Math-Net.Ru

Общероссийский математический портал

А. И. Мамонтов, Д. Г. Мещанинов, Алгоритм распознавания полноты в функциональной системе $L(\mathbb{Z})$, *Дискрет. матем.*, 2014, том 26, выпуск 1, 85–95

DOI: <https://doi.org/10.4213/dm1269>

Использование Общероссийского математического портала Math-Net.Ru подразумевает, что вы прочитали и согласны с пользовательским соглашением
<http://www.mathnet.ru/rus/agreement>

Параметры загрузки:

IP: 54.198.67.100

26 апреля 2023 г., 17:16:46

Алгоритм распознавания полноты в функциональной системе $L(\mathbb{Z})$

© 2014 г. А. И. Мамонтов,¹ Д. Г. Мещанинов²

Предложен алгоритм, распознающий полноту (относительно суперпозиции) в функциональной системе полиномов первой степени над кольцом $\mathbb{Z}$. Оценены временная и емкостная сложности алгоритма. Алгоритм можно применять и для решения нескольких иных задач: распознавания свойств одного полинома и проверки полноты некоторых специальных систем.

Работа выполнена при финансовой поддержке РФФИ (проекты 13-01-00684, 11-01-00792а).

Ключевые слова: функциональная система, полиномы над кольцом целых чисел, полнота, временная и емкостная сложность алгоритмов.

В настоящей работе мы продолжаем исследование функциональной системы $P(\mathbb{Z})$ полиномов над кольцом $\mathbb{Z}$ с операциями суперпозиции [1–7] и ее подсистемы $L(\mathbb{Z})$, состоящей из полиномов только нулевой и первой степеней. Важнейшими в любой универсальной алгебре являются вопросы полноты и выразимости. Функциональные системы $P(\mathbb{Z})$ и $L(\mathbb{Z})$ (замкнутые классы в счетнозначной логике) конечно-порождаемы, множества $\{1, x - y, xy\}$ и, соответственно, $\{1, x - y\}$ образуют их базисы, поэтому критерий полноты в этих алгебрах можно сформулировать в терминах предполных классов (максимальных собственных подалгебр). Установлено, что в функциональной системе $P(\mathbb{Z})$ множество всех предполных классов имеет мощность континуума и проблема полноты алгоритмически неразрешима. В более узкой системе $L(\mathbb{Z})$ предполные классы образуют счетное множество, все они описаны в [4, 5], что позволило вывести критерий полноты в этой алгебре. Несмотря на бесконечность множества всех предполных классов, проблема полноты в $L(\mathbb{Z})$ оказалась алгоритмически разрешимой, об этом сообщалось в [4]. Алгоритм распознавания полноты в $L(\mathbb{Z})$ и оценки его временной и емкостной сложностей составляют основной результат данной работы.

Алгоритмическая сложность распознавания полноты является важной характеристикой функциональных систем, она интенсивно изучается в многозначных логиках [8–11]. Решение проблемы полноты сводится к распознаванию свойств функций, определяющих принадлежность предполным классам. Самостоятельный интерес представляют и свойства, не сохраняющиеся при суперпозициях. Эффективные алгоритмы распознавания свойств дискретных функций, в частности, представленных полиномами над некоторыми кольцами, актуальны с практических позиций, они заслужили значительное внимание многих авторов [12–19].

^{1,2} Место работы: НИУ "Московский энергетический институт"
e-mail: MamontovAI@yandex.ru, MeshchaninovDG@mpei.ru

Мы рассматриваем также алгоритмическую сложность распознавания свойств полиномов системы $L(\mathbb{Z})$. Анализируемые нами задачи представляют собой подзадачи (распознавание свойств одной функции, раздел 2) и частные случаи (относительная полнота, раздел 3) общей проблемы полноты, для их решения применяются фрагменты основного алгоритма из раздела 1. Полнота относительно фиксированного класса K (не обязательно замкнутого) означает полноту системы, содержащей класс K . Критерии относительной полноты чрезвычайно важны, если множество всех предполных классов неизвестно или трудно обозримо, а также в случае бесконечной порожденности алгебры (примеры — в [3]). Примерами критериев относительной полноты являются также теоремы Слупецкого [20], С. В. Яблонского [21], Саломая [22] для конечнозначных логик, Г. П. Гаврилова [23] для счетнозначной логики. А. И. Мамонтовым найдены критерии относительной полноты в бесконечно-порожденной функциональной системе $P(\mathbb{Q})$ полиномов первой степени над полем $\mathbb{Q}$ [24].

Полученные нами полиномиальные оценки сложности свидетельствуют об эффективности сконструированных алгоритмов.

