УДК 004.93.51

Т.М. Косовская

НЕКОТОРЫЕ ЗАДАЧИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА, ДОПУСКАЮЩИЕ ФОРМАЛИЗАЦИЮ НА ЯЗЫКЕ ИСЧИСЛЕНИЯ ПРЕДИКАТОВ, И ОЦЕНКИ ЧИСЛА ШАГОВ ИХ РЕШЕНИЯ

Косовская Т.М. **Некоторые задачи искусственного интеллекта, допускающие формализацию на языке исчисления предикатов, и оценки числа шагов их решения.**

Аннотация. Ряд задач искусственного интеллекта, включающих в себя такие задачи, как распознавание образов, медицинская диагностика, анализ рынка, сведены к доказательству выполнимости формул исчисления предикатов, имеющих простую структуру. Рассмотрены некоторые алгоритмы решения этих задач и доказаны верхние оценки числа шагов этих алгоритмов.

Ключевые слова: искусственный интеллект, исчисление предикатов, сложность алгоритмов, NP-полнота.

Kosovskaya T.M. **Some artificial intelligence problems permitting formalization by means of predicate calculus language and upper bounds of their solution steps.**

Abstract. Some artificial intelligence problems including such ones as pattern recognition, medical diagnostics, market analysis is reduced to the proof of satisfiability of predicate calculus formulas with a symple structure. Some algorithms solving such problems are regarded and the upper bounds of their steps are proved.

Keywords: artificial intelligence, predicate calculus formulas, complexity of algorithm, NP-compleetness.

1. Введение. При решении задач искусственного интеллекта, включающих в себя такие задачи, как распознавание образов, медицинская диагностика и выбор лечения, анализ рынка и выбор действий в рыночных условиях, всё чаще используются язык и методы доказательства математической логики [10]. Достаточно хорошо разработаны методы формализации таких задач при использовании языка исчисления высказываний (булевы функции) [2, 3].

Никакие потенциально бесконечные классы объектов, описание которых существенным образом задается формулами исчисления предикатов, с помощью булевых уравнений описать нельзя. В [10] на стр. 550 приведена экспоненциальная оценка числа пропозициональных переменных в пропозициональных формулах, моделирующих формулу исчисления предикатов в конечной области для задач планирования $T \cdot |Act| \cdot |O|^P$, где T — число временных этапов, $|Act|$ — число схем действий, $|O|$ — число объектов в проблемной области, P — максимальная арность (число параметров) любой схемы действий. В статье для одного из предложенных алгоритмов оценка числа шагов решения сформулированных задач, описанных формулами исчисления предикатов, имеет в точности такой же вид. Однако используемая память существенно меньше.

Решение рассматриваемых в работе задач сведено к доказательству теорем простого подмножества формул исчисления предикатов [4]. Доказательство таких теорем может быть проведено, например, посредством применения метода резолюций [9, 10, 12] или в секвенциальных исчислениях высказываний и предикатов [4, 7]. Поскольку метод резолюций и секвенциальные исчисления хорошо приспособлены для их компьютерной реализации, то оценки числа шагов алгоритмов построения вывода в настоящей работе приводятся именно для этих методов доказательства.

2. Постановка задачи. Пусть имеется множество Ω конечных множеств $\omega = \{\omega_1, \dots, \omega_t\}$, которые в дальнейшем будут называться объектами. Частью τ объекта ω называется любое его подмножество (не обязательно собственное). Пусть также на частях τ задан набор предикатов $p_1, \dots, p_n$, характеризующих свойства и отношения между элементами объекта ω.

Логическим описанием $S(\omega)$ объекта ω называется набор всех истинных постоянных формул вида $p_i(\overline{\tau})$ или $\neg p_i(\overline{\tau})$, выписанных для всех возможных частей τ объекта ω. [1]

Целевыми формулами называются такие формулы $A_k(\overline{x})$ со свободными переменными $\overline{x}$, что

1. $A_k(\overline{x})$ содержит в качестве атомарных только формулы вида $p_i(\overline{y})$, где $y \subseteq x$;
2. $A_k(\overline{x})$ не содержит кванторов;
3. $A_k(\overline{x})$ имеет вид дизъюнкции элементарных конъюнкций.

С помощью построенных описаний объектов и целевых формул предлагается решать следующие задачи.

[1]Здесь и далее посредством $\overline{x}$ обозначается список элементов конечного множества x, соответствующий некоторой перестановке номеров его элементов. Тот факт, что элементами списка $\overline{x}$ являются элементы множества y, будем записывать в виде $x \subseteq y$.

Задача идентификации. *Проверить, удовлетворяет ли объект ω или его часть целевой формуле $A_k(\overline{x})$ и предъявить эту часть объекта.*

Задача классификации. *Найти все такие номера k, что верна формула $A_k(\overline{\omega})$.*

Задача анализа. *Найти и классифицировать все части τ объекта ω, для которых $A_k(\overline{\tau})$.*

Решение задач идентификации, классификации и анализа для распознавания сложного объекта сведено в [5] к доказательству соответственно секвенций:[2]

$$S(\omega) \vdash \exists \overline{x}_{\neq}\ A_k(\overline{x}), \tag{1}$$

$$S(\omega) \vdash \bigvee_{k=1}^{M} A_k(\overline{\omega}), \tag{2}$$

$$S(\omega) \vdash \bigvee_{k=1}^{M} \exists \overline{x}_{\neq}\ A_k(\overline{x}), \tag{3}$$

где целевые формулы $A_k(\overline{x})$ являются описаниями классов Ω_k ($\Omega = \bigcup_{k=1}^{K} \Omega_k$) распознаваемых объектов и удовлетворяют условию: если для некоторого списка (упорядоченного множества) $\overline{\omega}$ всех элементов множества ω истинна формула $A_k(\overline{\omega})$, то $\omega \in \Omega_k$.

