МОСКОВСКИЙ ОРДЕНА ЛЕНИНА, ОРДЕНА ОКТЯБРЬСКОЙ РЕВОЛЮЦИИ И ОРДЕНА ТРУДОВОГО КРАСНОГО ЗНАМЕНИ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ им. М.В. ЛОМОНОСОВА

Механико-математический факультет

А.В. Михалев, Е.В. Панкратьев

КОМПЬЮТЕРНАЯ АЛГЕБРА

ВЫЧИСЛЕНИЯ В ДИФФЕРЕНЦИАЛЬНОЙ И РАЗНОСТНОЙ АЛГЕБРЕ

Издательство
Московского университета
1989

ББК 22.14
М 69
УДК 512.622

Р е ц е н з е н т ы :

Доктор физ.-мат. наук В.Н.Латышев
Доктор физ.-мат. наук Г.М.Кобельков

Печатается по постановлению
Редакционно-издательского совета
Московского университета

Михалев А.В., Панкратьев Е.В.
Компьютерная алгебра. Вычисления в дифференциальной и разностной алгебре: Учебное пособие. – М.: Изд-во МГУ, 1989. – 97 с.
ISBN 5-211-01431-6.
М 69

Вторая книга из серии учебных пособий по курсу "Компьютерная алгебра". Рассматривается одна из основных задач компьютерной алгебры – задача представления данных. Основное внимание уделяется представлениям полиномиальных, дифференциальных и разностных модулей. Соответствующие методы получили название теории базисов Гребнера. Сформулировано несколько эквивалентных определений базисов Гребнера, приведены алгоритмы их вычисления, некоторые приложения, среди которых – вычисление характеристических многочленов Гильберта.

Для студентов механико-математического факультета МГУ.

ББК 22.14

077(02)-89-заказное
ISBN 5-211-01431-6

Введение

Компьютерная алгебра – современная область математики, находящаяся на стыке алгебры и компьютерной математики. Представление о рассматриваемых ею задачах и используемых методах можно получить из книг [4 и 22].

Настоящее пособие является составной частью серии учебных пособий по компьютерной алгебре (см. [9, 16]), подготовленных на основе спецкурсов, читаемых на механико–математическом факультете МГУ. Оно посвящено конструктивным методам в теории дифференциальных и разностных колец и модулей и их приложениям. Излагаемые результаты находятся на стыке дифференциальной и разностной алгебры с одной стороны и компьютерной алгебры с другой. Обзор литературы по дифференциальной и разностной алгебре содержится в [15].

В 1952 г. А.Эйнштейн в работе [17] ввел понятие "жесткости" системы уравнений поля следующим образом: "... систему уравнений поля следует выбирать так, чтобы полевые величины определялись этой системой как можно более жестким образом. Чтобы применять этот принцип, нам нужен метод, который позволял бы дать меру жесткости системы уравнений. Поступим следующим образом: разложим переменные поля вблизи точки P в ряд Тейлора (предполагается аналитический характер поля). Коэффициенты разложения, которые представляют собой не что иное, как производные элементов поля в точке P, распадаются на группы соответственно порядку дифференцирования. В каждом порядке дифференцирования мы на первых порах получаем набор коэффициентов, которые можно бы было выбрать произвольно, если бы поле не должно было удовлетворять системе уравнений. Благодаря наличию системы дифференциальных уравнений (и уравнений, получаемых из них путем дифференцирования по координатам) число независимых коэффициентов уменьшается, так что в каждой группе уже меньшее число коэффициентов может быть выбрано произвольно. Количество "свободных" коэффициентов в каждой группе непосредственно дает меру "слабости" системы уравнений и, таким образом, определяет и "жесткость" системы". Далее в этой же статье А.Эйнштейн при-

водит несколько примеров вычисления "жесткости" системы уравнений. Вычисления проводятся с помощью эвристических методов, без строгого математического обоснования. В одном из примеров допущена ошибка, исправленная Эйнштейном в следующей работе.

Соответствующие математические определения были сформулированы в рамках дифференциальной алгебры Э.Колчиным в 1964 г. [26]. Центральным понятием в построенной теории является понятие дифференциального размерностного многочлена – математический объект, соответствующий "жесткости" системы по Эйнштейну. В докладе на Московском математическом конгрессе [27] Колчин наметил программу работ в области дифференциальной алгебры, в которой изучение дифференциальных размерностных многочленов заняло заметное место. Колчин разработал метод вычисления дифференциального размерностного многочлена, опирающийся на теорию характеристических множеств простых дифференциальных идеалов. Основы теории характеристических множеств в дифференциальной алгебре были заложены ранее основателем дифференциальной алгебры Дж. Риттом [36]. Джонсон [25] показал, что дифференциальный размерностный многочлен для расширения дифференциального поля совпадает с характеристическим многочленом Гильберта для градуированного модуля над кольцом многочленов, ассоциированного с модулем кэлеровых дифференциалов этого расширения. Техника характеристических множеств переносится на дифференциальные модули и используется для нахождения "хорошей" системы образующих модуля дифференциалов.

Важную роль в этой теории играют дифференциальные модули, систематическое изучение которых было начато в работе Ю.И.Манина [32]. Современное состояние этого направления отражено в обзоре авторов [15].

По исторически сложившейся традиции для большинства результатов дифференциальной алгебры через некоторое время находится разностный аналог. Так получилось и с теорией дифференциальной размерности. Разностные аналоги характеристических множеств, разностные размерностные многочлены и методы их вычисления получены в работах [1, 11]. Многие результаты могут быть сформулированы и в более общем виде для дифференциально-разнос-

тных полей и модулей (см., например, [3]).

С другой стороны, использование ЭВМ в алгебранческих вычислениях, в частности, в коммутативной алгебре, привело к развитию конструктивных методов в теории полиномиальных идеалов. Основы алгоритмических методов в теории полиномиальных идеалов заложены в работе Херман [23]. Существенную роль в развитии этих методов сыграла статья Хиронаки [24] и работы Б. Бухбергера, который в 1965 г. [19] ввел понятие базиса Гребнера полиномиального идеала и предложил алгоритм его вычисления. Начиная с 1976 года, методы базиса Гребнера были уточнены, обобщены и проанализированы во многих работах. Современное состояние теории базисов Гребнера освещено, например, в работах [2, 9, 22 и 33]. Оказалось, что существует тесная связь между теорией характеристических множеств в дифференциальной алгебре и теорией базисов Гребнера в коммутативной алгебре. Методы, основанные на вычислении базисов Гребнера и их применении, используются в большинстве систем компьютерной алгебры.

Разработка алгоритмов и исследование размерностных свойств систем дифференциальных уравнений с использованием ЭВМ была начата авторами в [13]. Результаты исследований, выполненных за прошедшее десятилетие на механико-математическом факультете (см. [5, 7, 8, 10, 14, 30, 35]), легли в основу данного курса.

Пособие начинается с рассмотрения вопросов представления данных для основных алгебранческих областей: кольца целых чисел, поля рациональных чисел, конечных полей, кольца многочленов и поля рациональных функций, алгебранческих и трансцендентных расширений полей. Рассмотрение задачи представления данных для фактор-колец кольца многочленов и для полиномиальных модулей приводит к введению понятия базисов Гребнера полиномиальных идеалов, которое вводится во втором параграфе. Основные результаты излагаются в параграфе 3, в котором вводятся определения дифференциальных и разностных модулей, понятие базисов Гребнера обобщается на случай подмодулей свободных дифференциальных или разностных модулей, формулируется и доказывается теорема, дающая различные характеризации базисов Гребнера, вводится понятие минимальных базисов Гребнера и обсуждается их связь с поня-

тием характеристических множеств, используемых в дифференциальной алгебре. В четвертом параграфе эти результаты формулируются в виде алгоритмов в абстрактной математической формулировке. В качестве приложений построенной теории в параграфе 5 рассматривается теория алгебраической и дифференциальной размерности (характеристические многочлены Гильберта и дифференциальные размерностные многочлены). В заключительном, шестом параграфе описывается специализированная система компьютерной алгебры для вычислений в дифференциальных и разностных модулях. Алгоритмы, сформулированные в параграфе 4, излагаются в виде, ориентированном на конкретную реализацию, приводятся примеры вычислений.

1. ПРОБЛЕМА ПРЕДСТАВЛЕНИЯ ДАННЫХ

Разнообразие структур данных, используемых в компьютерной алгебре, выдвигает на первый план задачу представления данных в ЭВМ. При аналитических вычислениях (вручную или с использованием ЭВМ) используются элементы таких множеств, как кольцо целых чисел, поле рациональных чисел, кольцо вычетов по некоторому модулю, кольца многочленов, различные элементарные функции. Объекты этих множеств допускают неоднозначную запись в виде алгебраических выражений, т.е. представление в памяти ЭВМ. Из различных вариантов представления нужно, по возможности, выбрать оптимальное относительно некоторых критериев. Какими же критериями руководствуются математики при выборе представления конкретных элементов?

Наиболее существенным является требование о том, чтобы выбор представления был каноническим, т.е. в множестве всех эквивалентных выражений нужно выбрать единственное выражение, которое представляло бы этот класс эквивалентности. При этом предполагается, что известен алгоритм проверки эквивалентности двух выражений. В действительности такой алгоритм имеется не всегда. Это является одной из причин, почему иногда на представления налагаются более слабые требования, чем то, что они канонические.

Одним из таких условий является условие нормальности.

Как правило, рассматриваемая структура данных снабжена некоторым набором арифметических операций, часть из которых определена не для всех значений аргументов. В частности, недопустимо деление на нуль. Тем самым нуль приобретает некоторое особое положение, и возрастает значение задачи определения равенства элемента нулю. Представление, в котором все эквивалентные нулю выражения представляются одним и тем же образом (0), называется нормальным. Любое каноническое представление является нормальным, но обратное верно не всегда. Однако во многих случаях наличие нормального представления позволяет построить каноническое. Если же рассматриваемая структура данных такова, что в ней имеются кроме нуля и другие "особые" элементы, то определение

нормального представления должно быть усложнено.

Ключевое для канонического представления понятие эквивалентности объектов может быть самым различным. Могут использоваться различные определения эквивалентности объектов даже на одном и том же множестве. Например, на множестве многочленов с коэффициентами из конечного поля можно рассматривать отношение функционального равенства, а можно рассматривать отношение эквивалентности между многочленами, рассматриваемыми как элементы кольца многочленов с коэффициентами из заданного поля.

Другим требованием, предъявляемым к выбору представления, является требование естественности. Что понимается под этим требованием? Давайте рассмотрим пример неестественного представления, основанного на методе Брауна. Предположим, что мы умеем определять, являются ли два выражения эквивалентными, и что наша ЭВМ обладает неограниченной памятью. В процессе появления выражений мы будем сравнивать кажде новое выражение с уже встречавшимися нам, которые хранятся в памяти ЭВМ. Если среди ранее встречавшихся выражений имеется эквивалентное исходному, то исходное выражение переписывается в форме эквивалентного ему выражения, уже хранящегося в памяти ЭВМ, в противном случае его форма объявляется каноническим представителем в данном классе эквивалентности и запоминается. К преимуществам такого метода следует отнести его универсальность - метод работает всегда, когда есть алгоритм проверки эквивалентности двух выражений, недостатком метода является его неестественность, т.е. представление конкретного элемента зависит от того, в какой последовательности элеметов он появляется (и в каком месте). Представление называется естественным, если преставление каждого элемента определяется одними и теми же правилами, не зависящими от того, в какой последовательности появляется этот элемент.

Ниже будут рассмотрены некоторые из основных структур данных, используемых в компьютерной алгебре, и для них рассмотрена проблема представления данных. Хорошо известно, что в общем случае эта проблема неразрешима, т.е существуют отношения эквивалентности, для которых не существует алгоритма выбора канонического представителя в множестве эквивалентных выражений, и

даже проверки эквивалентности двух выражений.

