

УДК 519.71

О СЛОЖНОСТИ ПРОБЛЕМЫ ВЫПОЛНИМОСТИ СИСТЕМЫ ФУНКЦИОНАЛЬНЫХ БУЛЕВЫХ УРАВНЕНИЙ ^{*)}

В. С. Фёдорова

Аннотация. Рассматриваются функциональные булевы уравнения и проблема распознавания выполнимости для них, которая состоит в следующем: существуют ли булевы функции, удовлетворяющие данному функциональному уравнению. Устанавливаются верхняя и нижняя оценки сложности проблемы выполнимости системы функциональных булевых уравнений, тем самым обосновывается невозможность её решения методом, который существенно проще непосредственного перебора.

Ключевые слова: функциональное булево уравнение, выполнимость, сложность.

Введение

Один из часто используемых способов задания функций и множеств функций в математике — задание с помощью систем функциональных уравнений. Этот способ применяется в теории рекурсивных функций и теории автоматов и обладает простотой и краткостью записи. При определении функциональных уравнений используются функциональные и индивидуальные переменные, а также различные функциональные и индивидуальные константы и, возможно, функционалы и операторы. Немало подобных функциональных уравнений можно найти и в теории функций многозначной логики (особенно в теории булевых функций [9, 10]). С помощью функциональных уравнений можно определять множества монотонных, самодвойственных, линейных и многих других функций.

В настоящей работе рассматриваются функциональные булевы уравнения и проблема распознавания выполнимости для них, которая состоит в следующем: существуют ли булевы функции, удовлетворяющие данному функциональному уравнению. Во всех случаях разрешимых

^{*)}Исследование выполнено при финансовой поддержке гранта Президента РФ (МД–757.2011.9).

проблем выполнимости существует тривиальный алгоритм решения проблемы выполнимости, перебирающий все возможные значения входящих в формулу переменных и вычисляющий соответствующие значения формулы. Такой переборный алгоритм предоставляет достаточно грубую верхнюю оценку временной сложности решения проблемы распознавания выполнимости. Наиболее трудной задачей, как правило, является получение нетривиальной нижней оценки временной сложности решения упомянутой проблемы. Для получения нижних оценок используется моделирование абстрактных вычислительных устройств формулами рассматриваемого типа. Величина нижней оценки при этом может существенно зависеть как от выбранных вычислительных устройств, так и от способа моделирования вычислений на этих устройствах. Чаще всего в качестве абстрактного вычислительного устройства используются различные варианты машин Тьюринга (например, с ограничениями на время или зону работы), а моделирование состоит в описании с помощью формул процесса вычисления на этих машинах.

Целью данной работы является получение верхней и нижней оценок временной сложности решения проблемы распознавания выполнимости для функциональных булевых уравнений.

1. Основные понятия и терминология

Дадим необходимые определения. Пусть $E_2 = \{0, 1\}$, P_2 — множество всех функций на E_2 (множество булевых функций), $P_2^{(n)}$ — множество всех n -местных булевых функций.

В определении языка функциональных уравнений придерживаемся терминологии из [7]. В качестве *функциональных констант* будем использовать булевы функции-константы 0 и 1.

Наряду с функциональными константами рассматриваем функциональные булевы переменные (коротко, функциональные переменные). Для обозначения n -местных функциональных переменных используем символы $\varphi_i^{(n)}$. Областью значений функциональной переменной $\varphi_i^{(n)}$ служит множество $P_2^{(n)}$. В случае, когда это не приводит к недоразумению, верхние индексы у функциональных переменных будем опускать.

Помимо функциональных переменных используем обычные *индивидуальные переменные* x_1, x_2, \dots с областью значений E_2 . Иногда для лучшего понимания структуры формулы в качестве индивидуальных переменных будем использовать переменные y, z, v, w .

Определим понятие *терма*. Всякая индивидуальная переменная и функциональная константа 0 или 1 есть терм. Если t_1, \dots, t_n — термы, $\varphi_j^{(n)}$ —

функциональная переменная, то выражение $\varphi_j^{(n)}(t_1, \dots, t_n)$ есть терм.

Равенством называем любое выражение вида $t_1 = t_2$, где t_1, t_2 — термы. Равенства $t_1 = t_2$ и $t_2 = t_1$ в дальнейшем не различаем. Равенства называем также *функциональными булевыми уравнениями*.

Пусть $t_1 = t_2$ — функциональное булево уравнение и $\varphi_{i_1}^{(n_1)}, \dots, \varphi_{i_m}^{(n_m)}$ — функциональные переменные, входящие в уравнение $t_1 = t_2$. *Решением уравнения* $t_1 = t_2$ называем систему булевых функций $\{f_{j_1}^{(n_1)}, \dots, f_{j_m}^{(n_m)}\}$, которая после замены каждой функциональной переменной $\varphi_{i_s}^{(n_s)}$ соответствующей функциональной константой $f_{j_s}^{(n_s)}$ превращает уравнение $t_1 = t_2$ в тождество (относительно всех входящих в уравнение индивидуальных переменных).

Пусть T — конечная система уравнений. *Решением системы уравнений* T называем систему булевых функций, которая является решением каждого уравнения, входящего в T . Будем говорить, что данные значения всех функциональных переменных, входящих в систему T , *выполняют* эту систему, если эти значения функциональных переменных принадлежат множеству решений данной системы. Конечная система уравнений T *выполнима*, если на множестве P_2 существуют значения всех функциональных переменных, которые выполняют систему T .

