На правах рукописи

Петухова Нина Дмитриевна

Разработка и оценка числа шагов работы алгоритмоврешения задач логико-предметного распознавания образов с использованием тактик обратного метода Маслова

05.13.17 – Теоретические основы информатики

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
кандидата физико-математических наук

Санкт-Петербург
2016

Работа выполнена на кафедре «Математики» Федерального государственного бюджетного образовательного учреждении высшего профессионального образования «Санкт-Петербургский государственный морской технический университет»

Научный руководитель: **Косовская Татьяна Матвеевна,**
доктор физико-математических, доцент, профессор кафедры «Информатики» федерального государственного бюджетного образовательного учреждения высшего образования «Санкт-Петербургский государственный университет» (СПб ГУ).

Официальные оппоненты: **Бельтюков Анатолий Петрович,**
доктор физико-математических наук, профессор, профессор федерального государственного бюджетного образовательного учреждения высшего профессионального образования «Удмуртский государственный университет».

Пак Вадим Геннадьевич,
кандидат физико-математических наук, доцент, научный сотрудник федерального государственного автономного образовательного учреждения высшего образования «Санкт-Петербургский политехнический университет Петра Великого».

Ведущая организация:

Защита состоится «___» _______ 2016 года в ___:___ на заседании диссертационного совета Д 212.232.51 на базе Санкт-Петербургского университета по адресу: ____. С диссертацией можно ознакомиться по адресу ____ и на сайте ____

Автореферат разослан «___» _______ 2016 года.

Ученый секретарь диссертационного совета
Д 212.232.51 д.ф.-м.н., профессор Демьянович Юрий Казимирович

Общая характеристика работы

Актуальность темы.Во многих задачах искусственного интеллекта исследуемый объект характеризуется не только своими глобальными признаками, задающими его свойства как единого целого, но и свойствами его частей и отношений между ними. При распознавании контурного изображения объект может быть представлен как множество его вершин и вида соединений этих вершин. В задачах диагностики объект задаётся набором своих составляющих, а также зависимостями между этими элементами, позволяющими объекту функционировать.

Такой подход ещё в 70-ые годы XX века был сформулирован в литературе. Однако, в те годы, практического применения он практически не получил в силу высокой вычислительной сложности возникающих при этом задач, которые являются NP-трудными. Тем не менее, ввиду постоянно возрастающей производительности вычислительной техники, в последние 15 лет интерес к этому вопросу возрос.

При изучении алгоритмов одно из ключевых мест занимает анализ времени их работы. Первым шагом является нахождение асимптотических оценок: верхней и нижней границ работы алгоритмов.

В настоящее время для множества задач найдены только такие алгоритмы, которые перебирают экспоненциально зависящее от длины записи исходных данных количество вариантов. Такие задачи, как правило,являются NP-трудными. Ясно, что при больших размерах решаемых задач применение алгоритмов с экспоненциальной оценкой иррационально, хотя для задач небольшого объема экспоненциальные решения могут оказаться лучше полиномиальных (например, при $n \geq 100$ экспоненциальная оценка 1.1^n лучше, чем $10n^3$).

В данной работе построены и доказаны оценки числа шагов работы алгоритмов решения задач логико-предметного распознавания образов, использующих тактики обратного метода, который был предложен в середине 60ых годов советским логиком С.Ю. Масловым (1932 – 1983).

Обратный метод С.Ю. Маслова не является менее эффективным методом автоматического доказательства теорем, чем, например, метод резолюций.При практическом применении, как показано вдиссертации, он является даже более эффективным, но не получил широкого распространения в программировании. Одной из причин этого является сложность обратного метода в оригинальном изложении С.Ю. Маслова.

В диссертации предложена конкретизация обратного метода для частного случая формул исчисления предикатов, к которым сводится решение некоторых задач искусственного интеллекта. А также разработаны алгоритмы решения задач логико-предметного распознавания образов, использующие тактики обратного метода Маслова, и доказаны оценки числа их шагов. Эффективность этих алгоритмов проиллюстрирована на примерах. Представленные алгоритмы пригодны для программной реализации.

Степень разработанности темы исследования. Термин «обратный метод» впервые введен С.Ю. Масловым в 1964 году. Автором был разработан метод установления выводимости для формул классического исчисления предикатов без равенства и функциональных символов.