1.1. КОЛЬЦО ЦЕЛЫХ ЧИСЕЛ

Проблема представления целых чисел не составляет особой сложности. Множество десятичных целых чисел обычно описывается следующим образом:

```
целое число == <натуральное число> | 0 | - <натуральное число>
натуральное число == <значащая цифра> |
                     <значащая цифра> <последовательность цифр>
значащая цифра    == 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
последовательность цифр == <цифра>  |
                           <цифра> <последовательность цифр>
цифра             == 0 | <значащая цифра>
```

Выписанное определение целых чисел дает однозначность представления каждого такого числа и аналогичное определение (только, может быть, с другим основанием) используется в большинстве систем компьютерной алгебры. Пользуясь таким представлением, удобно реализовать арифметические операции над целыми числами. При этом сложение и вычитание являются относительно "дешевыми" операциями, а умножение и деление - "дорогими". При оценке сложности арифметических операций следует учитывать как стоимость элементарной операции (одноразрядной), так и количество одноразрядных операций для выполнения какого-либо действия над многозначными числами. Сложность умножения и деления обусловлена, в первую очередь, тем, что с ростом длины числа (его записи в какой-либо системе счисления) количество элементарных операций увеличивается по квадратичному закону, в отличие от линейного для сложения и вычитания. Кроме того, то, что мы обычно называем алгоритмом деления многозначных чисел, в действительности основано на переборе (часто весьма значительном) возможной очередной цифры частного, при этом недостаточно просто воспользоваться правилами деления однозначных чисел. При большом основании системы счисления (часто оно может иметь по-

рядок 2^{30}) этот способ малоэффективен.

Рассмотренное представление не является единственным каноническим представлением целых чисел. Для выбора канонического представления можно воспользоваться единственностью разложения натурального числа на простые множители. Такое представление целого числа может быть применено в тех задачах, где используются только операции умножения и деления, так как они становятся очень "дешевыми", однако несоизмеримо возрастает стоимость операций сложения и вычитания, что препятствует использованию подобного представления. В некоторых задачах отказ от канонического представления дает значительный выигрыш в быстродействии, в частности, может использоваться частичное разложение числа на множители. Особенно полезен аналогичный метод при работе не с числами, а с многочленами.

Если известно, что при работе программы все встречающиеся в вычислениях целые числа ограничены по абсолютной величине некоторой заданной константой, то можно использовать для задания таких чисел их систему вычетов по модулям некоторых взаимно простых чисел, произведение которых превосходит упомянутую константу. Вычисления с классами вычетов выполняются, как правило, быстрее, чем арифметика многократной точности. А арифметикой многократной точности при таком подходе нужно пользоваться только при вводе или выводе информации.

Отметим, что наряду с каноническими представлениями в системах компьютерной алгебры используются и другие представления. В частности, желательно, чтобы наличие или отсутствие знака '+' перед целым числом не влияло на восприятие его компьютером. Таким образом, для положительных чисел получается неоднозначное представление, хотя форма отрицательных чисел определена однозначно.

```
<положительное целое>    ==  <натуральное число> |
                               + <натуральное число>
<отрицательное целое>    == - <натуральное число>
```

Другое требование - на восприятие числа не должно влиять

наличие нулей перед первой значащей цифрой.

1.1.1. Упражнения

а) Оценить количество одноразрядных умножений, требующихся для умножения m-значного числа на n-значное.

б) Показать, что два двузначных числа можно перемножить, используя только 3 умножения однозначных чисел.

в) Найти алгоритм деления длинных чисел, не требующий большого перебора при нахождении первой цифры частного.

1.2. РАЦИОНАЛЬНЫЕ ЧИСЛА

Множество рациональных чисел $\mathbb{Q}$ определяется как фактор-множество множества пар $(a,b) \in \mathbb{Z}\times\mathbb{Z}$, $b\neq 0$, по отношению эквивалентности: $(a,b) \simeq (c,d) \Leftrightarrow ad - bc = 0$. Если у нас фиксирована каноническая форма целого числа, то каноническую форму рационального числа мы можем получить, например, выбирая из эквивалентных пар целых чисел (a,b) такую, у которой $b>0$ и НОД(a,b)

1. Все сказанное выше о представлении целых чисел относится и к представлению рациональных чисел.

1.3. ПРИБЛИЖЕННЫЕ ВЫЧИСЛЕНИЯ

Хотя выше и отмечалось, что в компьютерной алгебре вычисления обычно производятся точно, без округления, тем не менее в ней рассматриваются и задачи, требующие приближенного решения (например, нахождение вещественных корней многочлена). В отличие от численного анализа ответ в таких задачах представляется не в виде числа, а в виде интервала на вещественной оси (области в комплексной плоскости). С такими интервалами можно производить арифметические действия, соответствующая арифметика известна под названием интервальной. Как правило, интервальная арифметика комбинируется с арифметикой многократной точности, поскольку требуемая точность обычно весьма высока.

1.3.1. Упражнения

а) Привести пример аксиомы поля вещественных чисел, не вы-

полняющейся при работе с числами типа *REAL* на фортране.

б) Какие аксиомы поля вещественных чисел не выполняются при работе с числами типа *REAL* на фортране?

в) Привести пример аксиомы поля вещественных чисел, не выполняющейся для интервальной арифметики.

г) Какие аксиомы поля вещественных чисел не выполняются для интервальной арифметики?

1.4. АЛГЕБРАИЧЕСКИЕ ЧИСЛА

1.4.1. Определение. Алгебраическим числом называется число α, являющееся корнем многочлена от одной переменной с целыми коэффициентами. Если старший коэффициент этого многочлена равен 1, то алгебраическое число называется целым.

1.4.2. Упражнения

а) Показать, что $\sqrt{2}$, $\sqrt{3}+\sqrt{2}$, $\sqrt{2+\sqrt{5}}$ – целые алгебраические числа.

б) Показать, что целые алгебраические числа образуют кольцо.

в) Показать, что алгебраические числа образуют поле.

1.4.3. Утверждение. Существуют алгебраические числа, не выражающиеся через радикалы.

Доказательство этого утверждения основано на теории Галуа и может быть найдено в учебниках по алгебре.

Таким образом, в поле алгебраических чисел можно выделить подполя алгебраических чисел, порожденных простыми радикалами (простые радикальные расширения), вложенными радикалами (вложенные радикальные расширения) и соответствующие подкольца в кольце целых алгебраических чисел.

Представление алгебраических чисел представляет собой значительно более трудную задачу. Если речь идет об одном алгебраическом числе, то для его задания нужно знать минимальный многочлен, корнем которого является данное число. В большинстве алгебраических задач несущественно различие между различными корнями одного и того же неприводимого многочлена. Однако в за-

дачах, где используются различные метрические свойства, часто приходится для задания алгебраического числа указывать не только соответствующий неприводимый многочлен, но и интервал на вещественной оси или область в комплексном пространстве, содержащую единственный корень указанного многочлена. При этом арифметические операции над алгебраическими числами оказываются очень трудоемкими. Нахождение минимального многочлена для суммы или произведения алгебраических чисел представляет собой нетривиальную задачу, методы его нахождения будут описаны ниже, при изучении базисов Гребнера.

При работе с конкретным полем алгебраических чисел используется представление чисел этого поля, связанное с фиксированием примитивного элемента и однозначностью представления элементов этого поля через фиксированный примитивный элемент. Упомянутые выше сложности возникают при необходимости производить операции над элементами из различных конечных расширений поля рациональных чисел. Эти сложности настолько значительны, что часто отказываются от выбора примитивного элемента и рассматривают поля алгебраических чисел как расширения поля рациональных чисел с многими образующими. В частности, такое представление обычно используется при работе с радикальными расширениями, т.е. с расширениями поля рациональных чисел, получаемыми последовательным присоединением радикалов некоторых элементов (возможно, с вложениями).

1.4.4. Упражнения

а) Найти минимальный многочлен над $\mathbb{Q}$ для $\sqrt{2}+\sqrt{3}$.

б) Построить каноническое представление для поля $\mathbb{Q}(\sqrt{2},\sqrt{3},\sqrt{5})$.

в) Построить алгоритм получения канонического представления для простых радикальных расширений (полей алгебраических чисел, порожденных несколькими радикалами без вложений).

г) Построить алгоритм получения канонического представления для вложенных радикальных расширений.

1.5. ТРАНСЦЕНДЕНТНЫЕ ЧИСЛА

Большинство систем компьютерной алгебры допускает работу с трансцендентными числами e и π, для которых фиксированы соответствующие свойства тригонометрических, логарифмических и показательных функций. Вычисления в трансцендентных расширениях производятся так же, как в полях рациональных функций. Задание конкретного трансцендентного числа какими-либо метрическими или функциональными свойствами и проверка его алгебраической независимости с уже имеющимися величинами представляет собой алгоритмически неразрешимую задачу.

1.6. МНОГОЧЛЕНЫ

Действия с многочленами лежат в основе любой системы компьютерной алгебры. Пусть K – некоторое кольцо, задача представления элементов которого уже решена. Представление элементов кольца многочленов $K[x]$ можно выбирать различными способами. Наиболее распространенным является представление многочлена в виде последовательности коэффициентов, упорядоченной по возрастанию или убыванию степеней одночленов. Представление многочленов, при котором запоминаются все коэффициенты, включая нулевые, называется **плотным**. Плотное представление используется в задачах, где рассматриваемые многочлены имеют сравнительно небольшое количество нулевых коэффициентов. Если степени многочленов достаточно высоки, а количество ненулевых коэффициентов мало (**разреженные** многочлены), то удобнее использовать **разреженное** представление многочленов, в котором указываются только ненулевые коэффициенты и соответствующие степени одночленов. При этом алгоритмы работы с такой формой записи становятся более сложными, но значительно экономится память ЭВМ, а во многих случаях и время работы программы.

Многочлены от многих переменных можно рекурсивно рассматривать как многочлены от одной переменной, но с коэффициентами из кольца многочленов от меньшего числа переменных (рекурсивное представление). А можно на множестве одночленов ввести отноше-

ние порядка и записывать слагаемые в соответствии с выбранным порядком. Наиболее часто используются следующие три отношения порядка:

- лексикографическое упорядочение мономов, получающееся из фиксированного порядка на множестве переменных;
- упорядочение мономов по степеням, а мономы одной и той же степени упорядочиваются лексикографически;
- упорядочение мономов по степеням, а мономы одной и той же степени упорядочиваются в обратном лексикографическом порядке, т.е. при равенстве степеней большим считается вектор с *меньшей* первой координатой, при равенстве первых координат - с *меньшей* второй и т.д. Может показаться, что этот порядок совпадает с предыдущим, для "отраженных" векторов. При $n=2$ это действительно так, однако в общем случае эти два отношения порядка различаются более существенно, что продемонстрировано в примере 1.6.1б, где предполагается, что $x > y > z$.

1.6.1. Примеры

а) Пусть переменные x и y упорядочены так, что $x > y$. Тогда многочлен $(x+y)^2+x+y+1$ с учетом соответствующих порядков записывается в виде:

$x^2+2xy+x+y^2+y+1$ (лексикографический порядок);

$x^2+2xy+y^2+x+y+1$ (по степени, затем лексикографический),

$y^2+2xy+x^2+y+x+1$ (по степени, затем обратный лексикографический).

б) Рассмотрим разложение многочлена $(x+y+z)^3$. Из однородности многочлена следует, что первые два из рассматриваемых порядков для этого многочлена совпадают. Выпишем его представление с использованием второго и третьего порядка

$x^3+3x^2y+3x^2z+3xy^2+6xyz+3xz^2+y^3+3y^2z+3yz^2+z^3$ и

$x^3+3x^2y+3xy^2+y^3+3x^2z+6xyz+3y^2z+3xz^2+3yz^2+z^3$ соответственно.

Можно пользоваться как плотной (когда записываются все коэффициенты от самого старшего до самого младшего или наоборот), так и разреженной (когда записываются только ненулевые коэффициенты и соответствующие степени) формой записи. В отличие от многочленов от одной переменной, для которых используется как разреженное, так и плотное представление, для многочленов от

нескольких переменных плотное представление почти не используется, поскольку уже при сравнительно небольших степенях количество коэффициентов в этих многочленах весьма велико и почти все они нулевые.

Отметим, что для многочленов задача разложения на множители представляет еще большую сложность, чем для целых чисел, и поэтому не всегда представляется целесообразным раскрывать при умножении скобки, часто бывает полезным в ущерб каноничности записи хранить многочлен в виде произведения.