При решении задач медицинской диагностики в качестве множеств ω выступают пациенты, рассматриваемые как множество своих частей (органов), а исходные предикаты p_i — свойства этих частей и отношения между ними, обычно называемые симптомами. Целевые формулы в этом случае могут задавать классы заболеваний или синдромы. Решение задач идентификации, классификации и анализа в этом случае сводится к доказательству тех же самых формул и соответствуют таким задачам, как «обладает ли

---

[2] Для того, чтобы записать, что значения для переменных списка $\overline{x}$, удовлетворяющие формуле $A(\overline{x})$, различны, вместо формулы

$$\exists x_1 ... \exists x_m (\&_{i=1}^{m} \&_{j=i+1}^{m} (x_i \neq x_j)\ \&\ A(x_1, ..., x_m))$$

будет использоваться обозначение

$$\exists \overline{x}_{\neq} A(\overline{x}).$$

пациент заданным заболеванием (синдромом) и какие именно органы влекут это заболевание?», «какими заболеваниями (синдромами) страдает заданный орган?», «какие области пациента какими заболеваниями (синдромами) обладают?».

При решении задач анализа рынка в качестве множеств ω выступают множества субъектов и объектов рынка, исходные предикаты p_i — свойства этих объектов и субъектов, а также отношения между ними. Целевые формулы могут задавать желаемые (или негативные) ситуации на рынке. Решение задач идентификации, классификации и анализа в этом случае сводится к доказательству тех же самых формул и соответствуют таким задачам как «какие объекты рынка влияют на то, что при имеющемся состоянии рынка будет получена заданная ситуация?», «какие ситуации следуют из заданных свойств объектов рынка?“, „какие подмножества объектов рынка к каким ситуациям приводят?».

Можно привести ещё множество задач искусственного интеллекта, решение которых является решением сформулированных задач и сводится к доказательству секвенций (1–3).

3. Методы решения. Прежде всего заметим, что для того, чтобы уметь доказывать секвенции (1–3), достаточно уметь доказывать секвенцию вида

$$S(\omega) \vdash \exists \overline{x}_{\neq}\ A(\overline{x}), \tag{4}$$

где $A(\overline{x})$ — элементарная конъюнкция атомарных формул и их отрицаний.

Это вытекает из следующего: так как целевые формулы не содержат кванторов, то каждую из них можно привести в дизъюнктивную нормальную форму (ДНФ). Пусть $A_k(\overline{x}_k) \Leftrightarrow \vee_j A_k^j(\overline{x}_k^j)$, где $A_k^j(\overline{x}_k^j)$ — элементарные конъюнкции, содержащие в качестве своих переменных переменные списка $\overline{x}_k^j$, являющегося подсписком списка переменных $\overline{x}_k$. Кроме того, $\exists x(A \vee B) \Leftrightarrow (\exists x A \vee \exists x B)$.

Таким образом, для доказательства секвенции (1) достаточно доказать хоть при одном j секвенцию $S(\omega) \vdash \exists \overline{x}_{k\neq}^j\ A_k^j(\overline{x}_k^j)$.

Для доказательства секвенции (2) достаточно доказать хоть при одном j и хоть при одном k секвенцию $S(\omega) \vdash A_k^j(\overline{\omega})$.

Для доказательства секвенции (3) достаточно доказать хоть при одном j и хоть при одном k секвенцию $S(\omega) \vdash \exists \overline{x}_{k\neq}^j\ A_k^j(\overline{x}_k^j)$.

Следует отметить, что, вообще говоря, доказательство следования формулы $\exists x A(x)$ из списка других формул не обеспечивает нахождения тех значений x, для которых $A(x)$ верна при сформулированных условиях. Для нахождения таких значений требуются конструктивные методы доказательства.[3]

Широко распространён переборный метод доказательства секвенции вида (4). Он заключается в переборе всех возможных наборов значений из множества констант ω и проверке выполнимости формулы $A(\overline{x})$ на каждом таком наборе.

Другими методами конструктивного доказательства секвенции вида (4) являются, например, секвенциальные исчисления предикатов [4, 7], метод резолюций для исчисления предикатов [10, 12], обратный метод Маслова для исчисления предикатов [8, 11]. Эти методы позволяют предъявить набор значений переменных, выполняющий заданную формулу.

4. NP-трудность рассматриваемых задач. Задачи вида (1–3) в теории сложности алгоритмов принято называть задачами распознавания, так как в качестве ответа они имеют два значения «Да» и «Нет». Если задача распознавания имеет вид $\exists Y\ P(X,Y)$, то задачу вида $(?Y)\ P(X,Y)$ (при каких Y верна формула $P(X,Y)$) принято называть задачей поиска. Если задача распознавания $\exists Y\ P(X,Y)$ NP-полна, то соответствующая ей задача поиска $(?Y)\ P(X,Y)$ NP-трудна [1].

