ПОЛИНОМИАЛЬНЫЕ НИЖНИЕ ОЦЕНКИ СЛОЖНОСТИ УСТАНОВЛЕНИЯ РАЗРЕШИМОСТИ ЛОГИКО-АРИФМЕТИЧЕСКИХ УРАВНЕНИЙ

Н. К. КОСОВСКИЙ

Math. – Mech. Faculty, Leningrad University Petrodvorets, Leningrad, USSR 198904

(Поступило 1. 10. 1982)

Abstract. It is proved that every predicate calculated by a Turing machine that runs in bounded time is representable in the form: "a logic-arithmetical equation has a solution in natural numbers whose lengths are not greater than the value of some function of the length of the predicate argument". A logic-arithmetical equation is an equation of the form $P+P''=0$ where P is a polynomial with integer coefficients and P'' is a sum of arithmetical conjunctions. The arithmetical conjunction is defined on binary strings as the bit by bit logical product (known from computer hardware). The time bounding function may be any polynomial or any greater function. The solution length bounding function is a quadratic polynomial of the time bounding function.

Our theorem gives a logic-arithmetical representation of every predicate from the famous class NP. For predicates from NP there is a strict hierarchy with respect to the degree of polynomials. So, using the result of [3], for any n we give a degree n polynomial lower bound for some NP-complete problem. These NP-complete problems have the following form: "a logic-arithmetical equation in natural numbers whose lengths are not greater than the value of some polynomial considered as a function of the length of the predicate argument, is solvable". Numerous NP-complete problems are presented in [2] but without such polynomial lower lounds.

Логико-арифметическим уравнением называем уравнение вида $P+P''=0$, где P – полином с целыми коэффициентами, а P'' – сумма арифметических конъюнкций, которые определяются над двоичными записями чисел аналогично логическому умножению в ЭВМ. По существу в статье предлагается некоторая строгая иерархия всех предикатов, вычислимых недетерминированными машинами Тьюринга за полиномиальное число шагов. Полиномами достаточно высокой степени являются нижние оценки рассматриваемых переборных задач. Это особенно важно с практической точки зрения, поскольку задача с нижней оценкой сложности, являющейся полиномом высокой степени, практически невычислима на ЭВМ.

Предлагаемая классификация предикатов позволяет упростить получение нижних оценок числа шагов недетерминированных машин Тьюринга, вычисляющих предикаты, задаваемые логическим образом посредством предваренной нормальной формы с префиксом, содержащим только ограниченные кванторы существования. Это позволяет связать теоретико-числовое представление предикатов и вычисление их недетерминированными машинами Тьюринга.

Настоящее сообщение расширяет характеризацию предикатов каждого класса иерархии Гжегорчика, полученную в [1], на подклассы класса всех переборных предикатов.

Приводятся простые примеры переборных (т.е. NP-полных) задач, для которых доказываются полиномиальные верхние и нижние оценки времени их решения недетерминированными машинами Тьюринга (эти задачи невозможно решать и детермированными машинами Тьюринга за полиномиальное число шагов). Приводимые примеры переборных задач имеют следующий вид: разрешимо ли логико-арифметрическое уравнение в натуральных числах, длины которых не превосходят предварительно заданного ограничения с помощью функции, вычисляющей $C'' \cdot |X|^C$ по X.

Полученные результаты представляют интерес при изучении кибернетических аспектов теории алгоритмов и для получения доказательств невозможности эффективизации некоторых переборных алгоритмов.

Обширна литература, посвященная переборным задачам (см., например, обзор [2]). Интересные примеры теоретико-числовых переборных задач приведены в работе [3]. В работе [4] приводятся примеры теоретико-числовых задач, решение которых возможно недетерминированными машинами Тьюринга за полиномиальное число шагов.

Доказательство формулируемой ниже теоремы использует некоторые конструкции, предложенные в [5]. Другие нижние оценки сложности дискретных задач получены автором в [6].