Интерес к обратному методу установления выводимости появился в связи с серией работ С.Ю. Маслова и группы математической логики ЛОМИ АН СССР, опубликованных в период с 1694 по 1972 годы. В 1973 году Н. Нильсон написал книгу,посвященную методам поиска решений в пространстве состояний. В 1976 году Р. Дуда и П. Харт изложили исистематизировали методы распознавания образов и дали анализ пространственных сцен по их плоскому изображению. В 1981 году Г.Е. Минц, назвав обратный метод «резолютивными исчислениями», применил его к неклассическим пропозициональным логическим исчислениям. В 1982 году М. Гэри и Д. Джонсон написали книгу, посвященную вопросам сложности решения задач.В 1985 году А.А. Воронков рассмотрел обратный метод для классической логики без равенства. В 1989 году выходит работа В. Лифшица в которой представлены основные идеи обратного метода в форме, подчеркивающей его связь с методом резолюций. В 1992 году А.А. Воронковым и А.Н. Дегтяревым были предложены метод свободных переменных для классической логики с равенством, а также ряд критериев избыточности. В 1996 году А.А. Воронковприводит описание обратных исчислений свободных переменных для неклассических предикатных логик. А в 1996 году Т. Таммет предложил реализацию обратного метода для предикатной интуиционистской логики. В 2001 году В.В. Бурлуцким была предложена реализация обратного метода для модальной логики КТ. В 2004 году В.П. Оревков применил обратный метод и доказал разрешимость нового хорновскогофрагмента исчисления предикатов. В 2005 году Д.С. Ларионов разработал конкретизацию обратного метода для автоэпистемической логики, которая применялась при построении экспертных систем. Г.Е. Минц в 2010 году доказал разрешимость класса E, используя обратный метод. Кроме того, в 2010 году была предложена реализация обратного метода с применением оператора получения благоприятного набора. А в 2015 году В.А. Филипповским была предложена и реализована система автоматического поиска доказательств теорем, основанная на обратном методе, а также проведено сравнение обратного метода и метода резолюций.

Целью диссертационной работы является научное обоснование иразработкаалгоритмов решения задач логико-предметного распознавания образов на основе обратного метода Маслова, а также доказательство асимптотических оценок числа шагов работы этих алгоритмов. Для выполнения поставленной цели необходимо было решить следующие **задачи**:

1. Провести исследование обратного метода, муравьиных алгоритмов, параллельных вычислений и задачи нахождения наибольшей общей подформулы.

2. Сформулировать конкретизацию обратного метода для решения задач логико-предметного распознавания образов, допускающих формулировку средствами языка исчисления предикатов.

3. На основе сформулированного метода разработать алгоритм решения задач логико-предметного распознавания образов.

4. Модифицировать этот алгоритм, используя тактики муравьиных алгоритмов и параллельных вычислений.

5. Использовать последний алгоритм для решения задачи нахождения наибольшей общей подформулы.

6. Получить асимптотические оценки всех построенных алгоритмов.

Цели и задачи диссертационной работы соответствуют области исследований паспорта специальности 05.13.17 – «Теоретические основы информатики»: пунктам 2 (Исследование информационных структур, разработка и анализ моделей информационных процессов и структур), 5 (Разработка и исследование моделей и алгоритмов анализа данных, обнаружения закономерностей в данных и их извлечениях разработка и исследование методов и алгоритмов анализа текста, устной речи и изображений) и 7(Разработка методов распознавания образов, фильтрации, распознавания и синтеза изображений, решающих правил. Моделирование формирования эмпирического знания).

Объектами исследования диссертационной работы являются формулы исчисления предикатов, к которым сводятся многие задачи искусственного интеллекта, и обратный метод Маслова. **Предметом исследования** являются разработка и реализация алгоритмов решения задач искусственного интеллекта.

Методология работы основана на методах математического моделирования, анализа и синтеза теоретического и практического материала. Для решения задач, поставленных в работе, использовались методы следующие теоретические **методы**: построения математических моделей сложных систем, теории сложности алгоритмов, теории распознавания образов. В практической части работы применялись методы построения и анализа алгоритмов.

Научная новизна данной работы заключается в следующем.

1. Сформулирована конкретизация обратного метода для решения задач логико-предметного распознавания образов, допускающих формулировку средствами языка исчисления предикатов.

2. Разработан алгоритм IMA решения задач логико-предметного распознавания образов, основанный на обратном методе. На основе этого алгоритма построен алгоритм IAPTA, использующий тактики муравьиных алгоритмов и параллельных вычислений. Сформулирован алгоритм PHIAPTA для решения задачи нахождения наибольшей общей подформулы. Получены асимптотические оценки всех построенных алгоритмов.

Теоретическая и практическая значимость работы. Полученные результаты исследования применимы при разработке систем распознавания образов, объекты исследования в которых характеризуются свойствами своих элементов и отношениями между ними, вследствие чего они допускают формализацию на языке исчисления предикатов. Разработанные алгоритмы отличаются легкостью практической реализации.