Основными объектами нашего рассмотрения являются функции $f : \mathbb{Z}^n \rightarrow \mathbb{Z}$, задаваемые полиномами

$$f(x_1, \dots, x_n) = a_0 + a_1x_1 + \dots + a_nx_n, \quad a_0, a_1, \dots, a_n \in \mathbb{Z}. \quad (1)$$

Каждый такой полином полностью определяется вектором коэффициентов $(a_0, a_1, \dots, a_n)$ и представляет ровно одну функцию. Мы отождествляем функцию f , реализующий ее полином вида (1) и все конгруэнтные и равные f функции (т. е. получаемые из f переименованием переменных, а также введением и изъятием фиктивных переменных). Множество всех таких функций мы обозначаем через $L(\mathbb{Z})$. Такое же обозначение применяем и для функциональной системы (алгебры) $L(\mathbb{Z})$ с операциями суперпозиции. Буквами p, q , возможно, с индексами, обозначаем простые числа, $\text{НОД}(c_1, \dots, c_k)$ — наибольший общий делитель целых $c_1, \dots, c_k$. Понятие наибольшего общего делителя несколько модифицируем, вводя равносильные обозначения $\Delta(c_1, \dots, c_k)$ и

$$\bigtriangleup_{i=1}^k c_i$$

для наибольшего общего делителя лишь ненулевых из чисел $c_1, \dots, c_k$ и полагая $\Delta(0, \dots, 0) = 0$. Символами $[F]$ обозначаем замыкание множества F относительно суперпозиции.

В [5] найдены все предполные в $L(\mathbb{Z})$ классы, они позволили сформулировать критерий полноты в конечно-порожденной функциональной системе $L(\mathbb{Z})$. Перечислим их.

1. Класс L^+ функций, у которых все коэффициенты при переменных неотрицательны.

2. Класс D , содержащий все константы и унарные функции, а также все n -местные ($n \geq 2$) функции, имеющие $\text{НОД}(a_1, \dots, a_n) > 1$.

3. Классы $C(p_1 \cdots p_r)$, где $p_1, \dots, p_r$ — различные простые числа, $r \geq 1$, состоят из функций, у которых все коэффициенты $a_1, \dots, a_n$ кратны некоторому p_i , $1 \leq i \leq r$, или все коэффициенты $a_1, \dots, a_n$ кроме, возможно, одного кратны $p_1 \cdots p_r$. Каждый класс $C(p_1 \cdots p_r)$ содержит все константы и унарные функции.

4. Классы $S(p)$ функций, у которых $a_1 + \dots + a_n \equiv 1 \pmod{p}$.

5. Классы $U(b, p)$, $b = 0, 1, \dots, p-1$, состоят из функций, сохраняющих множество $\{c \in \mathbb{Z} \mid c \equiv b \pmod{p}\}$.

1. Алгоритм распознавания полноты

Пусть $F = \{f_1, \dots, f_m\}$ — конечная система функций из $L(\mathbb{Z})$. Требуется определить, полна ли система F в $L(\mathbb{Z})$ (предполагается ответ „Да“ или „Нет“). Каждую из функций системы F считаем зависящей от одного множества переменных $\{x_1, \dots, x_n\}$ и представляем в виде

$$f_i(\tilde{x}) = a_{i0} + a_{i1}x_1 + \dots + a_{in}x_n, \quad i = 1, \dots, m,$$

при этом некоторые из коэффициентов a_{ij} , $j = 0, \dots, n$, могут быть нулевыми.

Введём характеристику $NV(f)$, принимающую значение 1, если функция f существенно зависит не менее, чем от двух переменных, и 0 в противном случае.

Для описания алгоритма введём две процедуры, работающие с неотрицательными целыми числами и их списками.

Процедура ДЕЛ($i; c, l$) имеет входной параметр $i \in \{1, \dots, m\}$ — номер такой функции из системы F , что $NV(f_i) = 1$. Процедура возвращает число $c = \Delta(a_{i1}, \dots, a_{in})$ и список l всех чисел $d \neq 1$, удовлетворяющих условию:

$$\exists k \in \{1, \dots, n\} : \left(a_{ik} \neq 0, \quad d \mid \bigwedge_{1 \leq j \leq n, j \neq k} a_{ij} \right).$$

Процедура ПЕРЕС(c_1, c_2, l_1, l_2) получает на входе числа c_1, c_2 и списки l_1, l_2 . В результате её работы число c_1 заменяется на $\Delta(c_1, c_2)$, число c_2 не изменяется, список l_2 заменяется на пустой, а изменённый список l_1 образуется из чисел d , удовлетворяющих хотя бы одному из условий:

- 1) $d \in l_1 \cap l_2$,
- 2) $d \in l_1$ и $(d, c_2) \neq 1$,
- 3) $d \in l_2$ и $(d, c_1) \neq 1$.

Алгоритм распознавания полноты представляет собой процедуру (назовём её *процедурой А*), состоящую из следующих действий.