1.7. РАЦИОНАЛЬНЫЕ ФУНКЦИИ

Поле рациональных функций $K(x_1, \ldots, x_n)$, где K – поле, обычно определяется как поле частных кольца многочленов $K[x_1, \ldots, x_n]$. Имея каноническое представление элементов кольца многочленов, каноническое представление рациональных функций можно получить наложением условия взаимной простоты числителя и знаменателя и нормировкой знаменателя (приравниванием старшего коэффициента знаменателя единице). Часто, однако, поле K само представляется как поле частных некоторой области целостности, например, поле $\mathbb{Q}(x)$ можно представить как поле частных кольца $\mathbb{Q}[x]$, а также как поле частных кольца $\mathbb{Z}[x]$. При представлении поля $\mathbb{Q}[x]$ как поля частных кольца $\mathbb{Z}[x]$ далеко не всегда можно в представлении рациональной функции приравнять старший коэффициент знаменателя единице. Множество обратимых элементов в $\mathbb{Z}$ состоит всего из двух элементов (1 и -1) и нормировку знаменателя можно осуществить, фиксировав знак старшего коэффициента.

1.8. ОБОБЩЕННЫЕ МНОГОЧЛЕНЫ И РАЦИОНАЛЬНЫЕ ФУНКЦИИ

В дифференциальной алгебре имеется ряд объектов, которые можно рассматривать как обобщение кольца многочленов. К ним относятся алгебры Вейля, кольца дифференциальных операторов и кольца дифференциальных многочленов. Подробное описание колец дифференциальных операторов и колец дифференциальных многочленов будет дано ниже, здесь отметим только, что элементы алгебры

Вейля или кольца дифференциальных операторов могут быть представлены в виде суммы одночленов от фиксированного конечного множества переменных с коэффициентами из фиксированного поля, т.е. так же, как и элементы кольца многочленов (допускается как плотная, так и разреженная запись). Сложности вычислений в таких кольцах связаны с некоммутативностью умножения. Некоммутативность умножения особенно сказывается при рассмотрении тел частных для этих колец (существование их доказывается в курсах по теории колец): правое частное двух элементов может не совпадать с их левым частным, правые множители отличаются от левых множителей, приходится рассматривать правые и левые наибольшие общие делители и наименьшие общие кратные, которые для взаимно простых элементов не совпадают с их произведениями.

В этом параграфе приводятся основные определения и результаты из дифференциальной и разностной алгебры, которые понадобятся нам в дальнейшем.

1.8.1. Определение. Оператор δ, действующий на некотором кольце, называется оператором дифференцирования, если $\delta(a+b) = \delta a + \delta b$ и $\delta(ab) = (\delta a)b + a\delta b$ для всех элементов a,b этого кольца. Дифференциальным кольцом называется коммутативное кольцо R с конечным множеством $\Delta = (\delta_1, \ldots, \delta_n)$ операторов дифференцирования кольца R таких, что $\delta\delta' a = \delta'\delta a$ ($a \in R$, $\delta \in \Delta$, $\delta' \in \Delta$); если $n=1$, то дифференциальное кольцо R называется обыкновенным, если $n > 1$, то R называется кольцом с частными производными. Если кольцо R является полем, то мы говорим о дифференциальном поле.

1.8.2. Примеры и упражнения

1.8.2.1. Любое кольцо можно рассматривать как дифференциальное кольцо с нулевым дифференцированием.

1.8.2.2. Кольцо многочленов $R[x]$ от одной переменной над обыкновенным дифференциальным кольцом можно превратить в обыкновенное дифференциальное кольцо, произвольным образом задав значение $\delta(x)$. Показать, что значением $\delta(x)$ продолжение дифференцирования δ на кольцо $R[x]$ определяется однозначно. Анало-

гичное утверждение верно для кольца формальных степенных рядов.

1.8.2.3. Пусть R – дифференциальное кольцо без делителей нуля, F – его поле частных. Показать, что F можно единственным образом превратить в дифференциальное поле, содержащее дифференциальное кольцо R.

1.8.2.4. Поле мероморфных в некоторой области вещественного n-мерного пространства функций можно рассматривать как дифференциальное поле, с множеством дифференцирований $\Delta = \{\partial/\partial x_1\}$.

1.8.2.5. Важным примером дифференциального кольца является кольцо дифференциальных многочленов, которое будет определено ниже.

1.8.3. Определение. Пусть R – дифференциальное кольцо с множеством операторов дифференцирования Δ. Под дифференциальным модулем над R или дифференциальным R-модулем мы понимаем R-модуль M, на котором действуют операторы множества Δ в соответствии со следующими правилами:

$$\delta(u + v) = \delta u + \delta v, \quad \delta(au) = (\delta a)u + a\delta u$$
$$(\delta \in \Delta,\ u \in M,\ v \in M,\ a \in R).$$

Дифференциальный модуль можно рассматривать как левый модуль над кольцом косых многочленов $R[\Delta] = R[\delta_1, \ldots, \delta_n]$ дифференциального типа. Это кольцо мы будем называть **кольцом линейных дифференциальных операторов.** Пусть T обозначает свободную коммутативную полугруппу (записанную мультипликативно), порожденную элементами из множества Δ. Каждый элемент кольца $R[\Delta]$ может быть единственным способом выражен в виде конечной суммы

$$\sum_{\theta \in T} a_\theta \theta = \sum_{i_1, \ldots, i_n} a_{i_1 \ldots i_n} \delta^{i_1} \delta_2^{i_2} \ldots \delta_n^{i_n},$$

умножение образующих определяется правилами:

$\delta_i \delta_j = \delta_j \delta_i$, $\quad \delta_i a = a\,\delta_i + \delta_i(a)$, где $\quad \delta_i \in \Delta$, $a \in R$,

и на все кольцо $R[\Delta]$ распространяется по линейности.

В предположении, что для кольца коэффициентов R каноническое представление фиксировано, каноническое представление кольца дифференциальных операторов $R[\Delta]$ получается так же, как и

для кольца многочленов достаточно упорядочить полугруппу T. Обычно рассматриваются отношения порядка, перечисленные в пункте 1.6.

Кольцо дифференциальных операторов обладает многими свойствами, аналогичными свойствам кольца многочленов, в частности, в нем нет делителей нуля, для любых двух элементов существует наибольший общий делитель (правый или левый, причем в общем случае они не совпадают) и наименьшее общее кратное (правое и левое, снова, возможно, не совпадающие), если F - δ-поле, то в $F[\delta]$ имеется алгоритм Евклида.

Кольцо $R[\Delta]$ иногда называют кольцом дифференциальных многочленов, но мы будем придерживаться терминологии, принятой в монографии [28], где кольцом дифференциальных многочленов и дифференциальным кольцом R называется кольцо $R\{\Delta\} = R\{y_1, \ldots, y_r\}$ многочленов от счетного множества переменных $\{\theta y_j \mid \theta \in \Theta, 1 \le j \le r\}$ над кольцом R. В кольце $R\{\Delta\}$ операторы дифференцирования из множества Δ действуют на коэффициентах по определению дифференциального кольца, а на образующих θy_j по правилу $\delta(\theta y_j) = (\delta\theta) y_j$.

Кольцо дифференциальных многочленов с точки зрения теории колец представляет собой кольцо коммутативных многочленов от счетного множества переменных (каждый конкретный многочлен зависит только от конечного числа переменных). Таким образом, для решения задачи представления данных кольца дифференциальных многочленов и поля рациональных дифференциальных функций достаточно упорядочить кольцевые образующие. Кольцо дифференциальных многочленов является дифференциальным кольцом, т.е. наряду с операциями сложения и умножения в нем имеются унарные операции дифференцирования, переводящие кольцевую образующую в другую кольцевую образующую, отношение порядка на множестве кольцевых образующих выбирается так, чтобы оно было согласовано с дифференцированиями.

Аналогичные объекты - кольца разностных операторов и кольца разностных многочленов - рассматриваются в разностной алгебре. Кольцо разностных операторов отличается от кольца дифференциальных операторов коммутационными соотношениями, все сказан-

ное выше о кольце дифференциальных операторов можно повторить и для кольца разностных операторов. В кольце разностных многочленов вместо операторов дифференцирования рассматриваются операторы трансляции, которые также переводят кольцевые образующие друг в друга, но на поле коэффициентов являются не дифференцированиями, а автоморфизмами (соответственно, для них можно рассматривать отрицательные степени).

1.8.4. Определение. Разностное кольцо определяется как кольцо R с фиксированным конечным множеством $\Delta = \{\tau_1, \ldots, \tau_n\}$ взаимно коммутирующих мономорфизмов кольца R в себя. Если все мономорфизмы τ_i – изоморфизмы, то R называется инверсным разностным кольцом. Если $n = 1$, то R называется обыкновенным разностным кольцом, в противном случае R называется кольцом с частными разностями. Элементы множества Δ называются операторами трансляции.

1.8.5. Примеры и упражнения

1.8.5.1. Любое кольцо можно рассматривать как разностное кольцо с тождественным изоморфизмом.

1.8.5.2. Кольцо многочленов $R[x]$ от одной переменной над обыкновенным разностным кольцом можно превратить в обыкновенное разностное кольцо, произвольным образом задав значение $\tau(x)$. Показать, что значением $\tau(x)$ трансляция кольца $R[x]$ определяется однозначно.

1.8.5.3. Показать, что не любой автоморфизм τ кольца $R[x]$ продолжается на кольцо формальных степенных рядов.

1.8.5.4. Пусть R – разностное кольцо без делителей нуля, F – его поле частных. Показать, что F можно единственным образом превратить в разностное поле, содержащее разностное кольцо R.

1.8.5.5. Поле формальных (сходящихся) рядов Лорана $K((x))$ можно рассматривать как обыкновенное разностное поле с оператором трансляции τ таким, что $\tau(x) = \kappa \cdot x$, где κ – произвольный ненулевой элемент поля K.

1.8.5.6. Важным примером разностного кольца является кольцо разностных многочленов, которое будет определено ниже.

1.8.6. Определение. Пусть R – разностное кольцо с множеством операторов трансляции Δ. Под разностным модулем над R, или разностным R-модулем, мы понимаем R-модуль M, на котором действуют операторы из множества Δ в соответствии со следующими условиями:

$$\tau(u+v) = \tau u + \tau v,\ \tau(au) = (\tau a)\tau u,$$
$$(\tau \in \Delta,\ u \in M,\ v \in M,\ a \in R).$$

Разностный модуль M можно рассматривать как левый модуль над кольцом косых многочленов $R[\Delta] = R[\tau_1, \ldots, \tau_n]$ разностного типа. Это кольцо мы будем называть кольцом разностных операторов. Пусть T – свободная коммутативная полугруппа (записываемая мультипликативно), порожденная элементами множества Δ. Каждый элемент кольца $R[\Delta]$ может быть единственным образом записан в виде конечной суммы

$$\sum_{\vartheta \in T} a_\vartheta \vartheta = \sum_{i_1, \ldots, i_n} a_{i_1 \ldots i_n} \tau_1^{i_1} \tau_2^{i_2} \ldots \tau_n^{i_n},$$

умножение образующих задается соотношениями:

$$\tau_i \tau_j = \tau_j \tau_i,\quad \tau_i a = \tau_i(a)\,\tau_i,\quad \text{где } \tau_i \in \Delta,\ a \in R,$$

и по линейности распространяется на все кольцо $R[\Delta]$. Это кольцо иногда называют кольцом разностных многочленов, но мы будем придерживаться терминологии, принятой в монографии [21], где кольцом разностных многочленов над разностным кольцом R называют разностное кольцо $R\{\Delta\} = R\{y_1, \ldots, y_r\}$ многочленов от счетного множества неизвестных $\{\vartheta y_j : \vartheta \in T,\ 1 \le j \le r\}$ над R. В кольце $R\{\Delta\}$ операторы трансляции из множества Δ действуют на коэффициентах по определению разностного кольца, а на образующих ϑy_j – по правилу: $\delta(\vartheta y_j) = (\delta\vartheta) y_j$.