Для доказательства NP-трудности рассматриваемых в статье задач в [6] сформулированы две задачи и доказана их NP-полнота.

Выполнимость в конечной интерпретации.

Дано: *Пусть заданы конечное множество объектов*

$$\omega = \{\omega_1, \ldots, \omega_t\}$$

и набор истинных постоянных атомарных формул вида $p_i^{\alpha_{i,\overline{\tau}}}(\overline{\tau})$, *где* $i = 1, \ldots, n$, $\tau \subseteq \omega$, $\alpha_{i,\overline{\tau}}$ *— логические константы. Пусть также дана бескванторная формула* $A(\overline{y})$, *представленная в виде дизъюнкции элементарных конъюнкций атомарных формул* ($\overline{y} = (y_1, \ldots, y_a)$ *– список предметных переменных, входящих в формулу).*

Вопрос: *Существует ли набор значений для* $\overline{y}$ *из* ω^a, *для которого формула* $A(\overline{y})$ *истинна?*

То есть верно ли, что

$$\exists \overline{x} A(\overline{x}).$$

[3]Здесь под конструктивностью понимается предъявление тех значений переменных, которые обеспечивают справедливость формулы вида $\exists x P(x)$.

Эта задача NP-полна, так как она принадлежит классу NP и задача КЛИКА [1] полиномиально сводится к ней.

Выполнимость на заданном множестве.

Дано: *Пусть заданы конечное множество объектов* $\omega = \{\omega_1, \ldots, \omega_t\}$ *и набор истинных постоянных атомарных формул вида* $p_i^{\alpha_{i,\overline{\tau}}}(\overline{\tau})$, *где* $i = 1, \ldots, n$, $\tau \subseteq \omega$, $\alpha_{i,\overline{\tau}}$ *— логические константы. Пусть также дана бескванторная формула* $A(\overline{y})$, *представленная в виде дизъюнкции элементарных конъюнкций атомарных формул* ($\overline{y} = (y_1, \ldots, y_t)$ *— список предметных переменных, входящих в формулу).*

Вопрос: *Существует ли набор различных значений для* $\overline{y}$ *из* ω^t, *для которого формула* $A(\overline{y})$ *истинна?*

То есть верно ли, что

$$\exists \overline{y}_{\neq} A(\overline{y}).$$

Эта задача NP-полна, так как она принадлежит классу NP и задача «Гамильтонов цикл» [1] полиномиально сводится к ней.

Следствиями NP-полноты этих задач будут следующие три утверждения.

Утверждение 1. *Если целевая формула* $A_k(\overline{x})$ *является исходным данным для решающего алгоритма, то задача идентификации NP-трудна.*

Доказательство. Это следует из того, что задача идентификации является задачей поиска, соответствующей задаче «Выполнимость в конечной интерпретации».

Утверждение 2. *Если целевая формула* $A_k(\overline{x})$ *является исходным данным для решающего алгоритма, то задача классификации NP-трудна.*

Доказательство. Это следует из того, что задача классификации является задачей поиска, соответствующей задаче «Выполнимость на заданном множестве».

Утверждение 3. *Если целевая формула* $A_k(\overline{x})$ *является исходным данным для решающего алгоритма, то задача анализа NP-трудна.*

Доказательство. Это следует из того, что задача анализа является многократным решением задачи идентификации.

Тем самым доказано, что в настоящее время нельзя гарантировать наличие полиномиального алгоритма заданной степени, решающего поставленные задачи в случае, если целевые формулы являются исходными данными для этого алгоритма. Однако стоит проанализировать, какие же параметры исходных данных находятся в показателе степени экспоненты при использовании различных алгоритмов.

5. Обозначения. Для получения оценок числа шагов работы алгоритмов, решающих задачу (4), введём следующие обозначения:

$A(\overline{x})$ — элементарная конъюнкция с предикатами $p_1, ..., p_n$ и предметными переменными $\overline{x}$;

$|A|$ — число различных вхождений предметных переменных в формулу $A(\overline{x})$;

a_{2i-1} — число вхождений предиката p_i в формулу $A(\overline{x})$ без отрицания;

a_{2i} — число вхождений предиката p_i в формулу $A(\overline{x})$ с отрицанием;

$a = \sum_{i=1}^{2n} a_i$ — общее число атомарных формул в $A(\overline{x})$;

m — число аргументов в $A(\overline{x})$;

η_i — число аргументов предиката p_i $(1 \leq i \leq n)$;

$\eta = \max_i(\eta_i)$;

$|S|$ — число различных вхождений предметных констант в $S(\omega)$;

s_{2i-1} — число вхождений предиката p_i в $S(\omega)$ без отрицания;

s_{2i} — число вхождений предиката p_i в $S(\omega)$ с отрицанием;

$s = \underbrace{\max}_{1 \leq i \leq 2n} (s_i)$.

