель $|A + \lambda B|$ имеет вид $|A| + |B| \cdot \lambda^3$. Тогда, заменив все ненулевые элементы какой-нибудь строки матрицы A на модуль числа $\frac{|B|}{|A|}$ и положив λ равным 1 или -1, также сделаем матрицу $A + \lambda B$ вырожденной. Итак, мы получили систему матриц $A + \lambda B, B, C$ с таким же рангом, как и у исходной системы, в которой число невырожденных матриц на 1 меньше. Если $|C| \neq 0$, то таким же образом получим систему $A + \lambda B, B, C + \lambda B$, рангом не больше $rg(A + \lambda B) + rgB + rg(C + \lambda B) \leqslant 2 + 3 + 2 = 7$.

Третий шаг. Пусть A, B, C — система матриц, где A — невырожденная матрица с элементами из $\{0, 1\}$, B, C — вырожденные матрицы с рациональными элементами. Применяя выкладки, аналогичные выкладкам из шагов 1 и 2, можно заменить невырожденную матрицу A на линейную комбинацию системы A, B, C над полем F для некоторого λ, которая является вырожденной матрицей. ∎

Работа выполнена при поддержке гранта РФФИ № 09–01–00701, и гранта Президента РФ № МД-757.2011.9.

Список литературы

[1] *Алексеев В. Б.* Логические полукольца и их использование для построения быстрых алгоритмов // Вестн. Моск. ун-та. Сер. 1. Матем. Мех. — 1997. — № 1. — С. 22–29.

Сложность обобщенной распределенной полиномиальной модели на базе системы многочленов над конечными полями

С. В. Шалагин

Sshalagin@mail.ru

Казанский государственный технический университет им. А. Н. Туполева

Определены оценки сложности реализации обобщенной распределенной полиномиальной модели (ОбРПМ), описываемой на основе полиномиальных преобразований над полем Галуа и его расширениями. Предложены модели и методы синтеза устройств, выполняющих как стохастические, так и теоретико-полиномиальные преобразования, на базе указанной ОбРПМ.

Представление распределенных полиномиальных моделей (РПМ) на основе операций в конечных полях привлекательно тем, что открываются возможности определения преобразований стохастических процессов алгебраическими выражениями, допускающими распараллеливание, а также оперативного варьирования свойств моделируемых случайных процессов путем изменения коэффициентов полиномов над конечным полем [1]. РПМ определим как модели, реализующие полиномиальные преобразования на основе распределенных вычислений — способов решения трудоемких вычислительных задач с использованием двух и более вычислителей. Однако существует проблема снижения оценок временной сложности при вычислении многочленов, а также при выполнении отдельных операций над конечными полями. Данное обстоятельство определяет предпосылки для создания обобщенной РПМ (ОбРПМ) как для дискретных стохастических процессов (ДСП) класса марковских и их функций [1–4], так и для теоретико-полиномиальных преобразований (ТПП) — дискретных преобразований, определяемых в различных базисах с применением полиномиальной алгебры [5]. При определении ОбРПМ в общем случае применимы дискретные детерминированные нелинейные функции (ДДНФ) вида $\psi\,(x_1,\ \ldots,\ x_N)\ =\ y$, где $x_1,\ \ldots,\ x_N,\ y$ — n-разрядные двоичные числа. Известен метод реализации ДДНФ $\psi\,(x_1,\ \ldots,\ x_m)$ как суммы элементарных полиномов над $GF(2^n)$ вида

$$f(q_1,\ \dots,\ q_m) = \sum_{i_1=0}^{w} \cdots \sum_{i_m=0}^{w} a_{i_1\dots i_m} q_1^{i_1} \dots q_m^{i_m}, \\ w = 2^n - 1, a_{i_1\dots i_m},\ q_1,\ \dots,\ q_m \in GF(2^n). \tag{1}$$

Примем в качестве меры сложности элементарные схемы – вычислители, реализующие произвольную булеву функцию от двух переменных. Определен частный случай ПФ вида (1) от m переменных над $GF(2)$ — полином Жегалкина. Порядки оценок временной и емкостной сложности ПФ вида (1), представленной системой многочленов над $GF(2)$ согласно [6], составляют $O(\log(m \cdot n))$ и $O\left(m \cdot n^2\right)$ соответственно [4].

Обозначим $U(\psi(x_1,\ \dots,\ x_m))$ — множество всевозможных значений ДДНФ (1), $I(x_j)$ — множество всевозможных значений переменной x_j, $j = \overline{1,\ m}$, на входе (1). Имеет место [4]

Утверждение 1. *ПФ вида* (1) *над* $GF(2^n)$ *принадлежит к классам* $A(p)$, $p = \overline{1,\ n}$, *и* $B(d_1\ \dots\ d_m)$, $d_j \in [1,\ n]$, $j = \overline{1,\ m}$, *если для реализуемой ей ДДНФ* $\psi(x_1,\ \dots,\ x_m)$: $|U(\psi(x_1,\ \dots,\ x_m))| \in \left(2^{k-1},\ 2^k\right]$, $|I(x_1)| \in \left(2^{d_1-1},\ 2^{d_1}\right]$, $|I(x_2)| \in \left(2^{d_2-1},\ 2^{d_2}\right]$, $\dots$, $|I(x_m)| \in \left(2^{d_m-1},\ 2^{d_m}\right]$.

Предложен метод синтеза вычислителей ДДНФ, принадлежащей классам $A(n)$ и $B(d_1, \dots, d_m)$, $d_j = n$, $j = \overline{1,\ m}$, *на основе системы из* l *ПФ*, включающий три этапа [4]:

1) представление множества значений и m множеств значений переменных указанной ДДНФ l k-разрядными векторами, $n = k \cdot l$ [7];
2) определение l ДДНФ, принадлежащих классам $A(k)$ и $B(d_1, \dots, d_{m\cdot l})$, $d_j = k$, $j = \overline{1,\ m \cdot l}$, позволяющих определить отображения φ_i, $i = \overline{1,\ l}$, для $m{\cdot}l$ множеств элементов $GF(2^k)$ в одно множество элементов $GF(2^k)$;
3) вычисление коэффициентов для l ПФ от $m{\cdot}l$ переменных каждая, определенных над $GF(2^k)$, которые соответствуют отображениям φ_i, $i = \overline{1,\ l}$, полученным на этапе 2).

Рассмотрим ОбРПМ генераторов ДСП класса марковских. Общим случаем данных ДСП является стохастическая функция неоднородной цепи Маркова (СФ НЦМ). Справедливо

Утверждение 2. *РПМ, определяющая СФ НЦМ, имеет вид*

$$A_{FNMC} = (X,\, Q,\, Y,\, z_1,\, z_2,\, f_1(x,\, z_1,\, q_t) = q_{t+1},\, f_2(z_2,\, q_{t+1}) = y,\, q_0)\,,$$

где X и Y — входной и выходной алфавиты, Q — множество состояний, z_1 и z_2 — дискретные стохастические величины (ДСВ) с заданным законом распределения, f_1 и f_2 — ПФ вида (1), $q_t \in Q$ — состояние НЦМ в момент времени t.

Предложен метод синтеза генератора СФ НЦМ, для которого вычисление элементов сводится к вычислению значений ПФ вида (1) от m переменных ($m = 2,\, 3$) над $GF(2^n)$, включающий 2 этапа:

1) представление ПФ вида f_1 и f_2 системой многочленов согласно предложенному методу синтеза ДДНФ;
2) генераторы, определяющие ДСВ z_1 и z_2, представимы РПМ согласно методу, определенному в [3].

Оценки временной сложности вычисления СФ НЦМ на основе РПМ будут уменьшены примерно в $\overline{T}$ раз, где $\overline{T} = \max\{t_1 + t_{f_1},\, t\} + t_{f_2}$, где t_1, t_2, t_{f_1} и t_{f_2} — оценки временной сложности вычисления ДСВ z_1 и z_2, а также ПФ вида f_1 и f_2.

На базе ОбРПМ представимы ТПП на примере определенных их подклассов: дискретных преобразований Фурье (ДПФ), Хартли (ДПХ) и цифровой фильтрации (ЦФ). Согласно [7], ДПФ описывается как функция вида $C_F = f_F(I)$, а ДПХ — как $C_H = f_H(I)$. Компоненты I и C_H представимы одним, а C_F — двумя n-разрядными двоичными числами. Алгоритм ЦФ с бесконечным временем отклика (IIR) порядка $p = \max\,(M,\, N)$ представим ПФ от $M + N + 1$ переменных над $GF(2^n)$, а алгоритм ЦФ с конечным временем отклика (FIR) — как ПФ от $M + 1$ переменных над полем $GF(2^n)$, если входной сигнал как для IIR, так и для FIR — n-разрядный двоичный вектор. При этом справедливы соотношения [7]:
для ДПФ:

$$T_{\text{ДПФ}} =]\log_2(Nn)[\,, \quad Q_{\text{ДПФ}} = \sum_{k=1}^{2N} n\,(Nn - 1) = 2Nn\,(Nn - 1)\,;$$

для ДПХ:

$$T_{\text{ДПХ}} =]\log_2(Nn)[\,, \quad Q_{\text{ДПХ}} = \sum_{k=1}^{N} n\,(Nn - 1) = Nn\,(Nn - 1)\,;$$

для IIR:

$$T_{\mathrm{IIR}} = \left]\, \log_2 \left(n\left(M+N+1\right)\right) \right[\,, \quad Q_{\mathrm{IIR}} = n\left(n\left(M+N+1\right)-1\right);$$

для FIR:

$$T_{\mathrm{FIR}} = \left]\, \log_2 \left(n\left(M+1\right)\right) \right[\,, \quad Q_{\mathrm{FIR}} = n\left(n\left(M+1\right)-1\right).$$

Список литературы

[1] *Захаров В. М., Шалагин С. В., Нурутдинов Ш. Р., Соколов С. Ю.* Полиномиальное представление конечноавтоматных случайных последовательностей над полем Галуа // Вестник КГТУ им. А. Н. Туполева. — 2003. — № 2. — С. 24–28.

[2] *Захаров В. М., Шалагин С. В.* Параллельные марковские модели над полем $GF(2^n)$ // Тр. Восьмой Межд. конф. «Высокопроизводительные параллельные вычисления на кластерных системах». — Казань: Издательство КГТУ им. А. Н. Туполева, 2008. — С. 155–160.

[3] *Шалагин С. В.* Полиномиальные модели генераторов дискретных случайных величин // 6-я ежегодная Межд. научно-практич. конф. «Инфокоммуникационные технологии глобального информационного общества»: Сб. трудов. — Казань: Центр оперативной печати, 2008. — С. 159–171.

[4] *Шалагин С. В.* Обобщенная распределенная полиномиальная модель нелинейных преобразований над потоками чисел в конечных полях // Сб. тр. III Всерос. науч. конф. «Информационные технологии в системе экономической безопасности России и ее регионов»: Тезисы докладов. — Казань: ИГМА-пресс, 2010. — С. 186–192.

[5] *Крот А. М.* Дискретные модели динамических систем на основе полиномиальной алгебры. — Минск: Навука і тэхніка, 1990.

[6] *Шалагин С. В.* О представлении нелинейных полиномов над конечным полем распределенной вычислительной системой // Нелинейный мир. — 2009. № 5. — С. 376–379.

[7] *Шалагин С. В.* Представимость моделей анализа и фильтрации цифровой информации над полем Галуа // Дискретные модели в теории управляющих систем: Электронный сб. материалов 8-й Межд. конф. — Москва, 2009. — С. 426–429. `www.dmconf.ru/dm8/proceedings.pdf`

О сравнении базисов при формульном представлении булевых функций

И. К. Шаранхаев

goran5@mail.ru

Бурятский государственный университет, Улан-Удэ

Под базисом понимаем конечную полную систему булевых функций.

Булева функция называется бесповторной в базисе B, если ее можно представить в этом базисе формулой, в которой каждая переменная встречается не более одного раза. В противном случае она называется повторной в B.

Булева функция называется слабоповторной в базисе B, если ее любая остаточная подфункция является бесповторной, а сама она повторна в базисе B.

Под сложностью $L(\Phi)$ формулы Φ понимаем число всех вхождений переменных в Φ. Сложностью $L_B(f)$ булевой функции f в базисе B называется наименьшее значение $L(\Phi)$ при условии, что формула Φ в базисе B представляет функцию f.

При сравнении базисов по сложности формульных представлений булевых функций на множестве всех базисов вводится частичный порядок: $B_1 \leqslant B_2$, если существует число c, такое, что $L_{B_1}(f) \leqslant cL_{B_2}(f)$ для любой булевой функции f, говорят, что B_1 предшествует B_2. Если $B_1 \leqslant B_2$ и $B_2 \leqslant B_1$, то базисы B_1 и B_2 называются эквивалентными. Если $B_1 \leqslant B_2$ и $B_2 \nleqslant B_1$, то пишем $B_1 < B_2$ и говорим, что B_1 строго предшествует B_2. Также говорим, что B_1 непосредственно предшествует B_2, если $B_1 < B_2$ и не существует базиса B, такого, что $B_1 < B < B_2$.