Кольца дифференциальных и разностных многочленов с точки зрения теории колец представляют собой кольца коммутативных многочленов от счетного множества переменных (каждый конкретный многочлен зависит только от конечного числа переменных). Таким образом, для решения задачи представления данных в кольце диф-

2*–1432

ференциальных (разностных) многочленов и поле рациональных дифференциальных (разностных) функций достаточно упорядочить кольцевые образующие. Кольцо дифференциальных многочленов является дифференциальным кольцом, т.е. наряду с операциями сложения и умножения в нем имеются унарные операции дифференцирования, переводящие кольцевую образующую в другую кольцевую образующую, отношение порядка на множестве кольцевых образующих выбирается так, чтобы оно было согласовано с дифференцированиями. Аналогично, кольцо разностных многочленов является разностным кольцом.

1.9. ВЕКТОРНЫЕ ПРОСТРАНСТВА И МОДУЛИ

В вычислительной математике и в алгебре понятие векторного пространства над некоторым полем K играет ключевую роль. При фиксированном базисе пространства задача представления данных не составляет какой-либо сложности – два вектора совпадают тогда и только тогда, когда совпадают все их координаты в фиксированном базисе. Соответствующая структура данных – "вектор элементов типа K с индексом $1..n$" – является одной из базисных структур данных в программировании. Для случая, когда коэффициенты образуют кольцо R, не являющееся полем, положение существенно сложнее – аналогом понятия векторного пространства (множество, замкнутое относительно сложения, вычитания и умножения на элементы кольца с естественными аксиомами сложения и умножения) является в этом случае понятие модуля. Важным частным случаем модуля является свободный модуль над кольцом R (R-модуль). В частности, любое кольцо с единицей можно рассматривать как свободный модуль над самим собой, любой идеал кольца является его подмодулем. С точки зрения задачи представления данных соответствующая структура данных также может быть при фиксированном базисе описана как "вектор элементов типа R с индексом $1..n$".

С другой стороны, свободный модуль над алгеброй обобщенных многочленов можно рассматривать как бесконечномерное векторное пространство над основным полем. В качестве базиса этого прос-

транства удобно выбрать всевозможные произведения мономов из кольца обобщенных многочленов на модульные образующие. Любой элемент модуля содержится в некотором конечномерном подпространстве, порожденном каким-то модмножеством базисных векторов. Выбор базисных векторов и отношения порядка на множестве базисных векторов определяет каноническую форму любого элемента свободного модуля над кольцом обобщенных многочленов.

К сожалению, множество свободных модулей незамкнуто относительно модульных гомоморфизмов, т.е. как подмодули, так и фактор-модули свободного модуля не обязаны быть свободными модулями.

1.9.1. Упражнения

1.9.1.1. Привести пример идеала в кольце многочленов от двух переменных, не являющегося свободным модулем над этим кольцом.

1.9.1.2. Привести пример фактор-кольца кольца многочленов от одной переменной, не являющегося свободным модулем над этим кольцом.

Основная часть пособия будет посвящена задаче представления данных для полиномиальных, дифференциальных и разностных модулей, важным частным случаем которых являются фактор-кольца кольца многочленов.

1.10. ЗАДАЧИ ДЛЯ РЕАЛИЗАЦИИ НА ЭВМ

1.10.1. Реализовать структуру данных "длинные целые" (Z) со следующей системой команд:

1. Скопировать <вх:/1> в <вых:/2>
2. <вых:/3> := <вх:/1> +|-|* <вх:/2>
3. <вых:/3> := <вх:/1> * 2**<вх:*N*:*integer*>
4. <вых:/3,/4> := частное и остаток от деления <вх:/1> на <вх:/2>
5. <вых:/3,/4> := частное и остаток от деления <вх:/1> на 2**<вх:*N*:*integer*>

6. <вых: *I3*> <вых: *N4*: *integer*> :– частное и остаток от деления <вх: *I1*> на <вх: *N2*: *integer*>
7. <вых: *I3*> := <вх: *I1*> +|–|* <вх: *N2*: *integer*>
8. сравнить <вх: *I1*, *I2*> : > | · | <
9. длина(<вх: *I1*>) : *integer* !память, занимаемая числом !в байтах (или словах)
10. знак(<вх: *I1*>) 1|–1|0
11. битовая_длина(<вх: *I1*>) : *integer**4
12. <вых: *I3*> := НОД(<вх: *I1*>,<вх: *I2*>)
13. <вых: *I2*> := <вх: *N1*: *integer*>
13. <вых: *I2*> := <вх: *S1*: строка>
14. <вых: *S1*: строка> := <вх: *I2*>

1.10.2. На базе структуры данных *Z* реализовать "рациональные числа" (*Q*) с системой команд:
(*integer* и *real* – соответствующие структуры языка программирования)

1. Скопировать <вх: *Q1*> в <вых: *Q2*>
2. <вых: *Q3*> := <вх: *Q1*> + <вх: *Q2*>
3. <вых: *Q3*> := <вх: *Q1*> – <вх: *Q2*>
4. <вых: *Q3*> := <вх: *Q1*> * <вх: *Q2*>
5. <вых: *Q3*> := <вх: *Q1*> / <вх: *Q2*> ! *Q2* ≠ 0
6. сравнить <вх: *Q1*, *Q2*> > | | <
7. знак(<вх: *Q1*>) : 1|–1|0
8. Числитель <вх: *Q1*: *Q*> <вых: *I1*: *Z*>
9. Знаменатель <вх: *Q1*: *Q*> <вых: *I1*: *Z*>
10. <вых: *Q1*: *Q*> := <вх: *I1*: *Z*>
11. <вых: *Q1*: *Q*> := <вх: *I1*: *Z*>/<вх: *I2*: *Z*>
12. Целая часть <вх: *Q1*: *Q*> <вых: *I1*: *Z*>
13. Значение <вх: *Q1*: *Q*> <вых: *R1*: *Z*>

1.10.3. Сформулировать систему команд и реализовать структуру данных "кольцо вычетов" (Z/mZ).

1.10.4. Сформулировать систему команд и реализовать структуру данных "конечное поле" ($GF(q)$) (не обязательно простое).

1.10.5а. Сформулировать систему команд и реализовать структуру

данных "вещественные_числа" (R).

1.10.5б. Сформулировать систему команд и реализовать структуру данных "p-адические_числа" (Q_p).

1.10.6. Реализовать "кольцо_вычетов" (Z/mZ) на основе многомодульной арифметики.

1.10.7. Сформулировать систему команд и реализовать структуру данных "кольцо_многочленов_от_одной_переменной_над_K" ($K[x]$) для $K=Z|Q|GF(q)|Z/mZ|Q_p$.

1.10.8. Сформулировать систему команд и реализовать структуру данных "кольцо_многочленов_от_n_переменных_над_K" ($K[x_1,...,x_n]$) для $K=Z|Q|GF(q)|Z/mZ|Q_p$.

1.10.9. Сформулировать систему команд и реализовать структуру данных "поле_рациональных_функций_от_одной_переменной_над_K" ($K(x)$) для $K=Q|GF(q)|Q_p$.

1.10.10. Сформулировать систему команд и реализовать структуру данных "поле_рациональных_функций_от_n_переменных_над_K" ($K(x_1,...,x_n)$) для $K=Q|GF(q)|Q_p$.

1.10.11а. Сформулировать систему команд и реализовать структуру данных "простые_радикальные_расширения".

1.10.11б. Сформулировать систему команд и реализовать структуру данных "вложенные_радикальные_расширения".

1.10.11в. Сформулировать систему команд и реализовать структуру данных "алгебранческие_числа".

1.10.12. Сформулировать систему команд и реализовать структуру данных "дифференциальные_операторы" ($F[\Delta]$), где F – поле рациональных функций от одной или n переменных над Q.

Составить и реализовать алгоритм нахождения правого (левого) деления с остатком для кольца $F[\delta]$, где F – поле рациональных функций от одной переменной над Q.

Составить и реализовать алгоритм нахождения правого (левого) НОД и НОК для кольца $F[\Delta]$, где F – поле рациональных функций от одной или нескольких переменных с рациональными коэффициентами.

1.10.13. Сформулировать систему команд и реализовать структуру данных "разностные_операторы" ($F[\Delta]$), где F – поле рациональных функций от одной или n переменных над Q.

Составить и реализовать алгоритм нахождения правого (левого) деления с остатком для кольца $R[\tau]$, где R – поле рациональных функций от одной или нескольких переменных с рациональными коэффициентами.

Составить и реализовать алгоритм нахождения правого (левого) НОД и НОК для кольца $R[\Delta]$, где R – поле рациональных функций от одной или нескольких переменных с рациональными коэффициентами.

1.10.14. Сформулировать систему команд и реализовать структуру данных "дифференциальные_многочлены" ($F\{y_1,\ldots,y_m\}$), где F – одно из рассматривавшихся ранее дифференциальных колец.

1.10.15. Сформулировать систему команд и реализовать структуру данных "разностные_многочлены" ($F\{y_1,\ldots,y_m\}$), где F – одно из рассматривавшихся ранее разностных колец.

1.10.16. Сформулировать систему команд и реализовать структуру данных "тригонометрические_функции".

1.10.17. Сформулировать систему команд и реализовать структуру данных "усеченные ряды Тейлора".

1.10.18. Сформулировать систему команд и реализовать структуру данных "кватернионы" над полем Q.

2. КОЛЬЦО РЕГУЛЯРНЫХ НА АЛГЕБРАИЧЕСКОМ МНОГООБРАЗИИ ФУНКЦИЙ

Следующей рассматриваемой задачей является задача выбора канонического представления для элементов кольца регулярных на некотором алгебранческом многообразии функций. Это кольцо представляет собой фактор-кольцо кольца многочленов $R = K[x_1, \ldots, x_n]$ по некоторому идеалу I. Предполагаем, что идеал I задан конечной системой образующих: $I = (f_1, \ldots, f_m)$. Теорема Гильберта о базисе утверждает, что таким образом может быть задан любой идеал кольца многочленов R. Любой элемент фактор-кольца R/I – это смежный класс элементов кольца R относительно идеала I. При фиксированном каноническом представлении элементов кольца R задача о представлении элементов фактор-кольца R/I сводится к задаче выбора канонического представителя в смежном классе. Будем пытаться решить ее в следующей формулировке: в кольце многочленов $R = K[x_1, \ldots, x_n]$ дано конечное множество элементов $f_1, \ldots, f_m$. Требуется построить алгоритм, который для любого многочлена $g \in R$ выбирал бы канонического представителя в соответствующем смежном классе по идеалу I.

Кольцо многочленов R представляет собой бесконечномерное векторное пространство над полем K, базис которого образует счетное множество мономов $T = \{x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n} \mid i_1 \geq 0, \ldots, i_n \geq 0\}$. Идеал I, а следовательно, и фактор-кольцо R/I, также являются векторными K-пространствами. Наша задача состоит в построении отображения i: R/I в R, правого обратного к каноническому гомоморфизму π: R --> R/I, т.е. $\pi \cdot i\,(x) = x$ для любого $x \in R/I$. Таким образом, мы получаем разложение R в прямую сумму векторных пространств I и $i(R/I)$. Задачу выбора канонического представления решает тогда отображение $i \cdot \pi$: R --> R, получающееся проектированием прямой суммы векторных пространств на одно из слагаемых. Достаточно выбрать новый базис кольца R, рассматриваемого как векторное K-пространство, пересечение которого с идеалом I представляет базис векторного пространства I.

2.1. Пример. Пусть идеал I является мономиальным, т.е. по-

рожден мономами $f_1, \ldots, f_m$. Тогда $T \cap I$ является базисом векторного пространства I, а $T \setminus (T \cap I)$ – базисом фактор-кольца R/I, рассматриваемого как векторное пространство. Каноническое представление получается, если в разложении любого многочлена по базису T отбрасывать элементы, принадлежащие I.

Хотя только что рассмотренный пример носит частный характер, он указывает на общий подход к решению поставленной задачи: выбрать такой базис векторного пространства R, пересечение которого с идеалом I представляет собой базис векторного пространства I.