Степень достоверности и апробация результатов. Результаты диссертационной работы были представлены и обсуждались на научных конференциях: СПИСОК-2012, Санкт-Петербург, Россия, 25-27 апреля 2012; XV Международная научная конференция «Informationtheoriesandapplications» (ITA 2012), Варна, Болгария, 18 июня-6 июля 2012; СПИСОК-2013, Санкт-Петербург, Россия, 23-26 апреля 2013; XVI Международная научная конференция «Informationtheoriesandapplications» (ITA 2013), Варна, Болгария, 29 июня-11 июля 2012, СПИСОК-2016, Санкт-Петербург, Россия, 26-29 апреля 2016.

Основные результаты диссертации были получены в рамках выполнения исследований при финансовой поддержке РФФИ (проект 14-08-01276).

Публикации результатов. По теме диссертации опубликовано 7 работ, из них 3 статьи в научных журналах из перечня российских рецензируемых журналов, в которых должны быть опубликованы основные научные результаты диссертаций на соискание учёных степеней доктора и кандидата наук [1, 2, 3], 2 статьи в зарубежных научных журналах на английском языке [4, 5] и тезисы 2 докладов [6, 7]. В статьях [1, 2, 4 и 5] диссертанту принадлежит построение алгоритмов решения задач логико-предметного распознавания образов, основанных на обратном методе Маслова и нахождение асимптотических оценок этих алгоритмов. В статье [3] диссертантом предложено применение построенных ранее алгоритмов для решения задачи нахождения наибольшей общей с точностью до имён аргументов подформулы двух элементарных конъюнкций атомарных формул.

Личный вклад автора. Результаты, представленные в диссертационной работе, получены соискателем либо самостоятельно, либо при его непосредственном участии.

Положения, выносимые на защиту

1. Построен алгоритм IMA поиска вывода формул, к доказательству которых сводятся многие задачи Искусственного Интеллекта, основанный на тактиках обратного метода Маслова. Получены асимптотические оценки числа шагов работы этого алгоритма.

2. Построен алгоритм IAPTA поиска вывода формул, к доказательству которых сводятся многие задачи Искусственного Интеллекта, основанный на тактиках обратного метода Маслова, а также тактиках муравьиных алгоритмов с использованием идей параллельных вычислений. Получены асимптотические оценки числа шагов работы этого алгоритма.

3. Построен алгоритм PHIAPTA поиска наибольшей общей подформулы, основанный на алгоритме IAPTA. Получены асимптотические оценки числа шагов работы этого алгоритма.

Структура и объем диссертации. Работа состоит из введения, четырех глав, разбитых на разделы подразделы, заключения, списка литературы и трехприложений. Общий объем рукописи составляет 157 страниц текста, включая 88 страниц непосредственно текста диссертации и 69 страниц приложений. Текст содержит 21 рисунок. Библиографический список включает 105 наименований.

Содержание работы

Во введении обосновывается актуальность темы исследования, формулируется цель работы, дается обзор литературы по изучаемой проблеме, приводится краткое содержание работы по главам.

В первой главе приведен обзор современного состояния предметной области по рассматриваемой в диссертации теме. В разделе 1.1 рассматривается постановка задач Искусственного Интеллекта при логико-предметном подходе. В разделе 1.2 приведена формулировка обратного метода С.Ю. Маслова для решения задач логико-предметного распознавания образов. В разделах 1.3-1.6 даны обзоры тем алгоритмов муравья, параллельных вычислений и неполной выводимости предикатных формул соответственно.

Во второй главе сформулирован обратный метод Маслова для решения задач логико-предметного распознавания образов, построен алгоритм IMA решения таких задач на основе обратного метода, получены асимптотические оценки этого алгоритма.

В разделе 2.1 сформулирован обратный метод для доказательства выводимости формул вида

$$\exists(x_1,...,x_n)_{\neq}\left(\underset{i=1}{\overset{\delta}{\&}} D_i\left(a_1,...,a_k,x_1,...,x_n\right)\right) \qquad (1)$$

Эта формула является частным случаем формул, для которых разработан обратный метод Маслова. В (1) дизъюнкты $D_i\left(a_1,..., a_k, x_1,..., x_n\right)$ имеют вид $D_i = \left(\vee\neg S\left(a_1,...,a_k\right)\vee P_{k_1}\left(x_1,...,x_n\right)\right)$,где $\vee\neg S\left(a_1,...,a_k\right)$ – обозначение для дизъюнкции отрицаний атомарных формул, входящих в $S\left(a_1,..., a_k\right)$.

Определение 1. *Любой список Г формул вида* $D_i\left(a_1,..., a_k, c_1,..., c_n\right)$ *является**F-набором** для формул вида* (*1*)*, если формулы в этом списке не повторяются*.