6. Алгоритм полного перебора. Задача (4) равносильна тому, что формула $A(\overline{x})$ истинна при подстановке некоторых различных элементов множества ω в качестве соответствующих значений входящих в нее предметных переменных. Она может быть решена полным перебором с помощью последовательного выделения всех частей $\tau \subseteq \omega$ (подмножеств ω, число элементов в которых совпадает с числом предметных переменных формулы $A_k(\overline{x})$) и различного упорядочивания их элементов для присвоения значений этих элементов переменным, входящим в $A_k(\overline{x})$. Такой алгоритм будем называть алгоритмом полного перебора. [4]

Теорема 1. *При использовании алгоритма полного перебора задача проверки того, что замыкание элементарной конъюнкции квантором существования $\exists \overline{x}_{\neq} A(\overline{x})$ следует из совокупности*

---

[4] В следующей теореме $\mathtt{A}_t^m$ — число размещений из t по m.

постоянных атомарных формул или их отрицаний $S(\omega)$, а также нахождение тех наборов значений переменных, существование которых утверждается в формуле, может быть решена не более чем за

$$\mathtt{A}_t^m \cdot (|A| + \sum_{i=1}^{2n} a_i \cdot s_i)$$

шагов, где под шагом понимается подстановка значения переменной в формулу или сравнение конъюнкта формулы с формулой совокупности $S(\omega)$ на их графическое совпадение.

Доказательство. Число наборов значений $\overline{\tau}$ предметных переменных $\overline{x}$, для которых требуется проверить истинность формулы $A(\overline{x})$ — это число размещений из t по m $\mathtt{A}_t^m$.

Для каждого из $\mathtt{A}_t^m$ наборов $\overline{\tau}$ требуется произвести не более, чем $\sum_{i=1}^{n}(a_{2i-1}+a_{2i})\eta_i = |A|$ подстановок констант в формулу $A(\overline{x})$ вместо соответствующих переменных списка $\overline{x}$. Для вычисления значения формулы с подстановкой $A_k(\overline{\tau})$ необходимо произвести не более чем $\sum_{i=1}^{2n} a_i \cdot s_i$ сравнений конъюнктов с формулами из $S(\omega)$. Всего $\mathtt{A}_t^m \cdot (|A| + \sum_{i=1}^{2n} a_i \cdot s_i)$ шагов.

Следствие 1 теоремы 1. *Если максимальное число вхождений предметных переменных в элементарные конъюнкции k-ой целевой формулы равно $|A_k|$, число дизъюнктивных членов k-ой целевой формулы равно, J_k то число шагов решения задачи идентификации при использовании алгоритма полного перебора составляет*

$$O(J_k \cdot t^{m_k} \cdot |A_k| \cdot |S|).$$

Следствие 2 теоремы 1. *Число шагов решения задачи классификации при использовании алгоритма полного перебора составляет*

$$O(t! \cdot |A| \cdot |S|).$$

Следствие 3 теоремы 1. *Если m — максимальное число переменных в элементарных конъюнкциях, входящих в целевые формулы, то при использовании алгоритма полного перебора число шагов решения любой из сформулированных выше задач составляет*

$$O(t^m \cdot |A| \cdot |S|).$$

7. Алгоритм проверки выводимости секвенции в секвенциальном исчислении предикатов. При построении вывода в секвенциальном исчислении предикатов секвенции

$$S(\omega) \vdash \exists \overline{x}_{\neq}\ A(\overline{x})$$

в первую очередь производим m применений правила введения квантора существования в сукцедент, заменяя при этом каждое вхождение предметной переменной x_j из списка $\overline{x}$ на метапеременную для терма T_j. Затем производим $a-1$ применений правила введения конъюнкции в сукцедент, получая a секвенций вида $S(\omega) \vdash {p_i}^{\alpha}(T_{j_1}, ..., T_{j_{\eta_i}})$, где α — это константа *true* или *false*. То есть антецеденты всех этих секвенций совпадают, а в сукцедентах стоят атомарные формулы или их отрицания, полученные из конъюнктивных членов формулы $A(\overline{x})$ заменой предметных переменных на соответствующие метапеременные для термов. Получение сукцедентов этих секвенций даже на машине Тьюринга можно осуществить за линейное от длины записи формулы $A(\overline{x})$ число шагов $|A|$.

Для того, чтобы секвенция $S(\omega) \vdash \exists \overline{x}_{\neq} A(\overline{x})$ была выводима необходимо и достаточно, чтобы все полученные секвенции являлись аксиомами, т.е. нашлись такие различные значения из ω для $T_1, ..., T_m$, что при подстановке этих значений в формулы из сукцедента каждая из них графически совпала бы с одной из формул списка $S(\omega)$.

Для получения оценки числа шагов описанного алгоритма докажем следующую лемму.

Лемма 1. *Число шагов проверки того, что каждая секвенция совокупности из a секвенций вида $S(\omega) \vdash {p_i}^{\alpha}(T_{j_1}, ..., T_{j_{\eta_i}})$ является аксиомой, а также нахождения значений метапеременных для термов, обеспечивающих свойство секвенций быть аксиомами, не превосходит*

$$s_1^{a_1} \cdot \ldots \cdot s_{2n}^{a_{2n}} \cdot (\sum_{i=1}^{n}(a_{2i-1} + a_{2i})\eta_i +$$

$$+ a^2 \cdot \eta^2) + \sum_{i=1}^{n}(s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i,$$

где под шагом понимается операция сравнения двух постоянных атомарных формул с одним и тем же предикатом на их графическое совпадение или выписывание уравнения вида $T = \omega_j$.

Доказательство. Пусть секвенции и множество$S(\omega)$ упорядочены по номеру предиката, входящего в сукцедент секвенции, причем для каждого i сначала идут атомарные формулы с предикатом p_i, а затем формулы с отрицанием предиката p_i.

