Math-Net.Ru

All Russian mathematical portal

N. G. Parvatov, Finding the subsets of variables of a partial Boolean function which are sufficient for its implementation in the classes defined by predicates, *Diskretn. Anal. Issled. Oper.*, 2020, Volume 27, Issue 1, 110–126

DOI: <https://doi.org/10.33048/daio.2020.27.664>

Use of the all-Russian mathematical portal Math-Net.Ru implies that you have read and agreed to these terms of use

<http://www.mathnet.ru/eng/agreement>

Download details:

IP: 54.224.187.69

April 26, 2023, 12:56:39

НАХОЖДЕНИЕ МНОЖЕСТВ ПЕРЕМЕННЫХ ЧАСТИЧНОЙ
БУЛЕВОЙ ФУНКЦИИ, ДОСТАТОЧНЫХ ДЛЯ ЕЁ
РЕАЛИЗАЦИИ В КЛАССАХ, ЗАДАВАЕМЫХ ПРЕДИКАТАМИ

Н. Г. Парватов

Томский государственный университет,
пр. Ленина, 36, 634050, Томск, Россия

E-mail: ngparvatov@yandex.ru

Аннотация. Для заданного класса K частичных булевых функций и произвольной частичной булевой функции f от n переменных множество U её переменных называется достаточным для реализации в классе K , если в этом классе найдётся функция, доопределяющая f и зависящая от переменных из множества U . В статье рассматривается задача нахождения всех множеств, достаточных для реализации функции f в классе K . Для ряда классов, задаваемых отношениями, предлагаются алгоритмы, решающие указанную задачу со сложностью $O(2^n n^2)$ битовых операций. В том числе построены алгоритмы указанной сложности для классов P_2^* всех частичных булевых функций и M_2^* всех частичных монотонных функций. Предлагаемые алгоритмы основаны на использовании преобразований Уолша — Адамара и Мёбиуса. Библиогр. 21.

Ключевые слова: частичная булева функция, достаточное множество переменных, преобразование Уолша — Адамара, преобразование Мёбиуса.

Введение

Пусть $E_2 = \{0, 1\}$ и $E_2^* = \{0, 1, *\}$. Функция $f: E_2^n \rightarrow E_2^*$ называется *частичной булевой функцией* от n переменных; при этом число n может принимать значения, равные $0, 1, 2, \dots$. Пусть K — класс частичных булевых функций и f — некоторая частичная булева функция от n переменных. Множество U , состоящее из натуральных чисел $u_1, \dots, u_k$, где $1 \leq u_1 < \dots < u_k \leq n$, назовём *достаточным для реализации функции f в классе K* , если в нём найдётся функция g от k переменных такая, что для любых значений $x_1, \dots, x_n$ из множества E_2 выполняется условие

$$(1) \quad g(x_{u_1}, \dots, x_{u_k}) = f(x_1, \dots, x_n) \text{ или } f(x_1, \dots, x_n) = *.$$

В этом случае говорим также, что функция g реализует функцию f .

Далее рассматривается задача нахождения системы множеств, достаточных для реализации в классе K частичной булевой функции от n переменных. Нас будет интересовать битовая сложность алгоритма, решающего данную задачу для произвольной частичной булевой функции от n переменных. При этом предполагается, что класс K фиксирован, функция на входе алгоритма задаётся вектором своих значений и система множеств на выходе определяется вектором значений её характеристической функции.

Уточним понятия вектора значений и характеристической функции. Будем считать, что для каждого конечного множества A , содержащего k элементов, определены две инъективные функции

$$C_A: A \rightarrow E_2^m, \quad D_A: E_k \rightarrow A,$$

где $m = \lceil \log_2 k \rceil$ и $E_k = \{0, \dots, k-1\}$. Вектором значений дискретной функции $f: A \rightarrow B$ будем называть двоичный набор

$$(C_B(f(D_A(0))), \dots, C_B(f(D_A(k-1)))).$$

Так, вектор значений частичной булевой функции f от n переменных есть набор

$$(C_{E_2^*}(f(D_{E_2^n}(0))), \dots, C_{E_2^*}(f(D_{E_2^n}(2^n-1)))).$$

Всюду далее предполагается, что дискретные функции, появляющиеся на входах и выходах алгоритмов, задаются векторами своих значений.

Характеристической функцией системы B подмножеств множества чисел $\{1, \dots, n\}$ будем называть булеву функцию

$$\chi: E_2^n \rightarrow E_2, \quad x \mapsto \chi(x) = [\{i \mid x_i = 1\} \in B].$$

Иными словами, функция χ на произвольном наборе $x = (x_1, \dots, x_n)$ из множества E_2^n принимает значение $\chi(x)$, совпадающее с логическим значением $\{i \mid x_i = 1\} \in B$. В соответствии с определением, сделанным выше, вектором значений характеристической функции χ является двоичный набор

$$(C_{E_2}(\chi(D_{E_2^n}(0))), \dots, C_{E_2}(\chi(D_{E_2^n}(2^n-1)))).$$

Характеристическую функцию системы множеств, достаточных для реализации функции f в классе K , будем обозначать в дальнейшем через $\chi(f, K)$.

Рассматриваемая далее задача нахождения достаточных множеств возникает естественно при решении проблем анализа и синтеза дискретных управляющих систем, как в [1]. Данная задача тесно связана с задачами распознавания свойств функций, заданных векторами значений.

Отметим в связи с этим работы [2–6], содержащие существенные результаты в этом направлении. Так, в [2] предложен метод разложения для построения алгоритмов распознавания, получены алгоритмы сложности $O(2^n \sqrt{n} \log n)$ для распознавания частичной монотонности и поляризуемости, алгоритм сложности $O(2^n)$ для распознавания доопределимости до линейной функции. В [3] построен алгоритм линейной сложности распознавания полиномиальности для функций над кольцом классов вычетов по составному модулю. В [4–6] развит метод ступенчатых билинейных алгоритмов для задач распознавания, построены алгоритмы для ряда классов.