Определение 2. *Пусть* $\alpha_1,...,\alpha_l$ *– список констант из списка* $a_1,...,a_k$. $\beta_1,...,\beta_l$ *– список переменных и констант из списка* $a_1,...,a_k$*. Рассмотрим систему равенств:*

$$\begin{cases}\alpha_1 = \beta_1\\ \quad\dots\\ \alpha_l = \beta_l\end{cases} \qquad (2)$$

Пусть $u_1,...,u_p$ *– список без повторений всех переменных, входящих в равенства системы* (2). *Систему* (2) *будем называть* ***системой уравнений в переменных*** $u_1,...,u_p$ *.**Решением системы уравнений***(2) *будем называть всякий набор значенийконстант* $\gamma_1,...,\gamma_p$ *из списка* $a_1,...,a_k$ *такой, что в результате одновременной замены всех переменных* $u_1,...,u_p$ *на константы* $\gamma_1,...,\gamma_p$ *соответственно левые и правые части каждого равенства системы совпадут.* ***Система уравнений*** **(2)** ***не имеет решений****, если в списке* $\gamma_1,...,\gamma_p$ *есть повторения, или если в системе уравнений* (2) *приходится сравнивать разные константы.*

Определение 3. *F-набор называется* ***пустым*** □*, если все формулы, входящие в него, не имеют переменных и тавтологичны.*

Определение 4.*F-набор называется* ***тупиковым****, если в него входит хотя бы одна формула, не имеющая переменных и являющаяся ложной или не являющаяся ни тавтологией ни противоречием.*

В разделе 2.2 сформулирован алгоритм IMA (inversemethodalgorithm), использующий тактику обратного метода.

Алгоритм IMA (inverse method algorithm)

1. Строим δ-членный F-набор, формулы в котором не повторяются. То есть в формуле (1) удаляем все знаки конъюнкций и составляем список элементарных дизъюнкций вида $\vee \neg S\left(a_1,...,a_k\right) \vee P_{k_i}\left(x_1,...,x_{n_i}\right)$.

2. Присваиваем значения переменным следующим образом:

2.1. Отменяем все пометки об удалениях предикатных формул из *S(ω)* (если они были).

2.2. Берем формулу $\vee \neg S\left(a_1,...,a_k\right) \vee P_{k_i}\left(t_1,...,t_m\right)$ из рассматриваемого F-набора, содержащую m-местный предикатный символ такой, что набор $t_1,...,t_m$ содержит хотя бы одну переменную.

2.3. Ищем в *S(ω)*формулу $P_{k_i}\left(v_1,...,v_m\right)$ при наборе констант $v_1,...,v_m$ с отрицанием перед выбранным в п. 2.2 предикатным символом. Если нашли подходящую формулу – помечаем её как удаленную и переходим к пункту 2.4, если её не существует, то переходим к пункту 3.

2.4. Решаем систему уравнений, отождествляющую списки переменных $t_1,...,t_m$ и констант $v_1,...,v_m$. В случае, если эта система имеет решение, перейти к пункту 2.5, если система решений не имеет, то перейти к пункту 2.3.

2.5. Заменяем во всем F-наборе переменные из списка $t_1,...,t_m$ на их значение, полученные в пункте 2.4.

2.6. В полученном F-наборе удаляем повторения формул.

2.7. Если получился пустой F-набор, то алгоритм заканчивает работу.

2.8. Если получился тупиковый F-набор – перейти к пункту 3.

2.9. Если в F-наборе все формулы, имеющие переменные помечены как удаленные – переходим к пункту 4.

2.10. Если в F-наборе существуют элементарные дизъюнкции, имеющие переменные, которым еще не присвоено значение, то перейти к пункту 2.2.

3. Возвратная часть алгоритма.

3.1. Отменяем последнее действие пункта 2.5, если это возможно, и переходим к пункту 2.3

3.2. Если отмена последнего действия пункта 2.5 невозможна, то помечаем атомарную формулу $P_{k_i}(t_1,...,t_m)$ как удаленную и переходим к пункту 2.

4. Если все формулы помечены как удаленные значит, формула не выводима. Алгоритм заканчивает работу.

В разделе 2.3 получены асимптотические оценки числа шагов работы алгоритма IMA.

Определение 5.*Однимшагомработы алгоритма* *называется как присвоение переменной значения (решение одного уравнения вида x=a), так и проверка на графическое совпадение атомарных формул или подстановка значений переменных в атомарные формулы, имеющие вхождения данных переменных.*

Пусть l – наибольшее количество аргументов в атомарной формуле;
s+1 – количество атомарных формул в каждом из дизъюнктов;
s_k – количество вхождений в исходную формулу атомарных формул с k-м предикатным символом.