2.2. Утверждение. Пусть M – векторное пространство (возможно, бесконечномерное) и $M' \subseteq M$ – его подпространство. Предположим, что базис Γ векторного пространства M выбран таким образом, что $\Gamma' = \Gamma \cap M'$ представляет собой базис пространства M'. Тогда каноническое представление фактор-пространства M/M' в M получается, если базис пространства M/M' отождествить с $\Gamma'' = \Gamma \setminus \Gamma'$.

Доказательство получается немедленно из разложения векторного пространства M в прямую сумму векторных пространств с базисами Γ' и Γ'', которые изоморфны пространствам M' и M'' соответственно. □

Пусть идеал I порожден многочленами $f_1, \ldots, f_n$. Обозначим $F = \{f_1, \ldots, f_n\}$. Тогда счетное множество многочленов $T \cdot F = \{\theta \cdot f_1 \mid \theta \in T, f_1 \in F\}$ порождает векторное пространство I, однако эти многочлены не являются линейно независимыми. Наша ближайшая задача состоит в построении достаточно простого алгоритма выбора в множестве $T \cdot F$ линейно независимого подмножества. Для этого построим отображение φ: $T \cdot F \dashrightarrow T$ такое, что прообразы различных элементов из T линейно независимы, и выберем в прообразе каждого элемента единственного представителя (если этот прообраз не пуст). Получим систему Σ линейно независимых векторов в идеале I, которая, однако, может не порождать идеал I как векторное пространство.

Следующими задачами являются проверка, порождает ли полу-

чившееся линейно независимое множество векторное пространство I, и если ответ отрицательный, то пополнение его до базиса.

Предположим, что множество T упорядочено таким образом, что:

а) $1 < \theta$ для любого монома θ;

б) если $\theta_1 < \theta_2$, то $\theta_1 \cdot t < \theta_2 \cdot t$ для любого монома t. (2.1)

Как уже сказано в параграфе 1.6, наиболее часто используются следующие три отношения порядка:

- лексикографическое упорядочение мономов, получающееся из фиксированного порядка на множестве переменных;
- упорядочение мономов по степеням, а мономы одной и той же степени упорядочиваются лексикографически;
- упорядочение мономов по степеням, а мономы одной и той же степени упорядочиваются в обратном лексикографическом порядке.

Отображение φ ставит в соответствие любому многочлену f его старший моном (присутствующий в f с ненулевым коэффициентом).

2.3. Упражнение. Показать, что многочлены с различными старшими мономами линейно независимы.

2.4. Упражнение. Показать, что свойство системы Σ порождать или не порождать векторное пространство I не зависит от выбора представителей в прообразах элементов из T.

2.5. Упражнение. Показать, что система Σ порождает векторное пространство I тогда и только тогда, когда полугруппа, порожденная в T старшими мономами элементов множества F, совпадает с полугруппой старших мономов элементов идеала I.

2.6. Упражнение. Показать, что система Σ порождает векторное пространство I тогда и только тогда, когда идеал, порожденный старшими мономами элементов множества F, совпадает с ассоцинированным градуированным идеалом идеала I (относительно фильтрации с одномерными факторами, определяемой введенным отношением порядка).

Рассматриваемая ситуация укладывается в следующую более общую схему: имеется градуированное некоторым вполне упорядоченным множеством векторное пространство grM с одномерными од-

нородными компонентами. Фиксирован базис Г этих компонентов. На пространстве M рассматривается фильтрация, совместная с градуировкой. Выбирается множество Г' элементов фильтрованного пространства M такое, что при переходе к градуированному пространству grM различные элементы множества Г' переходят в различные элементы множества Г. Тогда множество $\Gamma' \cup (\Gamma \setminus \mathrm{gr}\Gamma')$ является базисом пространства M и определяет разложение пространства M в прямую сумму подпространств M' и M'', где M' – пространство с базисом Г', а пространство M'' изоморфно факторпространству M/M', следовательно, определяет каноническое представление пространства M/M' в M.

В случае кольца многочленов градуировка осуществляется полугруппой N_0^n, где N_0 – множество неотрицательных целых чисел, в дальнейшем мы будем рассматривать также градуировку множеством $N_0^n \cdot (Z/kZ)$, где к свободной коммутативной полугруппе добавляется конечная. Такое множество соответствует, например, свободному конечнопорожденному модулю над кольцом многочленов (не обязательно коммутативных). Конечная компонента соответствует образующим свободного модуля. Примеры упорядочений рассматривались в пункте 1.9.

Вернемся к рассмотрению полиномиальных идеалов. Как уже отмечалось, в качестве базиса Г выбирается множество мономов T, утверждение о том, что R является градуированным векторным пространством с базисом T означает, что любой многочлен можно записать в виде $f = a_0 m_0 + \sum_j a_j m_j$, $j \geq 1$, где $m_0 > m_j$ для всех $j \geq 1$. Переход от фильтрации к градуировке означает выделение старшего одночлена: $\mathrm{gr}(f) = a_0 m_0$.

В частности, такое представление имеет место для всех образующих f_1 идеала I, причем мы можем выбрать эти образующие так, чтобы старшие коэффициенты у них были равны 1, так как мы предполагаем, что K – поле:

$$f_i = m_{i0} + \sum_j a_{ij} m_{ij}, \quad i = 1..m. \qquad (2.2)$$

В качестве Г' можно выбрать любое подмножество $\Sigma \subset T \cdot F$, где $F = \{f_1, \ldots, f_n\}$ – произвольная система образующих идеала I, руководствуясь двумя требованиями: во-первых, различные эле-

менты множества Σ должны иметь разные старшие мономы; во-вторых, система Σ должна быть максимальна в том смысле, что для любого элемента $\xi \in T \cdot F$ существует элемент $\sigma \in \Sigma$ с таким же старшим мономом. Например, можно включить в Σ множество $T \cdot f_1$, далее добавить к нему те элементы множества $T \cdot f_2$, старшие мономы которых отличаются от старших мономов всех элементов, уже включенных в множество Σ и т.д.

2.7. Определение. Систему образующих F идеала I назовем базисом Гребнера этого идеала, если подмножество Σ, введенное выше, образует базис векторного пространства I.

Из сформулированных выше упражнений следует корректность определения базиса Гребнера, т.е. независимость его от конкретного выбора множества Σ.

2.8. Пример. Пусть I – главный идеал, порожденный многочленом f. Тогда f является базисом Гребнера идеала I.

2.9. Пример. Многочлены $f_1 = x^2 - 1$ и $f_2 = x^3 - 1$ не составляют базис Гребнера порождаемого ими идеала. Доказать.

В следующих примерах рассматривается кольцо многочленов $K[x_1, \ldots, x_n]$, содержащее идеал I, заданный множеством образующих $F=(f_1, \ldots, f_m)$, предполагается, что одночлены в записи элементов f_1 упорядочены в соответствии с одним из введенных выше отношений порядка и нормированы таким образом, что их старшие коэффициенты равны 1.

2.10. Пример. Если $I = K[x_1, \ldots, x_n]$, то F является базисом Гребнера идеала I тогда и только тогда, когда $1 \in F$.

2.11. Пример. Если I – максимальный идеал, то F – базис Гребнера тогда и только тогда, когда для любой переменной x_1 найдется элемент $f(i) \in F$ со старшим мономом x_1.

Следует заметить, что введенное выше определение базиса Гребнера не является конструктивным: не указан алгоритм, как можно проверить, что некоторая система многочленов представляет базис Гребнера порождаемого ими идеала, и тем более не дан алгоритм, позволяющий для идеала, заданного некоторой системой образующих, построить его базис Гребнера.

В следующем параграфе определение базиса Гребнера будет дано в более общей ситуации, а также будут приведены алгоритмы проверки, является ли данная система образующих идеала его базисом Гребнера, и в случае отрицательного ответа, алгоритм, позволяющий пополнить эту систему до базиса Гребнера.

3. БАЗИСЫ ГРЕБНЕРА ДЛЯ ПОЛИНОМИАЛЬНЫХ, ДИФФЕРЕНЦИАЛЬНЫХ И РАЗНОСТНЫХ МОДУЛЕЙ

В настоящем параграфе излагаются основные результаты теории базисов Гребнера для подмодулей свободных полиномиальных, дифференциальных и разностных модулей. В качестве основных примеров рассматриваются идеалы в кольце многочленов от нескольких переменных. Будем придерживаться следующих обозначений.

Пусть κ – поле, $P=\kappa[X_1, \ldots, X_n]$ – кольцо многочленов над κ, или κ – дифференциальное поле с множеством операторов дифференцирования $\Delta = \{\delta_1, \ldots, \delta_n\}$ и $P=\kappa[\delta_1, \ldots, \delta_n]$ – кольцо дифференциальных операторов, или κ – разностное поле с множеством операторов трансляции $\Delta = \{\tau_1, \ldots, \tau_n\}$ и $P=\kappa[\tau_1, \ldots, \tau_n]$. Символы κ, P и n в дальнейшем будут использоваться только для обозначения одного из перечисленных здесь объектов. На множестве $\Gamma = \mathbb{N}_0^n$ выберем отношение порядка, удовлетворяющее для всех $\alpha_1, \alpha_2 \in \mathbb{N}_0^n$ следующим условиям:

$$\alpha_1 \neq 0 \Longrightarrow 0 < \alpha_1, \tag{3.1}$$

$$\alpha_1 < \alpha_2 \Longleftrightarrow (\beta \in \mathbb{N}_0^n \Longrightarrow \alpha_1 + \beta < \alpha_2 + \beta). \tag{3.2}$$

Отношение $<$ на $\mathbb{N}_0^n$ канонически соответствует упорядочению $<_T$ в кольце P множества термов (мономов)

$$T=\{X_1^{i_1} \ldots X_n^{i_n} : (i_1, \ldots, i_n) \in \mathbb{N}_0^n\},$$

или $T=\{\delta_1^{i_1} \ldots \delta_n^{i_n} : (i_1, \ldots, i_n) \in \mathbb{N}_0^n\}$,

или $T=\{\tau_1^{i_1} \ldots \tau_n^{i_n} : (i_1, \ldots, i_n) \in \mathbb{N}_0^n\}$ соответственно.

Примеры соответствующих отношений порядка (лексикографическое упорядочение, упорядочение по степени, затем лексикографическое; упорядочение по степени, затем обратное лексикографическое) рассматривались в п. 1.6.

Введем отношение частичного порядка $<_M$ на T следующим образом:

$$\varphi_1 <_M \varphi_2 \Longleftrightarrow \exists\, \varphi \in T: \varphi_2 = \varphi\varphi_1. \tag{3.3}$$

Из (1.1) и (1.2) следует, что отношение $<_T$ совместно с $<_M$, т.е

$$\varphi_1 <_M \varphi_2 \Rightarrow \varphi_1 <_T \varphi_2 \qquad (3.4)$$

Пусть P^m - свободный P-модуль. Определим множество термов в P^m

$$T_m \quad \{(t,\zeta) \quad \zeta \in T,\ t \in \{1, \ldots, m\}\}.$$

Частичный порядок $<_M$ на T_m определим условием.

$$t_1 <_M t_2 \iff \exists\, \zeta \in T:\ t_2 = \zeta\, t_1.$$

Упорядочение $<_T$ термов из T_m – это отношение линейного порядка, удовлетворяющее соотношениям:

$$t_1 <_M t_2 \quad \Rightarrow \quad t_1 <_T t_2,$$

$$\forall\, \zeta \in T:\ t_1 <_T t_2 \iff \zeta\, t_1 <_T \zeta\, t_2.$$

Упорядочение, удовлетворяющее выписанным условиям, можно получить, исходя из упорядочения одночленов в кольце коэффициентов, рассматривая элементы свободного модуля как структуру данных "вектор элементов типа обобщенный многочлен с индексом $1..m$", т.е. сравнивая термы вида (t,ζ) по их первой координате t, и только в случае равенства ее для двух термов переходить к сравнению мономов из кольца обобщенных многочленов. Однако в дифференциальной алгебре чаще используется следующее отношение порядка.

