22 *J. Algebra*, 2012, 360, p. 21–52.

23 Jantzen J.C. *Representations of algebraic groups*, Boston, Pure and Applied Mathematics, 131, 1987.

24 Vinberg Ye.B., Onishchik A.L. *Seminar po gruppam Li i algebraicheskim gruppam*, Moscow: Nauka, 1988.

УДК 510.52 + .58

И.В.Латкин[1], А.В.Селиверстов[2]

[1]Восточно-Казахстанский государственный технический университет им. Д. Серикбаева, Усть-Каменогорск;
[2]Институт проблем передачи информации им. А.А. Харкевича РАН, Москва, Россия
(E-mail: slvstv@iitp.ru)

Вычислительная сложность фрагментов теории поля комплексных чисел

В статье обсуждена вычислительная сложность формул в предварённой форме с ограничением на число перемен кванторов. В частности, говорится о теории алгебраически замкнутых полей. Доказано, что распознавание вершин многомерного куба на гиперплоскости сводится к распознаванию особых точек на гиперповерхности, построенной за полиномиальное время. Более того, доказаны некоторые соотношения между классами сложности. Даны рекомендации по улучшению концепции сложности, а также связи с теорией моделей.

Ключевые слова: вычислительная сложность, теория первого порядка, комплексные числа, перемена кванторов, иерархия классов.

1. Введение. Оптимизация функционала на множестве вершин многомерного куба является важной задачей, имеющей практические приложения в народном хозяйстве. Конкретные примеры и обычно используемые методы решения можно найти в работах [1–3]. Мы продемонстрируем сводимость таких задач к исследованию решений систем алгебраических уравнений над алгебраически замкнутым полем, которые легко интерпретируются в соответствующем языке первого порядка. Ввиду этого мы рассмотрим также разрешающие алгоритмы для фрагментов теории поля комплексных чисел, состоящих из формул, которые представимы в предварённой форме с числом перемен кванторов, ограниченной некоторой фиксированной величиной. Практическая ценность такого перехода объясняется возможностью более эффективного использования стандартных пакетов программ для символьных вычислений. Среди них Maple и свободно распространяемый пакет SINGULAR (http://www.singular.uni-kl.de). Поскольку многие задачи дискретной оптимизации являются алгоритмически трудными, поиск новых методов решения и эффективное использование готовых пакетов программ могут сократить как время работы вычислительных устройств, так и усилия, необходимые для разработки алгоритмов и отладки их программной реализации.

При формализации многих комбинаторных задач возникают универсально-экзистенциальные формулы. Последнее обстоятельство играет важную роль, в частности, они остаются истинными при переходе к индуктивному пределу, что позволяет использовать методы теории моделей [4, 5] при работе с алгебраически незамкнутыми полями. Покажем это на простом примере. Поле комплексных чисел элементарно эквивалентно алгебраическому замыканию поля рациональных чисел, которое является индуктивным пределом конечных алгебраических расширений. Поэтому замкнутая универсально-экзистенциальная формула, истинная в каждом конечном расширении поля рациональных чисел, будет истинной в поле комплексных чисел. Возможность выразить нужное свойство универсально-экзистенциальной формулой существенно зависит от класса иерархии, которому принадлежит задача распознавания. Поэтому методы теории моделей оказываются тесно связанными с исследованием вычислительной сложности.

2. Применение базисов Грёбнера. Напомним, что каждая полная рекурсивно аксиоматизируемая теория первого порядка разрешима. Примерами таких теорий служат теория алгебраически замкнутого поля фиксированной характеристики, теория вещественно замкнутых полей, теория плотных

линейных порядков без концевых элементов, арифметика Пресбургера. Чистая теория равенства разрешима на полиномиально ограниченной памяти и полна в этом классе [6; 336]. Но для большинства разрешимых теорий сложность разрешающего алгоритма очень велика. Например, известен алгоритм дважды экспоненциального времени для теории поля комплексных чисел. С другой стороны, любой разрешающий алгоритм теории алгебраически замкнутого поля фиксированной характеристики требует использовать (по меньшей мере) экспоненциальную память [7]. Нижние границы сложности арифметики Пресбургера и её фрагментов обсуждаются в [8]. Высокая сложность и у многих неполных теорий [9].

Использование базисов Грёбнера является весьма общим методом решения многих задач, связанных с системами алгебраических уравнений над алгебраически замкнутым полем [10; 128]. История их возникновения и, в частности, вклад в развитие теории выдающегося российского математика А.И. Ширшова описаны в [11]. Алгоритмы вычисления базисов Грёбнера входят во многие пакеты для символьных вычислений, включая Maple и Singular. Однако применение этих методов часто оказывается малоэффективным из-за появления в ходе вычислений многочленов очень высокой степени [12–14]. Время работы соответствующих алгоритмов дважды экспоненциальное или ещё выше.