Полученные в этой статье нижние оценки сложности вычисления некоторых задач точны с точностью до возведения в квадрат.

1. Основные понятия и результаты

Будем писать $T(X, Y)$, если «ни в одном разряде двоичной записи чисел X, Y не стоит одновременно единица» (см., например, [5]). Характеризующую предикат T функцию будем обозначать посредством M:

$$\forall XY\,(M(X, Y) = 0 \Leftrightarrow T(X, Y)), \quad \forall XY\,(M(X, Y) \leq 1).$$

Арифметической конъюнкцией будем называть функцию, которая по двоичной записи чисел вычисляет число, в двоичной записи которого стоит единица в том и только в том разряде, в котором в двоичной записи аргументов стоит одновременно 1. Обозначим эту операцию посредством $\wedge$. Легко убедиться, что

$$\forall XY\,(T(X, Y) \Leftrightarrow (X \wedge Y = 0)).$$

Рассматриваются недетерминированные одноленточные машины Тьюринга с потенциально бесконечной только вправо лентой.

Пусть $|X|$ обозначает длину двоичного представления числа X, при этом все нули, предшествующие первой единице, не учитываются.

Теорема. Во-первых, если при некоторых положительных натуральных числах C, C'' предикат Π вычислим недетерминированной машиной Тьюринга с ограничением на число шагов, задаваемое функцией, вычисляющей по X выражение $C'' \cdot \Phi(|X|)^C$, то этот предикат при некотором C'' представим в виде

$$\exists X''\left(\leq 2^{C''\cdot\Phi(|X|)^{2\cdot C}}\right)\exists \overline{Y}\,(\leq X'')\,(P(\overline{Y}, X)+P''(\overline{Y})=0), \qquad (*)$$

где $\Phi(X)\geq X, P$ — полином с целыми коэффициентами, P'' — сумма функций $M, \overline{Y}$ — список переменных, и при устранении ограничения на первый слева квантор получается эквивалентное представление предиката Π.

Во-вторых, каждый предикат, представимый в виде $(*)$, вычислим недетерминированной машиной Тьюринга за число шагов, выражаемое функцией, вычисляющей по X выражение $C'' \cdot \Phi(|X|)^{4\cdot C}$ при некотором C''. □

Заметим, что здесь и далее в формулах, выделенных в строку, будут опускаться кванторы всеобщности по свободным переменным для натуральных чисел. Списки свободых и связанных переменных для натуральнных чисел обозначаются надчеркнутыми буквами.

Список переменных не превосходит некоторого выражения, если каждая переменная такого списка не превосходит этого выражения.

В результате, в частности, получается строгая иерархия предикатов, вычислимых недетерминированными машинами Тьюринга за полиноминальное число шагов (в зависимости от степени полинома). Так как для недетермированных машин Тьюринга имеется строгая иерархия по времени их решения в зависимости от степени полинома, ограничивающего число шагов, то в результате получим строгую иерархию предикатов вида: «логико-арифметическое уравнение имеет решение в натуральных числах, длина которых ограничена полиномом» (в зависимости от степени полинома).

В частности, полином $C \cdot |X|^K$ является нижней оценкой числа шагов, необходимых недетермированной машине Тьюринга для решения вопроса: разрешимо ли логико-арифметическое уравнение в натуральных числах, длины которых не превосходят параметра, ограниченного с помощью функции, вычисляющей выражение $C'' \cdot |X|^{2\cdot K+2}$ по X при некотором C''. Сказанное верно и для детерминированных машин Тьюринга с лентой такого же типа, как и у недетерминированных машин Тьюринга. Строгая классификация элементарных по Кальмару функций с ростом натурального числа K, если в теореме в качестве функции Φ взять функцию, вычисляющую по X выражение $2^{\cdot^{\cdot^{\cdot^{X}}}}$, где цифра 2 повторена K раз. Теорема позволяет получить представление всех преди-

катов класса NP, если в теореме в качестве Φ взять полиномиальные функции.