Задача нахождения достаточных множеств поставлена в [7]. Предложенный там метод позволяет строить нетривиальные по сложности алгоритмы для ряда классов. В том числе получены алгоритмы сложности $O(3^n)$ для класса P_2^* всех частичных булевых функций и сложности $O(3^n n^{-1/2} \log n)$ для класса M_2^* всех частичных монотонных булевых функций.

В данной статье для рассматриваемой задачи описываются алгоритмы битовой сложности $O(2^n n^2)$, основанные на использовании преобразований Уолша — Адамара и Мёбиуса и применимые для ряда классов, заданных отношениями. В том числе построены алгоритмы, применимые для классов P_2^* и M_2^* . По способу построения предлагаемые алгоритмы близки алгоритмам из [4–6] для распознавания принадлежности дискретной функции классу, заданному отношениями.

Следует заметить, что преобразования Уолша — Адамара и Мёбиуса широко используются в различных областях математики, включая комбинаторику [8, 17], теорию кодирования [9] и криптографию. В том числе преобразование Уолша — Адамара используется при декодировании кодов Рида — Маллера [9–12] и при изучении максимально нелинейных функций [13]. Преобразование Мёбиуса, определяемое в разной общности, лежит в основе важнейшего комбинаторного принципа (принципа обращения Мёбиуса). Это преобразование возникло в работах [14, 15] по теории групп и было систематически обобщено для комбинаторных приложений в [16]. В настоящее время известно большое число приложений различных вариантов этого преобразования [8, 17]. В данной работе указанные преобразования находят ещё одну область применения.

1. Основной результат

Пусть β — отношение на E_2 ; класс всех частичных булевых функций, сохраняющих это отношение, обозначается через $\text{Pol}^*(\beta)$. Рассматриваемые далее алгоритмы применимы для ряда классов, описываемых отношениями. Введём в рассмотрение соответствующие отношения.

Пусть m — произвольное целое положительное число. Для произвольной булевой константы s через $\alpha_s^{(m)}$ обозначим отношение на множестве E_2 арности m , определённое следующим образом:

$$(x_1, \dots, x_m) \in \alpha_s^{(m)} \Leftrightarrow x_1 \oplus \dots \oplus x_m = s.$$

В частности, отношение $\alpha_0^{(2)}$ совпадает с отношением равенства, а класс $\text{Pol}^*(\alpha_0^{(2)})$ совпадает с классом P_2^* всех частичных булевых функций.

Классы $\text{Pol}^*(\alpha_0^{(2m)})$ при $m = 1, 2, \dots$ бесконечно убывают по включению и пересекаются по классу частичных булевых функций, доопределяемых до линейных.

Для набора $c \in E_2^m$ через $\varepsilon_c^{(m)}$ обозначим отношение $E_2^m \setminus \{c\}$. В частности, при $m = 2$ и $c = (1, 0)$ отношение $\varepsilon_c^{(m)}$ совпадает с порядком $\leq$, а класс $\text{Pol}^*(\varepsilon_c^{(m)})$ совпадает с классом частичных булевых функций, доопределяемых до монотонных.

При произвольном $m = 1, 2, \dots$ и $c = (1, \dots, 1)$ функции из класса $\text{Pol}^*(\varepsilon_c^{(m)})$ называются m -неразделёнными. Любые m наборов, на которых такая функция принимает значение 1, имеют общую единичную компоненту. Данные классы бесконечно убывают по включению и пересекаются по классу всех неразделённых частичных булевых функций.

Сформулируем основной результат данной статьи.

Теорема 1. Пусть m — целое положительное число, $c \in E_2^m$, s — двоичная константа и $\beta \in \{\alpha_s^{(m)}, \varepsilon_c^{(m)}\}$. Существует алгоритм сложности $O(2^n n^2)$, который по произвольной частичной булевой функции f от n переменных вычисляет её характеристическую функцию $\chi(f, \text{Pol}^*(\beta))$.

Построение алгоритмов, о которых говорится в теореме, опирается на использование логических и арифметических функций, определяемых в разд. 2 и 3. В разд. 4 и 5 описываются быстрые алгоритмы вычисления арифметических функций для рассматриваемых задач, основанные на преобразованиях Уолша — Адамара и Мёбиуса. В разд. 6 приводится доказательство теоремы 1.

2. Логическая функция

Пусть m — целое положительное число, c — набор из множества E_2^m , β — отношение арности m на множестве E_2 и f — частичная булева функция от n переменных. Логической функцией тройки (f, β, c) назовём функцию

$$H(f, \beta, c): E_2^n \rightarrow E_2, \quad x \mapsto H(f, \beta, c)(x),$$

которая на наборе $x = (x_1, \dots, x_n)$ из множества E_2^n принимает значение $H(f, \beta, c)(x)$, равное 1, в том и только том случае, если в множестве E_2^n

найдётся m наборов

$$v_1 = (v_{11}, \dots, v_{1n}), \dots, v_m = (v_{m1}, \dots, v_{mn})$$

таких, что

$$(2) \quad (f(v_1), \dots, f(v_m)) = c;$$

(3) для любого номера i , где $1 \leq i \leq n$, выполняется условие

$$(v_{1i}, \dots, v_{mi}) \in \beta \Leftrightarrow x_i = 0.$$

Логической функцией пары (f, β) назовём функцию

$$H(f, \beta): E_2^n \rightarrow E_2, \quad x \mapsto \bigvee_c H(f, \beta, c)(x)[c \notin \beta],$$

где дизъюнкция вычисляется по всем наборам c из множества E_2^m . Иными словами, эта функция на наборе x из множества E_2^n принимает значение $H(f, \beta)(x)$, равное 1, в том и только том случае, если в множестве E_2^m найдётся m наборов $v_1, \dots, v_m$, для которых выполняются условия (3), а также

$$(4) \quad (f(v_1), \dots, f(v_m)) \notin \beta.$$

Логические функции представляют интерес, так как имеет место

Лемма 1. Пусть β — отношение арности m на множестве E_2^m . Существует алгоритм битовой сложности $O(n2^n)$, который для любой частичной булевой функции f от n переменных находит характеристическую функцию $\chi(f, \text{Pol}^*(\beta))$ по логическим функциям $H(f, \alpha_0^{(2)})$ и $H(f, \beta)$.