$t_1 \quad (i_1,\varphi_1) <_T t_2 \quad (i_2,\varphi_2)$ тогда и только тогда, когда

либо $\deg \varphi_1 < \deg \varphi_2$,

либо $\deg \varphi_1 = \deg \varphi_2$ и $i_1 < i_2$,

либо $\deg \varphi_1 = \deg \varphi_2$, $i_1 = i_2$ и $\varphi_1 < \varphi_2$ относительно лексикографического порядка при фиксированной нумерации переменных.

3.1. Определение. Любой элемент $f \in P^m \setminus (0)$ допускает единственное представление в виде конечной суммы:

$$f = \sum_{i=1}^{r} c(f,t_i)t_i, \quad 0 \neq c(f,t_i) \in \kappa,\ t_i \in T_m, \qquad (3.5)$$

$$t_r <_T t_{r-1} <_T \cdots <_T t_1.$$

Положим по определению $\mathrm{Hterm}(f) := t_1$, $\mathrm{Hcoeff}(f)$: $c(f,t_1)$. Оп-

ределим Hterm(0) := 0, Hcoeff(0) := 0. Аналогично, для любого конечного множества $B \subset P^m$, Hterm(B) := {Hterm(f) : $f \in B$} и для любого подмодуля $U \subset P^m$, Hterm(U) обозначает подмодуль, порожденный множеством {Hterm(f): $f \in U$}.

3.2. Определение. Пусть $B \subset P^m \setminus \{0\}$ – конечное множество образующих некоторого P-модуля $U \subset P^m$ (без потери общности можно предположить, что Hcoeff(g) = 1 для любого $g \in B$). Определим процесс редукции следующим образом: $f \xrightarrow[B]{} f'$, если $f, f' \in P^m$, и существуют $t \in T_m$, $\zeta \in T$, $g \in B$ такие, что $c = c(f,t) \neq 0$, $t = \zeta\,\mathrm{Hterm}(g)$, $f' = f - c\zeta g$.

В дальнейшем, будем опускать указание на множество B, если это не приведет к двусмысленности, или если выбор множества B не существенен.

Символ $\xrightarrow[B]{*}$ обозначает транзитивное, а $\xrightarrow[B]{+}$ – рефлексивно-транзитивное замыкание отношения $\xrightarrow[B]{}$. Элемент f называется нередуцируемым, если не существует элемента $f' \neq f$ такого, что $f \xrightarrow[B]{} f'$.

3.3. Примеры

3.3.1. Пусть n=0, т.е. P^m – векторное пространство над полем κ. Если v – вектор, у которого первая ненулевая координата стоит на i-м месте, то редукция вектора w относительно v приводит к обнулению i-й координаты вектора w путем вычитания соответствующего кратного вектора v.

3.3.2. Пусть κ – поле, n=1 и m=1, т.е рассматривается кольцо многочленов от одной переменной. Пусть B={x^3–1, x^2–1}. Тогда $x^4 \xrightarrow[B]{} x$ и $x^4 \xrightarrow[B]{} x^2$.

3.4. Определение. Нормальную редукцию (алгоритм нормальной формы) $\Rightarrow_C$ можно определить, предполагая, что редукция элементов осуществляется по однозначному правилу, например, редуцируемые термы выбираются в порядке убывания относительно полного упорядочения термов, при фиксированном терме соотношения выбираются в том порядке, как они располагаются в множестве G.

3.5.Определение. Частичную редукцию определим как нормальную редукцию, осуществляемую только ... тех пор, пока реду-

цируется старший терм.

3.6. Лемма. Если $f \xrightarrow[B]{*} f'$, то элементы f и f' принадлежат одному и тому же смежному классу P^m/U.

Доказательство. $g \in B$, следовательно, $c\zeta g \in U$.

3.7. Лемма. Для любого конечного множества многочленов F и для любого многочлена g цепочка $g \xrightarrow[F]{} g_1 \xrightarrow[F]{} \dots \xrightarrow[F]{} g_k$ обрывается после конечного числа шагов. Следовательно, для любого элемента g существует (не обязательно единственный) нередуцируемый элемент g' такой, что $g \xrightarrow[F]{*} g'$.

Доказательство. Предположим противное. Выберем среди всех бесконечных цепочек редукций цепочку, начинающуюся с многочлена g с минимальным относительно введенного отношения порядка старшим одночленом t. Возможны две ситуации: либо на некотором шаге редукции одночлен t редуцируется и оставшаяся часть цепочки начинается с многочлена, все слагаемые которого меньше, чем t; либо t не редуцируется ни на каком шаге редукции. В обоих случаях получается противоречие с минимальностью выбранной цепочки: в первом случае можно выбрать хвост исходной цепочки, остающийся после редуцирования t; во втором – вычесть из всех элементов цепочки одночлен t.

3.8. Предложение. Множество нередуцируемых относительно отношения $\xrightarrow[B]{}$ элементов является векторным κ-пространством.

Доказательство. Нужно проверить, что если f и g – нередуцируемые элементы и $c \in \kappa$, то элементы $f + g$ и cf также нередуцируемы. Это немедленно следует из того, что в $f + g$ и cf присутствуют с ненулевыми коэффициентами только те слагаемые, которые присутствуют в f и g.

3.9. Определение. На прямом произведении $P^m \times P^m$ определим функцию S: $S(f,f') := 0$, если $f = 0$, $f' = 0$ или HOK(Hterm(f), Hterm(f')) не определен, в остальных случаях $S(f,f') :=$ Hcoeff(f') φf – Hcoeff(f) $\zeta f'$, где $\varphi,\zeta \in T$ и φHterm(f) = HOK(Hterm(f), Hterm(f')) = ζHterm(f').

Следующее определение является непосредственным обобщением

определения базисов Гребнера полиномиальных идеалов (2.7) на подмодули свободных полиномиальных, дифференциальных и разностных модулей.

3.10. Определение. Пусть $U \subset P^m$ – подмодуль свободного модуля P^m, $G \subset U$ – конечное множество, $<_T$ – упорядочение множества термов T_m. Множество G называется базисом Гребнера (G-базисом) подмодуля U, если для любого ненулевого элемента $f \in U$ имеется представление Гребнера (G-представление):

$$f = \sum_{i=1}^{r} c_i \varphi_i g_i, \quad 0 \neq c_i \in \kappa, \quad \varphi_i \in T, \ g_i \in G, \qquad (3.6)$$

$$\varphi_i \operatorname{Hterm}(g_i) >_T \varphi_{i+1} \operatorname{Hterm}(g_{i+1}),$$

откуда, в частности, следует, что

$$\operatorname{Hterm}(f) = \varphi_1 \operatorname{Hterm}(g_1).$$

Недостатком введенного определения является то, что для одного и того же элемента могут существовать различные G-представления. Например, если $g_1 = t^2 - 1$, $g_2 = t^3 - 1$, то $(t^2 - 1)(t^3 - 1) = t^3 \cdot g_1 - g_1 = t^2 \cdot g_2 - g_2$ – два различных G-представления одного и того же многочлена. С другой стороны, достаточно сложно проверить, что некоторый элемент не допускает G-представления. От этих недостатков можно избавиться, если потребовать, чтобы любой одночлен мог появляться в G-представлении в качестве старшего терма слагаемого $\varphi_1 g_1$ не более чем для одного элемента $g_1 \in G$. В частности, можно предполагать, что элементы множества G упорядочены, и при выборе линейно независимых элементов вида $\varphi_1 g_1$ мы руководствуемся правилами, сформулированными в определении нормальной редукции 3.4. Представление такого вида мы будем называть нормальным G-представлением.

Для формулировки основного результата настоящего раздела введем некоторое обозначение и докажем три леммы.

3.11. Обозначение. Для элементов $f, f' \in P^m$ будем писать $f \nabla f'$, если существует элемент $f'' \in P^m$ такой, что $f \xrightarrow{*} f''$ и $f' \xrightarrow{*} f''$.

3.12. Лемма. Пусть $f, f', f'' \in P^m$ и $f \xrightarrow{*} f'$. Тогда $f + f'' \nabla f' + f''$.

Доказательство. Пусть $f \quad f'+c\cdot\eta\cdot g$, где $\mathrm{Hterm}(\eta\cdot g) \quad \varphi$ и $c = c(\varphi,f)\neq 0$, $c(\varphi,f')=0$. Если $c'' \quad c(\varphi,f'')$, то $f'' \quad c''\cdot\eta\cdot g+h$, где $c(\varphi,h)=0$, тогда $f+f'' = f'+c\cdot\eta\cdot g+c''\cdot\eta\cdot g+h \quad f'+h+(c+c'')\cdot\eta\cdot g \longrightarrow^{*} f'+h$ и $f'+f'' = f'+h+c''\cdot\eta\cdot g \longrightarrow^{*} f'+h$.

3.13. Лемма. Если множество G порождает подмодуль $U \subset P^m$ и $f - f' \in U$, то существует целое $s \geq 0$ и элементы $f=f_0,f_1, \ldots, f_s=f'$ такие, что для всех i от 1 до s либо $f_{i-1} \longrightarrow f_i$, либо $f_i \longrightarrow f_{i-1}$.

Доказательство. Поскольку G порождает модуль U, элемент $f-f'$ можно представить в виде суммы

$$\sum_{i=0}^{r} c_i\cdot\eta_i\cdot g_i,$$

где c_i – коэффициенты, $\eta_i \in T$, $g_i \in G$ (могут совпадать и различных значениях i). Доказательство леммы будем вести индукцией по минимальной длине такого представления r. Если r 0, то $f f'$ и утверждение леммы выполнено. Для произвольного r мы можем предполагать, что $\varphi = \mathrm{Hterm}(\eta_r\cdot g_r) \geq_M \mathrm{Hterm}(\eta_i\cdot g_i)$ для всех i. Положим $f_1=f-c(\varphi,f)\cdot\eta_r\cdot g_r$, $f_2=f_1-(c_r-c(\varphi,f))\cdot\eta_r\cdot g_r$. Тогда $f \longrightarrow f_1 \longleftarrow f_2$ и $f_2 - f' \quad f - f' - c_r\cdot\eta_r\cdot g_r \quad \sum_{i=0}^{r-1} c_i\cdot\eta_i\cdot g_i$, так что можно применить предположение индукции. □

3.14. Определение. Будем говорить, что отношение редукции $\longrightarrow$ удовлетворяет условию слияния, если для любого элемента из $f \longrightarrow^{*} f'$ и $f \longrightarrow^{*} f''$ следует, что $f' \nabla f''$.

3.15. Определение. Будем говорить, что отношение редукции $\longrightarrow$ удовлетворяет локальному условию слияния, если для любого элемента f из $f \longrightarrow f'$ и $f \longrightarrow f''$ следует, что $f' \nabla f''$.

3.16. Определение. Будем говорить, что отношение редукции $\longrightarrow$ удовлетворяет псевдолокальному условию слияния, если для всех $f,f',f'' \in P^m$ таких, что $f \longrightarrow f'$ и $f \longrightarrow f''$ существует целое $s \geq 0$ и элементы $f'=f_0,f_1, \ldots, f_s=f''$ такие, что $f \longrightarrow^{*} f_i$ и $f_{i-1} \nabla f_i$ для всех $i \in 1..s$.

3.17. Лемма. Если нетерово отношение $\longrightarrow$ удовлетворяет псевдолокальному условию слияния, то отношение $\longrightarrow$ удовлетворя-

ет условию слияния.

Доказательство. Воспользуемся "нетеровой" индукцией, т.е. покажем, что если утверждение леммы верно для всех g таких, что $f \longrightarrow g$, то оно верно и для f. Такой индукции достаточно для доказательства леммы, поскольку в противном случае некоторый элемент f, для которого утверждение леммы не выполняется, мог бы быть выбран в качестве первого элемента бесконечной цепочки $f \longrightarrow f_1 \longrightarrow \ldots \longrightarrow f_n \longrightarrow \ldots$, для всех элементов которой утверждение леммы также не выполняется.

Итак, фиксируем f и предположим, что для всех элементов f^* таких, что $f \overset{*}{\longrightarrow} f^*$, утверждение леммы выполняется. Покажем, что оно выполняется и для f. Без потери общности мы можем предполагать, что данные элементы f' и f'' отличны от f, т.е. имеют место редукции $f \longrightarrow g_1 \overset{*}{\longrightarrow} f'$ и $f \longrightarrow g_2 \overset{*}{\longrightarrow} f''$. Элементы g_1 и g_2 удовлетворяют псевдолокальному условию слияния при некотором s.