Для языка функциональных уравнений сформулируем следующую алгоритмическую проблему: *по произвольной конечной системе уравнений T выяснить, является ли T выполнимой.*

2. Верхняя оценка сложности проблемы выполнимости

Рассмотрим систему функциональных булевых уравнений с функциональными константами 0 и 1. В дальнейших рассуждениях будет удобнее пользоваться записью этой системы в виде одного закодированного слова. Для того чтобы уменьшить его длину и избавиться от необходимости кодировать скобки, при записи термов будем пользоваться *польской инверсной записью* [1] (или *постфиксной нотацией* в отличие от классической инфиксной). Общие правила построения выражения в постфиксной нотации определяются индуктивным следующим образом:

- (i) если E — переменная или константа, то её польская инверсная запись есть E ;
- (ii) если $E = F(x_1, x_2, \dots, x_n)$, то её польская инверсная запись есть $x_1 x_2 \dots x_n F$;
- (iii) если $E = (E_1)$, то её польская инверсная запись есть E_1 в постфиксной нотации.

Таким образом, в польской инверсной записи все аргументы располагаются перед знаком операции. Выражение читается слева направо. Когда в нём встречается символ операции, выполняется соответствующая операция над необходимым количеством последних встретившихся перед ним аргументов в порядке их записи. Результат операции заменяет в выражении последовательность её аргументов и символ, её обозначающий, после чего выражение вычисляется дальше по тому же правилу. Результатом вычисления выражения становится результат последней вычисленной операции.

Если индексы у индивидуальных и функциональных переменных записывать в двоичной системе счисления, то каждое уравнение можно представить в виде слова в алфавите, содержащем символы 0, 1, x , φ , $=$, левую и правую скобки и запятую (индексы записываются в строку сразу после символов переменных x и φ : сначала верхний индекс, а затем через запятую нижний индекс). Полученные после этих преобразований уравнения закодируем словами в алфавите $\{0, 1, \Lambda\}$ по следующим правилам:

1) аргументы функциональных переменных разделяются пробелом Λ , между записью последнего аргумента и кодом функциональной переменной также ставится пробел Λ ;

2) знак $=$ заменяется парой пробелов $\Lambda\Lambda$, между уравнениями системы ставятся три пробела $\Lambda\Lambda\Lambda$;

3) символы функциональной константы 0 и индивидуальной переменной x кодируются нулём, символ функциональной константы 1 и функциональной переменной φ — единицей, после кода индивидуальной переменной без пробела записывается её двоичный номер, после кода функциональной переменной также без пробела записывается двоичный код числа её переменных и затем через пробел — двоичный код номера переменной.

В качестве иллюстрации закодируем функциональное булево уравнение $\varphi_3^{(3)}(x_1, x_2, 0) = \varphi_4^{(2)}(0, 1)$:

$$1) \varphi_{11}, 11(x_1, x_{10}, 0) = \varphi_{10}, 100(0, 1);$$

$$2) x_1, x_{10}, 0\varphi_{11}, 11 = 0, 1\varphi_{10}, 100;$$

$$3) 01\Lambda 010\Lambda 0\Lambda 111\Lambda 11\Lambda\Lambda 0\Lambda 1\Lambda 110\Lambda 100.$$

Легко заметить, что такое кодирование является эффективным и однозначным.

В дальнейшем, оценивая сложность проблемы выполнимости функциональных булевых уравнений, будем иметь в виду, что системы функциональных булевых уравнений задаются в предложенном выше коде,

а сложность проблемы выполнимости оценивается через длину этого кода.

Пусть в системе функциональных булевых уравнений содержатся m различных функциональных переменных, зависящих соответственно от n_1, n_2, \dots, n_m аргументов. Пусть также $n = \max\{n_1, n_2, \dots, n_m\}$. Тогда для проверки выполнимости этой системы методом полного перебора необходимо проверить не более

$$2^{2^{n_1}} \cdot 2^{2^{n_2}} \cdot \dots \cdot 2^{2^{n_m}} \leq (2^{2^n})^m \quad (1)$$

вариантов. Оценим сверху величины n и m через длину l кода системы функциональных булевых уравнений.

Подсчитаем длину записи всех функциональных переменных, входящих в систему. Пусть $r \geq 0$. Чтобы записать в двоичной системе счисления все числа от 1 до $2^{r+1} - 1$, требуется ровно $r2^{r+1} + 1$ двоичных символов. Докажем это утверждение математической индукцией.

БАЗИС ИНДУКЦИИ: если $r = 0$, то все числа от 1 до $2^{0+1} - 1 = 1$ можно записать с помощью одного двоичного символа.

ИНДУКТИВНЫЙ ПЕРЕХОД: пусть для $r = i - 1$ все числа от 1 до $2^i - 1$ записываются с использованием ровно $(i - 1)2^i + 1$ двоичных символов. Тогда для $r = i$ числа $1, 2, \dots, 2^i - 1, 2^i, \dots, 2^{i+1} - 1$ требуют для своей двоичной записи $(i - 1)2^i + 1 + 2^i(i + 1) = i2^{i+1} + 1$ символов.

Таким образом, запись только двоичных номеров m функциональных переменных требует не менее $(\log_2(m + 1) - 1)(m + 1) + 1$ двоичных разрядов. Из неравенства

$$(\log_2(m + 1) - 1)(m + 1) + 1 < l$$

следует, что m асимптотически не превосходит величины $l/\log_2 l$ [5]. Для n в качестве верхней оценки возьмём величину l , поскольку возможен случай, когда система функциональных уравнений состоит из единственной функциональной переменной, зависящей только от констант 0, 1. Тогда на основе неравенства (1) можно оценить сверху сложность алгоритма, решающего проблему выполнимости для систем функциональных булевых уравнений.