ДОКАЗАТЕЛЬСТВО. Для произвольной булевой функции

$$h: E_2^n \rightarrow E_2, \quad x \mapsto h(x)$$

функция

$$h^D: E_2^n \rightarrow E_2, \quad x \mapsto \overline{h(\bar{x})},$$

называется *двойственной*, а функция

$$h^M: E_2^n \rightarrow E_2, \quad x \mapsto \bigvee_u h(u)[u \leq x],$$

называется *монотонной мажорантой*; здесь дизъюнкция вычисляется по всем наборам u из множества E_2^n . Легко проверяются равенства

$$(h^M)_0 = (h_0)^M, \quad (h^M)_1 = (h_0)^M \vee (h_1)^M,$$

где дизъюнкция функций выполняется поточечно и через H_b обозначается подфункция, получаемая из функции H при фиксации первой переменной булевой константой b . Из записанных равенств видно, что монотонная мажоранта h^M вычисляется по функции h рекурсивным алгоритмом битовой сложности $L(n)$, где $L(n) \leq 2L(n-1) + 2^n$ для всех $n > 1$ и тогда $L(n) = O(n2^n)$.

Покажем, что

$$\chi(f, \text{Pol}^*(\beta)) = H(f, \alpha_0^{(2)})^{MD} \cdot H(f, \beta)^{MD},$$

где умножение функций выполняется поточечно. Этого будет достаточно для доказательства леммы, поскольку монотонная мажоранта булевой функции находится алгоритмом битовой сложности $O(n2^n)$ и двойственная функция находится алгоритмом сложности $O(2^n)$. Дальнейшее доказательство разобьём на две части.

1. Пусть для набора $y = (y_1, \dots, y_n)$ из множества E_2^n выполняется равенство

$$\chi(f, \text{Pol}^*(\beta))(y) = 1.$$

Иными словами, множество $U = \{i \mid y_i = 1\}$ является достаточным для реализации функции f в классе $\text{Pol}^*(\beta)$. Обозначим через $u_1, \dots, u_k$ элементы множества U . Сказанное выше означает, что в классе $\text{Pol}^*(\beta)$ найдётся реализующая функцию f функция g от k переменных такая, что для любого набора $x = (x_1, \dots, x_n)$ из множества E_2^n выполняется условие (1). Покажем, что

$$H(f, \beta)^{MD}(y) = H(f, \alpha_0^{(2)})^{MD}(y) = 1.$$

Предположим, что $H(f, \beta)^{MD}(y) = 0$. Тогда $H(f, \beta)^M(\bar{y}) = 1$ и для некоторого набора $x \leq \bar{y}$ имеем $H(f, \beta)(x) = 1$. Это означает, что для некоторых наборов $v_1, \dots, v_m$ из множества E_2^n выполняются условия (3) и (4). Тогда в множестве E_2^k для наборов

$$v'_1 = (v_{1u_1}, \dots, v_{1u_k}), \dots, v'_m = (v_{mu_1}, \dots, v_{mu_k})$$

выполняются условия

- (3') $(v'_{1i}, \dots, v'_{mi}) \in \beta$ для любого номера i из множества U ;
- (4') $(g(v'_1), \dots, g(v'_m)) = (f(v_1), \dots, f(v_m)) \notin \beta$.

Это противоречит тому, что функция g сохраняет отношение β . Полученное противоречие доказывает равенство $H(f, \beta)^{MD}(y) = 1$. Аналогично доказывается, что $H(f, \alpha_0^{(2)})^{MD}(y) = 1$; для этого в рассуждении выше нужно взять $\alpha_0^{(2)}$ вместо β .

2. Пусть

$$H(f, \alpha_0^{(2)})^{MD}(y) \cdot H(f, \beta)^{MD}(y) = 1.$$

Докажем, что $\chi(f, \text{Pol}^*(\beta))(y) = 1$. Иными словами, требуется доказать, что множество $U = \{i \mid y_i = 1\}$ является достаточным для реализации функции f в классе $\text{Pol}^*(\beta)$. Как и раньше, через $u_1, \dots, u_k$ обозначим

элементы множества U . Определим частичную булеву функцию g от k переменных, для набора $x = (x_1, \dots, x_k)$ из множества E_2^k положив

$$g(x) = \begin{cases} f(z), & \text{если } f(z) \neq * \text{ и } x_1 = z_{u_1}, \dots, x_k = z_{u_k} \\ & \text{для некоторого набора } z = (z_1, \dots, z_n) \in E_2^n, \\ * & \text{в противном случае.} \end{cases}$$

Проверим, что функция g определена корректно. Рассмотрим для этого пару наборов

$$v_1 = (v_{11}, \dots, v_{1n}), \quad v_2 = (v_{21}, \dots, v_{2n})$$

таких, что значения $f(v_1), f(v_2)$ не равны $*$. Предположим, что эти наборы содержат одинаковые значения $x_1, \dots, x_k$ в позициях с соответствующими номерами $u_1, \dots, u_k$, в частности,

$$(v_{1u_1}, v_{2u_1}) \in \alpha_0^{(2)}, \dots, (v_{1u_k}, v_{2u_k}) \in \alpha_0^{(2)}.$$

Определим двоичный набор

$$w = (w_1, \dots, w_n), \quad \text{где } w_j = [(v_{1u_j}, v_{2u_j}) \notin \alpha_0^{(2)}] \text{ для } 1 \leq j \leq k.$$