Наконец, полученные результаты позволяют предложить подход к решению открытой проблемы из [3]: верно ли, что $NP = \mathcal{D}$? В качестве следствия теоремы получаем, что $NP = \mathcal{D}$ тогда и только тогда, когда предикат T принадлежит классу $\mathcal{D}$.

2. Схема доказательства основной теоремы

Докажем вторую часть теоремы. Пусть некоторый предикат представим в виде, указанном в теореме. Недетерминированное вычисление характеристической функции такого предиката можно произвести следующим образом. Недетерминированным способом выписываем числа $\overline{Y}$ не более, чем за $M \cdot C'' \cdot \Phi(|X|)^{4 \cdot C}$ шагов, где M — число аргументов в P. Затем проверяем условие

$$P(\overline{Y}, X) + P''(\overline{Y}) = 0.$$

Для такой проверки достаточно $C'' \cdot (|X| + \sum |\overline{Y}| + |P''| + |P|)^2$ шагов, где $|P|$ и $|P''|$ — длины записей выражений P и P'' соответственно, $\sum |\overline{Y}|$ — сумма длин всех компонент вектора $\overline{Y}$.

Таким образом, недетерминированное вычисление предиката оказалось возможным за $C_1 \cdot \Phi(|X|)^{4 \cdot C}$ шагов при некотором C_1.

Докажем первую часть теоремы. Пусть некоторый предикат вычислим недетерминированной машиной Тьюринга за время, задаваемое функцией, вычисляющей по X выражение $C_1 \cdot \Phi(|X|)^C$.

Получим логико-арифметическое представление предикатов, вычислимых за указанное время недетерминированной машиной Тьюринга.

Пусть внешним алфавитом машины Тьюринга является алфавит $A = \{в_0, \ldots, в_M\}$. Буква $в_0$ считается символом пустой ячейки ленты машины Тьюринга. При достраивании ленты новыми ячейками предполагается, что в них находится буква $в_0$. В самой левой клетке ленты находится знак $*$, который не может быть ни стерт, ни вписан в другую клетку. Машина может находиться в одном из следующих состояний $p_0, \ldots, p_H$; состояние p_1 является начальным состоянием, а состояние p_0 — заключительным. Команда машины Тьюринга имеет один из следующих трех видов

$$p_K в = p_{K''} в'', \quad p_K = \vec{p}_{K''}, \quad p_K = \overleftarrow{p}_{K''},$$

где $в$ и $в''$ — буквы алфавита $A \cup \{*\}$, $K \geq 1$, $K'' \geq 0$, стрелка сверху означает сдвиг головки машины Тьюринга на одну ячейку в направлении, указанном стрелкой. Конфигурацией (или мгновенным описанием) будем называть всякое слово вида $* T \vec{p}_K T''$ и вида $* T \overleftarrow{p}_K T''$, где T и T'' — слова в алфавите A. Эти мгновенные описания соответствуют следующей записи на ленте: $* T T''$. Стрелка над состоянием в конфигурации указывает на букву, обозреваемую головкой машины Тьюринга.

Для дальнейшего удобно команды машины Тьюринга заменить подстановками (правилами нормальных алгорифмов) в алфавите $A \cup \{*\} \cup \cup \{\vec{p}_0, \ldots, \vec{p}_H, \overleftarrow{p}_0, \ldots, \overleftarrow{p}_H\}$. Команда первого вида заменяется двумя подстановками

$$\vec{p}_K\, в \to \vec{p}_{K''}\, в'', \quad в\, \overleftarrow{p}_K \to в''\, \overleftarrow{p}_{K''}.$$

Команда второго вида заменяется на серию подстановок вида

$$\vec{p}_K\, в \to в'\, \vec{p}_{K''}, \quad \overleftarrow{p}_K\, в \to \vec{p}_{K''}\, в$$

для любой буквы $в$ алфавита $A \cup \{*\}$. На аналогичную серию подстановок заменяется команда третьего вида

$$в\, \vec{p}_K \to в\, \overleftarrow{p}_{K''}, \quad в\, \overleftarrow{p}_K \to \overleftarrow{p}_{K''}\, в.$$

Заметим, что эти подстановки не изменяют длину слова, в которое производится подстановка.