Поскольку в разд. 3 нижняя оценка получена с использованием недетерминированных вычислительных устройств, покажем, как и с какой сложностью проблему выполнимости можно решить недетерминированным образом. Опишем коротко недетерминированную процедуру, которая по коду системы функциональных булевых уравнений проверяет её

выполнимость. Прежде всего, эта процедура недетерминированным образом генерирует (угадывает) вектор-строки булевых функций для всех функциональных переменных, входящих в рассматриваемую систему. Это занимает память, не превосходящую величины $m2^n$, и выполняется за время, линейное относительно $m2^n$. Затем, пользуясь кодом представленной системы уравнений, процедура проверяет выполнимость системы уравнений на множестве угаданных булевых функций. Эта часть работы процедуры может быть исполнена за время, квадратичное относительно $m2^n$. Таким образом, в случае выполнимости системы уравнений проверка её выполнимости может быть проведена за время $((l2^l)/\log_2 l)^2$. Детерминизация описанной процедуры приводит, разумеется, к экспоненциальному увеличению времени.

3. Нижняя оценка сложности проблемы выполнимости

Для получения нижней оценки временной сложности решения поставленной проблемы будут использоваться *конечные недетерминированные однородные структуры* [3, 4] (ОС), которые по вычислительным возможностям эквивалентны недетерминированным линейно ограниченными автоматам и недетерминированным машинам Тьюринга, работающим с линейной зоной [2]. Выбор ОС в качестве основных вычислительных устройств обусловлен чисто техническими причинами: работа ОС сравнительно просто моделируется функциональными булевыми уравнениями.

По произвольной ОС будет эффективно построена система функциональных булевых уравнений \mathcal{T} , которая выполнима в том и только том случае, когда ОС преобразует начальную конфигурацию в заключительную. Тем самым сложность проверки выполнимости системы уравнений \mathcal{T} будет оцениваться снизу сложностью преобразования начальной конфигурации ОС в заключительную и (в меньшей степени) сложностью построения системы \mathcal{T} по ОС.

Введём необходимые понятия [6]. Пусть $A = (Q, g)$ — конечный недетерминированный автомат с множеством состояний $Q = \{q_0, q_1, \dots, q_{r-1}\}$, двумя входами и двумя выходами и функцией переходов $g : Q^3 \rightarrow 2^Q \setminus \{\emptyset\}$ (входным алфавитом автомата A является алфавит Q).

Для любого натурального числа m через M_m обозначим *однородную структуру*, т. е. линейно упорядоченную последовательность из m копий A_1, A_2, \dots, A_m автомата A , в которой каждый автомат A_i , $1 < i < m$, связан с автоматами A_{i-1} и A_{i+1} . Крайние автоматы A_1 и A_m связаны только с автоматами A_2 и A_{m-1} соответственно.

ОС M_m работает в дискретном времени $t = 1, 2, \dots$. В каждый момент времени $t+1$ состояние автомата A_i , $1 < i < m$, определяется с помощью функции переходов g состояниями автоматов A_{i-1} , A_i , A_{i+1} в момент времени t . Будем считать, что при вычислении состояний автоматов A_1 и A_m вместо первого и третьего аргументов в g всегда подставляются значения q_1 и q_2 из Q соответственно.

Согласно приведённым определениям функционирование ОС M_m происходит следующим образом. В начальный момент времени автоматы A_1, A_2, \dots, A_m устанавливаются в некоторые состояния $q_{i1}, q_{i2}, \dots, q_{im}$. Назовём этот набор состояний *инициальным*. В следующий момент времени вектор-состоянием (или *конфигурацией*) ОС M_m будет набор

$$(q_{j1}, q_{j2}, \dots, q_{jm}),$$

где $q_{j1} \in g(q_1, q_{i1}, q_{i2})$, $q_{j2} \in g(q_{i1}, q_{i2}, q_{i3})$, \dots , $q_{jm-1} \in g(q_{im-2}, q_{im-1}, q_{im})$, $q_{jm} \in g(q_{im-1}, q_{im}, q_2)$. Затем к полученным состояниям вновь «применяется» g , и т. д.

Выделим состояние $q_0 \in Q$ и назовём его заключительным. Наложим ограничения на функцию переходов g : если хотя бы один из её аргументов равен q_0 , то значение функции g равно в точности $\{q_0\}$:

$$g(q_0, q_i, q_j) = g(q_i, q_0, q_j) = g(q_i, q_j, q_0) = \{q_0\},$$

где $q_i, q_j \in Q$. Таким образом, если все автоматы ОС M_m придут в заключительное состояние q_0 , то в дальнейшем конфигурация ОС M_m не меняется. В этом случае будем считать, что ОС M_m закончила работу, а конфигурацию (q_0, q_0, \dots, q_0) назовём *заключительной*.

Будем говорить, что ОС M_m *допускает* конфигурацию $(q_{i1}, q_{i2}, \dots, q_{im})$, если M_m способна преобразовать $(q_{i1}, q_{i2}, \dots, q_{im})$ в заключительную конфигурацию (q_0, q_0, \dots, q_0) . Очевидно, что при этом число тактов преобразования не превосходит величины r^m . Множество всех конфигураций из Q^m , допускаемых ОС M_m , обозначим через $\text{Rec}(M_m)$.

Пользуясь техникой из [2], нетрудно показать, что для любого недетерминированного автомата A множество

$$R(A) = \bigcup_{m \geq 1} \text{Rec}(M_m)$$

принадлежит $\text{NSPACE}(L(n))$ — классу языков, распознаваемых недетерминированными машинами Тьюринга, работающими с линейной зоной $L(n)$ [8].

Теорема 1. Существует алгоритм временной сложности $O(m \log m)$, который для любого недетерминированного автомата A сводит проблему принадлежности конфигурации множеству $R(A)$ к проблеме выполнимости некоторой системы функциональных булевых уравнений.