Теорема 1. (Нижняя оценка работы алгоритма IMA.)*Количество шагов решения задачи искусственного интеллекта, сведённой к доказательству следствия вида* $S(\omega) \Rightarrow \exists \bar{x}_{\neq} A(\bar{x})$ *при использовании алгоритма IMA, основанного на тактиках Обратного метода не менее* $O(sl)$*шагов.*

Теорема 2. (Верхняя оценка числа шагов работы алгоритма IMA.)*Количество шагов затрачиваемых на решение задачи искусственного интеллекта, сведённой к доказательству следствия вида* $S(\omega) \Rightarrow \exists \bar{x}_{\neq} A(\bar{x})$*при использовании алгоритма IMA, основанного на тактиках Обратного метода, а также нахождения значений для переменных, существование которых утверждается в заключении логического следования задачи не превосходит* $O\left(\delta \cdot \max_k s_k{}^{\delta} \cdot l\right)$ *шагов.*

В третьей главе описаны идеи применения муравьиных алгоритмов и параллельных вычислений для решения задач логико-предметного распознавания образов с помощью обратного метода Маслова.

Используя идеи муравьиных алгоритмов, создаем многопроцессорную систему, в которой действия между процессорами распределены поровну. Равное разделение действий необходимо для того, чтобы каждое действие имело возможность стать первым выполненным. Каждый процессор может выполнять следующие простые действия:

1) присвоение значения переменным;
2) замена всех вхождений переменной на её значение;
3) проверка формул на графическое совпадение;
4) отмена присвоения значения переменным;
5) отмена замены всех вхождений переменной;
6) изменение приоритета данного действия на 0.

В разделе 3.1 сформулирован алгоритм IAPTA, основанный на применении обратного метода, муравьиных тактик и параллельных вычислений.

АлгоритмIAPTA (Inverse Ant Parallel Tactic Algorithm)

1. Строим δ-членный F-набор, формулы в котором не повторяются. То есть переписываем без конъюнкций все дизъюнкты вида $\vee \neg S(\omega) \vee P_{k_i}\left(x_1,\dots,x_{n_i}\right)$ при $i=1,\dots,\delta$. Создаем популяцию из δ процессоров.

Каждой паре потенциально контрарных формул $P_{k_i}\left(x_1,\dots,x_{n_i}\right)$ и $\neg P_{k_i}\left(a_{j_1},\dots,a_{j_{n_i}}\right)$, входящих в один F-набор, назначаем приоритет их отождествления равным 1. Остальные приоритеты назначаем равными 0.

2. Копируем δ-членный F-набор $\delta - 1$ раз. Получаем ровно δ одинаковых F-наборов. Назначаем i-му процессору $(i=1,\dots,\delta)$ свою начальную формулу $P_{k_i}\left(x_1,\dots,x_{n_i}\right)$ – формулу, с которой данный процессор начинает свой итерационный цикл, и потенциально контрарную ей формулу из $S(\omega)$, имеющую приоритет, равный 1. Если каким-то двум процессорам назначены формулы, начинающиеся с одного и того же предикатного символа (таких формул не более δ), то назначаем для них разные формулы из $S(\omega)$, потенциально контрарные данной.

Если для каких-то двух процессоров не существует разных потенциально контрарных формул, то формула не выводима. Алгоритм заканчивает работу. Иначе, переходим к п. 3.3.

3. Параллельно работают δ процессоров; i-й процессор $(i=1,\dots,\delta)$ осуществляет присвоение значений переменным.

3.1. Если в рабочей формуле данного процессора нет переменных, то в качестве рабочей для этого процессора выбираем формулу из следующей элементарной дизъюнкции, содержащую хоть одну переменную.

3.2. Ищем среди формул в $S(\omega)$формулу $\neg P_{k_i}\left(a_{j_1},...,a_{j_{n_i}}\right)$, имеющую приоритет, равный 1, и потенциально контрарную формуле $P_{k_i}\left(t_1,...,t_{n_i}\right)$, с которой работает этот процессор. Если нашли подходящую формулу, то переходим к п. 3.3. Если ее нет, то переходим к п. 4.

3.3. Решаем систему уравнений вида $t_l = a_{j_l}$ $\left(l = 1,..., n_i\right)$, унифицирующую список переменных и констант со списком констант. В случае, если эта система имеет решение, то переходим к п. 3.4. Если система решений не имеет, то понизить приоритет этого действия до 0 и переходим к п. 3.2.

3.4. Записываем результаты, полученные разными процессорами, и проверяем их на непротиворечивость следующим образом. Если процессоры одновременно присваивают одним и тем же переменным разные значения, то такие результаты считаются противоречивыми. Если результаты действий двух процессоров не противоречат друг другу, то присвоение полученных значений переменных осуществляется в формулах обоих процессоров.

3.5. Заменяем в F-наборе каждого процессора вхождения переменных из списка на их значения, полученные в п. 3.3 и 3.4, если успешно пройдена проверка на непротиворечивость.