Таким образом, множество базисов разбито на классы эквивалентности. В [1] доказано, что в каждом классе базисов можно указать канонический вид класса, причем если базис B непосредственно предшествует базису $B^{'}$, то канонический вид B содержит на одну функцию больше, чем канонический вид $B^{'}$, а эта функция является слабоповторной в базисе каноническом для $B^{'}$.

О. Б. Лупановым замечено (результат сформулирован в [2]), что базис $B_0 = \{\vee, \cdot, -, 0, 1\}$ является наибольшим элементом при введенном порядке, то есть класс базисов, эквивалентных базису B_0, самый «плохой» по сложности реализаций булевых функций формулами. Эти базисы назовем базисами нулевого яруса. Отметим, что базис B_0 канонический для своего класса.

Базисами k-го яруса ($k > 0$) называются все базисы, непосредственно предшествующие всем базисам $(k-1)$-го яруса.

Введем обозначение $x^\sigma = \begin{cases} x, \text{ если } \sigma = 1 \\ \bar{x}, \text{ если } \sigma = 0 \end{cases}$. Функции f и g называются однотипными, если $f(x_1, \ldots, x_n) = g(x_{i_1}^{\sigma_1}, \ldots, x_{i_n}^{\sigma_n})$, где $(i_1, \ldots, i_n)$ — произвольная перестановка чисел от 1 до n.

Функции f и g называются обобщенно-однотипными, если f однотипна к g или $\bar{g}$. Очевидно, что на множестве булевых функций отношение обобщенной однотипности является отношением эквивалентности.

В работе [3] получено описание всех обобщенных типов функций, слабоповторных в базисе B_0, а как следствие — канонических базисов первого яруса. Заметим, что при добавлении к базису B_0 разных по типу слабоповторных функций получаются базисы из разных классов эквивалентности. Поэтому уже в первом ярусе счетное число различных классов эквивалентности базисов.

Естественно, возник вопрос изучения базисов второго яруса. Эти базисы удалось описать в работах [4–9]. Таким образом, описание базисов второго яруса завершено.

Список литературы

[1] *Черухин Д. Ю.* Алгоритмический критерий сравнения булевых базисов // Математ. вопросы кибернетики. — 1999. — № 8. — С. 77–122.

[2] *Субботовская Б. А.* О сравнении базисов при реализации функций алгебры логики формулами // ДАН СССР. — 1963. — Т. 149, № 4. — С. 784–787.

[3] *Стеценко В. А.* О предплохих базисах в P_2 // Математ. вопросы кибернетики. — 1992. — № 4. — С.139–177.

[4] *Перязев Н. А.* Слабоповторные булевы функции в бинарном базисе // Иркутский университет. Серия: Дискретная математика и информатика. Вып. 4. — Иркутск, 1998.

[5] *Кириченко К. Д.* Слабоповторные булевы функции в некоторых предэлементарных базисах // Иркутский университет. Серия: Дискретная математика и информатика. Вып. 13. — Иркутск, 2000.

[6] *Шаранхаев И. К.* О слабоповторных булевых функциях в одном предэлементарном базисе // Дискретный анализ и исследование операций. Серия 1. — 2003. — Т. 10, № 2. — С. 79–101.

[7] *Шаранхаев И. К.* О булевых базисах второго яруса // Известия вузов. Математика. — 2004. — Вып. 3. — С. 81–82.

[8] *Шаранхаев И. К.* Слабоповторные булевы функции в предэлементарном немонотонном базисе порядка 3 // Вестник Бурятского университета. Серия 13. — 2005. — Вып. 2. — С. 61–71.

[9] *Шаранхаев И. К.* О классификации базисов булевых функций // Вестник Бурятского университета. Серия 13. — 2006. — Вып. 3. — С. 61–67.

Задача оптимального управления для систем с нелокальными условиями

Я. А. Шарифов

Sharifov22@rambler.ru

Бакинский государственный университет

Исследование нелокальных задач представляет интерес как с точки зрения развития общей теории дифференциальных уравнений, так и с точки зрения приложений в математическом моделировании [1]. Объектом исследования в данной работе являются задачи оптимального управления в системах нелинейных обыкновенных дифференциальных уравнений первого порядка с граничными условиями:

$$\dot{x} = f\left(x, u, t\right), \quad x\left(t\right) \in R^n, \quad t \in T = \left[t_0, t_1\right], \tag{1}$$

$$Ax\left(t_0\right) + Bx\left(t_1\right) = C, \tag{2}$$

здесь $f\left(x, u, t\right)$ — заданная n-мерная вектор-функция, непрерывная по совокупности переменных вместе с производными по x и u до второго порядка включительно, $A \in R^n$, $B \in R^{n \times n}$, $C \in R^{n \times 1}$ — постоянные матрицы, $u\left(t\right)$ — r-мерный измеримый и ограниченный вектор управляющих воздействий на отрезке T. Предполагается, что почти всюду на этом отрезке управляющие воздействия удовлетворяют ограничению типа включения:

$$u\left(t\right) \in U, \quad t \in T, \tag{3}$$

где U — открытое множество из пространства R^r.

Целью задачи оптимального управления является минимизация функционала

$$J(u) = \varphi(x(t_0), x(t_1)) + \int\limits_T F(x,u,t)\,dt, \tag{4}$$

определенного на решениях краевой задачи (1), (2) при допустимых управлениях, удовлетворяющих условию (3). Здесь предполагается, что скалярные функции $\varphi(x,y)$ и $F(x,u,t)$ непрерывны по своим аргументам и имеют непрерывные и ограниченные производные по x и y до второго порядка включительно.

Пусть при некоторых условиях краевая задача (1), (2) при каждом допустимом управлении $u(t) \in U,\ t \in T$, имеет единственное решение $x(t,u)$.

Пусть $\{u, x = x(t,u)\}$ и $\{\tilde{u} = u + \Delta u,\ \tilde{x} = x + \Delta x = x(t,\tilde{u})\}$ — два допустимых процесса, где $\Delta u(t) = \varepsilon\delta u(t)$, ε — достаточно малое число, $\delta u(t)$ — некоторая кусочно-непрерывная функция. Тогда приращение функционала $\Delta J(u) = J(\tilde{u}) - J(u)$ при фиксированных функциях $u(t), \Delta u(t)$ есть функция параметра ε. Если справедливо представление

$$\Delta J(u) = \varepsilon\delta J(u) + \frac{1}{2}\varepsilon^2\delta^2 J(u) + o\left(\varepsilon^2\right), \tag{5}$$

то назовём $\delta J(u)$ первой вариацией, а $\delta^2 J(u)$ — второй вариацией функционала.

Пусть $H(\psi,x,u,t) = \langle\psi(t), f(x,u,t)\rangle - F(x,u,t)$ функция Понтрягина, а функция $\psi = \psi(t) \in R^n$ является решением следующей краевой задачи

$$\dot{\psi}(t) = -\frac{\partial H(\psi,x,u,t)}{\partial x}, \qquad t \in T,$$
$$A'\psi(t_1) + B'\psi(t_0) = -B'\frac{\partial\varphi}{\partial x(t_0)} - A'\frac{\partial\varphi}{\partial x(t_1)}.$$

Из (5) следует, что на оптимальном управлении $u^0(t)$ выполняются условия [2]

$$\delta J(u^0) = 0,\ \ \delta^2 J(u^0) \geqslant 0. \tag{6}$$

Теорема. *Если допустимое управление $u(t)$ удовлетворяет условию $\delta J(u^0)=0$, то для его оптимальности в задаче (1)–(4) необходимо, чтобы неравенство*

$$\delta^2 J(u) = -\left\{ \int_T \int_T \left\langle \delta' u(\tau) \frac{\partial' f(x,u,\tau)}{\partial u} R(\tau,s) \frac{\partial f(x,u,s)}{\partial u}, \delta u(s) \right\rangle d\tau ds + \right.$$
$$+ \int_T \left\langle \delta' u(\tau) \frac{\partial^2 H(\psi,\, x,u,t)}{\partial u^2}, \delta u(t) \right\rangle dt +$$
$$+2 \int_T \int_T \left\langle \delta u'(t) \frac{\partial^2 H(\psi,x,u,t)}{\partial x \partial u} \Phi(t) \left(A + B\Phi\left(t_1\right)\right)^{-1} \Phi^{-1}(s) \frac{\partial f(x,u,s)}{\partial u}, \delta u(s) \right\rangle dtds +$$
$$\left. + \int_T \left\langle \int_t^{t_2} \delta u'(\tau) \frac{\partial^2 H(\psi,x,u,\tau)}{\partial x \partial u} \Phi(\tau) d\tau \Phi^{-1}(t) \frac{\partial f(x,u,t)}{\partial u}, \delta u(t) \right\rangle dt \right\} \geqslant 0$$

выполнялось для всех $\delta u(t) \in L_\infty[t_0,t_2]$, где

$$R(\tau,s) = \Phi^{-1\prime}(\tau) \left[\Phi(t_1)\left(A+B\Phi(t_1)\right)^{-1}\right]' \frac{\partial^2 \varphi}{\partial x(t_1)^2} \Phi(t_1)\left(A+B\Phi(t_1)\right)^{-1} \Phi^{-1}(s) +$$
$$+\Phi^{-1\prime}(\tau)\left[\left(A+B\Phi(t_1)\right)^{-1} B\Phi(t_1)\right]' \frac{\partial^2 \varphi}{\partial x(t_0)^2} \left(A+B\Phi(t_1)\right)^{-1} B\Phi(t_1)\Phi^{-1}(s) +$$
$$+\Phi^{-1\prime}(\tau)\left[\left(A+B\Phi(t_1)\right)^{-1} B\Phi(t_1)\right]' \frac{\partial^2 \varphi}{\partial x(t_0)\, \partial x(t_1)} \Phi(t_1)\left(A+B\Phi(t_1)\right)^{-1}\Phi^{-1}(s) +$$
$$+\Phi^{-1\prime}(\tau)\left[\left(A+B\Phi(t_1)\right)^{-1} \Phi(t_1)\right]' \frac{\partial^2 \varphi}{\partial x(t_0)\, \partial x(t_1)} \Phi(t_1)\left(A+B\Phi(t_1)\right)^{-1} B\Phi^{-1}(s) +$$
$$+\Phi^{-1\prime}(\tau)\left[\left(A+B\Phi(t_1)\right)^{-1} B\Phi(t_1)\right]' \times$$
$$\times \int_T \Phi'(t) \frac{\partial^2 H(\psi,x,u,t)}{\partial x^2} \Phi(t)\, dt \left(A+B\Phi(t_1)\right)^{-1} B\Phi(t_1)\Phi^{-1}(s) +$$
$$+\Phi^{-1\prime}(\tau) \int_{\max(\tau,s)}^{t_1} \Phi'(t) \frac{\partial^2 H(\psi,x,u,\tau)}{\partial x^2} \Phi(t)\, dt \Phi^{-1}(s) -$$
$$-\Phi^{-1\prime}(\tau) \int_\tau^{t_1} \Phi'(\xi) \frac{\partial^2 H(\psi,x,u,\xi)}{\partial x^2} \Phi(\xi)\, d\xi \left(A+B\Phi(t_1)\right)^{-1} B\Phi(t_1)\Phi^{-1}(s) -$$

$$-\Phi^{-1}(\tau)\left[(A+B\Phi(t_1))^{-1}B\Phi(t_1)\right]'\int\limits_{\tau}^{t_1}\Phi'(\xi)\frac{\partial^2 H(\psi,x,u,\xi)}{\partial x^2}\Phi(\xi)\,d\xi\Phi^{-1}(s),$$

$\Phi(t)$ — *нормированная фундаментальная матрица вариационного уравнения.*

Список литературы

[1] *Нахушев А. М.* Задачи со смещением для уравнений в частных производных. — М.: Наука, 2006.

[2] *Габасов Р., Кириллова Ф. М.* Особые оптимальные управления. — М.: Наука, 1973.

Восстановление графа мозаичной структуры агентами

Н. К. Шатохина

nshatokh@rumbler.ru

Государственный университет информатики и искусственного интеллекта, Донецк

Рассмотрена задача описания структуры графа на основе информации, полученной при обходе его по границе. В работе предложен алгоритм восстановления структуры графа мозаичной структуры в виде формулы, отражающей состав исходного графа как комбинацию базовых структур. Приведены оценки его временной и емкостной сложности.

Задача описания структуры объекта на основе информации, полученной при обходе по его границе, — одна из задач анализа геометрических сред, графов и других дискретных структур. Известны работы, например [1], посвященные задаче распознавания структуры объекта, в которых рассматривался случай решения подобных задач с использованием одного автомата (далее — агента).