1. Если $a_{ij} \geq 0$ при всех $i = 1, \dots, m$ и $j = 1, \dots, n$, то выдать ответ „Нет“ и завершить работу (в этом случае $F \subset L^+$).
2. Для $i = 1, \dots, m$ вычислить $v_i = NV(f_i) \Delta(a_{i1}, \dots, a_{in})$. Если все $v_i \neq 1$, то выдать ответ „Нет“ и завершить работу ($F \subset D$).
3. Положить $j = 1$ и $c_1 = 0$.
4. Для $i = 1, \dots, m$, если $NV(f_i) = 1$, то выполнить действия:
 - 4.1. применить ДЕЛ($i; c_j, l_j$);
 - 4.2. если $j = 1$, то положить $j = 2$, иначе применить ПЕРЕС(c_1, c_2, l_1, l_2).
5. Если $c_1 \neq 1$ или список l_1 не пуст, то выдать ответ „Нет“ и завершить работу.
6. Для $i = 1, \dots, m$ положить $A_i = a_{i0}$, $B_i = a_{i1} + \dots + a_{in} - 1$.
7. Если $\Delta(B_1, \dots, B_m) \neq 1$, то выдать ответ „Нет“ и завершить работу ($F \subset S(p)$ для некоторого p).
8. Положить

$$w = \bigwedge_{1 \leq i < j \leq m} (A_i B_j - A_j B_i). \quad (2)$$

9. Если $w = 0$, то выдать ответ „Нет“ и завершить работу, иначе для $j = 0, 1, \dots, w - 1$ вычислять

$$d_j = \bigtriangleup_{i=1}^m (A_i + jB_i). \quad (3)$$

Если $d_j \neq 1$ при некотором j , то выдать ответ „Нет“ и завершить работу.

10. Выдать ответ „Да“ и завершить работу.

Мы покажем, что эта процедура является алгоритмом, и оценим его временную сложность.

Лемма 1. Если $F \notin L^+$ и $F \notin D$, то процедура A завершается на шаге 5 в том и только том случае, когда $F \subset C(p_1 \cdots p_r)$ при некоторых $p_1, \dots, p_r$.

Доказательство. Без ограничения общности полагаем $NV(f_i) = 1$ при $i = 1, \dots, m$. Первое выполнение цикла 4 состоит в применении $ДЕЛ(1; c_1, l_1)$ и присвоении значения $j = 2$. Класс $C(p_1 \cdots p_r)$ содержит все функции каждого из двух типов:

- 1) все коэффициенты при переменных функции кратны некоторому p_i , $1 \leq i \leq r$;
- 2) все коэффициенты при переменных функции, кроме, возможно, одного, кратны $p_1 \cdots p_r$.

Функция типа 1) принадлежит $C(p_1 \cdots p_r)$ тогда и только тогда, когда $\Delta(c_1, p_1 \cdots p_r) \neq 1$. Функция типа 2) принадлежит $C(p_1 \cdots p_r)$ тогда и только тогда, когда $p_1 \cdots p_r \in l_1$. Итак, $f_1 \in C(p_1 \cdots p_r)$ в точности при условии

$$\Delta(c_1, p_1 \cdots p_r) \neq 1 \text{ или } p_1 \cdots p_r \in l_1. \quad (4)$$

Если $m \geq 2$, то на втором проходе цикла 4 выполняются $ДЕЛ(2; c_2, l_2)$ и $ПЕРЕС(c_1, c_2, l_1, l_2)$. При этом $\{f_1, f_2\} \subset C(p_1 \cdots p_r)$ только в случае выполнения условия (4). Если $m \geq 3$, то остальные прохождения цикла 4 аналогичны второму. После m циклов убеждаемся, что условие (4) равносильно включению $F \subset C(p_1 \cdots p_r)$. Лемма доказана.

Лемма 2. Пусть $k \in \mathbb{Z}_+$, $A_i, B_i \in \mathbb{Z}$ при $i = 1, \dots, m$. Тогда

$$\bigtriangleup_{i=1}^m (A_i + kB_i) = \bigtriangleup_{i=1}^m (A_i + (k+w)B_i),$$

где w определено условием (2).

Доказательство. Пусть $t | \bigtriangleup_{i=1}^m (A_i + kB_i)$. Тогда t является делителем и числа

$$B_j(A_i + kB_i) - B_i(A_j + kB_j) = A_iB_j - A_jB_i$$

при всех $i, j = 1, \dots, m$. Следовательно, $t | w$ и $t | (A_i + (k+w)B_i)$ при всех i , откуда вытекает справедливость леммы.

Лемма 2 утверждает, что последовательность $d_0, d_1, d_2, \dots$, где d_j при $j \geq 0$ определены формулой (3), периодична с периодом w , если $w \neq 0$.