Существуют другие методы, не связанные с нахождением базисов Грёбнера и позволяющие определить совместность системы алгебраических уравнений с рациональными коэффициентами за экспоненциальное время. Впервые эта возможность была показана в [15], в дальнейшем метод был немного усовершенствован [16]. Наряду с этим развивались вероятностные алгоритмы [17–19]. Однако все известные вероятностные методы решения систем уравнений также требуют экспоненциального времени в общем случае. Хотя недавно для решения этой задачи описан эффективный вероятностный алгоритм, имеющий низкую сложность при некоторых дополнительных ограничениях на число мономов в уравнениях [19].

В [20] найден разрешающий алгоритм для формул в предварённой форме с ограниченным числом перемен кванторов, время работы которого экспоненциально зависит от длины формулы, но дважды экспоненциально — от числа перемен кванторов.

3. Иерархии задач по сложности. Напомним вкратце строение так называемой полиномиальной иерархии языков PH [6; 203–208, 21]. Она и другие ей подобные иерархии были введены для более точной классификации проблем по сложности их решения, так как каждую разумно поставленную проблему, ответ на которую может быть только «да» или «нет», можно сформулировать в терминах принадлежности к соответствующему языку.

Нижним (нулевым) уровнем этой иерархии объявляется класс языков, распознаваемых детерминированными машинами Тьюринга за полиномиальное время. Таким образом, языки этого класса считаются наиболее легко распознаваемыми. Насколько оправдана подобная точка зрения, мы обсудим позднее. Следующий, первый, уровень иерархии состоит из двух классов языков. Языки одного класса, обозначаемого как Σ_1^P, распознаются недетерминированными машинами Тьюринга за полиномиальное время, а другой класс — Π_1^P содержит все дополнения до языков первого, т.е. это просто классы NP и $coNP$ соответственно. Если уже определены классы уровня k, то уровень $k+1$ состоит из двух подуровней. На «нижнем» находится класс Δ_{k+1}^P, языки которого распознаются детерминированными машинами Тьюринга за полиномиальное время, с использованием языка L из класса Σ_k^P в качестве оракула, т.е. «подсказчика», который может сказать мгновенно, принадлежит ли данное слово языку L. «Верхний» подуровень состоит из двух классов — Σ_{k+1}^P и Π_{k+1}^P. В Σ_{k+1}^P собраны все языки, распознаваемые недетерминированными машинами Тьюринга за полиномиальное время, и тоже с использованием языка из класса Σ_k^P в качестве оракула. В Π_{k+1}^P — все дополнения до языков первого класса. Правильно ли подуровни названы «верхним» и «нижним», до сих пор неизвестно, вполне вероятно, что при достаточно больших k (два или более) все классы иерархии PH совпадают. Неизвестно также, верно ли, что $\Sigma_{k+1}^P \cap \Pi_{k+1}^P = \Delta_{k+1}^P$, хотя бы при одном k (см. ниже п. 5). Но, тем не менее, сходство в обозначении уровней этой иерархии и уровней арифметической иерархии в классической теории рекурсии не случайно, так как язык L принадлежит классу Σ_{k+1}^P тогда и только тогда, когда существуют такие многочлен $p(n)$ и вычислимый за полиномиальное время предикат $R(x, y_1, ..., y_{k+1})$, что

$$x \in L \Leftrightarrow \exists y_1 \forall y_2 \exists y_3 ... Q\, y_{k+1}\ \&_{j=1}^{k+1}[|\, y_j\,| \leq p(|\, x\,|)\ \&\ R(x, y_1, ..., y_{k+1})],$$

где Q — это $\exists$, если k чётное, и $\forall$, если k нечётное, а $|\, x\,|$ — длина слова x [7]. Для языков из класса Π_{k+1}^{P} кванторы $\exists$ и $\forall$ в (1) меняются местами.

Объединение всех классов языков, входящих в какие-то уровни, описанные выше, называют *PH*-иерархией.

По аналогии с полиномиальной иерархией PH [21] определяется экспоненциальная иерархия EH классов сложности, нулевой уровень которой состоит из языков, разрешимых за экспоненциальное время $poly(\exp(n))$, где число n означает длину входа, а основание степени может быть любым числом, большим единицы. Хотя это не доказано, широко распространено мнение о том, что иерархия EH невырожденная. Также рассматривается иерархия $EXP - H$, нулевой уровень которой состоит из языков, разрешимых за время $\exp(poly(n))$. Аналогично определяется иерархия $DoubleEXP - H$ для дважды экспоненциального времени.

4. Некоторые свойства иерархий. Отметим, что вырождение полиномиальной иерархии PH влечёт вырождение других аналогичных иерархий. Доказательство основано на методе, называемом набивкой, или накачкой. Однако обратная импликация не доказана.

Рассмотрим, например, взаимосвязь оценок сложности вычислений по памяти и по времени. Пусть *P-space* означает класс языков, которые распознаются с использованием полиномиальной памяти, а *EXP-space* — класс языков, допускаемых с использованием памяти $\exp(poly(n))$, где число n означает длину входа, а $DoubleEXP$ — класс языков, допускаемых за время $\exp(\exp(poly(n)))$.

Теорема 1. Равенство классов $P - space = EXP$ влечёт равенство классов *EXP-space* = = *DoubleEXP*.