Заметим, что $w \leq \bar{y}$ и выполняются соотношения

$$\begin{aligned} H(f, \alpha_0^{(2)})(w) &\leq H(f, \alpha_0^{(2)})^M(w) \\ &\leq H(f, \alpha_0^{(2)})^M(\bar{y}) = \overline{H(f, \alpha_0^{(2)})^{MD}(y)} = \bar{1} = 0. \end{aligned}$$

Значит, имеет место равенство $H(f, \alpha_0^{(2)})(w) = 0$, означающее в силу определения логической функции, что выполняется соотношение $(f(v_1), f(v_2)) \in \alpha_0^{(2)}$, т. е. значения $f(v_1), f(v_2)$ совпадают. Таким образом, функция g определена однозначно на наборе $x = (x_1, \dots, x_k)$, выбранном произвольно в множестве E_2^k .

Проверим, что функция g сохраняет отношение β . Для этого рассмотрим наборы

$$v_1 = (v_{11}, \dots, v_{1n}), \dots, v_m = (v_{m1}, \dots, v_{mn}).$$

Предположим, что

$$(v_{1u_1}, \dots, v_{mu_1}) \in \beta, \dots, (v_{1u_k}, \dots, v_{mu_k}) \in \beta.$$

Определим двоичный набор

$$w = (w_1, \dots, w_n), \quad \text{где } w_j = [(v_{1u_j}, \dots, v_{mu_j}) \notin \beta] \text{ для } 1 \leq j \leq k.$$

Заметим, что $w \leq \bar{y}$ и

$$H(f, \beta)(w) \leq H(f, \beta)^M(w) \leq H(f, \beta)^M(\bar{y}) = \overline{H(f, \beta)^{MD}(y)} = \bar{1} = 0.$$

Следовательно, имеет место равенство $H(f, \beta)(w) = 0$, означающее в силу определения логической функции, что набор значений $f(v_1), \dots, f(v_m)$

удовлетворяет отношению β , и тогда функция g сохраняет это отношение. Лемма 1 доказана.

Замечание 1. Методом из статьи [2] для вычисления монотонной мажоранты булевой функции от n переменных можно построить алгоритм битовой сложности $O(2^n \sqrt{n} \log n)$. Тогда и для сложности вычисления характеристической функции по логической верна та же верхняя оценка. Для дальнейшего, однако, достаточно оценки, установленной в лемме 1.

3. Арифметическая функция

Арифметической функцией тройки (f, β, c) назовём функцию

$$G(f, \beta, c): E_2^n \rightarrow \{0, \dots, 2^{mn} - 1\}, \quad x \mapsto G(f, \beta, c)(x),$$

которая на наборе $x = (x_1, \dots, x_n)$ из множества E_k^n принимает значение $G(f, \beta, c)(x)$, равное числу наборов $(v_1, \dots, v_m)$ в множестве $(E_2^n)^m$, для которых выполняются условия (2) и (3) из определения логической функции.

Арифметической функцией пары (f, β) назовём функцию

$$G(f, \beta): E_2^n \rightarrow \{0, \dots, 2^{mn} - 1\}, \quad x \mapsto \sum_c G(f, \beta, c)(x)[c \notin \beta],$$

где суммирование выполняется по всем наборам c из множества $E_2^m \setminus \beta$. Эта функция на наборе $x \in E_2^n$ принимает значение $G(f, \beta)(x)$, равное числу наборов $(v_1, \dots, v_m)$ в множестве $(E_2^n)^m$, для которых выполняются условия (3) и (4) из определения логической функции.

Ясно, что логическая функция вычисляется по арифметической для той же пары алгоритмом сложности $O(n2^n)$. В связи с этим будем интересоваться далее алгоритмами вычисления арифметических функций для отношений $\alpha_s^{(m)}$ и $\varepsilon_c^{(m)}$. Предлагаемые для этого алгоритмы основаны на том, что вычисление арифметических функций в преобразованном виде можно свести рекурсивно к аналогичным вычислениям с меньшими значениями параметра m и, в конце концов, со значением $m = 1$. Для отношений $\alpha_s^{(m)}$ это удаётся сделать с использованием преобразования Уолша — Адамара, а для отношения $\varepsilon_c^{(m)}$ нужным свойством обладает преобразование Мёбиуса. Возможность подобной рекурсии и вид преобразований, с помощью которых она реализуется, обусловлены строением рассматриваемых отношений. Аналогичный метод рекурсии используется в [6] при построении ступенчатых билинейных алгоритмов. Следует отметить, что рассматриваемые в данной статье задачи могут быть решены на основе модификации этого метода, но далее предлагается независимый способ решения.

4. Преобразования Уолша — Адамара

Для произвольной функции $F: E_2^n \rightarrow \mathbb{Q}$ определим функции

$$F^\Phi: E_2^n \rightarrow \mathbb{Q}, \quad x \mapsto \frac{1}{2^n} \sum_v F(v)(-1)^{(x,v)},$$

$$F^\phi: E_2^n \rightarrow \mathbb{Q}, \quad x \mapsto \sum_v F(v)(-1)^{(x,v)},$$

где суммирование в обоих случаях выполняется в поле рациональных чисел по всем наборам v из множества E_2^n , и

$$(x, v) = x_1 v_1 \oplus \cdots \oplus x_n v_n$$

есть скалярное произведение наборов $x = (x_1, \dots, x_n)$ и $v = (v_1, \dots, v_n)$ из множества E_2^n .