Лемма 3. Пусть $A_1, A_2, B_1, B_2, k \in \mathbb{Z}_+$ и $A_1B_2 = A_2B_1$. Тогда

$$\Delta(A_1 + kB_1, A_2 + kB_2) = \Delta(A_1, A_2) + k \Delta(B_1, B_2). \quad (5)$$

Доказательство. Если $A_1 A_2 B_1 B_2 = 0$, то справедливость равенства (5) легко проверить. В противном случае $\Delta(A_1, A_2) = d > 0$ и $\Delta(B_1, B_2) = e > 0$. Тогда $A_i = \alpha_i d$, $B_i = \beta_i e$, $\alpha_i \beta_i > 0$ при $i = 1, 2$, причём $(\alpha_1, \alpha_2) = (\beta_1, \beta_2) = 1$ и $\alpha_1 \beta_2 = \alpha_2 \beta_1$. Из этих условий следует, что $\alpha_1 = \beta_1$, $\alpha_2 = \beta_2$ и обе части соотношения (5) равны $d + ke$. Лемма доказана.

Лемма 4. Если $F \not\subset L^+$, $F \not\subset D$, $F \not\subset C(p_1 \dots p_r)$ и $F \not\subset S(p)$ при всех $r, p, p_1, \dots, p_r$, то процедура A завершается на шаге 9 в том и только том случае, когда $F \subset U(k, p)$ при некоторых p и k .

Доказательство. Условие $f_i \in U(k, p)$ равносильно соотношению

$$a_{i0} + \sum_{j=1}^n a_{ij} k \equiv k \pmod{p}.$$

Для $i = 1, \dots, m$ положим $A_i = a_{i0}$, $B_i = a_{i1} + \dots + a_{in} - 1$. Тогда $F \subset U(k, p)$ в том и только том случае, когда $p | d_k$.

Если число w , определяемое условием (2), отлично от нуля, то для поиска таких значений p, k достаточно по формуле (3) вычислить d_j только при $j = 0, 1, \dots, w-1$.

Если при некотором j получим $d_j \neq 1$, то найдется такое p , что $p | d_j$. Пусть k — наименьший неотрицательный вычет числа j по модулю p . Тогда $p | (A_i + kB_i)$ при всех $i = 1, \dots, m$ и, следовательно, $F \subset U(k, p)$.

Если $d_0 = d_1 = \dots = d_{w-1} = 1$, то $d_k = 1$ при всех $k \geq 0$, и тогда система F не содержится ни в одном классе $U(k, p)$.

При $w = 0$ рассмотрим

$$v_0 = \max_{1 \leq i \leq m, B_i \neq 0} \left\lfloor \frac{-A_i}{B_i} \right\rfloor + 1, \quad v = \max\{0, v_0\}.$$

Для такого v справедливы неравенства $1 - A_i/B_i \leq 1 + v$, $B_i^2(A_i/B_i + v) \geq 0$, $2B_i(A_i + vB_i) \geq 0$. Последнее из них приводит к соотношению

$$|2B_i + A_i + vB_i| = 2|B_i| + |A_i + vB_i|,$$

откуда, согласно лемме 2,

$$d_{v+2} = \bigtriangleup_{i=1}^m |A_i + (v+2)B_i| = d_v + 2 \bigtriangleup_{i=1}^m B_i.$$

В силу условия $F \not\subset S(p)$ для любого p имеем $\Delta(B_1, \dots, B_m) = 1$, поэтому $d_{v+2} \neq 1$. Для p такого, что $p | d_{v+2}$, положим k равным наименьшему неотрицательному вычету числа $v+2$ по модулю p . Тогда $p | (A_i + kB_i)$ при всех $i = 1, \dots, m$ и, следовательно, $F \subset U(k, p)$. Лемма доказана.

Теорема 1. Процедура A представляет собой алгоритм. Если все коэффициенты функций системы F ограничены по абсолютной величине константой t и максимальное количество переменных функций есть n , то размером задачи является $N = mnt$ и алгоритм имеет при реализации машиной с произвольным доступом к памяти временную сложность $O(N^2 \log^2 N)$ (двоичных операций) и емкостную сложность $O(N \log N)$ (битов).

Доказательство. Можно считать, что каждая функция системы F зависит от n переменных: при необходимости введем фиктивные переменные. Тогда каждую функцию можно задать вектором коэффициентов, имеющим длину $n + 1$, а за размер задачи принять величину $m(n + 1)t$, эквивалентную $N = mnt$ при $n \rightarrow \infty$. Оценим сложность каждого этапа процедуры A , принимая во внимание только арифметические операции с двоичным кодом и не учитывая операции обращения к памяти. Сложение и умножение целых чисел, не превосходящих t по абсолютной величине, требует $O(\log t)$ и $O(\log^2 t)$ битовых операций соответственно. Для вычисления НОД(a, b) применяем алгоритм Евклида, его временная сложность при условии $|a| \geq |b|$ оценивается величиной $O(\log^2 |a|)$, такую же сложность имеет вычисление $\Delta(a, b)$ [25].