Будем считать, что машина Тьюринга начинает работу с конфигурации $* x в_0 \ldots в_0$, где x — число, записанное в двоичной системе счисления с цифрами $в_1$, $в_2$, обозначающими соответственно нуль и единицу.

Пусть машина закончила работу за e шагов. Тогда существуют слова Φ, Γ, K, E и T, длина каждого из которых не превосходит $(|x|+e+2)\cdot(e+1)$ и такие, что

0. E — слово, полученное последовательным приписыванием друг к другу нескольких конфигураций,
1. $K = * \overleftarrow{p}_1 x в_0$,
2. Φ имеет вид $в_0 \ldots в_0$, что эквивалентно $\Phi в_0 = в_0 \Phi$,
3. $K \Phi E$ получается из T в результате последовательности детерминированных замен двубуквенных слов, состоящих из разных, не использованных ранее букв, на левые части подстановок, заменивших, как указано выше, команды машин Тьюринга,
4. $E\Gamma$ получается из T в результате последовательности недетерминированных замен тех же двубуквенных слов, что и в пункте 3 на правые части подстановок (вместо тех же слов, вместо которых были подставлены левые части соответствующих подстановок),
5. в Γ входит $\overleftarrow{p}_0$ или $\vec{p}_0$.

Используемые в пункте 3 детерминированные замены могут быть по способу Ю. В. Матиясевича сведены к заменам однобуквенных слов на однобуквенные, так как замены производятся вместо двубуквенных слов специального вида (или сведены к заменам пункта 4).

Важно, что используемые в пункте 4 замены с помощью введения не используемых ранее букв можно свести к недетерминированным заменам вида

$$X = [Y]^{AB}_{CE||C''E''} \qquad (**)$$

(вместо каждого вхождения слова AB в слово Y подставляется одно из слов CE, $C''E''$). При этом буквы A, B отличны друг от друга. Для

завершения доказательства достаточно арифметизировать все указанные условия, используя полиномы и предикат T.

Кодом слова P в K-буквенном алфавите, на буквы которого будем ссылаться по их номерам в последовательности вхождения их в список всех букв алфавита, будем называть систему из $K+1$-го числа, в которой каждая E-ая компонента при $E \leq K$ имеет двоичную запись, получаемую из P заменой каждого вхождения E-ой буквы на цифру 1, и заменой вхождений остальных букв на цифру 0, и в которой $K+1$-е число в двоичной записи имеет вид $10 \ldots 0$ (число нулей равно длине слова P). Ясно, что слово однозначно восстанавливается по своему коду и что код пустого слова имеет вид $0, \ldots, 0,1$.

Пусть $X_1, \ldots, X_K, X_{K+1}$ – код слова X, а $Y_1, \ldots, Y_K, Y_{K+1}$ – код слова Y. Легко видеть, что два слова совпадают тогда и только тогда, когда их коды покомпонентно совпадают и что кодом слова XY является следующая система чисел

$$X_1 \cdot Y_{K+1} + Y_1, \ldots, X_K \cdot Y_{K+1} + Y_K, X_{K+1} \cdot Y_{K+1}.$$

Понятия, которые необходимы для окончательной арифметизации и которые были предложены и использованы Ю. В. Матиясевичем, например, двуместный предикат «быть тенью», заданный на буквах, и т. п., могут быть легко выражены в предлагаемой кодировке. Заинтересованный читатель легко это может выполнить аналогично тому, как это сделал Ю. В. Матиясевич в своей работе на стр. 81.