Обозначим через $\text{Func}(E_2^n, \mathbb{Q})$ векторное пространство над полем $\mathbb{Q}$, состоящее из функций, отображающих множество E_2^n в поле рациональных чисел и рассматриваемых с поточечными операциями сложения и умножения на скаляры. Операции

$$\text{Func}(E_2^n, \mathbb{Q}) \rightarrow \text{Func}(E_2^n, \mathbb{Q}), \quad F \mapsto F^\Phi,$$

$$\text{Func}(E_2^n, \mathbb{Q}) \rightarrow \text{Func}(E_2^n, \mathbb{Q}), \quad F \mapsto F^\phi,$$

являются взаимно обратными линейными преобразованиями этого пространства. Они называются *обратным* и *прямым преобразованиями Уолша — Адамара*.

Преобразования Уолша — Адамара позволяют получить быстрый алгоритм вычисления арифметической функции для отношений $\alpha_s^{(m)}$, в основе чего лежит

Лемма 2. Пусть m, m_1, m_2 — целые положительные числа и c, c_1, c_2 — наборы из множеств $E_2^m, E_2^{m_1}, E_2^{m_2}$ соответственно такие, что имеют место равенства $m = m_1 + m_2, c = c_1 c_2$.

Тогда для любой частичной булевой функции f от n переменных и любой булевой константы s выполняется равенство

$$G(f, \alpha_s^{(m)}, c)^\phi = G(f, \alpha_s^{(m_1)}, c_1)^\phi \cdot G(f, \alpha_s^{(m_2)}, c_2)^\phi,$$

где умножение функций выполняется поточечно.

ДОКАЗАТЕЛЬСТВО. Пусть f — частичная булева функция от n переменных и x — набор из множества E_2^n . Достаточно доказать лемму для $s = 0$, так как

$$G(f, \alpha_1^{(m)}, c)^\phi(x) = G(f, \alpha_0^{(m)}, c)^\phi(\bar{x}).$$

Для произвольной булевой константы u через $N_u(m, c, x)$ обозначим число наборов $(v_1, \dots, v_m)$ в множестве $(E_2^n)^m$, для которых

$$(f(v_1), \dots, f(v_m)) = c, \quad (v_1, x) \oplus \dots \oplus (v_m, x) = u.$$

Имеют место равенства

$$\begin{aligned} N_0(m, c, x) &= N_0(m_1, c_1, x) \cdot N_0(m_2, c_2, x) + N_1(m_1, c_1, x) \cdot N_1(m_2, c_2, x), \\ N_1(m, c, x) &= N_0(m_1, c_1, x) \cdot N_1(m_2, c_2, x) + N_1(m_1, c_1, x) \cdot N_0(m_2, c_2, x), \\ G(f, \alpha_0^{(m)}, c)^\phi(x) &= N_0(m, c, x) - N_1(m, c, x), \end{aligned}$$

используя которые, получаем

$$\begin{aligned} &G(f, \alpha_0^{(m_1)}, c_1)^\phi(x) \cdot G(f, \alpha_0^{(m_2)}, c_2)^\phi(x) \\ &= (N_0(m_1, c_1, x) - N_1(m_1, c_1, x)) \cdot (N_0(m_2, c_2, x) - N_1(m_2, c_2, x)) \\ &= (N_0(m_1, c_1, x) \cdot N_0(m_2, c_2, x) + N_1(m_1, c_1, x) \cdot N_1(m_2, c_2, x)) \\ &\quad - (N_0(m_1, c_1, x) \cdot N_1(m_2, c_2, x) + N_1(m_1, c_1, x) \cdot N_0(m_2, c_2, x)) \\ &= N_0(m, c, x) - N_1(m, c, x) = G(f, \alpha_0^{(m)}, c)^\phi(x). \end{aligned}$$

Лемма 2 доказана.

Следствие 1. Пусть m — целое положительное число и s — набор из множества E_2^m , содержащий l единиц. Тогда для любой частичной булевой функции f от n переменных и любой булевой константы c выполняется равенство

$$G(f, \alpha_s^{(m)}, c)^\phi = (G(f, \alpha_s^{(1)}, 0)^\phi)^{m-l} \cdot (G(f, \alpha_s^{(1)}, 1)^\phi)^l.$$

Доказательство. Для доказательства нужно многократно воспользоваться разложением из леммы 2. Следствие 1 доказано.

Следствие 2. Пусть m — целое положительное число. Тогда для любой частичной булевой функции f от n переменных и любой булевой константы s выполняется равенство

$$\begin{aligned} G(f, \alpha_s^{(m)})^\phi &= \frac{1}{2} \left((G(f, \alpha_s^{(1)}, 0) + G(f, \alpha_s^{(1)}, 1))^\phi \right)^m \\ &\quad + (-1)^{s+1} \left((G(f, \alpha_s^{(1)}, 0) - G(f, \alpha_s^{(1)}, 1))^\phi \right)^m, \end{aligned}$$

где операции сложения, вычитания и умножения функций выполняются поточечно.

ДОКАЗАТЕЛЬСТВО. В силу определений арифметических функций и следствий 1 и 2 получаем

$$\begin{aligned}
G(f, \alpha_s^{(m)})^\phi &= \left(\sum_{c \notin \alpha_s^{(m)}} G(f, \alpha_s^{(m)}, c) \right)^\phi = \sum_{c \notin \alpha_s^{(m)}} (G(f, \alpha_s^{(m)}, c))^\phi \\
&= \sum_{\substack{l \equiv s+1 \\ (\text{mod } 2)}} \binom{m}{l} (G(f, \alpha_s^{(1)}, 0)^\phi)^{m-l} (G(f, \alpha_s^{(1)}, 1)^\phi)^l \\
&= \frac{1}{2} ((G(f, \alpha_s^{(1)}, 0)^\phi + G(f, \alpha_s^{(1)}, 1)^\phi)^m \\
&\quad + (-1)^{s+1} (G(f, \alpha_s^{(1)}, 0)^\phi - G(f, \alpha_s^{(1)}, 1)^\phi)^m) \\
&= \frac{1}{2} (((G(f, \alpha_s^{(1)}, 0) + G(f, \alpha_s^{(1)}, 1))^\phi)^m \\
&\quad + (-1)^{s+1} ((G(f, \alpha_s^{(1)}, 0) - G(f, \alpha_s^{(1)}, 1))^\phi)^m),
\end{aligned}$$

где суммирование выполняется по всем наборам c из множества E_2^m и по всем целым l таким, что $0 \leq l \leq m$. Следствие 2 доказано.