Доказательство. Очевидно, что любое множество класса *EXP-space* распознаваемо за дважды экспоненциальное время. Пусть *P-space* = *EXP*. Рассмотрим множество X слов в алфавите 0 и 1, распознаваемое за дважды экспоненциальное время $t(x)$ алгоритмом A. Обозначим Y множество слов с префиксом из $\log t(x)$ нулей, единицы и слова x. В слове из Y суффикс x однозначно восстанавливается по слову из Y: это символы правее самой левой единицы. Очевидная модификация алгоритма A допускает множество Y за экспоненциальное от длины входа время. По предположению множество Y принадлежит *P-space*. Следовательно, Y допускается алгоритмом, требующим памяти, экспоненциально ограниченной длиной суффикса x. Поскольку слова из Y однозначно определяются своими суффиксами x, составляющими множество X, то, таким образом, X принадлежит классу *EXP-space*.

Известно частичное вырождение иерархии AM, отражающей интерактивные взаимодействия с конечным числом раундов между вероятностной машиной (Артуром), работающей полиномиальное время, и машиной с неограниченными ресурсами (Мерлином) [22]. Неформально, Артур должен узнать истину, ведя диалог с Мерлином, обладающим гораздо большими возможностями, но при этом, проверяя, не обманывает ли его Мерлин. Конечное число раундов можно свести к частному случаю, когда Мерлин даёт все ответы сразу. При этом посредством диалога с полиномиальным числом перемежений вопросов и ответов можно моделировать работу произвольного алгоритма с полиномиально ограниченной памятью. Важное отличие AM от PH состоит в использовании Артуром вероятностного алгоритма. По аналогии с этим результатом можно было бы ожидать, что иерархия PH тоже вырождена и класс NP совпадает с двойственным классом $coNP$. Однако многочисленные попытки доказать или опровергнуть это утверждение не привели к успеху.

5. Полные языки. Для всякого сложностного класса языков, т.е. класса, выделяемого на основании «одинаковости» временной или ёмкостной сложности алгоритмов, которые распознают языки этого класса, важной характеристикой служат *полные* в этом классе языки. Такой язык L должен, во-первых, сам принадлежать этому классу, во-вторых, для каждого языка M из этого класса вопрос о принадлежности слов языку M должен полиномиально сводиться к аналогичному вопросу для языка L. Таким образом, полный для сложностного класса язык полностью его характеризует относительно сложности вычислений, можно поэтому сказать, что полный в данном классе язык — это его паспорт.

Ярким примером этому может служить класс NP, для него известно очень много полных языков [6; 64], часто называемых NP-полными. В их число входят языки, соответствующие таким признан-

но сложным задачам, как задача о существовании гамильтонова цикла, задачи о выполнимости формул исчисления высказываний и о возможности правильно раскрасить вершины графа в k цветов и многие другие широко известные задачи.

В последнее время список известных NP-полных проблем активно пополняется за счёт классических задач алгебры. Например, таковой является задача о вычислении геодезической длины элементов в свободной разрешимой группе ступени 2 и фиксированного ранга [23]. В то же время для некоторых, казалось бы, очень тесно связанных с этой задачей проблем равенства, сопряженности и степени имеются алгоритмы полиномиальной сложности [23]. Более того, эти алгоритмы являются полиномиальными не только от длины исследуемых слов, но также и от ранга и ступени разрешимости свободной разрешимой группы, что сильно контрастирует с давно известными алгоритмами для решения этих задач, основанными на вложении Магнуса. Недавно эти детерминированные полиномиальные алгоритмы удалось заметно упростить [24], интересно, что вероятностные аналоги этих алгоритмов, которые описаны там же, имеют в качестве верхней границы сложности многочлены, степени которых лишь на единицу меньше, чем у детерминированных алгоритмов.

Наличие полных языков для некоторых сложностных классов является открытой проблемой. Например, существование полного языка во всей иерархии PH равносильно тому, что она имеет только конечное число уровней, т.е. она является почти вырожденной.

В отличие от этого, уровни полиномиальной иерархии PH допускают простую характеристику на основе полных языков в каждом классе. Примеры полных языков из класса Δ_2^P можно найти в [25]. Для классов Σ_k^P и Π_k^P таковыми служат классы предварённых формул с соответствующим числом перемен кванторов в чистой теории равенства. Вся же чистая теория равенства является полной для класса $P-space$ [6; 213, 214], и неизвестно, принадлежит ли она иерархии PH. В последнем случае она была бы полной и там.

Отметим, что если иерархия PH (или аналогичная ей) не вырождена, то, помимо «стандартных» уровней иерархии, существуют и промежуточные классы. Аналогичная ситуация наблюдается в теории тьюринговых степеней неразрешимости. Можно ожидать, что существует много языков, которые принадлежат некоторому уровню иерархии, но не полны в нём и не принадлежат вложенным уровням. Тем не менее, известно лишь немного кандидатов из класса NP, для которых не доказана ни полиномиальная разрешимость, ни NP-полнота. Один из таких языков состоит из пар изоморфных графов.