Выполнение каждого из этапов 1, 2, 3 требует, как нетрудно проверить, $O(mn \log^2 t)$ операций и $O(mn \log t)$ единиц памяти.

Проанализируем этап 4.

Однократное применение процедуры $ДЕЛ(i; c, l)$ включает в себя:

- 1) вычисление $c = \Delta(a_{i1}, \dots, a_{in})$, что требует $O(n \log^2 t)$ операций и $O(\log t)$ единиц памяти;
- 2) вычисление величин $\Delta_k = \Delta_{j \neq k} a_{ij}$ для $k = 1, \dots, n$ (возможно, не всех), что требует $O(n^2 \log^2 t)$ операций и $O(n \log t)$ единиц памяти;
- 3) построение списка l делителей d (при этом $d \leq t$) чисел Δ_k , для чего достаточно $O(n^2 t \log^2 t)$ операций и $O(nt \log t)$ единиц памяти.

Таким образом, выполнение процедуры $ДЕЛ(i; c, l)$ имеет временную сложность $O(n^2 t \log^2 t)$ и емкостную сложность $O(nt \log t)$.

Далее, процедура $ПЕРЕС(c_1, c_2, l_1, l_2)$ имеет дело со списками l_1, l_2 длины не более t каждый и включает в себя:

- 1) вычисление $\Delta(c_1, c_2)$ ($O(\log^2 t)$ операций и $O(\log t)$ единиц памяти);
- 2) обновление списка l_1 , состоящее в построении $l_1 \cap l_2$ ($O(t)$ операций и памяти), вычислении НОД(d, c_2) для d из l_1 и НОД(d, c_1) для d из l_2 ($O(t \log^2 t)$ операций и $O(t \log t)$ единиц памяти);
- 3) преобразовании списка l_2 в пустой ($O(1)$ операций и памяти).

Таким образом, однократное выполнение процедуры $ПЕРЕС(c_1, c_2, l_1, l_2)$ имеет временную сложность $O(t \log^2 t)$ и емкостную сложность $O(t \log t)$.

На этапе 4 эти процедуры применяются для $i = 1, \dots, m$. Остальные действия этапа требуют лишь $O(1)$ операций и памяти. Следовательно, весь этап 4 имеет временную сложность $O(mn^2 t \log^2 t)$ и емкостную сложность $O(mnt \log t)$.

Временная и емкостная сложности этапа 5 составляют, как нетрудно проверить, $O(1)$.

На этапе 6 для $i = 1, \dots, m$ вычисляются значения $B_i = a_{i1} + \dots + a_{in} - 1$. Каждое из двух очередных слагаемых и результат их сложения не превосходят $nt + 1$ по абсолютной величине, поэтому временная и емкостная сложности этапа 6 составляют $O(mn \log(nt))$ и $O(m \log(nt))$ соответственно.

На этапе 7 производится вычисление $\Delta(B_1, \dots, B_m)$, что требует $O(m \log^2(nt))$ операций и $O(\log(nt))$ единиц памяти. Сравнение результата с 1 требует лишь $O(1)$ операций и памяти.

На этапе 8 при вычислении значения w по формуле (2) для $i, j = 1, \dots, m$ находятся величины $C_{ij} = A_i B_j - A_j B_i$ и $\Delta_{i,j=1}^m C_{ij}$. При этом $|A_i| \leq t$, $|B_j| \leq nt + 1$, $|C_{ij}| \leq 2t(nt + 1)$, следовательно, временная и емкостная сложности этапа 8 состав-

ляют $O(m^2 \log^2(nt))$ и $O(m \log(nt))$ соответственно. Отметим, что $w \leq 2t(nt+1)$, т. е. $w = O(nt^2)$.

На этапе 9 для $j = 1, \dots, w$, где $w = O(nt^2)$, вычисляются и сравниваются с 1 значения d_j . Для этого осуществляется mw сложений, mw умножений и mw операций нахождения НОД, причем $A_i + jB_i = O(n^2t^3)$, поэтому временная и емкостная сложности этапа 9 составляют $O(mnt^2 \log^2(nt))$ и $O(nt)$.

Этап 10 требует лишь $O(1)$ операций и памяти.

Суммируя сложности всех этапов, убеждаемся в конечности процедуры A и справедливости оценок $O(N^2 \log^2 N)$ для временной и $O(N \log N)$ для емкостной сложности алгоритма A . Теорема доказана.

2. Распознавание свойств одной функции

Если фиксированы $p, b, p_1, \dots, p_r$, то принадлежность заданной функции из $L(\mathbb{Z})$ каждому из классов L^+ , D , $S(p)$, $C(p_1 \cdots p_r)$, $U(b, p)$ может быть осуществлена на основе определений этих классов. Рассмотрим несколько иные задачи. Для фиксированной функции $f(x_1, \dots, x_n) \in L(\mathbb{Z})$ требуется ответить на следующие вопросы.