Следствие 3. Пусть m — целое положительное число и s — булева константа. Существует алгоритм битовой сложности $O(2^n n^2)$, который по произвольной частичной булевой функции f от n переменных вычисляет арифметическую функцию $G(f, \alpha_s^{(m)})$.

ДОКАЗАТЕЛЬСТВО. Заметим, что для любой функции $G: E_2^n \rightarrow \mathbb{Q}$ такой, что $n \geq 1$, и любого набора x из множества E_2^{n-1} верно

$$\begin{aligned}
(G^\phi)_0 &= (G_0)^\phi + (G_1)^\phi, & (G^\phi)_1 &= (G_0)^\phi - (G_1)^\phi, \\
(G^\Phi)_0 &= \frac{1}{2} ((G_0)^\Phi + (G_1)^\Phi), & (G^\Phi)_1 &= \frac{1}{2} ((G_0)^\Phi - (G_1)^\Phi),
\end{aligned}$$

где через G_b обозначается подфункция, получаемая из функции G при фиксации первой переменной булевой константой b . Отсюда видно, что вычисление прямого и обратного преобразований Уолша—Адамара для функции от n переменных требует $L(n)$ арифметических действий сложения и вычитания, где $L(n) \leq 2L(n-1) + 2^n$ и, таким образом, $L(n) = O(n2^n)$. Для функции $G = G(f, \alpha_s^{(m)})$ вычисление функции G^ϕ на основании следствия 2 потребует ещё $O(2^n)$ возведений в m -ю степень. Учитывая, что в этих действиях участвуют числа размера $O(n)$, получаем алгоритм битовой сложности $O(2^n n^2)$. Следствие 3 доказано.

5. Преобразования Мёбиуса

Для произвольной функции $F: E_2^n \rightarrow \mathbb{Q}$ определим функции

$$F^\mu: E_2^n \rightarrow \mathbb{Q}, \quad x \mapsto \sum_v F(v)[x \leq v],$$

$$F^\zeta: E_2^n \rightarrow \mathbb{Q}, \quad x \mapsto \sum_v F(v)[x \leq v](-1)^{\sum_{i=1}^n (v_i - x_i)},$$

где суммирование в обоих случаях выполняется по всем наборам $v \in E_2^n$.

Операции

$$\text{Func}(E_2^n, \mathbb{Q}) \rightarrow \text{Func}(E_2^n, \mathbb{Q}), \quad F \mapsto F^\mu,$$

$$\text{Func}(E_2^n, \mathbb{Q}) \rightarrow \text{Func}(E_2^n, \mathbb{Q}), \quad F \mapsto F^\zeta,$$

являются взаимно обратными линейными преобразованиями векторного пространства $\text{Func}(E_2^n, \mathbb{Q})$. Они называются *прямым* и *обратным преобразованиями Мёбиуса*.

Преобразования Мёбиуса позволяет построить быстрые алгоритмы вычисления арифметических функций для отношений $\varepsilon_c^{(m)}$, в основе чего лежит

Лемма 3. Пусть m, m_1, m_2 — целые положительные числа и c, c_1, c_2 — наборы из множеств $E_2^m, E_2^{m_1}, E_2^{m_2}$ соответственно такие, что имеют место равенства $m = m_1 + m_2, c = c_1 c_2$.

Тогда для любой частичной булевой функции f от n переменных выполняется равенство

$$G(f, \varepsilon_c^{(m)})^\mu = G(f, \varepsilon_{c_1}^{(m_1)})^\mu \cdot G(f, \varepsilon_{c_2}^{(m_2)})^\mu,$$

где умножение функций выполняется поточечно.

Доказательство. Пусть f — частичная булева функция от n переменных. В соответствии с определением преобразования Мёбиуса значение $G(f, \varepsilon_c^{(m)})^\mu(x)$ совпадает с числом наборов $(v_1, \dots, v_m)$ из множества $(E_2^n)^m$, для которых

$$(f(v_1), \dots, f(v_m)) = c, \quad v_{1i} = c_1, \dots, v_{mi} = c_m, \text{ если } x_i = 1.$$

С учётом этого очевидно равенство

$$G(f, \varepsilon_c^{(m)})^\mu(x) = G(f, \varepsilon_{c_1}^{(m_1)})^\mu(x) \cdot G(f, \varepsilon_{c_2}^{(m_2)})^\mu(x).$$

Лемма 3 доказана.

Следствие 4. Пусть m — целое положительное число и c — набор из множества E_2^m , содержащий l единиц. Тогда для любой частичной булевой функции f от n переменных выполняется равенство

$$G(f, \varepsilon_c^{(m)})^\mu = (G(f, \varepsilon_0^{(1)})^\mu)^{m-l} \cdot (G(f, \varepsilon_1^{(1)})^\mu)^l.$$

ДОКАЗАТЕЛЬСТВО. Для доказательства нужно несколько раз воспользоваться разложением из лемм 3. Следствие 4 доказано.

Следствие 5. Пусть m — целое положительное число и c — набор из множества E_2^m . Существует алгоритм битовой сложности $O(2^n n^2)$, который по произвольной частичной булевой функции f от n переменных вычисляет арифметическую функцию $G(f, \varepsilon_c^{(m)})$.