3.6. Если для какого-либо процессора получился пустой F-набор, то алгоритм заканчивает работу. Формула выводима и найден набор значений переменных, существование которых утверждалось в формуле.

3.7. Если получился тупиковый F-набор, то переходим к п. 4.

3.8. Если для всех процессоров приоритеты всех действий равны 0, то формула не выводима. Алгоритм заканчивает работу.

3.9. Если в F-наборе какого-либо процессора существуют формулы, имеющие переменные, которым еще не присвоено значение, то переходим к п. 3.1.

4. Возвратная часть алгоритма.

4.1. Отменяем последнее действие п. 3.5, если это возможно, и переходим к п. 3.2.

4.2. Если для какого-либо процессора отмена последнего действия п. 3.5 невозможна, то алгоритм заканчивает работу.

4.3. Если все процессоры закончили работу, но пустой F-набор не получен, то алгоритм заканчивает работу. Формула не выводима.

В разделе 3.2 получены асимптотические оценки числа шагов работы алгоритма IAPTA

Теорема 3. (Нижняя оценка числа шагов работы алгоритма*IAPTA*.) *Количество шагов доказательства логического следования вида*

$S(\omega)\Rightarrow\exists\bar{x}_{\neq}A_k(\bar{x})$ *и нахождения значений переменных, для которых это следование имеет место, при использовании алгоритма IAPTA, не менее 2δ(s + 2) + 2 шагов.*

Теорема 4. (Верхняя оценка числа шагов работы алгоритма*IAPTA*.) *Количество шагов доказательства логического следования вида* $S(\omega)\Rightarrow\exists\bar{x}_{\neq}A_k(\bar{x})$ *и нахождения значений переменных, для которых это следование имеет место, при использовании алгоритма IAPTA составляет* $O\left(l\cdot\left(\max_k s_k\right)^{\delta}\right)$ *шагов.*

В четвертой главе рассматривается решение задачи выделения максимальной общей подформулы с использованием алгоритма IAPTA.

В разделе 4.1 приведена постановка задачи выделения максимальной общей подформулы двух заданных элементарных конъюнкций $A(\bar{x})$ и $\tilde{A}(\bar{y})$, с точностью до имён переменных, как доказательства логического следования[1] $A(\bar{x})\Rightarrow\exists\bar{y}\tilde{A}(\bar{y})$ с нахождением такой подформулы $\tilde{A}'(\bar{y}')$ формулы $\tilde{A}(\bar{y})$, что имеет место следствие $A(\bar{x})\Rightarrow\exists\bar{y}'\tilde{A}(\bar{y}')$,а также общего унификатора λ формул $A(\bar{x})$ и $\tilde{A}'(\bar{y}')$, то есть такой подстановки имён переменных $\bar{y}'$ вместо некоторых имён переменных из списка $\bar{x}$, что $\tilde{A}'(\bar{y}')$ является подформулой $A(\bar{x})$ после применения унификатора λ.

Если в алгоритме IAPTA заменить действие «присвоение значений переменным» на «отождествление переменных», этот алгоритм можно модифицировать до алгоритма PHIAPTA, и применить и для решения задачи выделения максимальной общей подформулы.

В разделе 4.2 приведен алгоритм PHIAPTA выделения максимальной общей подформулы двух заданных элементарных конъюнкций с точностью до имён переменных.

АлгоритмPHIAPTA (Partial Hatchability Inverse Ant Parallel Tactic Algorithm)

1. Строим δ-членный F-набор, формулы в котором не повторяются. То есть переписываем без конъюнкций все дизъюнкты вида $A(\bar{x})\vee P_{k_i}(y'_1,\ldots,y'_{n_i})$, при $i=1,\ldots,\delta$. Создаем популяцию из δ процессоров.

Каждой паре потенциально контрарных формул $P_{k_i}(y'_1,\ldots,y'_{n_i})$, и $\neg P_{k_j}(x_1,\ldots,x_{n_j})$, входящих в один F-набор, назначаем приоритет их отождествления равным 1. Остальные приоритеты назначаем равными 0.

[1] Косовская Т. М. Частичная выводимость предикатных формул как средство распознавания объектов с неполной информацией // Вестник СПбГУ. Сер. 10. 2009. Вып. 1. С. 74-84.

2. Копируем δ-членный F-набор $\delta-1$ раз. Таким образом получаем ровно δ одинаковых F-наборов. Назначаем i-му процессору ($i=1,...,\delta$) свою начальную формулу $P_{k_i}\left(y'_1,...,y'_{n_i}\right)$ – формулу, с которой данный процессор начинает свой итерационный цикл, и потенциально контрарную ей формулу вида $\neg P_{k_j}\left(x_1,...,x_{n_j}\right)$ из $A(\bar{x})$, имеющую приоритет, равный 1. Для каждого процессора назначаем переменную $l_i=0$ ($i=1,...,\delta$), означающую длину фрагмента формулы $A(\bar{x})$ в формуле $\tilde{A}'(\bar{y}')$.