В настоящей работе предлагается алгоритм распознавания структуры сильносвязного неориентированного графа G без дыр, состоящего из равносторонних треугольников, с использованием двух агентов. Первый агент-исследователь (АИ) выполняет перемещения по исходному графу и передачу информации второму агенту-экспериментатору (АЭ). Второй агент по полученной информации описывает структуру графа.

Требуется разработать такие алгоритмы передвижения агента АИ по исходному графу G с передачей информации об этом и восстановления графа агентом АЭ по полученным данным в виде формулы, что АИ за конечное число шагов независимо от начального размещения обойдет часть внутренних вершин G и все граничные вершины, пошагово передавая информацию. АЭ по этой информации через конечное число шагов опишет граф P, изоморфный исходному графу G.

Основные определения

Пусть $E(v) = \{e_1, \dots\}$, $|E(v)| \leqslant 6$, обозначает множество ребер, инцидентных вершине v. На $E(v)$ определена функция $q : E(v) \to X = \{x, x^{+1}, x^{+2}, x^{+3}, x^{+4}, x^{+5}\}$, где показатель степени обозначает количество поворотов против часовой стрелки на угол 60° относительно ребра x. Введенная функция позволяет сопоставлять ребрам метки. Определим функцию $f : (E(v), q(v, w)) \to V$ так, что $f(v, q(v, w)) = w$, где (v, w) — ребро с меткой $q(v, w)$. Тогда всякое перемещение АИ по любому ребру $e = (v, w)$ (операцию перемещения) можно однозначно описать функцией q, а $x = q(v, w)$ задает направление перемещения и метку x ребра.

Лабиринтом $L = (G, t, q, f)$ называется граф G, на котором определена t — разметка вершин, q — разметка ребер и f — функция перемещения.

Маршрут по границам графа называется выпуклым, если для любых двух последовательных направлений перемещения $x^{l_i} x^{l_{i+1}}$ выполняется условие $0 \leqslant l_{i+1} - l_i \leqslant 3$, а граф соответственно — выпуклым.

Маршрут по границам графа называется вогнутым, если найдется хотя бы одна пара подряд следующих направлений перемещения $x^{l_i} x^{l_{i+1}}$, для которых выполняется условие $l_{i+1} - l_i \leqslant 4$, а граф — вогнутым. Вершина, инцидентная данным ребрам, называется вершиной перегиба.

Базовые мозаики

Рассматриваемые в данной работе графы имеют мозаичную структуру, состоящую из правильных треугольников. Частными случаями таких графов являются равносторонний треугольник, параллелограмм, трапеция и шестиугольник, которые называются базовыми структурами. Каждой такой структуре однозначно соответствуют строки направлений, порождаемые обходом против часовой стрелки $+M(v, k_1, k_2)$ и по часовой стрелке $-M(v, k_1, k_2)$, которые называются базовой и обратной базовой. Здесь k_1, k_2 — количество ребер одной из сторон структуры, для треугольника и шестиугольника $k_2 = 0$, v — начальная (опорная) вершина маршрута.

По количеству вершин (ребер), расположенных на границе базовых структур, нетрудно оценить количество внутренних вершин структуры, и, наоборот, по общему количеству вершин базовых структур нетрудно определить длину маршрутов по граничным вершинам графа.

Лемма 1. *Длина маршрута (количество ребер) p по граничным вершинам G оценивается как $\sqrt{12n-3} - 3 \leqslant p \leqslant 2n$.*

Идентификатором базовой структуры называется подстрока $M(v, k_1, k_2)$, полученная из $M(v, k_1, k_2)$ путем удаления последнего символа с учетом соответствующего коэффициента, то есть $M(v, k_1, k_2) = M(v, k_1, k_2)kx^d$, где d — степень последнего символа в $M(v, k_1, k_2)$.

Введены операции сокращения на подстроках на основе использования определений идентификаторов. Согласно этим операциям, как только в строке направлений встретится один из идентификаторов базовой структуры, происходит замена его ($-M(v, k_1, k_2)$ или $+M(v, k_1, k_2)$) на kx^d. Причем если это была обратная базовая структура, то происходит присоединение соответствующего подграфа к G. Если же встречается идентификатор базовой структуры, происходит удаление из G соответствующего подграфа.

Процесс распознавания состоит из двух алгоритмов — «Обход» и «Восстановление» для АЭ и АИ соответственно.

Алгоритмы распознавания графа

В алгоритме работы АИ предполагается, что агент случайным образом может быть помещен в любую вершину графа, поэтому алгоритм состоит из двух этапов. На первом этапе, если необходимо, АИ выходит на границу, а на втором этапе АИ обходит граф против часовой стрелки по граничным ребрам до тех пор, пока не возвратится в вершину выхода на границу.

Лемма 2. *Если в мозаичном графе n вершин, то максимальная длина пути по внутренним вершинам графа в выбранном направлении не превосходит величины $l = (n-1)/3$.*

Временная сложность алгоритма с учетом лемм составляет $O(n)$ шагов. Емкостная сложность $E(n)$ отсутствует, т.к. агенту ничего не требуется сохранять.

В алгоритме работы АЭ дважды анализируется строка направлений, сгенерированная и переданная АИ.

Вначале производится поиск точек перегиба. Если точки имеются, то определяется идентификатор обратной базовой структуры, содержащий очередную точку перегиба в качестве второй вершины маршрута. Если находится обратная базовая структура $-M(v, k_1, k_2)$, то производится преобразование строки согласно операции сокращения, присоединение данной структуры к графу с соответствующим количеством вершин, а в строку формулы добавляется $-M(v, k_1 k_2)$. В противном случае производится поиск идентификаторов базовых структур, расположенных до точки перегиба. Каждый раз, когда находится идентификатор, производится уменьшение строки согласно операции сокращения, отделение данной структуры от графа, и в строку формулы добавляется $+M(v, k_1, k_2)$.

Далее вновь просматривается сначала строка для поиска идентификаторов базовых структур. Каждый раз, когда находится идентификатор, производится преобразование строки согласно операции сокращения, отделение данной структуры от графа и добавление в формулу со знаком «+» соответствующего обозначения фрагмента. В результате преобразования исходная строка преобразуется в пустую строку.

Поскольку для каждой базовой структуры однозначно определено количество ее вершин, то фактически создается неявная нумерация этих вершин. Эта нумерация является биекцией, т. к. удаление/добавление базовых структур выполняется взаимно-однозначно в соответствии с их идентификаторами. Таким образом, справедливо следующее утверждение.

Теорема. *Выполняя алгоритмы распознавания, агенты распознают любой граф G с точностью до изоморфизма.*

В алгоритме работы АЭ осуществляются два просмотра исходной строки.

При подсчете временной сложности алгоритма работы АЭ учитывается, что распознавание структуры графа выполняется за счет двукратного просмотра исходной строки, длина которой не может превосходить $2n$ букв, т. е. временная сложность составит $4n$ шагов, то есть $O(n)$. Следовательно, суммарная временная сложность оценивается как $T(n) = O(n)$. Таким образом, совместная работа АИ и АЭ оценивается как $T(n) = O(n)$. Емкостная сложность $E(n)$ определяется сложностью строки направлений, мощностью множества граничных вершин $V' \subset V$ и строки формул, сложность которых оценивается величинами $O(n)$, $O(n)$, $O(n)$.

Список литературы

[1] *Кудрявцев В. Б., Ушчумлич Ш., Килибарда Г.* О поведении автоматов в лабиринтах // Дискретная математика. — 1992. — Т. 4, № 3. — С. 3–28.

О диагностике отождествления переменных в формулах булевых функций

В. И. Шевченко

vladimir-sh274@rambler.ru

Нижегородский государственный университет им. Н. И. Лобачевского

Рассматриваются схемы из функциональных элементов формульного типа [1], то есть схемы, в которых разветвления могут быть только на входах схемы, а к выходу каждого элемента схемы может быть присоединен только один вход только одного элемента. В любой из рассматриваемых схем один или несколько элементов могут находиться в неисправном состоянии. При этом если в исправном состоянии элемент реализует функцию $\varphi(x_1, x_2, \ldots, x_p)$, то при наличии в нем неисправностей он реализует функцию, которая получается из $\varphi(x_1, x_2, \ldots, x_p)$ отождествлением некоторых переменных. Для диагностики (поиска) этих неисправностей используются деревья решений (условные тесты) [2, 3]. В работе для различных конечных базисов для схем исследуются верхние и нижние оценки минимальной глубины деревьев решений в худшем случае.

Конечное множество булевых функций B будем называть *базисом*, а схему S, в которой каждому элементу сопоставлена функция из B, *схемой в базисе* B. Входы S и выходы ее элементов иногда будем называть вершинами. Предполагается, что каждая схема имеет хотя бы один вход, хотя бы один функциональный элемент и ровно один выход. Обозначим через $H(S)$ множество всех схем, которое содержит схему S и все схемы, которые могут быть получены из S при переходе некоторых ее элементов в неисправные состояния. Никаких других схем $H(S)$ не содержит. Множество различных булевых функций, реализуемых схемами из $H(S)$, обозначим через $F(S) = \{f_1, f_2, \ldots, f_m\}$. Разобъем $H(S)$ на подмножества

$$H_1(S), H_2(S), \ldots, H_m(S)$$

такие, что для $i = 1, \ldots, m$ все схемы из $H_i(S)$ реализуют одну и ту же булеву функцию f_i. Для удобства будем предполагать, что $S \in H_1(S)$.

Задача диагностики схемы S: для любой схемы U из $H(S)$ определить к какому из подмножеств $H_i(S)$, $i = 1, \ldots, m$, принадлежит U. Для решения этой задачи используются деревья решений (условные тесты).

Дерево решений Y для решения задачи диагностики схемы S представляет собой конечное ориентированное корневое дерево, в котором каждой вершине, не являющейся концевой, приписан двоичный набор из $\{0,1\}^n$, каждой концевой вершине — некоторое число из множества $\{1, \ldots, m\}$. Из каждой вершины, не являющейся концевой, исходят ровно две дуги, которым приписаны числа 0 и 1. Далее, для любой функции $f_i \in F(S)$, реализуемой схемами из $H_i(S)$, найдется полный путь (от корня до концевой вершины) $\gamma = v_1, u_1, \ldots, u_r, v_{r+1}$ такой, что вершине v_{r+1} приписано число i и, если при $q = 1, \ldots, r$ вершине v_q приписан набор $\alpha_q \in \{0,1\}^n$, а дуге u_q — число $\delta_q \in \{0,1\}$, то функция f_i — единственная функция в $F(S)$, которая на наборах $\alpha_1, \ldots, \alpha_r$ принимает значения $\delta_1, \ldots, \delta_r$ соответственно. Максимальная длина полного пути называется глубиной дерева решений Y и обозначается через $h(Y)$. Величина $d(S) = \min h(Y)$, где минимум берется по всем деревьям решений для диагностики S, называется *минимальной глуби-*

ной деревьев решений для диагностики схемы S. Обозначим через $d_B(N) = \max d(S)$, где максимум берется по всем схемам в базисе B, число вершин в которых не провосходит N.

Теорема. а) *Если базис* B *есть* $\{x \cdot y\}$ *или* $\{x \wedge y\}$, *или* $\{x \oplus y\}$, *то для* $N \geqslant 3$ *справедливо равенство:*

$$d_B(N) = \lceil N/2 \rceil.$$

б) *Если* $B = \{x \cdot y, x \vee y\}$ *или* B *есть непустое подмножество множества* $\{x(y \vee z), x \vee yz, xy \vee yz \vee xz\}$, *то для* $N \geqslant 2$ *имеют место неравества:*

$$2^{\lfloor (N-2)/6 \rfloor - 1} \leqslant d_B(N) \leqslant \binom{N+1}{\lceil N/2 \rceil + 1}.$$

в) *Если* $B = \{x_1 \cdot x_2, \bar{x}\}$ *или* $B = \{x_1 \vee x_2, \bar{x}\}$, *или* B *есть непустое подмножество множества* $\{x_1 \cdot \bar{x}_2, \bar{x}_1 \cdot \bar{x}_2, x_1 \vee \bar{x}_2, \bar{x}_1 \vee \bar{x}_2\}$, *то для* $N \geqslant 2$ *имеют место неравества:*

$$2^{\lfloor (N-2)/3 \rfloor - 1} \leqslant d_B(N) \leqslant 2^{\lceil N/2 \rceil}.$$

Список литературы

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

[2] *Мошков М. Ю./* Деревья решений. Теория и приложения. — Нижний Новгород: Изд-во ННГУ, 1994.

[3] *Яблонский С. В., Чегис И. А.* Логические способы контроля работы электрических схем. // Тр. Матем. ин-та АН СССР. — 1958. — Т. 51. — С. 270–360.