1. Существует ли класс $C(p_1 \cdots p_r)$, содержащий f ?
2. Существует ли класс $S(p)$, содержащий f ?
3. Существует ли класс $U(b, p)$, содержащий f ?

Очевидным представляется

Утверждение 1. Если $f \in D$, то $f \in C(p)$ для некоторого простого p .

Если выяснилось, что $f \in D$, то ответ на вопрос 1 положителен и дальнейшей проверки не требуется. Если же $f \notin D$, то для ответа на первый вопрос можно выполнить этапы 3–5 алгоритма A при $m = 1$ с временной сложностью $O(n^2t \log^2 t) = O(N_1^2 \log^2 N_1)$ и емкостной сложностью $O(nt \log t) = O(N_1 \log N_1)$, где $N_1 = nt$.

Ответ на вопрос 2 можно получить, выполнив этапы 6 и 7 алгоритма A при $m = 1$ с временной сложностью $O(n \log(nt)) + O(\log^2(nt)) = O(N_1 \log^2 N_1)$ и емкостной сложностью $O(\log N_1)$.

Для ответа на вопрос 3 можно выполнить этапы 6, 8, 9 алгоритма A при $m = 1$, но можно это сделать и проще.

Утверждение 2. Функция $f(x_1, \dots, x_n) = a_0 + a_1x_1 + \dots + a_nx_n$ не содержится ни в одном классе $U(b, p)$ тогда и только тогда, когда $a_0 \in \{1, -1\}$ и $a_1 + \dots + a_n = 1$.

Доказательство. Если $a_0 \notin \{1, -1\}$, то существует простой делитель p числа a_0 , и для него $f \in U(0, p)$.

Пусть $x_1 \equiv \dots \equiv x_n \equiv b \pmod{p}$ для некоторых p и b . В этом случае эквивалентно условию $f(x_1, \dots, x_n) \equiv b \pmod{p}$

$$(a_1 + \dots + a_n - 1)b \equiv -a_0 \pmod{p}. \quad (6)$$

Если $a_1 + \dots + a_n = 1$ и $a_0 \in \{1, -1\}$, то сравнение (6) невозможно.

Если $a_1 + \dots + a_n \neq 1$, то найдется простое p , не делящее $a_1 + \dots + a_n - 1$. Для такого p и $b \equiv -a_0(a_1 + \dots + a_n - 1)^{-1} \pmod{p}$ получим класс $U(b, p)$, содержащий f . Утверждение доказано.

Следствие 1. *Нахождение ответа на вопрос 3 для функции n переменных имеет временную сложность $O(n \log t)$ и емкостную сложность $O(1)$.*

3. Распознавание относительной полноты

Проблема *полноты относительно* заданного класса K (не обязательно замкнутого) состоит в выяснении того, полна ли система, содержащая класс K . Рассмотрим эту проблему в $L(\mathbb{Z})$ для классов $K = K_0, K_1, K_E, K_M, K_S$, где

K_0 — класс всех функций с коэффициентом $a_0 = 0$ (класс всех нечетных функций $f(x_1, \dots, x_n) = -f(-x_1, \dots, -x_n)$);

K_E — классы всех функций с коэффициентом a_0 , кратным E , $E \geq 2$;

K_1 — класс всех функций, зависящих не более чем от одной переменной;

K_M — класс всех функций, сохраняющих модуль, т. е. эквивалентность $x \sim y \Leftrightarrow |x| = |y|$ на $\mathbb{Z}$;

K_S — класс всех сюръекций (он не замкнут).

Утверждение 3. *Если $K_0 \subseteq F$, то система F полна в $L(\mathbb{Z})$ тогда и только тогда, когда она не содержится ни в одном из классов $U(0, p)$.*

Нетрудно проверить, что из всех предполных в $L(\mathbb{Z})$ классов только классы $U(0, p)$, где число p — произвольное простое, содержат K_0 .

Условие $f \in U(0, p)$ равносильно делимости на p коэффициента a_0 .

Следствие 2. *Если F_1 — конечная системы размера N , то полнота системы $K_0 \cup F_1$ распознаётся с временной сложностью $O(N)$ и емкостной сложностью $O(1)$.*

Утверждение 4. *Если $K_E \subseteq F$, то система F полна в $L(\mathbb{Z})$ тогда и только тогда, когда она не содержится ни в одном из классов $U(0, p)$, где $p|E$.*

Доказательство. Функция $x - y$ из K_E не принадлежит ни одному из классов $L^+, D, C(p_1 \dots p_r), S(p)$, поэтому система F не содержится ни в одном из указанных классов. Далее, пусть $f(x_1, \dots, x_n) \in K_E$. Если $E \not\equiv 0 \pmod{p}$, то для любого b функция f не принадлежит классу $U(b, p)$. Если $p|E$, то $p|a_0$ и условие $f \in U(b, p)$ эквивалентно сравнению (6). Класс K_E содержит функции f , для которых $a_1 + \dots + a_n \not\equiv 1 \pmod{p}$. Тогда из условия (6) следует $b = 0$. Таким образом, из всех предполных в $L(\mathbb{Z})$ классов только классы $U(0, p)$, где $p|E$, содержат K_E . Для полноты системы F необходимо и достаточно, чтобы она не содержалась ни в одном из таких классов $U(0, p)$. Утверждение доказано.