ДОКАЗАТЕЛЬСТВО. Закодируем состояния q_0, q_1, \dots, q_{r-1} автомата A двоичными наборами длины $l = \lceil \log_2 r \rceil$ так, чтобы код заключительного состояния q_0 являлся единичным булевым вектором, и построим по функции переходов g автомата A булевы отображения

$$G_1 : E_2^{2l} \rightarrow 2^{E_2^l}, \quad G_2 : E_2^{3l} \rightarrow 2^{E_2^l}, \quad G_3 : E_2^{2l} \rightarrow 2^{E_2^l}$$

следующим образом.

(g1) Если набор (e_1, e_2, \dots, e_l) кодирует состояние q_1 , то в соответствии с соглашением о функционировании автомата A_1 в ОС M_m отображение G_1 задаётся равенством

$$G_1(x_1, x_2, \dots, x_{2l}) = G_2(e_1, e_2, \dots, e_l, x_1, x_2, \dots, x_{2l}).$$

(g2) Если наборы (a_1, a_2, \dots, a_l) , (b_1, b_2, \dots, b_l) , (c_1, c_2, \dots, c_l) суть коды состояний q_a, q_b, q_c автомата A , $q_d \in g(q_a, q_b, q_c)$ и набор (d_1, d_2, \dots, d_l) является кодом состояния q_d , то

$$(d_1, d_2, \dots, d_l) \in G_2(a_1, a_2, \dots, a_l, b_1, b_2, \dots, b_l, c_1, c_2, \dots, c_l).$$

На остальных двоичных наборах длины $3l$ (если они есть) отображение G_2 определяется произвольным образом.

(g3) Если набор $(e'_1, e'_2, \dots, e'_l)$ кодирует состояние q_2 , то аналогично получаем

$$G_3(x_1, x_2, \dots, x_{2l}) = G_2(x_1, x_2, \dots, x_{2l}, e'_1, e'_2, \dots, e'_l).$$

Назовём конкатенацию кодов m состояний автоматов A_1, A_2, \dots, A_m кодом соответствующей конфигурации ОС M_m . Пусть B_1 — булев вектор длины lm , кодирующий некоторый набор состояний. Тогда при функционировании ОС M_m под действием отображений G_1, G_2, G_3 вектор B_1 будет преобразовываться с течением времени в векторы $B_2, B_3, \dots, B_{2^{lm}}$. Поскольку $2^{lm} \geq r^m$, нет необходимости рассматривать дальнейшие преобразования приведённых выше векторов. Объединим эти векторы $B_2, B_3, \dots, B_{2^{lm}}$ в один вектор B' длины $lm(2^{lm} - 1)$.

Пусть для упрощения изложения числа lm и l являются степенями двойки: $lm = 2^n$, $l = 2^s$, где n, s — натуральные. В этом случае длина

вектора B_1 равна 2^n , и его можно рассматривать как вектор значений некоторой булевой функции, зависящей от n переменных. В то же время длина вектора B' равна $lm(2^{lm} - 1) = 2^{n+lm} - 2^n$.

Построим систему функциональных булевых уравнений с функциональными константами 0, 1 и двумя основными функциональными переменными $\varphi_1^{(n)}(x_1, \dots, x_n)$, $\varphi_2^{(n+2^n)}(x_1, \dots, x_{n+2^n})$. Также будем использовать селекторные функции $e_i^l(x_1, \dots, x_i, \dots, x_l) = x_i$, где l, i — натуральные числа, $1 \leq i \leq l$. Предполагаем, что все встречающиеся в системе булевы функции заданы системами функциональных булевых уравнений с использованием функциональных констант 0, 1 [7].

В строящейся системе функциональных булевых уравнений функциональная переменная $\varphi_1^{(n)}$ будет «определять» код начальной конфигурации B_1 ОС M_m , а блоки длины lm вектора значений функциональной переменной $\varphi_2^{(n+2^n)}$ — коды последующих конфигураций ОС M_m . При этом последний блок длины lm в построениях не используется и потому может принимать любые значения, а предпоследний блок той же длины есть код заключительной конфигурации, т. е. состоит из одних единиц.

1. Уравнения для функциональной переменной $\varphi_1^{(n)}(x_1, x_2, \dots, x_n)$. Пусть код начальной конфигурации B_1 имеет вид $a_1 a_2 \dots a_{lm}$. Тогда

$$\begin{aligned} \varphi_1^{(n)}(0, 0, \dots, 0, 0) &= a_1, \quad \varphi_1^{(n)}(0, 0, \dots, 0, 1) = a_2, \\ \dots, \quad \varphi_1^{(n)}(1, 1, \dots, 1, 1) &= a_{lm}. \end{aligned} \quad (2)$$

Число использованных символов (по порядку, при фиксированном l) при кодировании системы функциональных уравнений (2) в алфавите $\{0, 1, \Lambda\}$ равно

$$lm(n + \log_2 n) \asymp mn \asymp m \log_2 m.$$

2. Уравнения для функциональной переменной $\varphi_2^{(n+2^n)}(x_1, \dots, x_{n+2^n})$. Обозначим через $T_{y+1=x}$ терм, описывающий следующее утверждение: число, имеющее двоичное представление $x_1 x_2 \dots x_{lm}$, непосредственно следует за числом с двоичным представлением $y_1 y_2 \dots y_{lm}$, или

$$(y_1 y_2 \dots y_{lm})_2 + 1 = (x_1 x_2 \dots x_{lm})_2,$$

причём $(11 \dots 1)_2 + 1 = (00 \dots 0)_2$. Тогда

$$\begin{aligned} T_{y+1=x} &= ((x_{lm} \sim \bar{y}_{lm}) \& (x_{lm-1} \sim (y_{lm-1} \oplus y_{lm} \& \bar{x}_{lm}))) \\ &\quad \& \dots \& (x_1 \sim (y_1 \oplus y_2 \& \bar{x}_2))), \end{aligned}$$

где \sim и \oplus — булевы функции эквивалентность и сложение по модулю 2 соответственно.