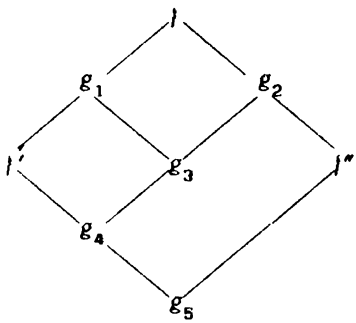

Рис. 1

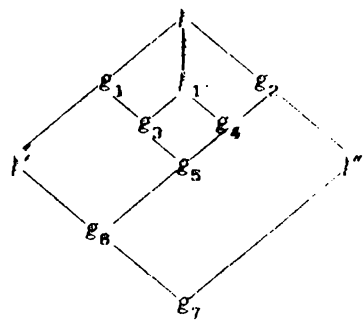

Рис. 2

Доказательство будем вести индукцией по s. Основание индукции по s предполагает s-1, т.е. g_1 и g_2 удовлетворяют локальному условию слияния. Из условия локального слияния следует, что существует g_3 такой, что $g_2 \overset{*}{\longrightarrow} g_3$ и $g_1 \overset{*}{\longrightarrow} g_3$. По предположению индукции для элементов f' и g_3 существует элемент g_4 такой, что $f' \overset{*}{\longrightarrow} g_4$ и $g_3 \overset{*}{\longrightarrow} g_4$, а также элемент g_5 такой, что $f'' \overset{*}{\longrightarrow} g_5$ и $g_4 \overset{*}{\longrightarrow} g_5$. Этот элемент удовлетворяет

условию леммы (см. рис. 1).

Переход от s к s+1 иллюстрируется следующей диаграммой (рис. 2). Пусть g_1 и g_2 удовлетворяют псевдолокальному условию слияния с цепочкой из s+2 элементов: $g_1=f_0$, f_1, ..., f_s, $f_{s+1}=g_2$. По предположению индукции элементы f_1 и g_2 удовлетворяют условию слияния (элемент g_4). Существование элементов g_5, g_6 и g_7 в приведенной диаграмме следует из предположения о том, что элементы, получающиеся редукцией элементов, следующих за f, в частности, g_1, f_1, g_2, удовлетворяют условию слияния □

Следующая теорема перечисляет ряд условий, равносильных определению базиса Гребнера. Следует отметить, что среди них содержатся условия 6' и 7', позволяющие за конечное число шагов проверить, является ли выписанная система образующих подмодуля U его базисом Гребнера.

3.18. Теорема. Пусть $U \subset P^m$ – подмодуль свободного модуля, $G \subset U$ – конечное множество, $<_T$ – отношение порядка на множестве термов T_m, предположим, что множество G нормализовано таким образом, что $\mathrm{Hcoeff}(g_1) = 1$ для всех $g_1 \in G$. Тогда эквивалентны следующие условия:

1. G является G-базисом модуля U

1'. Любой элемент модуля U допускает нормальное G-представление.

2. $\mathrm{Hterm}(G)$ порождает $\mathrm{Hterm}(U)$.

3. Для любого $f \in U$ имеет место $f \xrightarrow[G]{*} 0$.

3'. Для любого $f \in U$ имеет место $f \xrightarrow[G]{*} 0$.

4. Если $f - f' \in U$ и f, f' – нередуцируемы, то $f = f'$.

5. Если $f \in U$ и f – нередуцируем, то $f = 0$.

Следующие условия являются необходимыми для выполнения предыдущих и если множество G порождает U, то они являются и достаточными.

6. Если f, $f' \in G$ и $S(f,f') \neq 0$, то $S(f,f')$ допускает G-представление.

6'. Если $f,f' \in G$ и $S(f,f') \neq 0$, то $S(f,f')$ допускает нормальное G-представление.

7. Если $f,f' \in G$ и НОК(Hterm(f), Hterm(f')) определен, то в G существуют элементы $f = f_0, \ldots, f_1, \ldots, f_s = f'$ такие, что

$$\text{НОК}\,(\text{Hterm}(f_i)\colon i=0,\ldots,s) = \text{НОК}\,(\text{Hterm}(f),\text{Hterm}(f')) \qquad (3.7)$$

и каждый S-элемент $S(f_{i-1}, f_i)$, $i=1, \ldots, s$, допускает G-представление.

7'. Если $f,f' \in G$ и НОК(Hterm(f), Hterm(f')) определен, то в G существуют элементы $f = f_1, \ldots, f_1, \ldots, f_s = f'$, удовлетворяющие условию 3.7, такие, что каждый S-элемент $S(f_{i-1}, f_i)$, $i=1, \ldots, s$, допускает нормальное G-представление.

8. Если $f \xrightarrow[G]{*} f'$, $f \xrightarrow[G]{*} f''$, f',f'' – нередуцируемы, то $f' = f''$.

9. Если $f \xrightarrow[G]{*} f'$, $f \xrightarrow[G]{*} f''$, то существует $h \in P^m$ такой, что $f' \xrightarrow[G]{*} h$, $f'' \xrightarrow[G]{*} h$, т.е. $\xrightarrow[G]{}$ удовлетворяет условию слияния.

10. Для любых $f, f' \in G$ $S(f, f') \xrightarrow[G]{*} 0$.

10'. Для любых $f, f' \in G$ $S(f, f') \overset{*}{\Longrightarrow}_G 0$.

11. Если $f,f' \in G$ и НОК(Hterm(f), Hterm(f')) определен, то в G существуют элементы $f = f_0, \ldots, f_1, \ldots, f_r = f'$, удовлетворяющие условию (3.7) и такие, что для всех $i = 1,\ldots,r$ $S(f_{i-1}, f_i) \xrightarrow[G]{*} 0$.

11'. Если $f,f' \in G$ и НОК(Hterm(f), Hterm(f')) определен, то в G существуют элементы $f = f_0, \ldots, f_1, \ldots, f_r = f'$, удовлетворяющие условию (3.7) и такие, что для всех $i = 1,\ldots,r$ $S(f_{i-1}, f_i) \overset{*}{\Longrightarrow}_G 0$.

Доказательство. Докажем следующие импликации:

3 --> 10 --> 11

3' —> 10' —> 11' —> 11

3 --> 4 --> 5 --> 3' —> 3

3 --> 2 --> 1' —> 1 --> 6 --> 7 --> 11 --> 9 --> 3

1' —> 6' —> 7' —> 7

4 --> 8 --> 9

3 --> 10. Тривиально, поскольку $S(f,f') \in U$.

3' —> 10'. Аналогично.

10 --> 11. Достаточно положить r=1.

10' —> 11'. Аналогично.

11' —> 11. Тривиально.

3 --> 4. По предложению **3.8** множество нередуцируемых элементов является векторным пространством, значит, если f и f' нередуцируемы, то и их разность нередуцируема. Поскольку $f - f' \in U$, из **3.** и предыдущего замечания следует, что $f - f' = 0$, т.е. **4.**

4. --> 5. Полагаем $f' = 0$.

5. --> 3'. Достаточно применить леммы **3.6** и **3.7.**

3'. —> 3. Очевидно.

3. --> 2. Пусть $u \in U$ и $\mathrm{Hterm}(u) \notin \mathrm{Hterm}(G)$. Тогда элемент u не может редуцироваться к 0, что противоречит **3.**

2. --> 1'. Пусть существуют $0 \neq u \in U$, для которых нет нормального G-представления. Среди таких элементов выберем элемент с минимальным $\mathrm{Hterm}(u)$. По условию **2** можно применить шаг редукции, сокращающий $\mathrm{Hterm}(u)$. Полученное противоречие с минимальностью $\mathrm{Hterm}(u)$ доказывает **1'.**

1'. —> 1. Очевидно.

1. --> 6. Достаточно заметить, что если $f \in G$, $f' \in G$, то $S(f,f') \in U$.

1'. —> 6'. Аналогично.

6. --> 7. Очевидно.

6'. —> 7'. Также очевидно.

7'. —> 7. Очевидно.

7. --> 11. Пусть $1 \le i \le r$, $u = S(f_{i-1}, f_i)$ и

$$u = \sum_{j=1}^{r} c_j \varphi_j g_j, \quad 0 \neq c_j \in \kappa, \quad \varphi_j \in T, \ g_j \in G,$$

$$\varphi_1 \mathrm{Hterm}(g_1) >_T \varphi_{i+1}\mathrm{Hterm}(g_{i+1})$$

– G-представление элемента u. Положим $u_k = \sum_{j=k}^{r} c_j \varphi_j g_j$. Тогда $u = u_1 \xrightarrow{G} u_2 \xrightarrow{G} \dots \xrightarrow{G} u_r \xrightarrow{G} 0$.

11. —> 9. Ввиду леммы **3.17** достаточно доказать, что отношение редукции удовлетворяет псевдолокальному условию слияния. Пусть $f \longrightarrow f'$, $f \longrightarrow f''$. Это означает существование элементов g', g'' $\in G$, η', η'' $\in T$, $\varphi' = \eta' \mathrm{Hterm}(g')$, $\varphi'' = \eta'' \mathrm{Hterm}(g'')$ таких, что $f' = f - c'\eta' g'$, $f'' = f - c''\eta'' g''$, где $c' = c(f,\varphi') \neq 0$, $c'' = c(f,\varphi'') \neq 0$, но $c(f',\varphi') = c(f'',\varphi'') = 0$. Можно предполагать, что $\varphi'' \le_T \varphi'$. Обозна-

чим $R(\xi) = \xi - Hcoeff(\xi)Hterm(\xi)$ для любого $\xi \in P^m$.

Выделим в f слагаемое $c'\varphi'$, т.е. $f = f_1 + c'\varphi' + f_2$, где f_1 состоит из слагаемых, которые больше чем $c'\varphi'$, а f_2 – из слагаемых, меньших $c'\varphi'$. Нужно рассмотреть два случая: $\varphi'' < \varphi'$ и $\varphi'' = \varphi'$. В первом из них, полагая $f_2'' = f_2 - c''\eta''g''$ и $f_0 = f_1 - c'\eta' R(g')) + f_2''$, по лемме **3.13** получаем $f' = (f_1 - c'\eta' R(g')) + f_2 \nabla (f_1 - c'\eta' R(g')) + f_2'' = f_0$, откуда $f' \nabla f''$.

В случае, когда $\varphi' = \varphi''$, одновременно выполняются условия $Hterm(g') \leq_M \varphi'$ и $Hterm(g'') \leq_M \varphi''$. Поэтому определен $НОК(Hterm(g'), Hterm(g''))$. По условию 11 доказываемой теоремы в G существует последовательность $g' = g_0, \ldots, g_1, \ldots, g_t = g''$, удовлетворяющая условию 3.7 и такая, что $S(g_{i-1}, g_i) \longrightarrow^* 0$ для любого i. Значит, для любого i $Hterm(g_i) \leq_M \varphi'$, поэтому существуют $\eta_i \in T$ такие, что $\eta_i Hterm(g_i) = \varphi'$, $c'\varphi' \longrightarrow c'\eta_i R(g_i)$ и $f \dashrightarrow f_1 - c'\eta_i R(g_i))^* f_2 : h_i$, $i \in 1..t$. Покажем, что $h_{i-1} \nabla h_i$. Это следует из того, что $h_i - h_{i-1} = c'\eta_{i-1}R(g_{i-1}) - c'\eta_i R(g_i) = c'\vartheta S(f_i, f_{i-1}) \longrightarrow^* 0$, где $\vartheta \in T$ и удовлетворяет условию $\vartheta \cdot НОК(Hterm(g_i), Hterm(g_{i-1})) = \varphi'$. Следовательно, отношение $\longrightarrow$ удовлетворяет псевдолокальному условию слияния.