Следствие 3. *Если F_1 — конечная системы размера N , то полнота системы $K_E \cup F_1$ распознаётся с временной сложностью $O(N)$ и емкостной сложностью $O(1)$.*

Утверждение 5. *Если $K_1 \subseteq F$, то система F полна в $L(\mathbb{Z})$ тогда и только тогда, когда она не содержится ни в одном из классов D и $C(p_1 \dots p_r)$.*

Доказательство. Класс K_1 содержит все константы, функции $-x$ и $x + 1$. При этом константы не принадлежат классам $S(p)$, $-x \notin L^+$, функция $x + 1$ не входит в классы $U(b, p)$. С другой стороны, $K_1 \subset D$ и $K_1 \subset C(p_1 \cdots p_r)$ для всех $r, p_1, \dots, p_r$. Таким образом, система F не содержится ни в одном из классов $S(p)$, $U(b, p)$, L^+ , и ее полнота зависит только от включения в классы D и $C(p_1 \cdots p_r)$. Утверждение доказано.

Рассмотрим проблему полноты относительно класса K_M .

Утверждение 6 ([6]). *Полином произвольной степени с целыми, рациональными или вещественными коэффициентами сохраняет модуль тогда и только тогда, когда каждая его переменная имеет либо только четное (включая 0), либо только нечетное число вхождений в каждое слагаемое полинома (моном).*

Следствие 4. *Замкнутый класс K_M в функциональной системе $L(\mathbb{Z})$ состоит в точности из всех констант и всех функций вида ax , $a \in \mathbb{Z}$.*

Утверждение 7. *Если $K_M \subseteq F$, то система F полна в $L(\mathbb{Z})$ тогда и только тогда, когда она не содержится ни в одном из классов D и $C(p_1 \cdots p_r)$.*

Доказательство. Функция $-x$ из K_M не принадлежит классу L^+ , любая константа не принадлежит классам $S(p)$, константы C , удовлетворяющие условию $C \not\equiv b \pmod{p}$, не входят в классы $U(b, p)$. Таким образом, K_M не содержится в классах $L^+, S(p), U(b, p)$, но $K_M \subset D$ и $K_M \subset C(p_1 \cdots p_r)$ для всех $r \geq 1, p_1, \dots, p_r$. Для полноты системы F необходимо и достаточно, чтобы она не содержалась в классах D и $C(p_1 \cdots p_r)$. Утверждение доказано.

Если F_1 — конечная системы размера N , то для распознавания полноты систем $K_1 \cup F_1$ и $K_M \cup F_1$ можно применить этапы 2–5 алгоритм A с временной сложностью $O(N^2 \log^2 N)$ и емкостной сложностью $O(N \log N)$.

Проблема полноты относительно K_S решается просто.

Утверждение 8. *Если $K_S \subseteq F$, то система F полна в $L(\mathbb{Z})$.*

Доказательство. Сюръекция $x - y$ не входит ни в один из классов $L^+, D, C(p_1 \cdots p_r), S(p)$, сюръекция $x + 1$ не принадлежит ни одному из классов $U(b, p)$. Следовательно, класс K_S не содержится ни в одном из предполных в $L(\mathbb{Z})$ классов, поэтому $[K_S] = L(\mathbb{Z})$. Утверждение доказано.

Тем самым найдено достаточное условие полноты.

Для применения критериев полноты относительно K_0, K_E, K_M, K_1 полезно знать базисы в этих классах. Нетрудно проверить справедливость следующего утверждения.

Утверждение 9. *Системы $\{x - y\}$ и $\{E, x - y\}$ есть базисы в классах K_0 и K_E соответственно. Системы*

$$\{0, 1, -x\} \cup \left(\bigcup_p \{px\} \right), \quad \{0, -x, x - 1\} \cup \left(\bigcup_p \{px\} \right)$$

являются базисами бесконечно-порожденных классов K_M и, соответственно, K_1 .