Длина кода терма $T_{y+1=x}$ в алфавите $\{0, 1, \Lambda\}$ равна по порядку

$$lm \log_2 lm \asymp m \log_2 m.$$

Терм $T_1 = (x_1 x_2 \dots x_{lm} \sim 0)$ обеспечивает работу со всеми блоками длины lm вектора значений функциональной переменной φ_2 кроме последнего, не используемого в построении. Порядок его длины в закодированном в алфавите $\{0, 1, \Lambda\}$ виде — $m \log_2 m$.

Терм $T_2 = (x_1 \vee x_2 \vee \dots \vee x_{lm} \sim 1)$ позволяет работать со всеми блоками длины lm вектора значений функциональной переменной φ_2 кроме первого, который обрабатывается особым образом из-за необходимости задания начальной конфигурации. Его длина в закодированном виде по порядку также $m \log_2 m$.

Код состояния крайнего левого автомата A_1 ОС M_m есть какое-то значение булева отображения G_1 , взятого от кодов состояний автоматов A_1 и A_2 в предыдущий момент времени (терм T_{G_1} , приведённый ниже). Моменты времени определяются наборами переменных (x_1, \dots, x_{lm}) и (y_1, \dots, y_{lm}) , причём содержательно момент времени, определяемый набором переменных \tilde{x} , непосредственно следует за моментом времени, определяемым набором переменных \tilde{y} . (В последующих формулах для упрощения записи верхние индексы у переменных φ_1 и φ_2 опускаем.)

$$\begin{aligned} T_{G_1} = & \left(\left(\left(\varphi_2(x_1, \dots, x_{lm}, \underbrace{0, \dots, 0}_n) \sim e_1^l(G_1^1(\varphi_2(y_1, \dots, y_{lm}, \underbrace{0, \dots, 0}_n), \dots, \right. \right. \right. \\ & \left. \left. \left. \varphi_2(y_1, \dots, y_{lm}, \underbrace{(2l-1)_2}_n) \right) \right) \& \dots \& \left(\varphi_2(x_1, \dots, x_{lm}, \underbrace{(l-1)_2}_n) \right. \right. \\ & \left. \left. \sim e_1^l(G_1^1(\varphi_2(y_1, \dots, y_{lm}, \underbrace{0, \dots, 0}_n), \dots, \varphi_2(y_1, \dots, y_{lm}, \underbrace{(2l-1)_2}_n) \right) \right) \sim 1 \right) \\ & \vee \dots \vee \left(\left(\left(\varphi_2(x_1, \dots, x_{lm}, \underbrace{0, \dots, 0}_n) \sim e_1^l(G_1^r(\varphi_2(y_1, \dots, y_{lm}, \underbrace{0, \dots, 0}_n), \dots, \right. \right. \right. \\ & \left. \left. \left. \varphi_2(y_1, \dots, y_{lm}, \underbrace{(2l-1)_2}_n) \right) \right) \& \dots \& \left(\varphi_2(x_1, \dots, x_{lm}, \underbrace{(l-1)_2}_n) \right. \right. \\ & \left. \left. \sim e_1^l(G_1^r(\varphi_2(y_1, \dots, y_{lm}, \underbrace{0, \dots, 0}_n), \dots, \varphi_2(y_1, \dots, y_{lm}, \underbrace{(2l-1)_2}_n) \right) \right) \sim 1 \right), \end{aligned}$$

где $G_1^1, G_1^2, \dots, G_1^r$ — все возможные значения булева отображения G_1 на наборе переменных $(\varphi_2(y_1, \dots, y_{lm}, \underbrace{0, \dots, 0}_n), \dots, \varphi_2(y_1, \dots, y_{lm}, \underbrace{(2l-1)_2}_n))$,

причём если для какого-либо набора значений переменных y_1, \dots, y_{lm} отображение G_1 имеет r_1 значений, $r_1 < r$, то одно из принимаемых значений (любое) повторяется $r - r_1$ раз.

Подсчитаем порядок числа использованных в коде терма T_{G_1} символов:

$$r(lm \log_2 lm + n + \log_2(lm + n)) + 2l(lm \log_2 lm + n + \log_2(lm + n))l \asymp m \log_2 m.$$

Аналогично для крайнего правого автомата A_m ОС M_m :

$$\begin{aligned} T_{G_3} &= (((\varphi_2(x_1, \dots, x_{lm}, \underbrace{(lm - l)_2}_n) \sim e_1^l(G_3^1(\varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 2l)_2}_n), \\ &\quad \dots, \varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 1)_2}_n))) \& \dots \& (\varphi_2(x_1, \dots, x_{lm}, \underbrace{(lm - 1)_2}_n) \\ &\sim e_l^l(G_3^1(\varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 2l)_2}_n), \dots, \varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 1)_2}_n))) \sim 1) \\ &\vee \dots \vee ((\varphi_2(x_1, \dots, x_{lm}, \underbrace{(lm - l)_2}_n) \sim e_1^l(G_3^r(\varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 2l)_2}_n), \\ &\quad \dots, \varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 1)_2}_n))) \& \dots \& (\varphi_2(x_1, \dots, x_{lm}, \underbrace{(lm - 1)_2}_n) \\ &\sim e_l^l(G_3^r(\varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 2l)_2}_n), \dots, \varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 1)_2}_n))) \sim 1)), \end{aligned}$$

где $G_3^1, G_3^2, \dots, G_3^r$ — все возможные значения булева отображения G_3 на наборе переменных

$$(\varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 2l)_2}_n), \dots, \varphi_2(y_1, \dots, y_{lm}, \underbrace{(lm - 1)_2}_n)),$$

причём среди них, возможно, есть повторяющиеся (см. замечание после описания терма T_{G_1}).