Для каждого предиката p_i (входящего как без отрицания, так и с отрицанием) за один просмотр части длиной s_{2i-1} (или s_{2i}) множества $S(\omega)$ и a_{2i-1} (a_{2i} соответственно) просмотров части сукцедентов секвенций находим в $S(\omega)$ все постоянные формулы вида ${p_i}^{\alpha}(\omega_{l_1}, ..., \omega_{l_{\eta_i}})$ и для каждого вхождения предиката p_i выписываем $s_{2i-1}+s_{2i}$ систем из η_i уравнений вида $T_{j_k} = \omega_{l_k}$ при $k = 1, ..., \eta_i$, причем для атомарных формул, входящих в сукцедент секвенции без отрицания (так же, как и для атомарных формул, входящих в сукцедент секвенции с отрицанием), правые части этих систем совпадают.

То есть для каждого предиката p_i за $(s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i$ шагов (выписываний уравнений) получим формулу

$$\&_1^{a_i} \ \vee_1^{s_i} \ \&_{k=1}^{\eta_i} \ (T_{j_k} = \omega_{l_k}).$$

Повторяя эту процедуру с каждым предикатом при $i = 1, ..., n$, за $\sum_{i=1}^{n}(s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i$ шагов (выписываний уравнений) получим формулу для нахождения значений метапеременных для термов

$$\&_{i=1}^{n} \ \&_1^{a_i} \ \vee_1^{s_i} \ \&_{k=1}^{\eta_i} \ (T_{j_k} = \omega_{l_k}).$$

Приведем полученную формулу в ДНФ

$$\vee_1^{s_1^{a_1} \cdot \ldots \cdot s_{2n}^{a_{2n}}} \ \&_1^{a} \ \&_{k=1}^{\eta_i} \ (T_{j_k} = \omega_{l_k}).$$

Число шагов (выписываний соответствующих уравнений) приведения этой формулы в ДНФ не превосходит следующей функции от длины ее записи:

$$\sum_{1}^{s_1^{a_1} \cdot \ldots \cdot s_{2n}^{a_{2n}}} \sum_{i=1}^{2n} \sum_{j=1}^{a_i} \eta_i = \sum_{1}^{s_1^{a_1} \cdot \ldots \cdot s_{2n}^{a_{2n}}} \sum_{i=1}^{n} (a_{2i-1} + a_{2i}) \eta_i =$$

$$= s_1^{a_1} \cdot ... \cdot s_{2n}^{a_{2n}} \cdot \sum_{i=1}^{n} (a_{2i-1} + a_{2i})\eta_i.$$

После приведения этой формулы в ДНФ получаем $s_1^{a_1} \cdot ... \cdot s_{2n}^{a_{2n}}$ систем с m переменными, каждая из которых имеет не более $a \cdot \eta$ уравнений. Каждая такая система уравнений может быть решена за квадратичное от числа уравнений число шагов. Следовательно, проверка совместности каждой из этих систем, а также нахождение решения каждой совместной системы могут быть осуществлены не более чем за $(s_1^{a_1} \cdot ... \cdot s_{2n}^{a_{2n}}) \cdot a^2 \cdot \eta^2$ шагов.

Общее число шагов (сравнений двух выражений и выписываний уравнений вида $T_{j_k} = \omega_{l_k}$) не превосходит $s_1^{a_1} \cdot ... \cdot s_{2n}^{a_{2n}} \cdot (\sum_{i=1}^{n} (a_{2i-1} + a_{2i})\eta_i + a^2 \cdot \eta^2) + \sum_{i=1}^{n} (s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i$.

Следствие леммы 1. *число шагов проверки того, что каждая секвенция совокупности из a секвенций вида* $S(\omega) \vdash {p_i}^{\alpha}(T_{j_1}, ..., T_{j_{\eta_i}})$ *является аксиомой, а также нахождение значений метапеременных для термов, обеспечивающих свойство секвенций быть аксиомами, составляет*

$$O(t^{\sum_{i=1}^{n} \eta_i \cdot (a_{2i-1} + a_{2i})}),$$

где t *— число констант в* ω, η_i *— число аргументов у предиката* p_i, $(a_{2i-1} + a_{2i})$ *— число вхождений предиката* p_i *в совокупность секвенций.*

Теорема 2. *число шагов проверки того, что секвенция* $S(\omega) \vdash \exists \overline{x}_{\neq} A(\overline{x})$ *выводима в секвенциальном исчислении предикатов, а также нахождения значений для переменных* $\overline{x}$, *существование которых утверждается в правой части секвенции, не превосходит*

$$|A| + s_1^{a_1} \cdot ... \cdot s_{2n}^{a_{2n}} \cdot (\sum_{i=1}^{n} (a_{2i-1} + a_{2i})\eta_i + a^2 \cdot \eta^2) +$$

$$\sum_{i=1}^{n} (s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i,$$

где $|A|$ *— длина записи формулы* $A(\overline{x})$.