Если какие-то два процессора начинают работу с формулой, начинающейся с одного и того же предикатного символа, то назначаем для них разные формулы из $A(\bar{x})$, потенциально контрарные данной. Если для каких-то двух процессоров не существует разных потенциально контрарных формул, берем для них одинаковые потенциально контрарные формулы и переходим к п. 3.3.

3. Параллельно работают δ процессоров; i-ый процессор ($i=1,...,\delta$) осуществляет отождествление переменных следующим образом.

3.1. Если в рабочей формуле данного процессора нет переменных, которые еще не проходили процедуру отождествления, то в качестве рабочей для этого процессора выбираем формулу из следующей элементарной дизъюнкции, содержащую хоть одну «не отождествленную» переменную.

3.2. Ищем среди формул в $A(\bar{x})$ формулу $\neg P_{k_i}\left(x_{j_1},...,x_{j_{n_j}}\right)$, имеющую приоритет, равный 1, и потенциально контрарную формуле $P_{k_i}\left(t'_1,...,t'_{n_i}\right)$ (из $\tilde{A}'(\bar{y}')$), с которой работает этот процессор. Если нашли подходящую формулу, то переходим к п. 3.3. Если ее нет, то переходим к п. 4.

3.3. Решаем систему уравнений вида $t'_l=x_{j_l}$ ($l=1,...,n_i$), унифицирующую списки переменных. В случае, если эта система имеет решение, то увеличиваем длину текущего фрагмента l_i на единицу, запоминаем это число и текущее отождествление и переходим к п. 3.4. Если система решений не имеет, то понизить приоритет этого действия до 0 и переходим к п. 3.2.

3.4. Записываем результаты, полученные разными процессорами, и проверяем их на непротиворечивость.

3.5. Заменяем в F-наборе каждого процессора вхождения переменных из списка на их значения, полученные в п. 3.3 и 3.4, если успешно пройдена проверка на непротиворечивость.

3.6. Если для какого-либо процессора все переменные из $\tilde{A}'(\bar{y}')$ получили новое значение, и выведен пустой F-набор, то алгоритм заканчивает работу. Формула $\tilde{A}'(\bar{y}')$ полностью содержится в формуле $A(\bar{x})$).

3.7. Если в F-наборе какого-либо процессора отсутствуют формулы, имеющие переменные, которые еще проходили процедуру отождествления, то переходим к п. 4.

3.8. Если для всех процессоров приоритеты всех действий равны 0, то переходим к п. 4.3.

3.9. Если в F-наборе какого-либо процессора существуют формулы, имеющие переменные, которые еще не проходили процедуру отождествления, то переходим к п. 3.1.

4. Возвратная часть алгоритма.

4.1. Отменяем последнее действие п. 3.5, если это возможно, уменьшаем соответствующую величину l_i на 1, запоминаем это число и текущее отождествление и переходим к п. 3.2.

4.2. Если для какого-либо процессора отмена последнего действия п. 3.5 невозможна, то процессор заканчивает работу.

4.3. Если все процессоры закончили работу, то наибольшее число из всех l_i на всех шагах является размером наибольшей подформулы $\widetilde{A}'(\bar{y}')$ в $A(\bar{x})$. И соответствующие этому l_i отождествления являются наибольшей подформулой $\widetilde{A}'(\bar{y}')$ в $A(\bar{x})$.

В разделе 4.3 получены оценки числа шагов работы алгоритма PHIAPTA.

Теорема 5. (Нижняя оценка числа шагов работы алгоритма PHIAPTA) *Количество шаговвыделения максимальной общей подформулы двух элементарных конъюнкций* $A(\bar{x})$ *и* $\widetilde{A}(\bar{y})$, *при использовании алгоритма* PHIAPTA, *не менее* 4δ + 2δs − 1 + *l шагов*.

Теорема 6. (Верхняя оценка числа шагов работы алгоритма PHIAPTA.) *Количество шаговвыделения максимальной общей подформулы двух элементарных конъюнкций* $A(\bar{x})$ *и* $\widetilde{A}(\bar{y})$, *при использовании алгоритма* PHIAPTA, *не превосходит* $O\left(\delta\cdot l\cdot\left(\max\limits_{k} s_k\right)^{\delta}\right)$ *шагов*.

В приложениях A, B и C приведены модельные примеры применения алгоритмов IMA, IAPTA и PHIAPTA соответственно для решения задач логико-предметного распознавания образов.