Число использованных в коде терма T_{G_3} символов также равно по порядку $m \log_2 m$.

Таким образом, для блоков длины lm , определяемых набором переменных $(x_1, x_2, \dots, x_{lm})$, кроме первого (терм T_2) и последнего (терм T_1), коды состояний крайних левого (терм T_{G_1}) и правого (терм T_{G_3}) автоматов (соответственно первые и последние l символов конфигурации) определяются в соответствии с правилами функционирования ОС M_m при

условии, что блок, определяемый набором переменных $(y_1, y_2, \dots, y_{lm})$, является конфигурацией ОС M_m в предыдущий момент времени (терм $T_{y+1=x}$). Вышесказанное суммирует функциональное уравнение

$$T_1 \& T_2 \& T_{y+1=x} \rightarrow T_{G_1} \& T_{G_3} = 1. \quad (3)$$

Для наборов индивидуальных переменных $(v_1^1, \dots, v_{n-s}^1)$, $(v_1^2, \dots, v_{n-s}^2)$ и $(v_1^3, \dots, v_{n-s}^3)$ длины $n-s$ аналогичным терму $T_{y+1=x}$ образом выписываются термы $T_{v^1+1=v^2}$ и $T_{v^2+1=v^3}$, показывающие, что $(v_1^2, \dots, v_{n-s}^2)_2 = (v_1^1, \dots, v_{n-s}^1)_2 + 1$ и $(v_1^3, \dots, v_{n-s}^3)_2 = (v_1^2, \dots, v_{n-s}^2)_2 + 1$, с числом символов

$$n \log_2 n \asymp \log_2 m \log_2 \log_2 m.$$

Тогда терм T_{G_2} , утверждающий, что в конфигурации, имеющей номер (x_1, \dots, x_{lm}) , код любого не крайнего автомата с номером $(v_1^2, \dots, v_{n-s}^2)$ есть одно из значений булева отображения G_2 , взятого от кодов соответствующих трёх последовательно расположенных автоматов с номерами $(v_1^1, \dots, v_{n-s}^1)$, $(v_1^2, \dots, v_{n-s}^2)$ и $(v_1^3, \dots, v_{n-s}^3)$ в предыдущий момент времени, выглядит следующим образом:

$$\begin{aligned} T_{G_2} = & (((\varphi_2(x_1, \dots, x_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{0, \dots, 0}_s) \sim e_1^l(G_2^1(\varphi_2(y_1, \dots, y_{lm}, \\ & v_1^1, \dots, v_{n-s}^1, \underbrace{0, \dots, 0}_s), \dots, \varphi_2(y_1, \dots, y_{lm}, v_1^3, \dots, v_{n-s}^3, \underbrace{1, \dots, 1}_s)))) \\ & \& \dots \& (\varphi_2(x_1, \dots, x_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{1, \dots, 1}_s) \sim e_l^l(G_2^1(\varphi_2(y_1, \dots, y_{lm}, \\ & v_1^1, \dots, v_{n-s}^1, \underbrace{0, \dots, 0}_s), \dots, \varphi_2(y_1, \dots, y_{lm}, v_1^3, \dots, v_{n-s}^3, \underbrace{1, \dots, 1}_s)))) \sim 1) \\ & \vee \dots \vee (((\varphi_2(x_1, \dots, x_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{0, \dots, 0}_s) \sim e_1^l(G_2^r(\varphi_2(y_1, \dots, y_{lm}, \\ & v_1^1, \dots, v_{n-s}^1, \underbrace{0, \dots, 0}_s), \dots, \varphi_2(y_1, \dots, y_{lm}, v_1^3, \dots, v_{n-s}^3, \underbrace{1, \dots, 1}_s)))) \\ & \& \dots \& (\varphi_2(x_1, \dots, x_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{1, \dots, 1}_s) \sim e_l^l(G_2^r(\varphi_2(y_1, \dots, y_{lm}, v_1^1, \\ & \dots, v_{n-s}^1, \underbrace{0, \dots, 0}_s), \dots, \varphi_2(y_1, \dots, y_{lm}, v_1^3, \dots, v_{n-s}^3, \underbrace{1, \dots, 1}_s)))) \sim 1)), \end{aligned}$$

где G_2^1, \dots, G_2^r — все возможные значения булева отображения G_2 на

наборе

$$\begin{aligned}
& (\varphi_2(y_1, \dots, y_{lm}, v_1^1, \dots, v_{n-s}^1, \underbrace{0, \dots, 0}_s), \\
& \quad \dots, \varphi_2(y_1, \dots, y_{lm}, v_1^1, \dots, v_{n-s}^1, \underbrace{1, \dots, 1}_s), \\
& \varphi_2(y_1, \dots, y_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{0, \dots, 0}_s), \dots, \\
& \quad \varphi_2(y_1, \dots, y_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{1, \dots, 1}_s), \\
& \varphi_2(y_1, \dots, y_{lm}, v_1^3, \dots, v_{n-s}^3, \underbrace{0, \dots, 0}_s), \dots, \\
& \quad \varphi_2(y_1, \dots, y_{lm}, v_1^3, \dots, v_{n-s}^3, \underbrace{1, \dots, 1}_s)),
\end{aligned}$$

причём среди них, возможно, есть повторяющиеся.