О сложности задач целочисленного линейного программирования с ограниченными минорами

В. Н. Шевченко

shev@uic.nnov.ru

Нижегородский государственный университет им. Н. И. Лобачевского

Пусть $\mathbf{Z}$ — кольцо целых чисел, $\mathbf{Q}$ — поле рациональных чисел, $A = (a_{ij}) \in \mathbf{Z}^{m \times n}$ — целочисленная $m \times n$ матрица, $b \in \mathbf{Z}^m$

— целочисленный вектор-столбец, i-я компонента которого равна b_i $(i = 1, \ldots, m)$, J_+ — подмножество множества $\{1, \ldots, n\}$.

Рассмотрим систему линейных уравнений

$$Ax = b, \tag{1}$$

с неизвестными x_j $(j = 1, \ldots, n)$, где x_j — j-я координата вектора $x \in \mathbf{Q}^n$, и множество $\mathcal{M}(A, b, J_+)$ ее решений, удовлетворяющих условиям

$$x_j \geqslant 0 \quad \text{при} \quad j \in J_+, \tag{2}$$

Известно, что задача $\mathcal{P}_1$: найти вектор $z \in \mathcal{M}(A, b, J_+)$ или доказать, что $\mathcal{M}(A, b, J_+) = \varnothing$ — при $J_+ = \{1, ..., n\}$ является NP-полной и принадлежит классу P (задач, для которых имеются полиномиальные алгоритмы) при $J_+ = \varnothing$ (см., например, [1, 2]).

Выделены два класса подзадач $\mathcal{P}_1$, для которых имеются полиномиальные алгоритмы: один получается из псевдополиномиальных, а другой из квазиполиномиальных (см., например, [3, 4]) алгоритмов. Здесь предлагается третий класс — класс матриц $\mathfrak{A}_c$, состоящий из матриц A, любой минор для которых не превосходит по абсолютной величине некоторой константы c.

Теорема 1. *Если $A \in \mathfrak{A}_c$, то $\mathcal{P}_1 \in P$.*

Рассмотрим множества

$$P(A, b) = \{x \in \mathbf{Q}^n \mid Ax \leqslant b\}, \quad M(A, b) = P(A, b) \cap \mathbf{Z}^n$$

и задачу $\mathcal{P}_2$: найти $z \in M(A, b)$ или доказать, что $M(A, b) = \varnothing$.

Теорема 2. *Если $A \in \mathfrak{A}_c$, то $\mathcal{P}_2 \in P$.*

Нетрудно видеть, что теорему 2 достаточно доказать для случая, когда ранг A равен n (что и будем предполагать выполненным). Известно [4], что задача $\mathcal{P}_2$ принадлежит классу P при $m = n$ и NP-полна при $m = n+1$ (что позволяет говорить о ней как о простейшей NP-полной задаче целочисленного программирования).

Ключ к доказательству теоремы 2 дает

Лемма 3. *Если $A \in \mathfrak{A}_c$, $P(A, b) \neq \varnothing$, а $M(A, b) = \varnothing$, то существует полиномиальный алгоритм доказательства этого.*

Теоремы 1 и 2 сформулированы в [4] как гипотезы.

Работа выполнена при поддержке РФФИ, проект № 09-01-00545а.

Список литературы

[1] *Гэри М. Р., Джонсон Д. С.* Вычислительные машины и труднорешаемые задачи // М.: Мир, 1982.

[2] *Схрейвер А.* Теория линейного и целочисленного программирования // М.: Мир, 1991.

[3] *Lenstra H. W.* Integer programming with a fixed number of variables // Mathematics of operations research. — 1983. — V. 8, № 4. — P. 538–548.

[4] *Шевченко В. Н.* Качественные вопросы целочисленного программирования // М.: Физматлит, 1995.

Численное нахождение количественных характеристик некоторых $\{0,1\}$-матриц

В. Н. Шевченко, М. Е. Сморкалов

Нижегородский государственный университет им. Н. И. Лобачевского

Введение

Задача оценки средней величины миноров матриц некоторого типа играет важную роль во многих разделах математики, особенно в целочисленном программировании. Так, например, хорошо известно [1], что если матрица многоиндексной транспортной задачи неунимодулярная, то в некоторых случаях транспортный многогранник имеет вершины с дробными координатами и максимальный из знаменателей дробных компонент вершин многогранников не превосходит абсолютной величины базисного минора матрицы задачи.

Обозначим через $B_{n,k}$ булеву матрицу из n строк и $\binom{n}{k}$ столбцов, столбцами которой являются всевозможные булевы векторы, содержащие ровно k единиц. Заметим, что в случае $k = 2$ матрица $B_{n,k}$ представляет собой матрицу инцидентности полного графа. В случае $k \geqslant 3$ матрицу $B_{n,k}$ можно рассматривать как матрицу инцидентности полного гиперграфа.

Из [2] известно, как с ростом n ведет себя среднее значение модуля минора n-го порядка матрицы $B_{n,k}$ при $k = 2$ и $k \geqslant 8$. Поведение же этой величины при $3 \leqslant k \leqslant 7$ остается невыясненным, хотя известно, что среднее значение квадрата такого минора стремится к бесконечности. Задача ставится следующим образом: как, зная поведение среднего значения квадрата минора матрицы $B_{n,k}$ при $n \to \infty$, получить асимптотику модуля такого минора? Вообще, логично предположить, что если квадрат наудачу выбранного минора стремится к бесконечности, то к бесконечности стремится и модуль наудачу взятого минора. Однако, как выяснилось, это не всегда так. В данном докладе приводятся некоторые результаты, полученные для $k = 3, \ldots, 7$ с помощью методов математической статистики.

Нахождение среднего значения модуля минора матрицы $B_{n,k}$

Зафиксируем k ($k = 3, \ldots, 7$) и будем считать, что из всех возможных миноров порядка n матрицы $B_{n,k}$ наудачу выбирается минор и вычисляется его модуль. Тогда модуль таким образом выбираемого минора — случайная величина. Обозначим ее ξ. Данная случайная величина будет иметь следующий закон распределения вероятностей:

ξ	1	2	$\ldots$	m
P	$X_0(n)/M$	$X_1(n)/M$	$\ldots$	$X_0(m)/M$

Здесь $M = \binom{L}{n}$, $L = \binom{n}{k}$, $X_i(n)$ — количество квадратных подматриц порядка n матрицы $B_{n,k}$, модуль детерминанта которых равен i, m — максимально возможное значение детерминанта (по модулю). Математическое ожидание ξ вычисляется по формуле $\sum_{\nu=1}^{m} \frac{\nu X_\nu(n)}{M}$. Если бы нам был известен ряд распределения, то мы могли бы воспользоваться этой формулой и найти его теоретически. Однако, при $n \gg 1$ сделать это затруднительно, поэтому предлагается искать несмещенную оценку по выборке.

Для нахождения такой оценки зафиксируем k, сформируем матрицу $n \times n$ из различных случайных $(0,1)$-векторов, содержащих ровно по k единиц, и вычислим модуль ее определителя. Повторим эту процедуру N раз. В результате получим N модулей случайно выбранных миноров n-го порядка матрицы $B_{n,k}$. Обозначим их через $\overline{\nu_1}$, $\overline{\nu_2}, \ldots,$ $\overline{\nu_N}$ и найдем $\overline{M_\xi} = \frac{1}{N} \sum_{\nu=1}^{N} \overline{\nu_i}$. Известно, что $\overline{M_\xi} \approx M_\xi (M(\overline{M_\xi} - M_\xi)) = D_\xi / N$, при условии, что $M_\xi < \infty$ и дисперсия $D_\xi < \infty$. При фиксированном k будем увеличивать n и повторять описанный эксперимент.

Из результатов экспериментов следует, что при $4 \leqslant k \leqslant 7$ с ростом n среднее значение модулей миноров матриц растет экспоненциально от n. При $k = 3$ эксперимент показал рост среднего значения модуля минора при малых n ($n < 70$) и последующее убывание до нуля с ростом n.

Ниже приведена таблица с результатами эксперимента
для $k = 3, N = 10000$:

n	10	20	30	40	50	70	100	150	300
$\overline{M_\xi}$	1.76	1.82	1.96	2.76	3.09	3.61	0.79	0	0

для $k = 4, N = 10000$:

n	10	30	50	100	200	300
$\overline{M_\xi}$	4.01	208.79	7800	2.5E+7	5.9E+15	6.87E+21

для $k = 5, N = 10000$:

n	10	30	50	100	200	300
$\overline{M_\xi}$	8.1	4500	1.7E+6	4.2E+12	2.3E+25	4.9E+38

для $k = 6, N = 10000$:

n	10	30	50	100	200	300
$\overline{M_\xi}$	6.42	3.3E+4	1.5E+8	3.3E+16	3.2E+33	1.06E+51

для $k = 7, N = 10000$:

n	10	30	50	100	200	300
$\overline{M_\xi}$	3.85	2.3E+5	3.5E+9	4.2E+19	1.4E+40	2.8E+60

Подбор стандартного вероятностного распределения для распределения модулей миноров

Задача сравнения закона распределения некоторой случайной величины с известными законами распределения является очень важной. Зная, что исследуемая случайная величина имеет тот или иной закон распределения, мы можем использовать свойства этих распределений, например, находить вероятности того, что случайная величина попадет в тот или иной интервал.

В работе были проведены сравнения распределения исследуемой случайной величины (модуля наудачу выбранного минора матрицы $B_{n,k}$) со следующими стандартными распределениями:

— гамма-распределение;
— распределение Пуассона;
— отрицательное биномиальное распределение;
— показательное распределение;
— распределение Релея.

Для этого использовался критерий согласия Пирсона. В данной ситуации имело смысл проводить сравнение исследуемого распределения, очевидно дискретного, не только со стандартными дискретными, но и непрерывными распределениями, т. к. известно, что некоторые дискретные распределения при выполнении определенных условий можно аппроксимировать непрерывными. Например, биномиальное распределение с параметрами m и p при $m \gg 1$ и $mp(1-p) > 20$ можно аппроксимировать нормальным распределением с параметрами mp и $mp(1-p)$. Значение уровня значимости (вероятность отклонения правильной гипотезы) выбиралось равным 0.001. Величина выборки (N) во всех случаях бралась равной 10000. Количество интервалов, на которые разбивался отрезок, содержащий все результаты наблюдений, бралось равным $N^{1/3}$. Параметры тестовых распределений выбирались по принципу максимального правдоподобия с помощью встроенных функций Matlab.

По результатам экспериментов можно сделать вывод, что при $k = 3$ и при достаточно больших n распределение модуля наудачу выбранного минора матрицы $B_{n,3}$ аппроксимируется гамма-распределением, плотность которого задается формулой:

$$y = f(x|a,b) = \frac{1}{b^a \Gamma(a)} x^{a-1} e^{-x/b},$$

где $\Gamma(a)$ — гамма-функция от переменной a. Ниже приведена таблица с результатами эксперимента.

n	10	20	30	50	70
χ^2	9537.725	280	17.575	3.647	0.953
$\chi^2(1-q)$	39.252	31.264	26.124	24.322	13.816
Распределение подобрано верно	нет	нет	нет	да	да

При этом с ростом n при $n > 70$ отношение $\chi^2(1-q)$ к χ^2 продолжает расти и, стало быть, утверждение продолжает быть верным. Здесь

χ^2 — статистика Пирсона, а $\chi^2(1-q)$ — квантиль χ^2-распределения Пирсона.

Для $3 < k < 8$ подобрать стандартное распределение, которое бы аппроксимировало распределение модуля наудачу выбранного минора, не удалось.

Работа поддержана РФФИ, проект № 09-01-00545-а.

Список литературы

[1] *Шевченко В. Н.* Многогранники многоиндексных транспортных задач: алгебраический подход // Дискретный анализ и исследование операций. — 2004. — С. 64–70.

[2] *Шевченко В. Н.* Качественные вопросы целочисленного программирования. — М.: Физматлит, 1995.

О достаточности принципа максимума Понтрягина для одной нелинейной задачи оптимального управления

Г. В. Шевченко

shevch@math.nsc.ru

Институт математики им. С. Л. Соболева СО РАН, Новосибирск

Рассмотрена задача перевода нелинейной системы в нулевое состояние с минимизацией неотрицательного интегрального функционала. Частные случаи этой задачи — задачи минимизации расхода ресурсов и минимизации расхода энергии. Показано, что для этой задачи принцип максимума является необходимым и достаточным условием. Следует подчеркнуть, что при доказательстве достаточности принципа максимума не используются функции Кротова.