В [26] обнаружена связь проблемы равенства класса P пересечению классов NP и $coNP$ с проблемой поиска второго решения NP-полной задачи при условии, что некоторое решение уже известно. Если P не равен пересечению NP и $coNP$, то существует пример, когда поиск второго решения нельзя выполнить за полиномиальное время, зная первое решение. Если для каждого языка из пересечения NP и $coNP$ можно за полиномиальное время вычислить соответствующее решение, то второе решение можно найти за полиномиальное время.

Известный результат [20] показывает, что формулы с ограниченным числом перемен кванторов в теории поля комплексных чисел разрешимы алгоритмами экспоненциального времени, хотя это время зависит от числа перемен кванторов. Это может служить косвенным указанием на то, что либо экспоненциальная иерархия $EXP-H$ вырождена, либо сложность фрагментов с ограниченным числом перемен кванторов у теории поля комплексных чисел существенно ниже, чем у известных в настоящее время алгоритмов разрешения. Последнее обстоятельство может иметь важное теоретическое и практическое значение.

С другой стороны, даже совместность систем алгебраических уравнений с целыми коэффициентами, которая выражается в теории полей экзистенциальной формулой, является NP-трудной задачей. А именно, NP-полную задачу о разбиении множества целых чисел на две части с одинаковыми суммами за полиномиальное время можно свести к задаче распознавания особой точки на гиперповерхности в пространстве достаточно большой размерности.

6. Комбинированная мера сложности алгоритма. Вернёмся к вопросу о том, почему принято считать, что полиномиальные алгоритмы являются быстрыми. Для обоснования этого часто ограничиваются простым указанием на тот факт, что экспонента с любым основанием b, большим единицы, и показателем, линейно зависящим от аргумента n, станет больше значения любого многочлена $f(n)$ при всех n, начиная с некоторого значения m, зависящего от b и f. Поэтому при всех входах,

чья длина n больше m, алгоритм с верхней оценкой времени $f(n)$ будет работать быстрее алгоритма с экспоненциальной оценкой. То же верно, например, для субэкспоненциальной функции $n^{\log(n)}$. Это теоретическое обоснование не всегда согласуется с практикой, поскольку иногда экспоненциальные алгоритмы работают быстрее полиномиальных даже на достаточно длинных входах.

Одна из причин этого в следующем. При подсчёте времени работы берутся во внимание только «внешние» действия программы, а именно, сколько и каких операций произвела машина с исходными данными и теми, что хранятся в оперативной памяти (к примеру, сложений, вычитаний, умножений, сравнений, пересылок из одних ячеек памяти в другие и т.п.). Аналогом этому для машин Тьюринга служит подсчёт числа стираний-записываний и сдвигов головки. Но в действительности время работы тратится не только на эти операции, но и на поиск нужной команды. Проиллюстрируем это простым примером.

Рассмотрим сначала алгоритм T, который решает некоторую задачу полиномиальным сведением её к одному из k частных случаев, для каждого из которых имеется свой полиномиальный алгоритм решения: вначале посредством подпрограммы T_0 определяется, какая из подпрограмм $T_1,\ldots,T_k$ годится для данного входа, а затем происходит переход на эту подпрограмму. Предположим, что эту же задачу решает и алгоритм S, но путём прямого перебора экспоненциального количества вариантов. Пусть над входной цепочкой длины n для достижения результата каждая из программ T_i и S выполняет $f_i(n)$ и $F(n)$ действий соответственно. Если на входе x длины n требуется применить подпрограмму T_r и число n уже таково, что $F(n) > f_0(n) + f_r(n)$, то при расчётах на реальных ЭВМ это не обязательно означает, что алгоритм T, применённый к x, быстрее закончит вычисления, чем это сделает S.

Чтобы разобраться в ситуации, будем считать T и S программами машины Тьюринга. Как сейчас общепринято, мы отождествили *время работы* машин с *количеством действий на ленте*, т.е. с «внешней» сложностью, и проигнорировали те действия, которые нужны для перехода от подпрограммы T_0 к T_r — «внутреннюю» сложность исполнения T. Однако при громоздких программах T_i более реально было бы учесть и время перехода к подпрограмме T_r. Почти те же самые эффекты наблюдаются и в реальной вычислительной машине.

Для более адекватного описания сложности выполнения программ нужно ввести понятие *комбинированной* (или *агрегированной*) *меры сложности алгоритма*, которая учитывала бы и «внешнюю» и «внутреннюю» его сложность. Или точнее: нужно учитывать не только число шагов на ленте, но также и время поиска в программе очередной применимой команды, которое определяется *сложностью описания* (строения) всей программы, тогда многое станет на своё место. Однако если брать при этом в расчёт только длину программы, то этого будет явно не достаточно. Хотя сложность описания программы машины Тьюринга самой по себе, без учёта её действий на ленте, может быть, пожалуй, удовлетворительно охарактеризована *гёделевским* номером этой программы. Но, разумеется, при выполнении некоторых естественных требований на нумерацию. Например, гёделевский номер программы должен монотонно зависеть от числа её команд, количества внутренних состояний машины, мощности рабочего алфавита и многих других значимых параметров программы. Немаловажным обстоятельством является и величина номера: традиционная нумерация последовательностей, основанная на разложении натуральных чисел по степеням простых, иногда приводит к очень большим номерам для коротких последовательностей. Более эффективной, а значит и подходящей, может быть нумерация, основанная на канторовском перечислении пар и разбиении длинной последовательности на пару «начало–конец».