Длина термина T_{G_2} (по порядку) в закодированном в алфавите $\{0, 1, \Lambda\}$ виде равна

$$\begin{aligned}
& r(lm \log_2 lm + (n - s) \log_2(n - s) + s \\
& \quad + 3l(lm \log_2 lm + (n - s) \log_2(n - s) + s))l \asymp m \log_2 m.
\end{aligned}$$

Объединим описанные термы в функциональное уравнение, утверждающее, что для любой конфигурации с номером $(x_1, x_2, \dots, x_{lm})$, кроме первой (терм T_2) и последней (терм T_1), код состояния автомата под номером $(v_1^2, \dots, v_{n-s}^2)$ определяется в соответствии с правилами функционирования ОС M_m (терм T_{G_2}) по кодам состояний трёх последовательно расположенных автоматов с номерами $(v_1^1, \dots, v_{n-s}^1)$, $(v_1^2, \dots, v_{n-s}^2)$, $(v_1^3, \dots, v_{n-s}^3)$ (термы $T_{v^1+1=v^2}$ и $T_{v^2+1=v^3}$) при условии, что блок, определяемый набором переменных $(y_1, y_2, \dots, y_{lm})$, является конфигурацией ОС M_m в предыдущий момент времени (терм $T_{y+1=x}$):

$$T_1 \& T_2 \& T_{y+1=x} \& T_{v^1+1=v^2} \& T_{v^2+1=v^3} \rightarrow T_{G_2} = 1. \quad (4)$$

Следующая группа функциональных уравнений необходима для задания первого блока значений длины lm функциональной переменной φ_2 , который получается согласно законам функционирования ОС M_m из начальной конфигурации, задаваемой вектором значений функциональной переменной φ_1 . Эти уравнения в некотором смысле являются частными случаями уравнений (3) и (4), однако имеют другие зависящие от φ_1

аргументы булевых отображений G_1 , G_2 и G_3 . Как и в предыдущем случае, термы $T_{G_1}^1$ и $T_{G_3}^1$ описывают функционирование крайних левого A_1 и правого автоматов A_m ОС M_m соответственно в момент времени, следующий за начальным:

$$\begin{aligned}
T_{G_1}^1 = & (((\underbrace{\varphi_2(0, \dots, 0)}_{n+lm} \sim e_1^l(G_1^1(\underbrace{\varphi_1(0, \dots, 0)}_n, \dots, \varphi_1(\underbrace{(2l-1)_2}_n)))) \\
& \& \dots \& (\underbrace{\varphi_2((l-1)_2)}_{n+lm} \sim e_l^l(G_1^1(\underbrace{\varphi_1(0, \dots, 0)}_n, \dots, \varphi_1(\underbrace{(2l-1)_2}_n)))) \sim 1) \\
& \vee \dots \vee (((\underbrace{\varphi_2(0, \dots, 0)}_{n+lm} \sim e_1^l(G_1^r(\underbrace{\varphi_1(0, \dots, 0)}_n, \dots, \varphi_1(\underbrace{(2l-1)_2}_n)))) \\
& \& \dots \& (\underbrace{\varphi_2((l-1)_2)}_{n+lm} \sim e_l^l(G_1^r(\underbrace{\varphi_1(0, \dots, 0)}_n, \dots, \varphi_1(\underbrace{(2l-1)_2}_n)))) \sim 1)), \\
T_{G_3}^1 = & (((\underbrace{\varphi_2(0, \dots, 0, (lm-l)_2)}_{lm} \sim e_1^l(G_3^1(\varphi_1(\underbrace{(lm-2l)_2}_n), \dots, \\
& \underbrace{\varphi_1((lm-1)_2)}_n))) \& \dots \& (\underbrace{\varphi_2(0, \dots, 0, (lm-1)_2)}_{lm} \sim e_l^l(G_3^1(\varphi_1(\underbrace{(lm-2l)_2}_n), \\
& \dots, \varphi_1(\underbrace{(lm-1)_2}_n)))) \sim 1) \vee \dots \vee (((\underbrace{\varphi_2(0, \dots, 0, (lm-l)_2)}_{lm} \\
& \sim e_1^l(G_3^r(\varphi_1(\underbrace{(lm-2l)_2}_n), \dots, \varphi_1(\underbrace{(lm-1)_2}_n))) \& \dots \& (\underbrace{\varphi_2(0, \dots, 0, (lm-1)_2)}_{lm} \\
& \sim e_l^l(G_3^r(\varphi_1(\underbrace{(lm-2l)_2}_n), \dots, \varphi_1(\underbrace{(lm-1)_2}_n)))) \sim 1)).
\end{aligned}$$

Длина каждого из термов $T_{G_1}^1$ и $T_{G_3}^1$ не превосходит по порядку

$$r(n + lm + 2ln)l = O(m).$$

Аналогично, терм $T_{G_2}^1$ описывает функционирование любого не крайнего автомата ОС M_m в момент времени, следующий за начальным:

$$\begin{aligned}
T_{G_2}^1 = & (((\underbrace{\varphi_2(0, \dots, 0, v_1^2, \dots, v_{n-s}^2, 0, \dots, 0)}_{lm} \\
& \sim e_1^l(G_2^1(\varphi_1(\underbrace{v_1^1, \dots, v_{n-s}^1}_s, \underbrace{0, \dots, 0}_s), \dots, \varphi_1(\underbrace{v_1^3, \dots, v_{n-s}^3}_s, \underbrace{1, \dots, 1}_s))))))
\end{aligned}$$