Пусть управляемый объект описывается системой нелинейных обыкновенных дифференциальных уравнений вида

$$\dot{x} = A(t)x + f(t, x, u), \tag{1}$$

где A — квадратная матрица порядка n; x — n-мерный вектор фазового состояния объекта; $x(0) = x^0$ — заданное начальное состояние объекта; $f(t, x, u) : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^s \to \mathbb{R}^n$ — непрерывная по x и по u и непрерывно дифференцируемая по x функция; $f(t,x,u)\Big|_{x\neq 0,\, u\neq 0} \neq 0$, $f(t,0,u)\Big|_{u\neq 0} \neq 0$, $f(t,x,0)=0$; u — s-мерное измеримое управление, стесненное ограничением

$$u(t) \in U, \tag{2}$$

U — компактное тело из $\mathbb{R}^s$, содержащее внутри 0.

Предполагается, что система (1) управляема из начального состояния x^0 в начало координат.

Задача. *Требуется найти допустимое управление* $u^0(t)$ $(t\in[0,T])$, *переводящее систему* (1) *из* $x(0)=x^0$ *за фиксированное время* T *в нулевое конечное состояние* $x(T)=0$ *и минимизирующее функционал*

$$\mathcal{F}(u) = \int\limits_0^T W(u(t))dt, \tag{3}$$

где $W(u)$ — *неотрицательная функция* $W:\mathbb{R}^s\to\mathbb{R}$, *такая, что* $W(0)=0$ *и* $W(u)\Big|_{u\neq 0}\neq 0$.

Очевидно, что эта задача имеет решение только при $T \geqslant T_{\mathrm{opt}}$, где T_{opt} — время оптимального по быстродействию перевода системы (1) из начального состояния x^0 в начало координат. Существование оптимального по быстродействию управления для более общих систем, частный случай которых система (1), показано в [1]. Предположим, что $T > T_{\mathrm{opt}}$.

Частными случаями поставленной задачи являются известные задачи: минимизации расхода ресурсов [2], где $W(u) = \sum\limits_{j=1}^{s}\alpha_j u_j$, $\sum\limits_{j=1}^{s}\alpha_j \neq 0$, $\alpha_j \geqslant 0$, и минимизации энергии, где $W(u)=\sum\limits_{j=1}^{s}u_j^2$, а в качестве U берётся параллелепипед $\{u=(u_1,\dots,u_s)\mid u_j\leqslant M_j,\ j=\overline{1,s}\}$, $M_j>0$ — действительные числа.

Обозначим через $\mathfrak{R}(T)$ область достижимости системы (1) за время T из начального состояния $x(0)=x^0$ допустимыми управлениями. В силу того, что $T>T_{\mathrm{opt}}$, $f(t,x,u)\Big|_{x\neq 0,\, u\neq 0}\neq 0$, $f(t,0,u)\Big|_{u\neq 0}\neq 0$

и $f(t, x, 0) = 0$, имеет место включение $0 \in \operatorname{int} \mathfrak{R}(T)$. Через $\operatorname{int} A$ здесь и далее обозначается внутренность множества A. В силу непрерывности правой части системы (1) множество $\mathfrak{R}(T)$ компактно. Более того, поскольку система (1) управляема в начало координат, $\mathfrak{R}(T)$ — тело.

Введём согласно принципу максимума Л. С. Понтрягина [3] сопряжённую систему

$$\dot{\psi} = -(A^* + D(f))\psi(t), \tag{4}$$

где $D(f)$ — матрица частных производных функции f по x, и запишем для задачи функцию Понтрягина

$$H(\psi(t), x(t), u(t)) = -W(u(t)) + \langle \psi(t), A(t)x(t) \rangle + \langle \psi(t), f(t, x(t), u(t)) \rangle.$$

Здесь и далее через $\langle \cdot, \cdot \rangle$ обозначается скалярное произведение векторов.

Теорема 1 (принцип максимума Понтрягина). *Для оптимальности управления $u^*(t)$ $(t \in [0, T])$ и траектории $x^*(t)$ $(t \in [0, T])$ необходимо существование такой ненулевой вектор-функции $\psi^*(t)$, являющейся решением сопряжённой системы* (4) *при некотором вполне определённом граничном условии $\psi(T) = c^*$, что при почти всех $t \in [0, T]$ функция $H(\psi^*(t),\, x^*(t), u)$ по переменной $u \in U$ достигает в точке $u = u^*(t)$ максимума, т. е.*

$$H(\psi^*(t), x^*(t), u^*(t)) = \max_{u \in U} H(\psi^*(t), x^*(t), u).$$

Из последнего выражения получаем

$$u(t) = \arg \max_{u \in U} \{-W(u) + \langle \psi(t), f(t, x(t), u) \rangle\}. \tag{5}$$

В силу однородности по ψ системы (4) в дальнейшем можно ограничиться рассмотрением только граничных условий $\psi(T) = c$ с единичными нормами $\|c\| = 1$, используя вместо (5) следующее выражение:

$$u(t) = \arg \max_{u \in U} \{-\mu \cdot W(u) + \langle \psi(t), f(t, x(t), u) \rangle\}, \tag{6}$$

где μ — неотрицательное действительное число.

Обозначим через $u(t,c,\mu)$, $t \in [0,T]$, управление с компонентами, доставляющими максимум в (5) при некотором действительном числе $\mu \geqslant 0$.

Нетрудно показать, что при фиксированных t, $x(t)$ и $\psi(t)$ функция

$$g(t,c,\mu) = \max_{u\in U}\{-\mu \cdot W(u) + \langle \psi(t), f(t,x(t),u)\rangle\}$$

строго убывает по μ на некотором отрезке $[0,\mu_c(t)]$ и тождественно равна нулю при $\mu \geqslant \mu_c(t)$, где $\mu_c(t) = \max\{\delta(u) \mid u = u(t,c,\mu) \in U^+,\ \mu \geqslant 0\}$, $U^+ = \{u \in U \mid \langle \psi(t), f(t,x(t),u)\rangle > 0\}$, $\delta(u)$ таково, что $\gamma(t,c,\delta(u),u) = 0$.

Рассмотрим функцию $\mathcal{G}(c,\mu) = \mathcal{F}(u(c,\mu)) = \int\limits_0^T W(u(t,c,\mu))dt$, где $u(c,\mu) = u(t,c,\mu)$, $t \in [0,T]$. Нетрудно показать, используя строгое убывание $g(t,c,\mu)$ по μ, что $\mathcal{G}(c,\mu)$ строго убывает по μ на отрезке $[0, \mu_T(c) = \max\limits_{t\in[0,T]} \mu_c(t)]$ и тождественно равна нулю при $\mu > \mu_T(c)$.

Пусть $S \subset \mathbb{R}^n$ — единичная сфера с центром в начале координат. Каждому $c \in S$ при любом фиксированном $\mu \geqslant 0$ однозначно соответствуют управление $u_\mu = u_\mu(t) = u(t,c,\mu)$, $t \in [0,T]$, и точка $x_\mu = x(T,u_\mu)$ из области достижимости $\mathfrak{R}(T)$ системы (1), где $x(T,u_\mu)$ — решение задачи Коши (1), $x(0) = x^0$ при допустимом управлении $u = u_\mu$ в момент времени $t = T$. Рассмотрим множество $\Omega(\widetilde{\mu}) = \{x = x(T,u_\mu) \mid\ c \in S,\ \mu \geqslant \widetilde{\mu}\}$. Очевидно, что множество $\Omega(0)$ совпадает с областью достижимости $\mathfrak{R}(T)$ системы (1). Границей множества $\Omega(\widetilde{\mu})$ является множество

$$\partial\Omega(\widetilde{\mu}) = \{x = x(T,u_{\widetilde{\mu}}) \mid u_{\widetilde{\mu}} = u_{\widetilde{\mu}}(t) = u(t,c,\widetilde{\mu})\ (t \in [0,T]),\ c \in S\}.$$

Так как функция $\mathcal{G}(c,\mu)$ строго убывает по μ на отрезке $[0,\mu_T]$ при любом $c \in S$, справедливы включения $\Omega(\widetilde{\mu}_2) \subset \Omega(\widetilde{\mu}_1)$ при любых $\widetilde{\mu}_2 > \widetilde{\mu}_1 \geqslant 0$.

Из полученных включений следует, что существует наименьшее $\widetilde{\mu}^*$, такое, что начало координат лежит на границе множества $\Omega(\widetilde{\mu}^*)$, а при любом $\widetilde{\mu}$, $0 \leqslant \widetilde{\mu} < \widetilde{\mu}^*$ лежит внутри $\Omega(\widetilde{\mu})$. Это означает, что существует $c^* \in S$, такое, что $x(T,u(c^*,\widetilde{\mu}^*)) = 0$.

Рассмотрим отображение $\mathfrak{F}_\mu : S \to \partial\Omega(\mu)$. Пусть $\mu_1 > \mu_2 \geqslant 0$. Отображения $\mathfrak{N}_{\mu_1\mu_2} : \partial\Omega(\mu_1) \to \partial\Omega(\mu_2)$ и $\mathfrak{N}_{\mu_2\mu_1} : \partial\Omega(\mu_2) \to \partial\Omega(\mu_1)$ удовлетворяют соотношениям $\mathfrak{F}_{\mu_2} = \mathfrak{N}_{\mu_1\mu_2}\mathfrak{F}_{\mu_1}$ и $\mathfrak{F}_{\mu_1} = \mathfrak{N}_{\mu_2\mu_1}\mathfrak{F}_{\mu_2}$. Из этих равенств следует, что $\mathfrak{N}_{\mu_1\mu_2}\mathfrak{N}_{\mu_2\mu_1} = I$, где I — тождественное отображение. Поэтому $\mathfrak{N}_{\mu_1\mu_2} = \mathfrak{N}^{-1}_{\mu_2\mu_1}$. Это означает, что отображение $\mathfrak{N}_{\mu_1\mu_2}$ является изоморфизмом. Следовательно, и отображение $\mathfrak{N}_{0\widetilde{\mu}^*}$ — изоморфизм. Тогда точке $0 \in \partial\Omega(\widetilde{\mu}^*)$ соответствует единственная точка, лежащая на границе области достижимости $\mathfrak{R}(T)$. Тем самым показано, что принцип максимума Понтрягина для поставленной задачи является и достаточным условием оптимальности.

Работа выполнена при поддержке РФФИ, проект № 10–01–00035, и Сибирского отделения РАН, проект № 85.

Список литературы

[1] *Wang J. R., Xiang X., Wei W.* The constructive approach on the time optimal control of system governed by nonlinear equations on Banach spaces // Electronic J. Quantative Theory of Differential Equations. — 2009. — V. 45. — P. 1–10.

[2] *Шевченко Г. В.* Метод нахождения оптимального по минимуму расхода ресурсов управления для нелинейных стационарных систем // Автоматика и телемеханика. — 2009. — Т. 70, № 4. — С. 119–130.

[3] *Понтрягин Л. С., Болтянский В. Г., Гамкрелидзе Р. В., Мищенко Е. Ф.* Математическая теория оптимальных процессов. — М.: Физматгиз, 1983.

Свойства минимальных нумераций корневых ордеревьев с выделенной вершиной

Д. С. Шелухин

dmshell@rambler.ru

Нижегородский государственный университет им. Н. И. Лобачевского

Постановка задачи

Задача поиска минимальной нумерации графа. Пусть $G(V, E)$ — произвольный граф, содержащий n вершин; $A = a_1, a_2, \ldots, a_n$ — множество из n натуральных чисел. Нумерацией графа $G(V, E)$ будем называть взаимно-однозначное соответствие $\varphi : V \to A$, где множество A — нумерующая последовательность графа. При нумерации φ каждой вершине $v_i \in V$ сопоставляется номер

$\varphi(v_i)$, а каждому ребру $e = (v_i, v_j) \in E$ — число $\Delta_e^\varphi = |\varphi(v_i) - \varphi(v_j)|$ (длина ребра (v_i, v_j) при нумерации φ).

Длиной графа $G(V, E)$ будем называть величину $\Delta^\varphi G = \sum_{e \in E} \Delta_e^\varphi$. Нумерации, на которых достигается $\min\limits_{\varphi} \Delta^\varphi G = \Delta(G)$, будем называть минимальными нумерациями.

Нумерующие последовательности. Пусть $A = \{a_1, a_2, \ldots, a_n\}$, $a_i < a_{i+1}$, $i = 1, 2, \ldots, n-1$, — произвольная нумерующая последовательность. Каждое целое число, заключенное между a_1 и a_n и не входящее в A, будем называть пробелом. Общее количество пробелов в A обозначим через $P(A) = (a_n - a_1) - (n-1)$. Если $P(A) = 0$, то нумерующая последовательность называется сплошной. В [1] приводится доказательство следующей леммы:

Лемма 1. *Для любой нумерации $\varphi : V \to A$, $P(A) > 0$ графа $G(V, E)$ и любой $\widetilde{\varphi} : V \to \widetilde{A}$, $P(\widetilde{A}) = 0$, сохраняющей порядок нумерации φ (вершины $\varphi^{-1}(v_i)$ и $\widetilde{\varphi}^{-1}(v_i)$ совпадают при любом $i = \overline{1, n}$), имеет место $\Delta^\varphi G \geqslant \Delta^{\widetilde{\varphi}} G + P(A)$.*

Нумерации ориентированных графов. Нумерации с выделенной вершиной. Для ориентированных графов вводится понятие допустимости нумерации. Чтобы нумерация была допустимой, требуется, чтобы номера вершин вдоль всех путей монотонно возрастали. Очевидно, что не существует допустимых нумераций в графах, имеющих ориентированные циклы.