Два варианта возможного исправления недостатков традиционного способа подсчёта сложности сделаны в [27]. В первом из них предлагается совсем не учитывать выполнение «скользящих» команд, которые не меняют ни внутреннее состояние машины, ни запись на ленте, т.е. команд вида $q_i\alpha \to q_iR$ или $q_i\alpha \to q_iL$. Во втором — сложность выполнения «скользящей» команды предлагается считать равной 2^{-t+1}, где t — это номер выполнения данной команды, когда она исполняется несколько раз подряд. Например, если такая команда выполнилась однажды 5 раз подряд, а во второй — трижды, то все восемь её исполнений дадут прибавку к сложности $\left(2^0+2^{-1}+2^{-2}+2^{-3}+2^{-4}\right)$ + $+\left(2^0+2^{-1}+2^{-2}\right)$, а не восемь. Действительно, пройтись по массиву из двух десятков единиц, выпол-

няя $q_i 1 \to q_i R$, вроде бы заметно легче, чем написать их десяток. Недостаток обоих этих вариантов — тот, что и при очень длинном массиве единиц сложность всех исполнений «скользящей» команды останется маленькой, что опять противоречит интуиции.

7. Особые точки гиперповерхностей. Гиперповерхность, заданная формой f над полем характеристики нуль, *особая*, если совокупность частных производных первого порядка этой формы имеет нетривиальный нуль. Гладкость проективной гиперповерхности, заданной формой f степени d от $(n+1)$ переменной, выражается неравенством нулю дискриминанта, т.е. формы D от коэффициентов f [28]. Степень дискриминанта равна $(n+1)\,(d-1)^n$. При d=1 дискриминант равен ненулевой константе; при $d=2$ дискриминант пропорционален определителю матрицы квадратичной формы. Для форм f от двух переменных (при n=1), получаемых гомогенизацией многочленов $g(x)$ степени d от одной переменной, дискриминант D имеет степень $2d-2$ и совпадает с широко используемым дискриминантом многочлена g [29; 34]. Высокая степень дискриминанта иллюстрирует высокую вычислительную сложность элиминации кванторов в теории поля комплексных чисел.

Напомним геометрическую интерпретацию дискриминанта. Особые гиперповерхности в проективном пространстве соответствуют касательным гиперплоскостям к многообразию Веронезе. В свою очередь касательные гиперплоскости соответствуют точкам двойственного многообразия [30; 245–249]. Дискриминант определяет гиперповерхность в двойственном проективном пространстве, двойственную к многообразию Веронезе.

Назовем (-1, 1)-точкой всякую точку в проективном пространстве, чьи однородные координаты равны -1 или 1, с точностью до общего ненулевого множителя. Это вершины многомерного куба. Проверка принадлежности некоторой (-1, 1)-точки к данной гиперплоскости является NP-полной задачей. Подходы к ее решению, основанные на теореме Гильберта Nullstellensatz, обсуждаются в [31]. Отметим, что соответствующая задача оптимизации может быть решена псевдополиномиальным алгоритмом, основанным на методе динамического программирования [32]. Этот метод усовершенствован в работе [33].

Покажем, что задача о распознавании гиперплоскости, на которой не лежит никакая вершина многомерного куба, сводится к проверке гладкости комплексной проективной гиперповерхности нечётной степени, начиная с третьей. Это говорит о вычислительной трудности проверки гладкости таких гиперповерхностей, хотя для квадрики гладкость проверяется легко. С другой стороны, это может быть полезно для анализа близких комбинаторных задач [34], поскольку взаимное расположение особых точек связано некоторыми ограничениями.

Далее рассматриваются проективные гиперповерхности, которые заданы формами с коэффициентами из конечного алгебраического расширения поля рациональных чисел. Элементы конечного алгебраического расширения можно отождествить с многочленами ограниченной степени над полем рациональных чисел. Арифметические операции над этим полем сводятся к операциям над целыми числами, размер которых ограничен полиномом от длин записей исходных алгебраических чисел [10; 56–65; 35].

Теорема 2. Для любого нечётного числа d, начиная с трёх, существует детерминированный алгоритм, который получает на вход гиперплоскость H, заданную линейной формой $h = a_0 x_0 + \ldots + a_n x_n$, где n не меньше трёх, и за полиномиальное время выдает такую гиперповерхность S степени d, что особые (-1, 1)-точки на S взаимно однозначно соответствуют (-1, 1)-точкам, лежащим на H.