$$\begin{aligned}
& \& \dots \& (\varphi_2(\underbrace{0, \dots, 0}_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{1, \dots, 1}_s)) \\
& \sim e_l^l(G_2^1(\varphi_1(v_1^1, \dots, v_{n-s}^1, \underbrace{0, \dots, 0}_s), \dots, \varphi_1(v_1^3, \dots, v_{n-s}^3, \underbrace{1, \dots, 1}_s)))) \\
& \sim 1) \vee \dots \vee ((\varphi_2(\underbrace{0, \dots, 0}_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{0, \dots, 0}_s)) \\
& \sim e_l^l(G_2^r(\varphi_1(v_1^1, \dots, v_{n-s}^1, \underbrace{0, \dots, 0}_s), \dots, \varphi_1(v_1^3, \dots, v_{n-s}^3, \underbrace{1, \dots, 1}_s)))) \\
& \& \dots \& (\varphi_2(\underbrace{0, \dots, 0}_{lm}, v_1^2, \dots, v_{n-s}^2, \underbrace{1, \dots, 1}_s)) \\
& \sim e_l^l(G_2^r(\varphi_1(v_1^1, \dots, v_{n-s}^1, \underbrace{0, \dots, 0}_s), \dots, \varphi_1(v_1^3, \dots, v_{n-s}^3, \underbrace{1, \dots, 1}_s)))) \sim 1)).
\end{aligned}$$

Здесь $G_i^1, G_i^2, \dots, G_i^r$, $i = 1, 2, 3$, — все возможные значения отображения G_i на соответствующем наборе переменных, причём среди них, возможно, есть повторяющиеся.

Порядок числа использованных символов алфавита $\{0, 1, \Lambda\}$ в терме $T_{G_2}^1$:

$$r(lm + (n - s) \log_2(n - s) + s + 3l((n - s) \log_2(n - s) + s))l = O(m).$$

Таким образом, следующая система функциональных уравнений задаёт первый блок значений длины lm функциональной переменной φ_2 :

$$\begin{cases} T_{G_1}^1 = 1, \\ T_{G_3}^1 = 1, \\ (T_{v^1+1=v^2} \& T_{v^2+1=v^3}) \rightarrow T_{G_2}^1 = 1. \end{cases} \quad (5)$$

И, наконец, заключительное уравнение, которое только и может вызывать неразрешимость всей системы:

$$\varphi_2(\underbrace{1, 1, \dots, 1, 0}_{lm}, p_1, \dots, p_n) = 1. \quad (6)$$

Здесь p_1, \dots, p_n — произвольные различные индивидуальные переменные. Уравнение гарантирует, что предпоследний блок длины lm , описывающий последнюю рассматриваемую конфигурацию, состоит только из единиц: это означает, что данная конфигурация является заключительной. Длина этого уравнения равна $O(m)$.

Назовём *системой* \mathcal{T} объединение уравнений и систем уравнений (2)–(6). Система функциональных булевых уравнений \mathcal{T} является искомой по построению и имеет длину порядка $m \log_2 m$. Если числа lm и l не являются степенями двойки, то порядок длины системы \mathcal{T} , очевидно, не изменится.

Таким образом, описанный алгоритм для любого недетерминированного автомата A сводит проблему принадлежности слова длины m множеству $R(A)$ к проблеме выполнимости некоторой системы функциональных булевых уравнений за время порядка $m \log_2 m$, поскольку равномерное кодирование состояний автомата A может быть осуществлено эффективно с линейной сложностью. Теорема 1 доказана.

Следствие 1. *Нижняя оценка временной сложности недетерминированного распознавания выполнимости системы функциональных булевых уравнений на одноленточной машине Тьюринга, работающей с линейной зоной, по порядку не меньше $d^{\frac{l}{\log_2 l}}$, где l — длина входа, $d > 1$.*

ДОКАЗАТЕЛЬСТВО. Сложность решения проблемы принадлежности слова длины m множеству $R(A)$ по порядку логарифма совпадает со сложностью вычисления функций на одноленточных машинах Тьюринга, работающих с линейной зоной [2]. Таким образом, эту проблему нельзя решить за время, по порядку меньшее чем d^m , $d > 1$, т. е. существенно проще непосредственного перебора. Отсюда с учётом теоремы 1 получаем требуемое утверждение.

ЛИТЕРАТУРА

1. Ахо А. В., Сети Р., Ульман Д. Д. Компиляторы: принципы, технологии и инструменты. — М.: Вильямс, 2003. — 768 с.
2. Катериночкина Н. Н. Об эквивалентности некоторых вычислительных устройств // Кибернетика. — 1970. — № 5. — С. 27–31.
3. Кудрявцев В. Б., Подколзин А. С., Болотов А. А. Основы теории однородных структур. — М.: Наука, 1990. — 296 с.
4. Курода С. И. Классы языков и линейно ограниченные автоматы // Кибернетический сб. Вып. 9. — М.: Мир, 1972. — С. 36–51.
5. Ложкин С. А. Лекции по основам кибернетики. — М.: Изд-во фак-та вычисл. математики и кибернетики МГУ, 2004. — 147 с.
6. Марченков С. С. Итерация булевых (n, n) -операторов // Вест. МГУ. Сер. 15. Вычисл. математика и кибернетика. — 2006. — № 4. — С. 36–41.
7. Марченков С. С., Фёдорова В. С. О решениях систем функциональных булевых уравнений // Дискрет. анализ и исслед. операций. — 2008. — Т. 15, № 6. — С. 48–57.

8. Мучник А. А. Добавление переводчика к статьям «Об альтернировании, I, II» // Кибернетический сб. (новая серия). Вып. 20. — М.: Мир, 1983. — С. 141–158.
9. Ekin O., Foldes S., Hammer P. L., Hellerstein L. Equational characterizations of Boolean function classes // Discrete Math. — 2000. — Vol. 211. — P. 27–51.
10. Foldes S. Equational classes of Boolean functions via the HSP theorem // Algebra Univers. — 2000. — Vol. 44. — P. 309–324.

Фёдорова Валентина Сергеевна,
e-mail: FedorovaVS@cs.msu.ru

Статья поступила
18 июня 2012 г.