Доказательство. Как было показано выше, за m применений правила введения квантора существования в сукцедент и a

применений правила введения конъюнкции в сукцедент получаем, что выводимость указанной секвенции равносильна тому, что каждая из полученных в результате применений этих правил a секвенций является аксиомой. При этом получение этих секвенций возможно за один просмотр формулы $A(\overline{x})$. Используя лемму 1, получаем, что общее число шагов проверки выводимости секвенции $S(\omega) \vdash \exists \overline{x}_{\neq}\ A(\overline{x})$ в секвенциальном исчислении предикатов, а также нахождения значений для переменных $\overline{x}$, существование которых утверждается в правой части секвенции, не превосходит указанного выражения.

8. Алгоритм проверки неудовлетворимости множества предложений методом резолюций для исчисления предикатов. Для доказательства методом резолюций того, что

$$S(\omega) \vdash \exists \overline{x}_{\neq} A(\overline{x}),$$

необходимо построить множество предложений (элементарных дизъюнкций), соответствующих формуле $\neg\exists \overline{x}_{\neq} A(\overline{x})$. Так как $A(\overline{x})$ является элементарной конъюнкцией, то это множество предложений состоит из одной элементарной дизъюнкции (которую будем обозначать $\overline{A}(\overline{x})$) отрицаний конъюнктивных членов формулы $A(\overline{x})$. То есть приведение формулы $A(\overline{x})$ к множеству предложений можно осуществить за один просмотр этой формулы.

Для того, чтобы из множества предложений $\{S(\omega), \overline{A}(\overline{x})\}$ было выводимо пустое предложение $nill$ необходимо и достаточно, чтобы существовал такой общий унификатор (подстановка) $|^{x_1\ ,\ldots,x_m}_{\omega_{l_1},\ldots,\omega_{l_m}}$, что каждый дизъюнктивный член предложения $\overline{A}(\overline{x})$ при такой подстановке (т.е. $\overline{A}(\omega_{l_1}, \ldots, \omega_{l_m})$) совпал с одной из постоянных формул из $S(\omega)$.

Лемма 2. *число шагов поиска общего унификатора для множества $S(\omega)$ и предложения $\overline{A}(\overline{x})$, состоящего из a дизъюнктивных членов, не превосходит*

$$s_1^{a_1} \cdot \ldots \cdot s_{2n}^{a_{2n}} \cdot (\sum_{i=1}^{n} (a_{2i-1} + a_{2i})\eta_i + a^2 \cdot \eta^2) + \sum_{i=1}^{n} (s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i,$$

где под шагом понимается операция сравнения или выписывание уравнения вида $T = \omega_j$.

Доказательство. Для каждого предиката p_i за один просмотр части длиной s_{2i-1} (или s_{2i}) множества $S(\omega)$ и a_{2i-1} (соответственно a_{2i}) просмотров предложения $\overline{A}(\overline{x})$ находим в $S(\omega)$ все постоянные формулы вида $p_i{}^{\alpha}(\omega_{l_1},...,\omega_{l_{\eta_i}})$ ($\alpha = true$ или $\alpha = false$) и для каждого вхождения предиката p_i выписываем s_i систем из η_i уравнений вида $x_{j_k} = \omega_{l_k}$ при $k = 1,...,\eta_i$, причем для атомарных формул, входящих в предложение без отрицания (так же, как и для атомарных формул, входящих в сукцедент секвенции с отрицанием), правые части этих систем совпадают.

То есть для каждого предиката p_i за $(s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i$ шагов (выписываний уравнений) получим формулу

$$\&_1^{a_i} \ \vee_1^{s_i} \ \&_{k=1}^{\eta_i} \ (x_{j_k} = \omega_{l_k}).$$

Повторяя доказательство леммы 1 с заменой слов «метапеременные для термов T_j» на слова «предметные переменные x_j», получаем требуемую оценку числа шагов.

Следствие леммы 2. *число шагов нахождения общего унификатора для множества $S(\omega)$ и предложения $\overline{A}(\overline{x})$, состоящего из a дизъюнктивных членов, составляет*

$$O(t^{\sum_{i=1}^{n} \eta_i \cdot (a_{2i-1} + a_{2i})}),$$

где t — число констант в ω, η_i — число аргументов у предиката p_i, $(a_{2i-1} + a_{2i})$ — число вхождений предиката p_i в совокупность секвенций.

Теорема 3. *число шагов проверки методом резолюций для исчисления предикатов того, что из $S(\omega)$ следует $\exists\overline{x}_{\neq}\ A(\overline{x})$, а также нахождения значений для переменных $\overline{x}$, существование которых утверждается в формуле, не превосходит*

$$|A| + s_1^{a_1} \cdot ... \cdot s_{2n}^{a_{2n}} \cdot (\sum_{i=1}^{n} (a_{2i-1} + a_{2i})\eta_i + a^2 \cdot \eta^2) +$$

$$\sum_{i=1}^{n} (s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i,$$

где$|A|$ — длина записи формулы $A(\overline{x})$.

Доказательство. За один просмотр формулы $\exists\overline{x}_{\neq}\ A(\overline{x})$ убирая кванторный префикс, заменяя знак & на $\vee$, убирая знак $\neg$ и записывая его перед атомарными формулами, входящими без отрицания, можно получить предложение $\overline{A}(\overline{x})$.