Рассмотрим класс однокорневых ордеревьев. Задача поиска минимальной нумерации для этого класса графов приведена в [1]. Выделим в однокорневом ордереве $t(V, E)$ некоторую вершину $v^* \in V$ и рассмотрим нумерацию, минимизирующую функционал

$$\Delta^\varphi t + \varphi(v^*), \tag{1}$$

где $\varphi(v^*)$ — это номер, присвоенный выделенной вершине при нумерации φ, t — однокорневое ориентированное дерево. Нумерацию дерева, минимизирующую функционал (1), будем называть минимальной нумерацией с выделенной вершиной.

Основные свойства нумерации однокорневого ориентированного дерева с выделенной вершиной

Очевидно, что если выделенная вершина v^* совпадает с корнем дерева, то минимум функционала (1) будет достигаться тогда и только тогда, когда минимально $\Delta^{\varphi}t$ (т. к. $\varphi(v^*)$ всегда будет равно первому номеру нумерующей последовательности).

Рассмотрим случай, когда выделенная вершина не является корнем. Пусть для некоторого дерева t существует нумерация φ, минимизирующая (1). Так как нумерация φ соответствует ориентации дерева, то вершина $\varphi^{-1}(n)$ является висячей (где n — последний номер нумерующей последовательности). Рассмотрим путь σ_0 из корня дерева до вершины $\varphi^{-1}(n)$. Поддеревья, порожденные дугами, не принадлежащими σ_0, обозначим через $t_i^{\sigma_0}$, $i = \overline{1,k}$. Пусть также t_1 — это поддерево, содержащее поддеревья $t_1^{\sigma_0}, t_2^{\sigma_0}, \ldots, t_i^{\sigma_0}$, где i — номер поддерева, содержащего вершину v^*. Поддерево t_2 будет содержать в себе поддеревья $t_{i+1}^{\sigma_0}, \ldots, t_k^{\sigma_0}$ и вершину с последним номером из нумерующей последовательности. Справедлива следующая

Лемма 2. *Существует минимальная нумерация φ дерева t с выделенной вершиной v^*, при которой нумерующие последовательности поддеревьев t_1 и t_2 — сплошные.*

Доказательство. Для доказательства леммы рассмотрим некоторую минимальную нумерацию φ дерева t с выделенной вершиной v^*, при которой поддеревья t_1 и t_2 пронумерованы несплошными нумерующими последовательностями. Построим нумерацию $\overline{\varphi}$ так, чтобы она сохраняла порядок нумерации φ на каждом из поддеревьев t_1 и t_2, но нумеровала бы их сплошными нумерующими последовательностями (что всегда можно сделать, если исходная нумерующая последовательность — сплошная). Минимизируемый функционал можно записать в виде

$$\Delta^{\varphi}t + \varphi(v^*) = \Delta^{\varphi}t_1 + \Delta^{\varphi}t_2 + \Delta^{\varphi}e + \varphi(v^*),$$

где e — ребро, связывающее деревья.

Рассмотрим, что произойдёт, когда мы пронумеруем деревья сплошными последовательностями (перейдем от φ к $\overline{\varphi}$). Очевидно, что нумерации поддеревьев t_1 и t_2 улучшатся (по лемме 1), $\varphi(v^*)$ — не ухудшится. Остаётся рассмотреть ребро $e = (v_1^e; v_2^e)$. Здесь $v_1^e \in t_1$, а $v_2^e \in t_2$. Допустим, что при нумерации $\overline{\varphi}$ значение v_1^e уменьшилось на k_1. Обозначим исходную нумерующую последовательность дерева t_1 через $A_1 = \{1, a_2, \ldots, a_{n_1}\}$, а полученную сплошную нумерующую последовательность того же дерева через $B_1 = \{1, 2, \ldots, b_{n_1}\}$. В связи с тем, что мы не меняли порядок нумерации, в обеих последовательностях вершине v_1^e соответствует один и тот же номер с конца (обозначим его x). Если значение v_1^e уменьшилось, значит, изначальный номер a_{n-x+1} был больше, чем $b_{n-x+1} = n - x + 1$, ровно на k_1. Очевидно, что тогда верно: $a_{n_1} \geqslant n_1 + k_1$. Если теперь посчитать количество пробелов $P(A_1)$, то оно, очевидно, будет больше или равно k_1. Аналогично, если значение v_2^e увеличилось на k_2, то в поддереве t_2 было не менее k_2 пробелов. Следовательно, нумерация $\overline{\varphi}$ не хуже нумерации φ. ∎

Лемма 2 позволяет уменьшать размерность задачи, если цепь σ_0 выделена. Введем обозначения: пусть v_0 — корень дерева, а поддеревья $t(v_1), t(v_2), \ldots, t(v_p)$, $p \geqslant 1$, порождены вершинами всех веток, выходящих из v_0 соответственно по ребрам $(v_0, v_1), (v_0, v_2), \ldots, (v_0, v_p)$. Из леммы 2 получаем

Следствие. *Существует такая минимальная нумерация φ дерева t с выделенной вершиной v^*, при которой нумерующие последовательности всех поддеревьев $t(v_1), t(v_2), \ldots, t(v_p)$, $p \geqslant 1$, сплошные, кроме, разве что, поддерева, содержащего вершину v^*, а при выделении цепи в поддереве, содержащем вершину v^*, образующееся при этом поддерево t_2 также нумеруется сплошной нумерующей последовательностью.*

Теперь рассмотрим механизм выделения пути σ_0. Пусть количество вершин в поддереве, содержащем выделенную вершину v^* (пусть это будет поддерево $t(v_i)$), равно n. Предположим, что нумерация поддерева $t(v_i)$ при минимальной нумерации дерева t известна. Выделим внутри этого поддерева путь σ_0 (из корня в лист с максимальным номером) и разобьём его на поддеревья t_1 и t_2. Введём обозначение $n_2 = |t_2|$. Справедлива следующая

Теорема 3. *Существует минимальная нумерация φ дерева t с выделенной вершиной v^*, такая, что цепь σ_0 будет выделена в поддереве $t(v_i)$ тогда и только тогда, когда $\nexists k, |t(v_k)| > n_2$ и $|t(v_k)| > n/2$.*

Для определения значения параметра n_2 можно использовать следующую теорему.

Теорема 4. *Параметр n_2 равен мощности поддерева t_2, полученного при минимальной нумерации поддерева $t(v_i)$ с выделенной вершиной v^*, и является одинаковым для всех минимальных нумераций поддерева $t(v_i)$ с выделенной вершиной v^*.*

Список литературы

[1] *Иорданский М. А.* Задачи размещения графов. I // Комбинаторно-алгебраические методы в прикладной математике. Межвуз. сб. научн. трудов / Под ред. Ал. А. Маркова. — Горький: Изд-во Горьковского ун-та, 1983.

Некоторые оценки сложности двоичных решающих диаграмм

А. Е. Шиганов

df-dx@mail.ru

ООО «Нангейт», Москва

В работе изучается сложность реализации булевых функций двоичными решающими диаграммами, а также упорядоченными k-считывающими двоичными решающими диаграммами при $k \geqslant 4$.

Напомним основные определения. Двоичной решающей диаграммой (Binary Decision Diagram, BDD) от булевых переменных $x_1, \ldots, x_n$ называется ориентированный ациклический граф Σ с одним истоком (входом), в котором каждый сток (выход) помечается 0 или 1, каждой вершине, отличной от выходов, приписана одна из переменных $x_1, \ldots, x_n$ и из неё исходят две дуги с пометками $0, 1$ [1]. Двоичная решающая диаграмма Σ от переменных $x_1, \ldots, x_n$ реализует булеву функцию $f(x_1, \ldots, x_n)$, если для каждого набора $\alpha = (\alpha_1, \ldots, \alpha_n) \in \{0, 1\}^n$ ориентированный путь, который начинается из входа Σ и далее при прохождении через вершину x_i следует по дуге α_i, приводит в выход, помеченный $f(\alpha)$.

Двоичная решающая диаграмма Σ называется упорядоченной k-считывающей (k-OBDD), если произвольный ориентированный путь из входа в выход Σ разбивается на k сегментов, таких, что на каждом сегменте переменные встречаются в одном и том же порядке, одинаковом для всех сегментов и указанных путей, и на каждом сегменте каждая переменная встречается не более одного раза [2].

Сложностью $L(\Sigma)$ BDD Σ называется число её вершин, отличных от выходов. Обозначим через $L^{\mathrm{BDD}}(f)$ (соответственно $L^{k\text{-OBDD}}(f)$) минимальную сложность BDD Σ (соответственно упорядоченной k-считывающей BDD Σ), реализующей f. Определим функции Шеннона

$$L^{\mathrm{BDD}}(n) = \max_{f(x_1,\ldots,x_n)} L^{\mathrm{BDD}}(f), \quad L^{k\text{-}\mathrm{OBDD}}(n) = \max_{f(x_1,\ldots,x_n)} L^{k\text{-}\mathrm{OBDD}}(f).$$

Представление булевых функций с помощью BDD было впервые предложено в [3]. Там же была получена нижняя асимптотическая оценка $2^{n-1}/n$ и верхняя асимптотическая оценка $2^{n+2}/n$ для функции Шеннона $L^{\mathrm{BDD}}(n)$. Позднее в [4] для этой функции была установлена асимптотика $2^n/n$, причём относительная погрешность полученных оценок, т.е. отношение разности между верхней и нижней оценками функции Шеннона к ней самой, составляла $O(1/\sqrt{n})$.

В работе [2] приводятся асимптотические оценки функции Шеннона $L^{\mathrm{BDD}}(n)$ с относительной погрешностью вида $o(\log n/n)$. Также в [2] установлено асимптотическое значение $2^n/n$ для функции Шеннона $L^{k\text{-}\mathrm{OBDD}}(n)$ при $k \geqslant 2$, а верхняя и нижняя оценки имеют относительную погрешность вида $o(\log n/n)$.

Основные результаты настоящей работы представлены в следующих теоремах.

Теорема 1. *При $n = 1, 2, \ldots$ справедливы оценки*

$$\frac{2^n}{n}\left(1 - O\left(\frac{1}{n}\right)\right) \leqslant L^{\mathrm{BDD}}(n) \leqslant \frac{2^n}{n}\left(1 + \frac{\log\log n + O(1)}{n}\right).$$

Теорема 2. *Для любого натурального $k \geqslant 4$ при $n = 1, 2, \ldots$ справедливы оценки*

$$\frac{2^n}{n}\left(1 - O\left(\frac{1}{n}\right)\right) \leqslant L^{k\text{-}\mathrm{OBDD}}(n) \leqslant \frac{2^n}{n}\left(1 + \frac{\log\log n + O(1)}{n}\right).$$

Отметим, что полученные оценки функций Шеннона $L^{\mathrm{BDD}}(n)$ и $L^{k\text{-}\mathrm{OBDD}}(n)$ при $k \geqslant 4$ имеют относительную погрешность вида $\dfrac{\log\log n + O(1)}{n}$.

Работа выполнена при поддержке РФФИ, грант № 09-01-00817-а.

Список литературы

[1] *Wegener I.* The complexity of Boolean functions. — Teubner (Stuttgart)/Wiley (Chichester), 1987.

[2] *Ложкин С. А.* О сложности реализации произвольных булевых функций в некоторых классах BDD // Тр. Межд. школы-семинара «Дискретная математика и математическая кибернетика» (Ратмино, 31 мая — 3 июня 2001 г.). — М.: МАКС Пресс, 2001. — С. 18–19.

[3] *Lee C. Y.* Representation of switching circuits by binary-decision programs // Bell. Sys. Tech. J. — 1959. — V. 38. — P. 985–999.

[4] *Кузьмин В. А.* Оценка сложности реализации булевых функций программами простого типа // Методы дискретного анализа. Вып. 29. — 1976. — С. 11–39.

Декомпозиция недоопределенных данных

Л. А. Шоломов

sholomov@isa.ru

Институт системного анализа РАН, Москва

Задан алфавит $A_0 = \{a_0, a_1, \ldots, a_{m-1}\}$ *основных символов*. Пусть $M = \{0, 1, \ldots, m-1\}$ и каждому непустому $T \subseteq M$ сопоставлен символ a_T. Символы алфавита $A = \{a_T,\ T \subseteq M\}$ называются *недоопределенными*, и *доопределением* символа $a_T \in A$ считается всякий основной символ a_i, $i \in T$. Символ a_M, доопределимый любым основным символом, называется *неопределенным* и обозначается $*$.