ДОКАЗАТЕЛЬСТВО. Для функции $G: E_2^n \rightarrow \mathbb{Q}$, где $n \geq 1$, верно

$$\begin{aligned}(G^\mu)_0 &= (G_0)^\mu + (G_1)^\mu, & (G^\mu)_1 &= (G_1)^\mu, \\ (G^\zeta)_0 &= (G_0)^\zeta - (G_1)^\zeta, & (G^\zeta)_1 &= (G_1)^\zeta,\end{aligned}$$

где, по-прежнему, через G_b обозначается подфункция, получаемая из функции G при фиксации первой переменной булевой константой b . Отсюда видно, что выполнение прямого и обратного преобразований Мёбиуса требует выполнения $L(n)$ арифметических действий сложения и вычитания, где $L(n) \leq 2L(n-1) + 2^n$ и, таким образом, $L(n) = O(n2^n)$. Для функции $G = G(f, \varepsilon_c^{(m)})$ вычисление функции G^μ потребует ещё выполнения $O(2^n)$ умножений. Так как складываются и умножаются числа размера $O(n)$, получаем алгоритм битовой сложности $O(2^n n^2)$. Следствие 5 доказано.

Замечание 2. Вычисление арифметической функции $G(f, \varepsilon_c^{(m)})$, выполняемое на основании следствия 5, по сути реализует принцип включений и исключений. В явном виде данный комбинаторный принцип использовался в [18] для задачи распознавания свойства m -неразделённости булевой функции.

6. Доказательство теоремы 1

Теорема 1 непосредственно следует из леммы 1 и следствий 3 и 5 с учётом того, что для функции от n переменных и произвольного отношения логическая функция вычисляется по арифметической алгоритмом битовой сложности $O(n2^n)$.

Замечание 3. Некоторое уменьшение сложности предлагаемых алгоритмов возможно на основе использования более быстрых алгоритмов умножения целых чисел. Например, можно использовать алгоритм Шёнхаге — Штрассена [19] сложности $O(n \log(n) \log \log(n))$, алгоритм Фюрера [20] или модулярный алгоритм [21] сложности $n \log(n) 2^{O(\log^*(n))}$.

ЛИТЕРАТУРА

1. **Golubeva O. I.** Construction of permissible functions and their application for fault tolerance // Proc. 2019 Int. Sib. Conf. Control Communications (Tomsk, Russia, Apr. 18–20, 2019). Tomsk: TUSUR, 2019. P. 1–5.
2. **Вороненко А. А.** О методе разложения для распознавания принадлежности инвариантным классам // Дискрет. математика. 2002. Т. 14, № 4. С. 110–116.
3. **Selezneva S. N.** Constructing polynomials for functions over residue rings modulo a composite number in linear time // Computer science — Theory and applications. Heidelberg: Springer, 2012. P. 302–313. (Lect. Notes Comput. Sci.; Vol. 7353).
4. **Алексеев В. Б., Емельянов Н. Р.** Метод построения быстрых алгоритмов в k -значной логике // Мат. заметки. 1985. Т. 38, вып. 1. С. 148–156.
5. **Алексеев В. Б.** Ступенчатые билинейные алгоритмы и распознавание полноты в k -значных логиках // Изв. вузов. Математика. 1988. № 7. С. 19–27.
6. **Алексеев В. Б.** Логические полукольца и их использование для построения быстрых алгоритмов // Вестн. Моск. ун-та. Сер. 1. 1997. № 1. С. 22–29.
7. **Парватов Н. Г.** Порождение достаточных множеств аргументов частичной булевой функции // Вестн. Томск. гос. ун-та. Прил. 2007. № 23. С. 44–48.
8. **Сачков В. Н.** Комбинаторные методы дискретной математики. М.: Наука, 1977.
9. **Мак-Вильямс Ф. Дж., Слоэн Н. Дж. А.** Теория кодов, исправляющих ошибки. М.: Связь, 1979.
10. **Rushforth C. K.** Fast Fourier–Hadamard decoding of orthogonal codes // Inform. Control. 1969. Vol. 15, No. 1. P. 33–37.
11. **Малютин А. А.** Быстрое корреляционное декодирование некоторых подмножеств слов кода Рида — Маллера первого порядка // Дискрет. математика. 1990. Т. 2, вып. 2, С. 155–158.
12. **Карякин Ю. Д.** Быстрое корреляционное декодирование кодов Рида — Маллера // Пробл. передачи информации. 1987. Т. 23, вып. 2, С. 40–49.
13. **Токарева Н. Н.** Обобщения бент-функций. Обзор работ // Дискрет. анализ и исслед. операций. 2010. Т. 17, № 1. С. 33–62.
14. **Weisner L.** Abstract theory of inversion of finite series // Trans. Amer. Math. Soc. 1935. Vol. 38. P. 474–484.
15. **Hall P.** The Eulerian functions of a group // Q. J. Math. 1936. Vol. 7. P. 134–151.
16. **Rota G.-C.** On the foundations of combinatorial theory: I. Theory of Möbius functions // Z. Wahrscheinlichkeitstheor. Verw. Geb. 1964. Vol. 2. P. 340–368.
17. **Stanley R. P.** Enumerative combinatorics. Vol. 1. New York: Camb. Univ. Press, 1997. (Camb. Stud. Adv. Math.; Vol. 49).

18. **Парватов Н. Г.** О распознавании свойств дискретных функций схемами из функциональных элементов // Вестн. Томск. гос. ун-та. Прил. 2005. № 14. С. 233–236.
19. **Schönhage A., Strassen V.** Schnelle Multiplikation großer Zahlen // Computing. 1971. Vol. 7. P. 281–292. [German].
20. **Fürer M.** Faster integer multiplication // Proc. 39th Annu. ACM Symp. Theory Comput. (San Diego, CA, USA, June 11–13, 2007). New York: ACM, 2007. P. 57–66.
21. **De A., Kurur P. P., Saha C., Saptharishi R.** Fast integer multiplication using modular arithmetic // Proc. 40th Annu. ACM Symp. Theory Comput. (Victoria, Canada, May 17–20, 2008). New York: ACM, 2008. P. 499–506.