Используя лемму 2, получаем, что общее число шагов проверки методом резолюций для исчисления предикатов того, что из $S(\omega)$ следует $\exists\overline{x}_{\neq}\ A(\overline{x})$, а также нахождения значений для переменных $\overline{x}$, существование которых утверждается в формуле, не превосходит $|A| + s_1^{a_1} \cdot \ldots \cdot s_{2n}^{a_{2n}} \cdot (\sum_{i=1}^{n}(a_{2i-1} + a_{2i})\eta_i + a^2 \cdot \eta^2) + \sum_{i=1}^{n}(s_{2i-1} \cdot a_{2i-1} + s_{2i} \cdot a_{2i}) \cdot \eta_i$, где $|A|$ — длина записи формулы $A(\overline{x})$.

Следствие теорем 2 и 3. *Если $\widetilde{a}$ — максимальное число вхождений атомарных формул в элементарные конъюнкции, составляющие целевые формулы, s — максимальное число вхождений одного и того же предиката (только без отрицаний или только с отрицаниями) в описание объекта, D — число дизъюнктов в целевых формулах, используемых при решении задачи, то при использовании секвенциального исчисления предикатов или метода резолюций для исчисления предикатов число шагов решения любой из сформулированных выше задач составляет*

$$O(D \cdot s^{\widetilde{a}}).$$

9. Обсуждение полученных результатов. Как видно из доказанных теорем и следствий из них, алгоритм полного перебора и рассмотренные алгоритмы доказательства выводимости в исчислении предикатов дают экспоненциальные (от длины записи целевых) формул оценки числа шагов работы алгоритмов. Это вполне согласуется с NP-трудностью рассмотренных задач. Однако эти оценки имеют в показателе степени экспоненты существенно различные характеристики длины записи целевых формул. Так оценка числа шагов работы алгоритма полного перебора имеет вид $O(t^m \cdot |A| \cdot |S|)$. В показателе степени находится максимальное число аргументов в целевых формулах.

Для алгоритмов, основанных на поиске вывода в рассмотренных исчислениях предикатов оценка числа шагов имеет вид

$$O(D \cdot s^{\widetilde{a}})$$

. В показателе степени находится максимальное число вхождений атомарных формул в элементарные конъюнкции, составляющие целевые формулы.

Если целевые формулы исключить из списка исходных данных алгоритмов (т.е. для каждого фиксированного набора целевых формул строится свой алгоритм), то для каждого такого алгоритма число шагов его работы будет полиномиальным от длины записи описания объекта. При этом в зависимости от используемого алгоритма и структуры целевой формулы показатель степени полинома, ограничивающего число шагов работы алгоритма, будет различным.

Литература

1. *Гэри М., Джонсон Д.* Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982. 416 с.

2. *Журавлёв Ю.И.* Об алгебраических методах в задачах распознавания и классификации // Распознавание, классификация, прогноз. Математические методы и их применение. 1989. Вып. 1. С. 9–16.

3. *Закревский А.Д.* Выявление импликативных закономерностей в булевом пространстве признаков и распознавание образов // Кибернетика. 1982. № 1. С. 1–6.

4. *Клини С.* Математическая логика. М.: Мир, 1973. 480 с.

5. *Косовская Т.М., Тимофеев А.В.* Об одном новом подходе к формированию логических решающих правил // Вестник ЛГУ. 1985. № 8. С. 22—27.

6. *Косовская Т.М.* Доказательства оценок числа шагов решения некоторых задач распознавания образов, имеющих логические описания // Вестник Санкт-Петербургского университета. Сер. 1. 2007. Вып. 4. С. 82–90.

7. *Косовский Н.К.* Элементы математической логики и её приложения к теории субрекурсивных алгоритмов. Л.: Изд-во ЛГУ, 1981. 192 с.

8. *Маслов С.Ю.* Связь между тактиками обратного метода и метода резолюций // Записки научных семинаров ЛОМИ АН СССР. 1969. Т. 16. С. 137–146.

9. *Нильсон Н.* Искусственный интеллект. М.: Мир, 1973. 270 с.

10. *Рассел С., Норвиг П.* Искусственный интеллект: современный подход, 2-е изд.: Пер. с англ. М.: Издательский дом «Вильямс», 2006. 1408 с.

11. *Оревков В.П.* Обратный метод поиска вывода [Приложение 4] // *Адаменко А.Н., Кучуков А.М.* Логическое программирование и Visual Prolog. СПб: БХВ-Петербург, 2003. 992 с.

12. *Робинсон Дж.* Машинно ориентированная логика, основанная на методе резолюций // Кибернетический сборник, новая серия. 1970. Вып. 7. С. 194–218.

Косовская Татьяна Матвеевна — д-р. физ.-мат. наук, доцент; ст. н. с. лаборатории информационных технологий в управлении и робототехнике СПИИРАН, доцент кафедры математики С.-Петербургского государственного морского технического университета (СПбГМТУ), доцент кафедры теоретической кибернетики математико-механического факультета С.-Петербургского государственного университета (СПбГУ). Область научных интересов: логический подход к решению задач искусственного интеллекта, теория сложности алгоритмов. Число научных публикаций - 61. kosovtm@gmail.com; СПИИРАН, 14-я линия В.О., д. 39, г. Санкт-Петербург, 199178, РФ; р.т. +7(812)328-0421.

Tatiana M. Kosovskaya — Doctor of Computer Science, Associate Professor; senior researcher of Laboratory of Information Technologies in Control and Robototechniques, SPIIRAS, Associate Professor of Theoretical Cybernetics Chair, SPbSU, Associate Professor of Mathematics Chair, SPbSMTU. Research area: logical approach to the solving of Artificial Intelligence problems, theory of complexity of algorithms. Number of publications — 61. kosovtm@gmail.com; SPIIRAS, 14-th line V.O., 39, St. Petersburg, 199178, Russia; office phone +7(812)328-0421.