Доказательство. Сопоставим линейной форме h форму нечётной степени d следующим образом: $f = a_0 x_0^d + \ldots + a_n x_n^d$. Выходом алгоритма служит ограничение формы f на гиперплоскость H, которое определяет гиперповерхность S в пространстве H. Особыми точками на S служат точки касания гиперплоскости H с гиперповерхностью, заданной формой f.

Если в некоторой (-1, 1)-точке x обе формы h и f обращаются в нуль, то проективные гиперповерхности, заданные этими формами, касаются друг друга в x. Следовательно, S имеет особую точку x.

Градиенты форм ∇h и ∇f — коллинеарные в точках с координатами, удовлетворяющими равенствам $x_k^{d-1} = x_j^{d-1}$ для всех тех индексов k и j, для которых коэффициенты a_k и a_j не равны ну-

лю. Если все коэффициенты не равны нулю, то такие точки с вещественными координатами являются (-1, 1)-точками. Здесь существенно используется чётность числа d -1. Если же k-й коэффициент равен нулю, то соответствующая координата особой точки может быть любой. Но в этом случае гиперплоскость H одновременно содержит или не содержит обе (-1, 1)-точки, отличающиеся k-й координатой. Следовательно, особые (-1, 1)-точки на S взаимно однозначно соответствуют (-1, 1)-точкам, лежащим на H. Теорема доказана.

Замечание. Ограничение снизу на число n в условии теоремы связано с тем, что в случае $n = 2$ S содержит конечное число точек.

Следствие. Для любого нечётного числа d, начиная с трёх, множество гиперповерхностей степени d в конечномерных пространствах над конечным алгебраическим расширением поля рациональных чисел, содержащих особую (-1, 1)-точку, NP -полное.

Доказательство. Рассматриваемое множество гиперповерхностей принадлежит классу NP, поскольку, недетерминированно угадав (-1, 1)-точку, за полиномиальное время можно проверить, является ли эта точка особой. Полнота в классе NP следует из предыдущей теоремы. Следствие доказано.

Работа выполнена при частичной поддержке Комитета науки МОН РК (грант 0726/ГФ) и Российского фонда фундаментальных исследований (проект 13-04-40196-Н).

Список литературы

1 *Береснев В.Л., Мельников А.А.* Алгоритм ветвей и границ для задачи конкурентного размещения предприятий с предписанным выбором поставщиков // Дискретный анализ и исследование операций. — 2014. — Т. 21. — № 2. — С. 3–23.

2 *Каренов Р.С.* Методика анализа и оптимизации сетевого графика // Вестн. Караганд. ун-та. Сер. Математика. — 2013. — № 3 (71). — С. 53–65.

3 *Айсагалиев С.А., Айсагалиев Ж.К.* Исследование по математическому программированию // Вестн. КазНУ. Сер. мат., мех., инф. — 2013. — № 2 (77). — С. 4–20.

4 *Ешкеев А.Р.* Йонсоновские множества и их некоторые теоретико-модельные свойства // Вестн. Караганд. ун-та. Сер. Математика. — 2014. — № 2 (74). — С. 53–62.

5 *Ешкеев А.Р., Жолмагамбетова Б.Р.* Позитивно алгебраически простые модели в классе экзистенциально-замкнутых моделей выпуклых позитивных йонсоновских теорий // Вестн. Караганд. ун-та. Сер. Математика. — 2014. — № 2 (74). — С. 63–69.

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

7 *Heintz J.* Definability and fast quantifier elimination in algebraically closed fields // Theoretical Computer Science. — 1983. — Vol. 24. — P. 239–277.

8 *Fürer M.* The complexity of Presburger arithmetic with bounded quantifier alternation depth // Theoretical Computer Science. — 1982. — Vol. 18. — P. 105–111.

9 *Верещагин Н.К.* Новое доказательство разрешимости элементарной теории линейно упорядоченных множеств // Математические заметки. — 1990. — Vol. 47. — № 5. — P. 31–38.

10 *Кокс Д., Литтл Дж., О'Ши Д.* Идеалы, многообразия и алгоритмы. Введение в вычислительные аспекты алгебраической геометрии и коммутативной алгебры. — М.: Мир, 2000. — 687 с.

11 *Bokut L.A., Chen Y.* Gröbner–Shirshov bases and their calculation // Bull. of Mathematical Sciences. — 2014. — [ER]. Access mode: doi: 10.1007/s13373-014-0054-6.

12 *Mayr E.W., Meyer A.R.* The complexity of the word problems for commutative semigroups and polynomial ideals // Advances in Mathematics. — 1982. — Vol. 46. — № 3. — P. 305–329.

13 *Чистов А.Л.* Дважды экспоненциальная нижняя оценка на степень системы образующих полиномиального простого идеала // Алгебра и анализ. — 2008. — Т. 20. — № 6. — С. 186–213.

14 *Mayr E.W., Ritscher S.* Dimension-dependent bounds for Gröbner bases of polynomial ideals // J. of Symbolic Computation. — 2013. — Vol. 49. — P. 78–94.