В заключении приведены **итоги** выполненного исследования, которые заключаются в следующем:

1. Сформулирована и обоснована адаптация обратного метода Маслова для доказательства выводимости формул вида $\exists(x_1,\ldots,x_n)_{\neq}\left(\mathop{\&}\limits_{i=1}^{\delta} D_i(a_1,\ldots,a_k,x_1,\ldots,x_n)\right)$, к доказательству выводимости которых сводятся многие задачи Искусственного Интеллекта, объекты исследования в которых характеризуются свойствами своих

элементов и отношениями между этими элементами, а, следовательно, допускающие формализацию средствами языка исчисления предикатов.

2. Разработан алгоритм IMA выводимости формул вида $\exists(x_1,...,x_n)_{\neq}\left(\underset{i=1}{\overset{\delta}{\&}} D_i(a_1,...,a_k,x_1,...,x_n)\right)$, основанный на разработанной адаптации обратного метода. Доказаны асимптотические оценки числа шагов работы этого алгоритма.

3. Разработана модификация IAPTA алгоритма IMA, использующая тактики муравьиных алгоритмов и параллельных вычислений. Доказаны асимптотические оценки числа шагов работы этого алгоритма.

4. Обоснована возможность применения обратного метода Маслова для решения задачи выведения максимальной общей подформулы. Сформулирован алгоритм PHIAPTA выделения максимальной общей с точностью до имен аргументов подформулы двух элементарных конъюнкций. Получены асимптотические оценки числа шагов работы этого алгоритма.

Были сформулированы следующие **рекомендации** по применению результатов работы.

1. Сформулированная конкретизация обратного метода Маслова является простым и понятным средством доказательства выводимости формул вида $\exists(x_1,...,x_n)_{\neq}\left(\underset{i=1}{\overset{\delta}{\&}} D_i(a_1,...,a_k,x_1,...,x_n)\right)$, поэтому её можно применять для первоначального знакомства с обратным методом.

2. Построенные алгоритмы полностью готовы к программной реализации и могут быть применены для решения различных задач Искусственного Интеллекта в рамках логико-предметного подхода.

Так же были предложены **перспективыдальнейшей разработки тематики, такие как** построение модификаций предложенных алгоритмов для решения различных задач искусственного интеллекта, программную реализацию построенных алгоритмов, а также качественное сравнение полученного метода с существующими методами решения задач логико-предметного распознавания образов и теории выводимости, например, с методом резолюций.

Публикации автора по теме диссертации

Публикации в журналах из перечня российских рецензируемых научных журналов, в которых должны быть опубликованы основные научные результаты диссертаций на соискание учёных степеней доктора и кандидата наук

1. Косовская, Т. М. Решение задач логико-предметного распознавания образов с использованием тактик обратного метода Маслова / Т.М. Косовская, Н.Д. Петухова // Компьютерные инструменты в образовании – 2014. – Вып. 3. – с. 9-20. – ISSN: 20712340

2. Петухова, Н. Д. Применение тактик муравьиных алгоритмов для решения некоторых задач Искусственного Интеллекта / Н.Д. Петухова, Т.М. Косовская // Вестн. С.-Петерб. ун-та. Сер 10. – 2015. – Вып. 3. – С. 67-82. – ISSN 1811-9905
3. Петухова, Н. Д. Выделение максимальной общей предикатной подформулы с помощью обратного метода Маслова / Н.Д. Петухова // Компьютерные инструменты в образовании – 2015. – Вып. 4. – С. 17-25. – ISSN: 20712340

Прочие публикации

4. Kosovskaya, T. The Inverse Method for Solving Artificial Intelligence Problems in the Frameworks of Logic-Objective Approach and Bounds of its Number of Steps / T. Kosovskaya, N. Petuchova // International Journal «Information Models and Analyses» – 2012. – Vol. 1. – P. 84-93. – ISSN 1314-6416
5. Kosovskaya, T. The Maslov's Inverse Method and Ant Tactics for Exhaustive Search Decreasing / T. Kosovskaya, N. Petuchova // International Journal «Information Models and Analyses» – 2013. – Vol. 2. Num. 1. – P. 81-89. – ISSN 1314-6416
6. Петухова, Н. Д. Обратный метод для решения задач логико-предметного распознавания образов и оценки числа шагов его работы / Н.Д. Петухова // Материалы научной конференции по проблемам информатики СПИСОК-2012 – С. 90-94. –ISBN 978-5-9651-0686-8
7. Петухова, Н. Д. Обратный метод Маслова и муравьиная тактика решения задач Искусственного Интеллекта / Н.Д. Петухова // Материалы научной конференции по проблемам информатики СПИСОК-2013 – С. 64-69. – ISBN 978-5-9651-0779-7

выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные
выходные данные