Источник X, порождающий символы $a_T \in A$ независимо с вероятностями p_T, будем называть *недоопределенным источником*. Величину

$$\mathcal{H}(X) = \min_Q \Big\{ -\sum_{T \subseteq M} p_T \log \sum_{i \in T} q_i \Big\},$$

где $\log x = \log_2 x$, минимум берется по наборам $Q = (q_i,\ i \in M)$, $q_i \geqslant 0$, $\sum_{i \in M} q_i = 1$, назовем *энтропией источника* X. Для недоопределенных данных эта величина играет роль энтропии Шеннона [1].

Источники X и Y будем называть (информационно) *равносильными* и записывать $X \approx Y$, если для любого источника Z выполнено $\mathcal{H}(XZ) = \mathcal{H}(YZ)$. Будем говорить, что источник X (информационно) *не слабее* Y (Y *не сильнее* X), и записывать $X \gtrsim Y$, если $XY \approx X$. Можно показать, что $X \approx Y$ тогда и только тогда, когда $X \gtrsim Y$ и $X \lesssim Y$. Существует эффективный алгоритм проверки соотношений $X \approx Y$ и $X \gtrsim Y$. (Алгоритм считается эффективным, если его трудоемкость оценивается сверху полиномом от размера исходных данных.)

Скажем, что X *разлагается в произведение* источников $X_1, ..., X_k$, если $X \approx X_1 \dots X_k$. Недоопределенный источник с алфавитом $\{0, 1, *\}$ называется *простым.* Источник X, который разлагается в произведение простых источников $X_1, \dots, X_k$, назовем *разложимым*, а само произведение $X_1 \dots X_k$ будем называть *декомпозицией* источника X.

Пусть задан некоторый класс $\mathbf{S}$ источников. Источник $Y \in \mathbf{S}$ будем называть (нижней) *аппроксимацией источника X в классе* $\mathbf{S}$, если $Y \lesssim X$ и для всякого $Z \in \mathbf{S}$, такого, что $Z \lesssim X$, выполнено $Y \gtrsim Z$. Очевидно, что аппроксимирующий источник Y, если он существует, определен однозначно с точностью до равносильности.

Обозначим через $\mathbf{D}$ класс всех разложимых источников. Если источник Y аппроксимирует X в этом классе, то про всякую декомпозицию источника Y будем говорить, что она аппроксимирует источник X. Все декомпозиции, аппроксимирующие заданный источник X, равносильны (т. е. равносильны произведения $X_1 \dots X_s$).

Утверждение 1. *Для всякого источника X существует аппроксимирующая декомпозиция $X_1 \dots X_s$.*

Замечание. Двойственно к понятию нижней аппроксимации можно определить верхнюю аппроксимацию. В качестве приближения мы рассматриваем только нижнюю аппроксимацию, поскольку верхняя аппроксимация в классе $\mathbf{D}$ существует не всегда.

Возникают задачи алгоритмического характера, связанные с построением декомпозиций (точных или приближенных) и с их упрощением — уменьшением числа s источников, присутствующих в декомпозициях. Под алфавитом A источника X будем понимать множество всех символов a_T, для которых $p_T > 0$. Некоторый анализ показывает, что задачи декомпозиции и аппроксимации источника X, а также упрощения этих представлений фактически не зависят от величин вероятностей p_T (ненулевых) и эти задачи могут быть переформулированы применительно к алфавиту A недоопределенных символов. В терминах алфавитов они и решаются в данной работе, но полученные результаты будем формулировать для источников.

Утверждение 2. *Существует эффективный алгоритм построения по источнику X аппроксимирующей декомпозиции $X_1 \dots X_s$, в которой число s простых источников не превосходит мощности $|A|$ алфавита A.*

Если источник X разложим, то произведение $X_1 \dots X_s$ является его декомпозицией,

Рассмотрим теперь задачу упрощения декомпозиций. Под декомпозицией будем понимать произвольное произведение $X_1 \dots X_s$ простых источников. В частности, оно могло возникнуть как декомпозиция или аппроксимация некоторого источника X. Источник X_i, $i = 1, \dots, s$, назовем *устранимым*, если

$$X_1 \dots X_s \precsim X_1 \dots X_{i-1} X_{i+1} \dots X_s.$$

Утверждение 3. *Существует эффективный алгоритм проверки устранимости.*

Используя его, можно путем последовательного удаления устранимых источников эффективно построить *неизбыточную* декомпозицию, из которой нельзя удалить ни одного источника без потери равносильности. Задача о сложности неизбыточной декомпозиции (возникающей из заданной декомпозиции $X_1 \dots X_s$) состоит в том, чтобы узнать, можно ли из $X_1 \dots X_s$ выбрать не более заданного числа t источников, образующих равносильную декомпозицию.

Утверждение 4. *Задача о сложности неизбыточной декомпозиции NP-полна.*

Существуют примеры того, что в рамках заданной системы источников нельзя добиться хорошего результата и существенное уменьшение сложности декомпозиций связано с использованием других источников. В связи с этим рассмотрим вопрос перехода от декомпозиции $X_1 \dots X_s$ к равносильной декомпозиции $Y_1 \dots Y_t$, использующей, вообще говоря, другие источники. Скажем, что источник Y *выразим* через $X_1, \dots, X_s$, если $X_1 \dots X_s Y \precsim X_1 \dots X_s$. Свойство выразимости может быть эффективно проверено с помощью утверждения 3, поскольку источник Y выразим через $X_1, \dots, X_s$ тогда и только тогда, когда Y устраним из $X_1 \dots X_s Y$.

Утверждение 5. *Существует эффективный алгоритм проверки равносильности заданных декомпозиций $X_1 \dots X_s$ и $Y_1 \dots Y_t$.*

Это следует из того, что $X_1 \dots X_s \approx Y_1 \dots Y_t$ тогда и только тогда, когда каждый Y_j выразим через $X_1, \dots, X_s$, а каждый X_i — через $Y_1, \dots, Y_t$.

Если новая декомпозиция $Y_1 \dots Y_t$ не задана, то для ее формирования может быть использован найденный в работе метод порождения всех источников Y, выразимых через заданные источники $X_1, \dots, X_s$.

Работа выполнена при поддержке ОНИТ РАН по программе фундаментальных исследований (проект «Теория и методы эффективного использования недоопределенных данных»).

Список литературы

[1] *Шоломов Л. А.* Элементы теории недоопределенной информации // Прикладная дискретная математика. Приложение № 2. — 2009. — С. 18–42.

[2] *Шоломов Л. А.* Преобразование нечетких данных с сохранением информационных свойств // Дискретный анализ и исследование операций. Сер. 1. — 2005. — Т. 12, № 3. — С. 85–104.

О сложности реализации предикатов из некоторых классов предикатными схемами

М. С. Шуплецов

miklesh@shupletsov.ru

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Рассматривается задача синтеза [1, 2] и получения асимптотических оценок различной степени точности для сложности реализации предикатов из некоторых классов при помощи предикатных схем (см., например, [3, 4]) в базисах специального вида. Напомним, что для инвариантных классов булевых функций (см., например, [5]) в традиционных классах управляющих систем (схемы из функциональных элементов, контактные схемы и др.) поведение функции Шеннона для сложности указанных управляющих систем на уровне асимптотики было установлено О. Б. Лупановым [2]. Кроме того, для некоторых классов функций, связанных с автоматными языками, С. А. Ложкиным [6] были получены оценки высокой степени точности для сложности функции Шеннона в классе контактных схем и схем из функциональных элементов, имеющих ограниченную глубину ветвления.

Пусть $\mathcal{X} = \{x_1, \ldots, x_n, \ldots\}$ (соответственно $\mathcal{Y} = \{y_1, \ldots, y_p, \ldots\}$) — счетный упорядоченный алфавит полюсных (соответственно внутренних) переменных. *Базисом* назовём произвольную полную (см., например, [3]) конечную систему предикатов $\mathfrak{B} = \{\pi_1, \ldots, \pi_b\}$, в которой предикат π_i, $i = 1, \ldots, b$, существенно зависит от k_i переменных из $\mathcal{X}$ и ему сопоставлено положительное действительное число, характеризующее «вес» этого предиката.

Напомним, что предикатная схема в базисе $\mathfrak{B}$ представляет собой двудольный граф, у которого все вершины одной доли помечены символами базисных предикатных элементов из множества $\mathfrak{B}$, а вершины другой доли — символами переменных из алфавита $\mathcal{X} \cup \mathcal{Y}$. Функционирование предикатного элемента с k полюсами задается его характеристической функцией от k переменных, связанных с этими полюсами, и определяется тем, что элемент находится в допустимом состоянии, если данная функция равна 1. Схема находится в допустимом состоянии на некотором наборе значений полюсных переменных тогда и только тогда, когда существует набор значений внутренних переменных, такой, что все предикатные элементы, из которых построена схема, находятся в допустимых состояниях. При этом предполагается, что предикатная схема Σ реализует предикат π от её полюсных переменных, если множество допустимых наборов π совпадает с множеством тех наборов, на которых Σ находится в допустимом состоянии. Более подробное описание указанной модели представлено в работах [3, 4].

Пусть $\Pi_2(n)$ — множество всех булевых предикатов от n переменных $x_1, \ldots, x_n$, а Π_2 — множество всех булевых предикатов от переменных из $\mathcal{X}$. Для множества (класса) предикатов $Q \subseteq \Pi_2$ и натурального n через $Q(n)$ будем обозначать множество $Q \cap \Pi_2(n)$. При этом множество Q и связанную с ним последовательность $Q(1)$, $Q(2)$, … будем называть *классом предикатов*. Для класса предикатов Q введем следующую функцию:

$$\sigma_Q(n) = \frac{\log |Q(n)|}{2^n},$$

причём из определения следует, что $0 \leqslant \sigma_Q(n) \leqslant 1$ для всех n.

Пусть $\mathcal{U}_{\mathfrak{B}}$ — класс предикатных схем, построенных в полном предикатном базисе $\mathfrak{B}$. Тогда под сложностью $L(\Sigma)$ предикатной схемы Σ, $\Sigma \in \mathcal{U}_{\mathfrak{B}}$, понимается сумма «весов» её предикатных элементов, а под сложностью $L_{\mathfrak{B}}(\pi)$ предиката π — минимальная из сложностей реализующих его схем в базисе $\mathfrak{B}$. Для инвариантного класса Q введем обычным образом функцию Шеннона $L_{\mathfrak{B}}(Q(n))$ для сложности предикатных схем в классе $\mathcal{U}_{\mathfrak{B}}$ как максимальную сложность $L_{\mathfrak{B}}(\pi)$, $\pi \in Q(n)$. Отметим, что в работах [3, 4] для произвольного базиса $\mathfrak{B}$ на основе некоторых специальных характеристик предикатов введен приведенный вес $\rho_{\mathfrak{B}}$ и обобщенный приведенный вес $\widehat{\rho}_{\mathfrak{B}}$, которые определены для произвольного полного базиса, и $\rho_{\mathfrak{B}} = \widehat{\rho}_{\mathfrak{B}}$ для почти всех базисов.

Множество предикатов $Q \subseteq \Pi_2(n)$ назовем *инвариантным классом предикатов*, если множество Q замкнуто относительно операций добавления и изъятия фиктивных переменных, операции переименования полюсов без отождествления и операции подстановки булевых констант. Отметим, что введенное понятие является естественным аналогом понятия инвариантного класса булевых функций (см., например, [5]).

Лемма. *Пусть Q — инвариантный класс предикатов. Тогда существует предел*

$$\sigma_Q = \lim_{n\to\infty} \sigma_Q(n) = \lim_{n\to\infty} \frac{\log|Q(n)|}{2^n}.$$

При этом число σ_Q, называемое характеристикой класса Q, удовлетворяет неравенству $0 \leqslant \sigma_Q \leqslant 1$.

Инвариантный класс предикатов Q будем называть *нулевым*, если его характеристика $\sigma_Q = 0$.

Теорема 1. *Если $\mathfrak{B}$ — произвольный полный предикатный базис, такой, что $\rho_{\mathfrak{B}} = \widehat{\rho}_{\mathfrak{B}}$, то для любого ненулевого инвариантного класса Q выполнено асимптотическое равенство*

$$L_{\mathfrak{B}}(Q(n)) \sim \sigma_Q \rho_{\mathfrak{B}} \frac{2^n}{n}.$$

Пусть, далее, Λ — язык (множество слов) в алфавите $\{0, 1\}$, а $\Lambda(n)$ — множество тех слов, длина которых не более n. Под сложностью $L_{\mathfrak{B}}(\alpha)$ слова α, $\alpha \in \Lambda$, понимается минимальная сложность предикатных схем, реализующих такие предикаты, столбец значений характеристической функции которых имеет префикс α, а функция Шеннона $L_{\mathfrak{B}}(\Lambda(n))$ равна максимальной сложности $L_{\mathfrak{B}}(\alpha)$, $\alpha \in \Lambda(n)$. Язык Λ является экспоненциальным, если мощность множества его слов длины не более n равна $2^{(\sigma+o(1))n}$, $0 < \sigma = \sigma_\Lambda \leqslant 1$. Известно (см., например, [7]), что любой язык Λ, распознаваемый конечным автоматом, является либо полиномиальным, либо экспоненциальным.