15 *Чистов А.Л.* Алгоритм полиномиальной сложности для разложения многочленов и нахождение компонент многообразия в субэкспоненциальное время // Записки научных семинаров ЛОМИ. — Ленинград: Наука, 1984. — Т. 137. — С. 124–188.

16 *Chistov A.L.* An improvement of the complexity bound for solving systems of polynomial equations // Notes of scientific seminars PDMI. — New York, 2012. — 181. — 6. — P. 921–924.

17 *Schwartz J.T.* Fast probabilistic algorithms for verification of polynomial identities // J. of the ACM. — 1980. — Vol. 27. — P. 701–717.

18 *Giusti M., Lecerf G., Salvy B.* A Gröbner free alternative for polynomial system solving // J. of Complexity. — 2001. — Vol. 17. — P. 154–211.

19 *Herrero M.I., Jeronimo G., Sabia J.* Affine solution sets of sparse polynomial systems // J. of Symbolic Computation. — 2013. — Vol. 51. — P. 34–54.

20 *Григорьев Д.Ю.* Сложность разрешения теории первого порядка алгебраически замкнутых полей // Изв. АН СССР. Сер. матем. — 1986. — Т. 50. — № 5. — С. 1106–1120.

21 *Wrathall C.* Complete sets and the polynomial-time hierarchy // Theoretical Computer Science. — 1977. — Vol. 3. — P. 23–33.

22 *Babaiy L.* Trading group theory for randomness // Proc. of the 17th ACM Symposium on Theory of Computing (STOC). — 1985. — P. 421–429.

23 *Miasnikov A.G., Romankov V., Ushakov A., Vershik A.* The word and geodesic problems in free solvable groups // Transactions of the American Mathematical Society. — 2010. — Vol. 362. — № 9. — P. 4655–4682.

24 *Ushakov A.* Algorithmic theory of free solvable groups: Randomized computations // J. of Algebra. — 2014. — Vol. 407. — P. 178–200.

25 *Deiyneko V.G., Klinz B., Weginger G.J.* Uniqueness in quadratic and hyperbolic 0–1 programming problems // Operations Research Letters. — 2013. — Vol. 41. — P. 633–635.

26 *Найденко В.Г.* О сложности нахождения второго решения NP-полной задачи // Весці Нацыянальнай Акадэміі Навук Беларусі. Серыя фізіка-матэматычных навук. — 2012. — № 2. — С. 114–118.

27 *Латкин И.В., Латкина Л.П.* Разности Σ_2^0-множеств и практические алгоритмы // Материалы II Междунар. науч.-практ. конф. «Тенденции и перспективы развития современного научного знания». — М.: Ин-т стратегических исследований, 2012. — С. 44–56.

28 *Морозов А.Ю., Шакиров Ш.Р.* Новые и старые результаты в теории результатов // Теоретическая и математическая физика. — 2010. — Т. 163. — № 2. — С. 222–257.

29 *Прасолов В.В.* Многочлены. — М.: МЦНМО, 1999.

30 *Харрис Дж.* Алгебраическая геометрия. Начальный курс. — М.: МЦНМО, 2005. — 400 с. / Пер.: Harris J. Algebraic geometry. A first course. — New York: Springer-Verlag, 1992.

31 *Margulies S., Onn S., Pasechnik D.V.* On the complexity of Hilbert refutations for partition // J. Symb. Comput. — 2015. — Vol. 66. — P. 70–83.

32 *Papadimitriou C.H.* On the complexity of integer programming // J. ACM. — 1981. — Vol. 28. — № 4. — P. 765–768.

33 *Tamir A.* New pseudopolynomial complexity bounds for the bounded and other integer Knapsack related problems // Oper. Res. Lett. — 2009. — Vol. 37. — № 5. — P. 303–306.

34 *Горбунов К.Ю., Селиверстов А.В., Любецкий В.А.* Взаимное расположение параллельных гиперплоскостей, квадрик и вершин многомерного куба // Проблемы передачи информации. — 2012. — Т. 48. — № 2. — С. 113–120.

35 *Ноден П., Китте К.* Алгебраическая алгоритмика, с упражнениями и решениями. — М.: Мир, 1999. — С. 203, 204.

И.В.Латкин, А.В.Селиверстов

Кешен сандар өрісі теориясының есептелімділік күрделілік тұстары

Мақалада кванторлар өзгерісінің санына шектелген ескертілген түрдегі формулалардың есептелімділік күрделілі талданды. Дербес жағдайда алгебралық тұйық өрістер теориясымен жұмыс жасалды. Гипержазықтықтағы көпөлшемді кубтың төбелерін тану полиномиалдық уақытта құрылған гиперкеңістіктегі ерекше нүктелерді тануға келтірілді. Сондай-ақ күрделілік кластары арасындағы кейбір қатынастар дәлелденген. Авторлармен күрделіліктің жақсартылған тұжырымдамасы және модельдер теориясымен байланысы ұсынылған.

