УДК 519.712.2

В.А. МОЩЕНСКИЙ, Д.В. МОЩЕНСКИЙ

О ДВУХ ПРОБЛЕМАХ, ОСНОВАННЫХ НА ПРОБЛЕМЕ ВЫПОЛНИМОСТИ

Two new problems are defined: a problem of complete satisfiability and a problem of recognition satisfiability. Lower and upper bound of its time complexities arc proven.

Иногда важно знать не только, существует ли выполняющий набор для данной КНФ, т. е. выходное значение 1 проблемы выполнимости (ВЫП), но и как много таких наборов имеется для данной КНФ. Например, известно [1, с. 128], что по выполняющему набору для данной КНФ в соответствующем графе находится клика, причем для различных наборов - разные клики. Эта проблема представляет интерес в теории графов, что отражается следующими понятиями.

Проблема полной выполнимости (ПВЫП) состоит в следующем: по заданной КНФ от n переменных выдать все двоичные наборы длины n, на которых данная КНФ имеет значение 1; если же эта КНФ реализует булеву функцию-константу нуль, то выдать $\varnothing$

Проблема распознавания выполнимости (РВЫП) состоит в следующем: по заданным КНФ от n переменных и некоторому списку двоичных наборов длины n выдать «да», если хотя бы на одном наборе из этого списка данная КНФ имеет значение 1, и «нет» - в противном случае.

Укажем на следующие соотношения между названными проблемами. Если при каждом решении проблемы ВЫП при ответе «да» можно из такого решения извлечь и сам выполняющий набор (что не всегда имеется), то в этом случае многократное применение проблемы ВЫП позволяет получить решение проблемы ПВЫП. В самом деле для данной КНФ решается проблема ВЫП и при ответе «да» извлекается выполняющий двоичный набор $(a_1, a_2, ..., a_n)$. Затем строится КНФ $F_1 = F \wedge \left(x_1^{\overline{a_1}} \vee x_2^{\overline{a_2}} \vee ... \vee x_n^{\overline{a_n}}\right)$. Для КНФ F_1 указанный двоичный набор является нулевым, и поэтому, если F_1 имеет выполняющий двоичный набор $(b_1, b_2, ..., b_n)$, он будет отличен от $(a_1, a_2, ..., a_n)$. Затем строим КНФ $F_2 = F_1 \wedge \left(x_1^{\overline{b_1}} \vee x_2^{\overline{b_2}} \vee ... \vee x_n^{\overline{b_n}}\right)$ и т. д., получая все выполняющие наборы $(a_1, a_2, ..., a_n)$, $(b_1, b_2, ..., b_n)$, ... исходной КНФ F.

Проблема РВЫП дает решение проблемы ВЫП, если в проблеме РВЫП за список двоичных наборов взять все двоичные наборы соответствующей длины.

Оценки временной сложности вычислений. Как известно [1, с. 126], исходную КНФ можно задавать словом в алфавите $A=\{0,1,\overline{x},x,(,)\}$, что мы и будем делать. Еще напомним, что число разрядов в двоичной записи натурального числа $n(n\geq 1)$ равно $\lfloor \log_2 n \rfloor+1$, а в дальнейшем вместо $\log_2 n$ пишем log n.

Через $F(n,l(n))$ обозначим класс всех совершенных КНФ (СКНФ) от n переменных, каждая из них содержит $l(n)$ элементарных дизъюнкций (ЭД), i-я из которых имеет вид

$$x_1^{a_{i1}} \vee x_2^{a_{i2}} \vee \ldots \vee x_n^{a_{in}},\ 1\leq i\leq l(n),\ 1\leq l(n)\leq 2^n.$$

В начальной конфигурации вычислений такая ЭД задается словом в алфавите A вида

$$\left(z1z10\ldots zb_{r1}zb_{r2}\ldots b_{rk(r)}\ldots zb_{n1}b_{n2}\ldots b_{nk(n)}\right),$$

где $z\in\{x,\overline{x}\}$, $\sum\limits_{i=1}^{k(r)} b_{ri}2^{k(r)-i}=r(1\leq r\leq n)$ и $k(r)=\lfloor \log r \rfloor+1.$ Значит, длина записи в начальной конфигурации вычислений одной такой ЭД есть

$$2+n+\left(1+2+\ldots+\left(\lfloor \log r \rfloor+1\right)\right)=2+n+O\left(\log^2 n\right)=\Omega(n).$$

Следовательно, длина m (n) записи в начальной конфигурации вычислений любой СКНФ из множества $F(n,l(n))$ равна $l(n)\Omega(n)$ и $m(n)=\Omega(nl(n)).$

Лемма **1.** Суммарная длина $S(n)$ попарно различных n слов по порядку не меньше nlogn, т. е. $S(n)=\Omega(n \log n)$.

Доказательство. Если рассматриваемые слова в однобуквенном алфавите, то для их попарного различия они должны иметь попарно различные длины и $S(n)=\Omega\left(n^2\right)$.