Рекомендовано лаб. ИТУР СПИИРАН, зав. лаб. — А.В. Тимофеев, д-р техн. наук, проф.

Статья поступила в редакцию 08.12.2010.

РЕФЕРАТ

Косовская Т.М. **Некоторые задачи искусственного интеллекта, допускающие формализацию на языке исчисления предикатов, и оценки числа шагов их решения.**

Для задач, допускающих формализацию на языке исчисления предикатов, вводится понятие описания $S(\omega)$ объекта ω как множества атомарных формул или их отрицаний, истинных на частях объекта, а также понятие целевой формулы $A_k(\overline{x})$ — формулы в ДНФ, содержащей в качестве атомарных только формулы с предикатами, входящими в описания объектов. В статье рассматриваются алгоритмы и оценки числа их шагов, решающие следующие задачи.

Задача идентификации. *Проверить, удовлетворяет ли объект ω или его часть целевой формуле $A_k(\overline{x})$ и предъявить эту часть объекта.*

Задача классификации. *Найти все такие номера k, что верна формула $A_k(\overline{\omega})$.*

Задача анализа. *Найти и классифицировать все части τ объекта ω, для которых $A_k(\overline{\tau})$.*

Для таких областей искусственного интеллекта, как распознавание образов, медицинская диагностика, анализ рынка эти задачи сведены к проверке справедливости секвенций $S(\omega) \vdash \exists\overline{x}_{\neq}\ A_k(\overline{x})$, $S(\omega) \vdash \bigvee_{k=1}^{M} A_k(\overline{\omega})$, $S(\omega) \vdash \bigvee_{k=1}^{M} \exists\overline{x}_{\neq}\ A_k(\overline{x})$.

Доказано, что все эти задачи являются NP-трудными. Для некоторых алгоритмов, решающих эти задачи, доказаны верхние оценки числа их шагов.

Для алгоритма полного перебора эта оценка составляет $O(t^m|A||S|)$, где t — число элементарных частей в объекте исследования, m — число аргументов в целевой формуле, $|A|$ — число атомарных формул в целевой формуле, $|S|$ — число атомарных формул в описании объекта.

Для алгоритмов, основанных на построении вывода в исчислениях предикатов, эта оценка составляет $O(D \cdot s^{\widetilde{a}})$, где D — число дизъюнктов в целевых формулах, s — максимальное число вхождений одного и того же предиката (только без отрицаний или только с отрицаниями) в описание объекта, $\widetilde{a}$ — максимальное число вхождений атомарных формул в элементарные конъюнкции, составляющие целевые формулы.

Эти результаты позволяют выбирать алгоритм в зависимости от структуры исходных предикатов и целевых формул, так как при использовании различных алгоритмов показатель степени полинома, ограничивающего число шагов работы алгоритма, будет различным.

SUMMARY

Kosovskaya T.M. **Some artificial intelligence problems permitting formalization by means of predicate calculus language and upper bounds of their solution steps.**

For problems permitting their formalization in predicate calculus language, a notion of a description $S(\omega)$ of an object ω as a set of atomic formulas or their negations which are valid for parts of the object as well as a notion of a goal formula $A_k(\overline{x})$ as a formula in the DNF containing only ftomic formulas with the predicates from the object description are introduced. Algorithms and their upper bounds of steps solving the following problems are considered in the paper.

Identification Problem. *Whether the object ω or its part satisfies the goal formula $A_k(\overline{x})$ and extract such a part of the object.*

Classification Problem. *To find all such numbers k that $A_k(\overline{\omega})$ is valid.*

Analysis Problem. *To find and classify all parts τ of th object $\omega$for which $A_k(\overline{\tau})$ is valid.*

For Artificial Intelligence problems concerning partern recognition, medical diagnostics, market analysis these problems are reduced to the checking of sequents $S(\omega) \vdash \exists \overline{x}_{\neq} A_k(\overline{x})$, $S(\omega) \vdash \bigvee_{k=1}^{M} A_k(\overline{\omega})$, $S(\omega) \vdash \bigvee_{k=1}^{M} \exists \overline{x}_{\neq} A_k(\overline{x})$.

It is proved that all these problems are NP-hard. For some algorithms solving these problems upper bounds of their steps are proved.

For an exhaustion algorithm such a bound is $O(t^m \cdot |A| \cdot |S|)$, where t is a number of elementary parts in the object under consideration, m is a number of arguments in the goal formula, $|A|$ is a number of atomic formulas in the goal formula, $|S|$ is a number of atomic formulas in the object description.

For the algorithms based on the construction of derivation in a predicate calculus such a bound is $O(D \cdot s^{\widetilde{a}})$, where D is a number of disjuncts in the goal formulas, s is maximum of the numbers of occurences of a predicate (only without negations and only with negations), $\widetilde{a}$ is maximum of the numbers of atomic formulas in an elementary conjunction forming the goal formula.

These results allow to choose the algorithm un the connection of the structure of initial predicates and goal formulas. It may be made because the use of different algorithms provides the different exponent of a polynom bounding the number of algorithm steps.