I.V.Latkin, A.V.Seliverstov

Computational complexity of fragments of the theory of complex numbers

We discuss the computational complexity of formulas in prenex form with bounded quantifier alternation depth. In particular, we have to do with the theory of algebraically closed fields. It is proved that recognition of multidimensional cube vertices on the hyperplane can be reduced to recognition of singular points on a hypersurface constructed in polynomial time. Moreover, some relations between complexity classes are proved. Finally, we discuss how one can improve a concept of complexity as well as connection with the model theory.

Referenses

1 Beresnev V.L., Melnikov A.A. *J. of Applied and Industrial Mathematics*, 2014, 8, 2, p. 177–189.

2 Karenov R.S. *Bull. of University of Karaganda. Ser.Mathematics*, 2013, 3 (71), p. 53–65.

3 Aisagaliyev S.A., Aisagaliyev Zh.K. *KazNU Bull. Mathematics, Mechanics and Computer Sciences*, 2013, 2 (77), p. 4–20 (in Russian).

4 Yeshkeyev A.R. *Bull. of University of Karaganda. Ser. Mathematics*, 2014, 2 (74), p. 53–62 (in Russian).

5 Yeshkeyev A.R., Zholmagambetova B.R. *Bull. of University of Karaganda. Ser. Mathematics*, 2014, 2 (74), p. 63–69 (in Russian).

6 Garey M.R., Johnson D.S. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman & Co. New York, 1979.

7 Heintz J. *Theoretical Computer Science*, 1983, 24, p. 239–277.

8 Fürer M. *Theoretical Computer Science*, 1982, 18, p. 105–111.

9 Vereshchagin N.K. *Mathematical Notes*, 1990, 47, 5, p. 444–449.

10 Cox D., Little J., O'Shea D. Ideals, Varieties, and Algorithms. *An Introduction to Computational Algebraic Geometry and Commutative Algebra*, Moscow: Mir, 2000, 687 p.

11 Bokut L.A., Chen Y. *Bull. of Mathematical Sciences*, 2014. doi: 10.1007/s13373-014-0054-6.

12 Mayr E.W., Meyer A.R. *Advances in Mathematics*, 1982, 46, 3, p. 305–329.

13 Chistov A.L. *St. Petersburg Mathematical J*., 2009, 20, 6, p. 983–1001.

14 Mayr E.W., Ritscher S. *J. of Symbolic Computation*, 2013, 49, p. 78–94.

15 Chistov A.L. *Computational complexity theory. Part II, Zap. Nauchn. Sem. LOMI, 137*, Otdel., Leningrad, Leningrad: Nauka, 1984, p. 124–188.

16 Chistov A.L. *J. of Mathematical Sciences*, New York, 2012, 181, 6, p. 921–924.

17 Schwartz J.T. *J. of the ACM*, 1980, 27, p. 701–717.

18 Giusti M., Lecerf G., Salvy B. *J. of Complexity*, 2001, 17, p. 154–211.

19 Herrero M.I., Jeronimo G., Sabia J. *J. of Symbolic Computation*, 2013, 51, p. 34–54.

20 Grigor'yev D.Yu. *Mathematics of the USSR-Izvestiya*, 1987, 29, 2, p. 459–475.

21 Wrathall C. *Theoretical Computer Science*, 1977, 3, p. 23–33.

22 Babaiy L. *Proc. of the 17th ACM Symposium on Theory of Computing (STOC)*, 1985, p. 421–429.

23 Miasnikov A.G., Romankov V., Ushakov A., Vershik A. *Transactions of the American Mathematical Society*, 2010, 362, 9, p. 4655–4682.

24 Ushakov A. *J. of Algebra*, 2014, 407, p. 178–200.

25 Deineko V.G., Klinz B., Weginger G.J. *Operations Research Letters*, 2013, 41, p. 633–635.

26 Naidenko V.G. *Proc. of the National Academy of Sciences of Belarus. Series of Physikal-Mathematical sciences*, 2012, 2, p. 114–118.

27 Latkin I.V., Latkina L.P. *Transactions of II International theoretical and practical conference «The tendencies and prospects of the modern scientific knowledge»*, Moscow: Russian Institute for Strategic Studies, 2012, p. 44–56.

28 Morozov A.Yu., Shakirov Sh.R. *Theoretical and Mathematical Physics*, 2010, 163, 2, p. 587–617.

29 Prasolov V.V. *Polynomials*, Moscow: MTsNMO, 1999, 302 p.

30 Harris Dzh. *Algebraic geometry. A first course*, Moscow: MTsNMO, 2005; New York: Springer-Verlag, 1992.

31 Margulies S., Onn S., Pasechnik D.V. *J. Symb. Comput*., 2015, 66, p. 70–83.

32 Papadimitriou C.H. *J. ACM*, 1981, 28, 4, p. 765–768.

33 Tamir A. *Oper. Res. Lett*., 2009, 37, 5, p. 303–306.

34 Gorbunov K.Yu., Seliverstov A.V., Lyubetsky V.A. *Problems of Information Transmission*, 2012, 48, 2, p. 185–192.

35 Naudin P., Quitté C. *Algorithmique algébrique, avec exercices corrigés*, Paris: Masson, 1992, p. 203, 204.