Если же эти слова в r-буквенном алфавите $(r\geq 2)$, то таких попарно различных слов, длина каждого из которых не больше $\log_r \sqrt{n}-1,$ будет «мало» по сравнению с n. Следовательно, почти все из этих и различных слов будут иметь длину, большую указанной. Лемма доказана.

Лемма **2.** Любая одноленточная детерминированная машина Тьюринга, решающая проблему ПВЫП для некоторой СКНФ из класса F(n,l(n)), где $l(n)\leq n^p$ и p - константа, сделает для почти всех n шагов $\Omega\left(m^a(n)2^{m^b(n)}\right)$ при $a=2/(p+1)$ и $b=1/(p+1)$.

Доказательство. Для каждой СКНФ из класса F(n,l(n)) согласно определениям проблемы ПВЫП и этого класса в заключительной конфигурации на ленте будет написано двоичных n-наборов $2^n-l(n)\geq 2^n-n^p=\Omega\left(2^n\right)$ для всех n, начиная с некоторого, так как p - константа, т. е. для почти всех n; длина их записи есть $n\cdot\Omega\left(2^n\right)=\Omega\left(n2^n\right)$. Следовательно, будет использовано дополнительно для почти всех n новых клеток ленты $\Omega\left(n2^n\right)-m(n)=\Omega\left(n\left(2^n-l(n)\right)\right)=\Omega\left(n2^n\right)$ (новых по сравнению с имеющимися в начальной конфигурации). Из этих новых клеток не менее половины, т. е. $\Omega\left(n2^{n-1}\right)$, будут расположены по одну сторону от участка клеток в начальной конфигурации. Тогда согласно [2, лемма 1, с. 97] над этой половиной клеток возникнут попарно различные следы вычисления. Значит, число всех шагов вычисления согласно лемме 1 будет $S\left(\Omega\left(n2^{n-1}\right)\right)=\Omega\left(n^2 2^n\right)$, откуда и следует требуемое заключение. Лемма доказана.

Теорема 1. Любая k-ленточная детерминированная машина Тьюринга $(k\geq 2)$, решающая проблему ПВЫП для некоторой СКНФ из класса, указанного в лемме 2, сделает для почти всех n шагов $\Omega\left(m^c(n)2^{m^b(n)/2}\right)$, где $c = a/2$ и a, b из леммы 2.

Доказательство. Пусть некоторая k-ленточная детерминированная машина Тьюринга $(k\geq 2)$ решает указанную проблему за $t(n)$ шагов для почти всех n. Согласно [2, лемма 1, с. 143] для нее может быть построена 1-ДМТ, решающая эту же проблему с временной сложностью $O\left(t^2(n)\right)$ для всех n, а значит, и для почти всех n. Тогда согласно лемме 2 получаем $O\left(t^2(n)\right)\geq\Omega\left(m^c(n)2^{m^b(n)}\right)$, откуда и следует требуемое заключение. Теорема доказана.

Замечание. Можно указать и верхнюю оценку временной сложности для машин из теоремы 1, для чего используем 3-ДМТ, работающую следующим образом. На второй ленте поочередно выписываются все двоичные наборы, начиная с нулевого; на каждом из них проверяется значение исходной КНФ, записанной в начальный момент на первой ленте. Если получится значение 1, то этот набор записывается на третью ленту, а на второй строится следующий двоичный набор при помощи добавления 1 к имеющемуся. Если d(n)-длина записи исходной КНФ, то всего потребуется $O\left(d(n)\cdot n\cdot 2^n\right)=O\left(d^2(n)\cdot 2^n\right)$ шагов для всех n; и при $d(n)=\Omega\left(2^{n/k}\right)$, где k - натуральная константа и $k\geq 1$, это число шагов есть полином степени $k+2$ от d(n); заметим, что на длине ленты, удовлетворяющей этому соотношению, записываются все КНФ от n переменных.

Теорема 2. Проблема РВЫП принадлежит классу P.

Доказательство. Пусть в начальной конфигурации на первой ленте заданы некоторая КНФ от n переменных, некоторый список двоичных наборов длины n и длина их записи есть $d(n)$. Тогда в этой записи как число двоичных наборов, так и длина записи самой КНФ меньше $d(n)$. Теперь, используя еще одну ленту, детерминированным образом за $O\left(d(n)\cdot n^2\cdot\log n\cdot d(n)\right)=O\left(d^5(n)\right)$ шагов легко проверить, выполняет ли один из наборов этого списка данную КНФ. Теорема доказана.

1. Мощенский А.В., Мощенский В.А. Математические основы информатики. Мн., 2002.
2. Трахтенброт Б.А. Сложность алгоритмов и вычислений. Новосибирск, 1967.

Поступила в редакцию 14.05.07.

Владимир Андреевич Мощенский - кандидат физико-математических наук, доцент кафедры дискретной математики и алгоритмики.

Дмитрий Владимирович Мощенский - математик-программист НИЛ кафедры дискретной математики и алгоритмики.