Рассмотрим, далее, специальный класс базисов, который является подклассом обобщенно-проводящих базисов, введенных в работе [3]. Пусть $\varphi(x_1, \ldots, x_n)$, $n \geqslant 3$, — некоторый предикат с двумя выделенными переменными (не ограничивая общности рассуждений, будем считать, что это переменные x_1 и x_2) и для любого i, $i = 3, \ldots, n$, существует такой набор $\alpha^i = (\alpha_3, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_n)$, что характеристическая функция $\chi_\varphi(x_1, x_2, \alpha_3, \ldots, \alpha_{i-1}, x_i, \alpha_{i+1}, \ldots, \alpha_n)$ равна $(x_1 \oplus x_i \oplus \sigma_4) \vee x_2^{\sigma_2}$, а также существует набор $\beta = (\beta_3, \ldots, \beta_n)$, такой, что $\chi_\varphi(x_1, x_2, \beta_3, \ldots, \beta_n) = x_1^{\sigma_5} \vee x_2^{\sigma_6}$, где $\sigma_1, \ldots, \sigma_6$ — булевы константы. Предикат φ, обладающий указанными свойствами, будем называть *строго проводящим*. Полный предикатный базис $\mathfrak{B}$ назовём *строго обобщенно-проводящим*, если в множестве базисных предикатов, на которых достигается приведенный вес $\rho_{\mathfrak{B}}$ базиса $\mathfrak{B}$, найдётся строго проводящий предикат φ, имеющий максимальное число полюсов среди всех предикатов указанного множества.

Теорема 2. *Если $\mathfrak{B}$ — строго обобщенно-проводящий базис, то для любого экспоненциального языка Λ, распознаваемого конечным автоматом, справедлива следующая оценка высокой степени точности:*

$$L_{\mathfrak{B}}(\Lambda(n)) = \sigma_\Lambda \rho_{\mathfrak{B}} \frac{2^n}{n} \left(1 + \frac{\left(2 + \frac{1}{k_{\mathfrak{B}}-1}\right) \log_2 n \pm O\left(1\right)}{n}\right),$$

где $k_{\mathfrak{B}}$ — максимальное число полюсов у базисных предикатов, на которых достигается приведенный вес $\rho_{\mathfrak{B}}$ базиса $\mathfrak{B}$.

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг.

Список литературы

[1] *Shannon C. E.* The synthethis of two-terminal switching circuits // Bell Syst. Techn. J. — 1949. — V. 28, № 1. — P. 59–98.

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

[3] *Шуплецов М. С.* Оценки высокой степени точности для сложности предикатных схем в некоторых базисах // Учен. зап. Казан. ун-та. Сер. Физ.-матем. науки. — 2009. — Т. 151, кн. 2. — С. 173–184.

[4] *Шуплецов М. С.* Об одном подходе к синтезу предикатных схем на основе обобщенных переменных // Вестник Московского университета. Сер. 15. Вычислительная математика и кибернетика. — 2010. — № 4. — С. 24–30.

[5] *Яблонский С. В.* О классах функций алгебры логики, допускающих простую схемную реализацию // Успехи математических наук. — 1957. — Т. 12. Вып. 6. — С. 189–196.

[6] *Ложкин С. А.* Асимптотические оценки высокой степени точности для сложности функций, связанных с автоматными языками // Проблемы теоретической кибернетики. Тезисы докладов XII Международной конференции (Нижний Новгород, 17–22 мая 1999 г.). Часть II / Под редакцией О. Б. Лупанова. — М.: Изд-во механико-математического факультета МГУ, 1999. — С. 138.

[7] *Хомский Н., Миллер Д.* Языки с конечным числом состояний // Кибернетический сборник. Вып. 4. — М.: Изд-во ин. литературы, 1962. — С. 233–255.

О скорости сходимости квазигрупповых свёрток распределений вероятностей

А. Д. Яшунский

alexey.yashunsky@gmail.com

Институт прикладной математики им. М. В. Келдыша РАН, Москва

Как отмечено в обзоре [1], в последнее время увеличивается интерес к вопросам применения квазигрупп в криптологии, в частности — для построения поточных шифров. Использование квазигрупповых операций при поточном шифровании позволяет получать зашифрованные сообщения, в которых распределение символов близко к равномерному. Специальный вид используемых преобразований позволяет свести их исследование к рассмотрению подходящих цепей Маркова.

Подобное сведение используется также и при исследовании весьма близкой задачи о случайных блужданиях в конечной группе, подробно описанной, например, в [2]. При исследовании групп для сведения к марковским цепям существенно используется ассоциативность умножения в группе.

Однако, как будет показано далее, ни ассоциативность умножения, ни специальный вид формул не требуются для того, чтобы установить сходимость распределений значений квазигрупповой формулы к равномерному распределению.

Пусть $Q = \{1, 2, \ldots, q\}$ — множество, на котором задана операция квазигруппового умножения $i \cdot j = k$ (где $i, j, k \in Q$), имеющая однозначно определённые обратные операции правого и левого деления: $i = k/j$ и $j = i\backslash k$ (подробнее см. [3]). Будем рассматривать множество распределений вероятностей на квазигруппе Q:

$$\mathcal{P}^q = \{(p_1, \ldots, p_q) : \sum_{i=1}^{q} p_i = 1 \text{ и все } p_i \geqslant 0\}$$

и операцию *свёртки* распределений из $\mathcal{P}^q$. Для распределений $u, v \in \mathcal{P}^q$ определим свёртку $u * v \in \mathcal{P}^q$ следующим образом:

$$(u * v)_i = \sum_{j=1}^{q} u_j v_{j\backslash i}.$$

Содержательно такая свёртка выражает распределение значения результата при квазигрупповом умножении двух случайных элементов из Q, имеющих распределения u и v, соответственно.

Далее мы будем исследовать распределения, получающиеся в результате многократного применения операции свёртки к некоторому начальному распределению.

Следуя [4], для $u \in \mathcal{P}^q$ обозначим через $(u_{[1]}, u_{[2]}, \ldots, u_{[q]})$ распределение, получающееся перестановкой компонент распределения u таким образом, что

$$u_{[1]} \geqslant u_{[2]} \geqslant \ldots \geqslant u_{[q]}.$$

Кроме того, обозначим $\delta_u = u_{[1]} - u_{[q]}$.

Используя введённые обозначения, сформулируем одно из свойств свёртки:

Теорема 1 (Харди, Литтлвуд, Пойя [5]). *Пусть $u, v \in \mathcal{P}^q$. Тогда для любого $i = 1, \ldots, q$*

$$u_{[1]}v_{[q]}+u_{[2]}v_{[q-1]}+\ldots+u_{[q]}v_{[1]} \leqslant (u*v)_i \leqslant u_{[1]}v_{[1]}+u_{[2]}v_{[2]}+\ldots+u_{[q]}v_{[q]}.$$

Дополнительно (и это следует, в том числе, непосредственно из теоремы) имеет место неравенство

$$(u * v)_{[q]} \geqslant \max\{u_{[q]}, v_{[q]}\}. \tag{1}$$

Теорема 1 позволяет получить оценку значения δ_{u*v} через значения δ_u и δ_v.

Теорема 2. *Пусть $u, v \in \mathcal{P}^q$. Тогда*

$$\delta_{u*v} \leqslant \min\{(1 - v_{[q]})\delta_u, (1 - u_{[q]})\delta_v\}.$$

Доказательство. По определению, $\delta_{u*v} = (u * v)_{[1]} - (u * v)_{[q]}$. Используя неравенства из теоремы 1, оцениваем

$$\begin{aligned}\delta_{u*v} \leqslant (u_{[1]}v_{[1]} + u_{[2]}v_{[2]} + \ldots + u_{[q]}v_{[q]}) - \\ - (u_{[1]}v_{[q]} + u_{[2]}v_{[q-1]} + \ldots + u_{[q]}v_{[1]}) = \\ = u_{[1]}(v_{[1]} - v_{[q]}) + u_{[2]}(v_{[2]} - v_{[q-1]}) + \ldots + u_{[q]}(v_{[q]} - v_{[1]}) = \\ = (u_{[1]} - u_{[q]})(v_{[1]} - v_{[q]}) + (u_{[2]} - u_{[q-1]})(v_{[2]} - v_{[q-1]}) + \ldots\end{aligned}$$

Очевидно, что для каждого $i = 1, \ldots, q$ выполнено

$$u_{[i]} - u_{[q-i+1]} \leqslant \delta_u, \quad v_{[i]} - v_{[q-i+1]} \leqslant \delta_v,$$

поэтому

$$\delta_{u*v} \leqslant (v_{[1]} + v_{[2]} + \ldots - v_{[\lfloor (q+1)/2 \rfloor]} - \ldots - v_{[q]})\delta_u \leqslant (1 - v_{[q]})\delta_u.$$

Аналогично, $\delta_{u*v} \leqslant (1 - u_{[q]})\delta_v$, откуда вытекает утверждение теоремы. ∎

Пусть $\pi \in \mathcal{P}^q$ — некоторое начальное распределение. Построим по индукции множества распределений D_k следующим образом. Положим $D_0 = \{\pi\}$ и далее

$$D_{k+1} = \{u * v : u \in D_r, v \in D_s, \max\{r, s\} = k\}.$$

Содержательно множество D_k соответствует распределениям, которые могут быть получены из начального распределения π в результате свёртки «глубины» k.

Теорема 3. *Пусть π — начальное распределение и D_k — соответствующие множества распределений. Тогда для любого $w \in D_k$ имеет место неравенство*

$$\delta_w \leqslant (1 - \pi_{[q]})^k.$$

Доказательство. Из неравенства (1) и определения множеств D_k вытекает, что для любого $w \in D_k$ $w_{[q]} \geqslant \pi_{[q]}$. Это соотношение, неравенство из теоремы 2 и определение множеств D_k вместе позволяют получить доказываемое утверждение. ∎

Утверждение теоремы является нетривиальным только в том случае, если $\pi_{[q]} > 0$, в противном случае неравенство в теореме превращается в $\delta_w \leqslant 1$. Если же неравенство $\pi_{[q]} > 0$ выполнено, то теорема позволяет утверждать, что с ростом k величины δ_w для $w \in D_k$ стремятся к нулю, а сами распределения w при этом приближаются к равномерному распределению $(1/q, \ldots, 1/q) \in \mathcal{P}^q$, причём, в силу определения δ_w, выполнено

$$\max_i \left| w_i - \frac{1}{q} \right| \leqslant (1 - \pi_{[q]})^k.$$

Таким образом, отклонение распределения w от равномерного оценивается через «глубину» k распределения w и начальное распределение π.

Автор выражает благодарность О. М. Касим-Заде за полезные обсуждения и внимание к работе.

Работа выполнена при финансовой поддержке РФФИ и программы фундаментальных исследований ОМН РАН «Алгебраические и комбинаторные методы математической кибернетики и информационные системы нового поколения».

Список литературы

[1] *Şcerbacov V.* Quasigroups in cryptology // Computer Science Journal of Moldova. — 2009. — V. 17, № 2 (50). — P. 193–228.

[2] *Saloff-Coste L.* Random walks on finite groups // Probability on discrete structures. Encyclopaedia Math. Sci., 110 / Ed. H. Kesten. — Berlin: Springer, 2004. — P. 263–346.

[3] *Белоусов В. Д.* Основы теории квазигрупп и луп. — М.: Наука, 1967.

[4] *Маршалл А., Олкин И.* Неравенства. Теория мажоризации и ее приложения. — М.: Мир, 1983.

[5] *Hardy G. H., Littlewood J. E., Pólya G.* Inequalities. — Cambridge: University Press, 1934.

Авторский указатель

Проблемы теоретической кибернетики

Материалы XVI Международной конференции

(Нижний Новгород, 20–25 июня 2011 г.)

Оформление обложки: А. А. Пережогин

Формат 60×84 $^1/_{16}$. Бумага офсетная. Печать цифровая.
Усл.-изд. л. 34,2. Усл. печ. л. 33,6. Тираж 300 экз. Заказ №

Издательство Нижегородского госуниверситета им. Н. И. Лобачевского,
603950, Н. Новгород, пр. Гагарина, 23.

Редакционно-издательское управление (РИУ)
Нижегородского госуниверситета им. Н. И. Лобачевского,
603950, Н. Новгород, пр. Гагарина, 23.