Поэтому обратим внимание только на элемент существенной новизны: арифметизацию формулы для недетерминированной замены $(* *)$, в которой X и Y – переменные для слов, остальные буквы – переменные для букв рассматриваемого алфавита.

Арифметизация этой формулы может быть проведена с помощью арифметической конъюнкции, арифметической дизъюнкции с дальнейшим исключением их при использовании следующих соотношений. Для любых X, Y имеет место

$$X + Y = (X \vee Y) + (X \wedge Y).$$

Каковы бы ни были числа X, Y и X'', длина записи которых меньше длины X_{K+1}, выполняется

$$X = (Y \wedge X'') \Leftrightarrow T(X_{K+1} - 1 - Y, X) \wedge T(X_{K+1} - 1 - X'', X) \wedge \\ \wedge T(Y - X, X'' - X).$$

Здесь X_{K+1} должно быть последней компонентой системы чисел, являющейся кодом некоторого слова.

В качестве нижних индексов у X и Y будем использовать иногда буквы вместо номеров букв. Удобно посредством $X(A)$, $X(C)$, $X(C'')$, $Y(A)$, $Y(C)$ и $Y(C'')$ обозначить

$$X_A \wedge 2 \cdot X_B, \quad X_C \wedge 2 \cdot X_E, \quad X_{C''} \wedge 2 \cdot X_{E''},$$
$$Y_A \wedge 2 \cdot Y_B, \quad Y_C \wedge 2 \cdot Y_E, \quad Y_{C''} \wedge 2 \cdot Y_{E''},$$

соответственно.

Формула $(**)$ эквивалентна конъюнкции следующих утверждений.

Во-первых, $X_{K+1} = Y_{K+1}$, т. е. длины слов X и Y совпадают.

Во-вторых, в X нет вхождений слова AB, т. е. $X(A) = 0$.

В-третьих, $Y(A) \wedge (X(C) \vee X(C'')) = Y(A)$, т. е. вхождения AB в Y находятся только на тех местах, на которых в X имеются вхождения CE или $C''E''$.

В-четвертых, для всякого e, не превосходящего K, выполняется

$$Y(A) \vee [Y(A)/2] \vee Y_e = Y(A) \vee [Y(A)/2] \vee X_e,$$

т. е. Y_e совпадает с X_e за исключением мест, на которых имеются вхождения AB в Y. На этом заканчивается изложение схемы доказательства теоремы.

ЛИТЕРАТУРА

[1] *Виноградов А. К., Косовский Н. К.:* Иерархия диофантовых представлений примитивно рекурсивных предикатов. Сб. «Вычислительная техника и вопросы кибернетики, вып. 12, Ленинград, изд. ЛГУ. с. 99–107, 1975.»

[2] *Garey M. R., Johnson D. S.:* Computers and Intractability. A Guide to the Theory of NP-Completeness, San Francisco, W. H. Freeman and Company, 1979.

[3] *Manders K. L., Adleman L.:* NP-Complete Decision Problems for Binary Quadratics. Jour. Comp. Syst. Sci. v. 16, pp. 168–184, 1978.

[4] *Тарасов С. П., Хачиян Л. Г.:* Границы решений и алгоритмической сложности систем выпуклых диофантовых неравенств. «Доклады АН СССР, т. 255, № 2, с. 296–300, 1980.»

[5] *Матиясевич Ю. В.:* Новое доказательство теоремы об экспоненциально диофантовом представлении перечислимых предикатов. Записки научн. семинаров Ленинград. отд. Математического инст. АН СССР, т. 60, с. 75–92, 1976.

[6] *Kossovsky N. K.:* Some simple examples of problems with provable small lower bounds, VI International Congress of Logic, Methodology and Philosophy of Science, abstracts, section 1–4, Hannover, pp. 185–189, 1979.