Список литературы

1. Дарсалия В. Ш., Условия полноты для полиномов с натуральными, целыми и рациональными коэффициентами. *Фундам. и прикл. матем.* (1996) **2**, №2, 365–374.
2. Дарсалия В. Ш., Об алгоритмической разрешимости свойства выразимости для полиномов. *Фундам. и прикл. матем.* (1999) **5**, №3, 931–935.
3. Дарсалия В. Ш., Относительная полнота для функциональных систем полиномов. *Фундам. и прикл. матем.* (2002) **8**, №4, 967–977.
4. Мамонтов А. И., Мещанинов Д. Г., Проблема полноты в функциональной системе линейных полиномов с целыми коэффициентами. В сб.: *Труды VI Междунар. конф. "Дискретные модели в теории управляющих систем"*. Изд. отдел ф-та ВМиК МГУ, Москва, 2004, с. 50–52.
5. Мамонтов А. И., Мещанинов Д. Г., Проблема полноты в функциональной системе линейных полиномов с целыми коэффициентами. *Дискретная математика* (2010) **4**, 64–82.
6. Мещанинов Д. Г., Никитин И. В., Функционально замкнутые классы полиномов, сохраняющих некоторые эквивалентности на числовых множествах. *Вестник МЭИ* (2011) **6**, 14–23.
7. Мещанинов Д. Г., Никитин И. В., Классы сохранения пороговых разбиений в функциональных системах полиномов. *Вестник МЭИ* (2012) **6**, 132–141.
8. Алексеев В. Б., Емельянов Н. Р., Метод построения быстрых алгоритмов в k -значной логике. *Матем. заметки* (1985) **38**, №1, 148–156.
9. Емельянов Н. Р., Об одном подходе к построению эффективных алгоритмов распознавания полноты в многозначных логиках. *Матем. заметки* (1986) **39**, №5, 766–775.
10. Селезнева С. Н., О сложности распознавания полноты множеств булевых функций, реализованных полиномами Жегалкина. *Дискретная математика* (1997) **9**, №4, 24–31.
11. Селезнева С. Н., О сложности распознавания полноты систем функций k -значной логики, заданных полиномами. В сб.: *Труды VI Междунар. конф. "Дискретные модели в теории управляющих систем"*. Изд. отдел ф-та ВМиК МГУ, Москва, 2004, с. 76–79.
12. Горшков С. П., О сложности распознавания мультиаффинности, биюнктивности, слабый положительности и слабой отрицательности булевых функций. *Обозр. прикл. и промышл. матем.* (1997) **4**, №2, 216–237.
13. Селезнева С. Н., Полиномиальный алгоритм для распознавания принадлежности реализованной полиномом функции k -значной логики предполным классам самодвойственных функций. *Дискретная математика* (1998) **10**, №3, 64–72.
14. Voronenko A. A., On the complexity of monotonicity verification. *Proc. 15-th Ann. IEEE Conf. Comput. Complexity* (1999, Florence) 235–238.
15. Селезнева С. Н., Полиномиальный алгоритм для распознавания принадлежности функций k -значной логики, представленных полиномами, к предполным классам линейных функций. *Вестник МГУ. Сер. 15: Выч. матем. и киберн.*, №3, 2001, с. 40–43.
16. Вялый М. Н., Алгоритмические задачи с таблицами значений булевых полиномов. *Труды ИСП РАН* (2004) **6**, 51–64.
17. Бухман А. В., О сложности распознавания сохранения функциями многозначных логик, заданных полиномами, некоторых множеств. *Вестник МГУ. Сер. 15: Выч. матем. и кибернетика*, №3, 2011, с. 38–43.

18. Бухман А. В., Полиномиальный алгоритм распознавания функций, инвариантных относительно преобразования Мёбиуса. В сб.: *Материалы XI Междунар. сем. "Дискретная математика и ее приложения"*. Изд-во мех.-мат. ф-та МГУ, Москва, 2012, с. 188–190.
19. Бухман А. В., О применении обобщенных полиномов для построения алгоритмов распознавания свойств k -значных функций, заданных полиномами. *Дискретная математика* (2012) **24**, №3, 51–61.
20. Slupecki J., Kryterium pełności wielowartościowych systemów logiki zdań. *Comptes Rendus des Séances de la Société des Sciences et de Lettres de Varsovie. Cl. III* (1939) **32**, 102–109.
21. Яблонский С. В., Функциональные построения в k -значной логике. *Труды Матем. ин-та им. В. А. Стеклова АН СССР* (1958) **51**, 5–142.
22. Саломая А., Некоторые критерии полноты для множеств функций многозначной логики. *Киберн. сб.* (1964) **8**, 7–32.
23. Гаврилов Г. П., О функциональной полноте в счетнозначной логике. *Пробл. киберн.* (1965) **15**, 5–64.
24. Мамонтов А. И., Проблема относительной полноты в функциональной системе линейных полиномов с рациональными коэффициентами. *Вестник МЭИ* (2011) **6**, 133–142.
25. Коблиц Н., *Курс теории чисел и криптографии*. ТВП, Москва, 2001.

Статья поступила 03.02.2013.

Переработанный вариант поступил 03.09.2013.