Парватов Николай Георгиевич

Статья поступила

20 июня 2019 г.

После доработки —

5 ноября 2019 г.

Принята к публикации

27 ноября 2019 г.

FINDING THE SUBSETS OF VARIABLES
OF A PARTIAL BOOLEAN FUNCTION
WHICH ARE SUFFICIENT FOR ITS IMPLEMENTATION
IN THE CLASSES DEFINED BY PREDICATES

N. G. Parvatov

Tomsk State University,
36 Lenin Avenue, 634050 Tomsk, Russia
E-mail: ngparvatov@yandex.ru

Abstract. Given a class K of partial Boolean functions and a partial Boolean function f of n variables, a subset U of its variables is called *sufficient for the implementation of f in K* if there exists an extension of f in K with arguments in U . We consider the problem of recognizing all subsets sufficient for the implementation of f in K . For some classes defined by relations, we propose the algorithms of solving this problem with complexity of $O(2^n n^2)$ bit operations. In particular, we present some algorithms of this complexity for the class P_2^* of all partial Boolean functions and the class M_2^* of all monotone partial Boolean functions. The proposed algorithms use the Walsh–Hadamard and Möbius transforms. Bibliogr. 21.

Keywords: partial Boolean function, sufficient subset of variables, Walsh–Hadamard transform, Möbius transform.

REFERENCES

1. O. I. Golubeva, Construction of permissible functions and their application for fault tolerance, in *Proc. Int. Sib. Conf. Control and Communications (Tomsk, Russia, Apr. 18–20, 2019)* (TUSUR, Tomsk, 2019), pp. 1–5.
2. A. A. Voronenko, On a decomposition method for recognizing membership in invariant classes, *Diskretn. Mat.* **14** (4), 110–116 (2002) [*Discrete Math. Appl.* **12** (6), 607–614 (2002)].
3. S. N. Selezneva, Constructing polynomials for functions over residue rings modulo a composite number in linear time, in *Lecture Notes in Computer Science*, Vol. 7353 (Springer, Heidelberg, 2012), pp. 303–312.

4. **V. B. Alekseev** and **N. R. Emel'yanov**, A method for constructing fast algorithms in k -valued logic, *Mat. Zametki* **38** (1), 148–156, 171 (1985) [*Math. Notes* **38** (1), 595–600 (1985)].
5. **V. B. Alekseev**, Stepwise bilinear algorithms and recognition of completeness in k -valued logics, *Izv. Vyssh. Uchebn. Zaved. Mat.*, No. 7, 19–27 (1988) [*Soviet Math. (Izv. VUZ)* **32** (7), 31–42 (1988)].
6. **V. B. Alekseev**, Logical semirings and their usage for construction of quick algorithms, *Vestn. Mosk. Univ., Ser. I: Mat. Mekh.*, No. 1, 22–29 (1997) [*Moscow Univ. Math. Bull.* **52** (1), 22–28 (1997)].
7. **N. G. Parvatov**, Generating sufficient sets for partial Boolean functions, *Vestn. Tomsk. Gos. Univ., Suppl.*, No. 23, 44–48 (2007).
8. **V. N. Sachkov**, *Combinatorial Methods of Discrete Mathematics* (Nauka, Moscow, 1977).
9. **F. J. MacWilliams** and **N. J. A. Sloane**, *The Theory of Error-Correcting Codes* (North-Holland, Amsterdam, 1977; Svyaz', Moscow, 1979).
10. **S. K. Rushforth**, Fast Fourier–Hadamard decoding of orthogonal codes, *Inform. Control* **15** (1), 33–37 (1969).
11. **A. A. Malyutin**, Fast correlation decoding of some subsets of words of the first-order Reed–Muller code, *Diskretn. Mat.* **2** (2), 155–158 (1990) [*Discrete Math. Appl.* **2** (2), 155–158 (1992)].
12. **Yu. D. Karyakin**, Fast correlation decoding of Reed–Muller codes, *Probl. Peredachi Inform.* **23** (2), 40–49 (1987).
13. **N. N. Tokareva**, Generalizations of bent functions: A survey of publications, *Diskretn. Anal. Issled. Oper.* **17** (1), 34–64, 99 (2010) [*J. Appl. Ind. Math.* **5** (1), 110–129 (2011)].
14. **L. Weisner**, Abstract theory of inversion of finite series, *Trans. Amer. Math. Soc.* **38**, 474–484 (1935).
15. **P. Hall**, The Eulerian functions of a group, *Q. J. Math.* **7**, 134–151 (1936).
16. **G. C. Rota**, On the foundations of combinatorial theory. I. Theory of Möbius functions, *Z. Wahrscheinlichkeitstheorie und Verw. Gebiete.* **2**, 340–368 (1964).
17. **R. P. Stanley**, *Enumerative Combinatorics*, Vol. 1 (Camb. Univ. Press, Cambridge, 1997).
18. **N. G. Parvatov**, On recognizing of properties of discrete functions by Boolean circuits, *Vestn. Tomsk. Gos. Univ., Suppl.*, No. 14, 233–236 (2005).
19. **A. Schönhage** and **V. Straßen**, Schnelle multiplikation großer zahlen, *Computing* **7**, 281–292 (1971).
20. **M. Fürer**, Faster integer multiplication, In *Proc. 39th Annual ACM Symp. Theory of Computing, San Diego, CA, USA, June 11–13, 2007* (ACM, New York, 2007), pp. 57–66.
21. **A. De**, **C. Saha**, **P. Kurur**, and **R. Saptharishi**, Fast integer multiplication using modular arithmetic, in *Proc. 40th Annual ACM Symp. Theory of Computing, Victoria, Canada, May 17–20, 2008* (ACM, New York, 2008), pp. 499–506 [*SIAM J. Comput.* **42** (2), 685–699 (2013)].

Nikolay G. Parvatov

Received June 20, 2019

Revised November 5, 2019

Accepted November 27, 2019