9 $\longrightarrow$ 3. Если множество G порождает U, то по лемме **3.13** существуют элементы $f = f_0, \ldots, f_i, \ldots, f_s = 0$ такие, что для любого i либо $f_{i-1} \longrightarrow f_i$, либо $f_i \longrightarrow f_{i-1}$. Пусть k обозначает наибольший индекс, для которого не выполняется условие $f_i \longrightarrow^* 0$. Тогда $f_{k+1} \longrightarrow^* 0$ и $f_{k+1} \longrightarrow f_k$. По условию 9, $0 \nabla f_k$, откуда получаем противоречие с выбором k, поскольку 0 редуцируется только в самого себя.

4 $\longrightarrow$ 8. $f' - f'' = (f' - f) + (f'' - f) \in U$, следовательно, $f' \nabla f''$.

8 $\longrightarrow$ 9. Пусть $f \longrightarrow^* f'$ и $f \longrightarrow^* f''$. Выберем нередуцируемые f_1' и f_1'' такие, что $f' \longrightarrow^* f_1'$, $f'' \longrightarrow^* f_1''$. Из 8 следует, что $f_1' = f_1''$, т.е. отношение $\longrightarrow$ удовлетворяет условию слияния.

Поскольку вопрос о G-представимости элемента может быть решен алгоритмически, пункт **3.18.6'** дает нам возможность сформулировать алгоритм проверки, является ли данная система образующих подмодуля его базисом Гребнера. Пункт **7'** этой же теоремы позволяет нам оптимизировать полученный алгоритм, проверяя G-представимость не всего множества S-элементов, а только неко-

торого его подмножества. Детализированные алгоритмы приведены в следующем параграфе.

3.19. Примеры и упражнения

3.19.1. Показать, что многочлены

$$f_1 = x^3yz - xz^2,$$
$$f_2 = xy^2z - xyz,$$
$$f_3 = x^2y^2 - z^2$$

не составляют базис Гребнера порождаемого ими идеала (упорядочение по степени, затем лексикографическое, $x > y > z$).

3.19.2. Показать, что многочлены

$$f_1 = x^3yz - xz^2,$$
$$f_2 = xy^2z - xyz,$$
$$f_3 = x^2y^2 - z^2,$$
$$f_4 = x^2yz - z^3,$$
$$f_5 = xz^3 - xz^2,$$
$$f_6 = yz^3 - z^3,$$
$$f_7 = xyz^2 - xz^2,$$
$$f_8 = z^4 - x^2z^2,$$
$$f_9 = x^3z^2 - xz^2$$

образуют базис Гребнера идеала, введенного в предыдущем упражнении.

3.19.3. Показать, что, используя 3.18.11, в предыдущем упражнении достаточно рассмотреть S-элементы для пар (2,3), (2,4), (5,6), (4,7), (2,7), (5,7), (5,8), (6,8), (4,9), (5,9).

Если данная система элементов не является базисом Гребнера порождаемого ею подмодуля, то ее можно расширить, присоединяя поочередно элементы, получающиеся редуцированием S-элементов.

Базис Гребнера для любого подмодуля U определен неоднозначно. В частности, после присоединения к базису Гребнера модуля U любого элемента $h \in U$ снова получаем базис Гребнера модуля U. Естественно возникает вопрос о минимальных базисах Гребнера.

В дифференциальной алгебре принята следующая терминология.

3.20. Определение. Подмножество G $(g_i : i \in I)$ свободного модуля P^m называется авторедуцированным множеством, если любой элемент $g_i \in G$ нередуцируем относительно $G \setminus \{g_i\}$.

Из определения немедленно следует, что ведущие термы всех элементов, принадлежащих авторедуцированному множеству, различны

3.21. Предложение. *Любое авторедуцированное множество состоит из конечного числа элементов, следовательно, его элементы можно упорядочить по возрастанию ведущих термов.*

Доказательство. Предположим, что имеется бесконечное авторедуцированное множество. Выберем из него бесконечное подмножество элементов, старшие члены которых зависят от одной и той же модульной образующей (это можно сделать, поскольку модульных образующих у нас конечное число). Представляя эти старшие члены в виде векторов с неотрицательными целыми координатами, выберем бесконечное подмножество элементов, в которых первая координата не убывает, из них выберем бесконечное подмножество элементов, у которых не убывает вторая координата и т.д. После нескольких шагов такого процесса получим бесконечное множество элементов, у которых все координаты не убывают. Но это противоречит авторедуцированности множества. □

Зафиксировав отношение порядка $<$ на множестве термов T_m, можно ввести отношение частичного порядка на множестве авторедуцированных множеств.

Предположим, что $A = \{f_1, \ldots, f_p\}$ и $B = \{g_1, \ldots, g_q\}$ – авторедуцированные множества, элементы которых упорядочены по возрастанию ведущих термов. Будем считать, что $A < B$, если,

– либо существует i, $1 \le i \le \min(p,q)$ такое, что $\mathrm{Hterm}(f_1) < \mathrm{Hterm}(g_1)$ и $\mathrm{Hterm}(f_j) = \mathrm{Hterm}(g_j)$ для $j < i$,

– либо $p > q$ и $\mathrm{Hterm}(f_j) = \mathrm{Hterm}(g_j)$ для $1 \le j \le q$.

3.22. Лемма. *Любое множество авторедуцированных подмножеств содержит минимальный элемент относительно введенного частичного порядка. Минимальный элемент в множестве всех авторедуцированных подмножеств некоторого подмодуля U свободного P-модуля является базисом Гребнера модуля U.*

Доказательство. По предложению 3.21 мы можем предполагать,

что элементы в наших авторедуцированных множествах упорядочены по возрастанию старших термов. Зафиксируем минимальное значение Hterm для первых элементов рассматриваемых авторедуцированных множеств (это значение определено однозначно, поскольку множество термов вполне упорядочено). Обозначим его t_1. В системе авторедуцированных множеств $A = \{a_1, \ldots,\}$ рассмотрим подсистему множеств таких, что $Hterm(a_1) = t_1$. В ней найдем минимальное значение старшего терма вторых элементов, обозначим его t_2. Продолжая подобным образом, получим авторедуцированную систему термов, упорядоченную по возрастанию. По предложению 3.21 эта система должна обрываться на конечном шаге. Выбор системы старших термов осуществлялся таким образом, что всегда существовало авторедуцированное множество, старшие термы элементов которого имели вид $t_1, \ldots, t_1$. Авторедуцированное множество, соответствующее полной системе $t_1, \ldots, t_n$, является минимальным.

3.23. Определение. Базис Гребнера G модуля $U \subset P^m$ назовем авторедуцированным, если множество G авторедуцировано.

3.24. Предложение. Авторедуцированный базис Гребнера модуля U определен однозначно с точностью до умножения его элементов на константы из поля κ.

Доказательство. Среди всех авторедуцированных подмножеств модуля U выберем минимальное. Обозначим его A и предположим, что его элементы нормированы так, что все их старшие коэффициенты равны 1. Покажем, что этим условием множество A определено однозначно и что оно является базисом Гребнера модуля U.

Предположим, что $A = \{a_1, \ldots, a_r\}$ и $B = \{b_1, \ldots, b_s\}$ – два множества, удовлетворяющих сформулированным выше условиям. Из условия минимальности следует, что $r=s$ и $Hterm(a_1)=Hterm(b_1)$ для любого i. Предположим, что существует i, для которого $a_1 \neq b_1$. Тогда $A' = \{a_1, \ldots, a_{1-1}, a_1-b_1\}$ – авторедуцированное множество, элементы которого принадлежат U, и ранг которого ниже, чем ранг A, что противоречит минимальности A.

Для доказательства того, что A – базис Гребнера модуля U, воспользуемся условием 3.18.2. Предположим противное, тогда существует элемент $g \in U$, старший терм которого редуцирован отно-

сительно A. Можно предполагать, что он сам также редуцирован относительно A. Рассмотрим множество

$$A' = \{a_1 \in A \mid a_1 <_M g\} \cup \{g\}.$$

Это множество авторедуцировано и его ранг меньше ранга A, что противоречит предположению о минимальности A. □

3.25. Определение. Пусть $B = \{f_1, \ldots, f_s\}$ – G-базис модуля $U \subset P^m$, относительно некоторого упорядочения термов из T_m. Назовем базис B нередуцируемым, если ни для одного i, $1 \le i \le s$, не существует G-представления $f_i = \sum_{j \ne i} b_j f_j$, в противном случае B называем редуцируемым.

3.26. Определение. Назовем G-базис B минимальным, если не существует G-базиса B' модуля U, содержащего менее s элементов.

3.27. Предложение. Понятия минимальности и нередуцируемости G-базисов совпадают. Каждый авторедуцированный G-базис – минимален и каждый минимальный G-базис квазиавторедуцирован, т.е. множество его ведущих термов авторедуцировано (это множество определяется модулем U однозначно).

Доказательство. Очевидно, что редуцируемый G-базис не является минимальным. Таким образом для доказательства предложения достаточно показать, что старшие термы элементов нередуцируемого базиса определены однозначно. Доказательство проходит во многом аналогично доказательству предложения 3.24 и оставляется в качестве упражнения.

3.28. Примеры.

3.28.1. Пусть k – поле и n 0. Если матрица системы линейных многочленов приведена к ступенчатому виду, т.е. нет строк, которые начинаются с одного и того же столбца, то эта система представляет собой базис Гребнера порождаемого ею идеала. Показать, что обратное, в общем случае, неверно, т.е. могут существовать базисы Гребнера векторного подпространства, первые ненулевые элементы различных векторов которых стоят в одном и том же столбце.

3.28.2. Пусть k – поле, n m 1. Множество B является бази-

сом Гребнера порождаемого им идеала тогда и только тогда, когда оно содержит НОД всех своих элементов. Минимальный базис Гребнера в этом случае состоит из одного элемента.

3.28.3. Базис Гребнера в примере 3.19.2 не является минимальным. После удаления из него первого элемента он становится минимальным и даже авторедуцированным.

Понятие базиса Гребнера вводилось нами как задача представления данных для вычислений на ЭВМ. Тем не менее базисы Гребнера имеют многочисленные чисто математические приложения. Некоторые из них рассматриваются Бухбергером в статье [2]. К ним относятся различные результаты о размерности множества решений системы алгебраических уравнений, решение систем линейных уравнений с полиномиальными коэффициентами, в частности, вычисление сизигий полиномиальных идеалов и т.д. В параграфе 5 мы подробно рассмотрим задачу вычисления по базису Гребнера наиболее тонкого инварианта алгебраического многообразия, описывающего его размерностные свойства многочлена Гильберта, и его аналога в дифференциально-разностной алгебре - дифференциально-разностного размерностного многочлена. Сейчас же сформулируем в качестве упражнений несколько приложений базисов Гребнера к теории нелинейных алгебраических уравнений.

3.29. Упражнения

1. Показать, что система алгебранческих уравнений не имеет решений в алгебраическом замыкании поля коэффициентов тогда и только тогда, когда базис Гребнера идеала, порожденного этой системой, содержит константу.

2. Показать, что система алгебраических уравнений из $k[x_1,\ldots,x_n]$ имеет конечное множество решений в алгебраическом замыкании поля коэффициентов тогда и только тогда, когда базис Гребнера идеала, порожденного этой системой, содержит для любого $i \in 1..n$ многочлен со старшим мономом, являющимся степенью x_i.

3.30. Задача. Построить теорию базисов Гребнера для идеалов в кольце многочленов с коэффициентами из кольца Z.

4. Основные алгоритмы вычисления базисов Гребнера в дифференциальных и разностных модулях

Математическая формулировка основных алгоритмов

Сначала основные результаты предыдущего параграфа сформулируем в виде алгоритмов, а во второй части параграфа приведем формулировку алгоритмов, ориентируемую на конкретную реализацию.

Пусть P – кольцо многочленов над полем k от n неизвестных или кольцо дифференциальных (разностных) операторов с n образующими над дифференциальным (разностным) полем. Предположим, что задано отношение порядка $<$ на множестве $\mathbb{N}_0^n$ и упорядочение $<_T$ множества термов T_m, совместное с $<$.

Пусть $G = \{g_1, \ldots, g_s\}$ – конечное множество элементов свободного P-модуля P^m таких, что $\mathrm{Hcoeff}(g_i) = 1$ для всех $g_i \in G$, U – подмодуль модуля P^m, порожденный множеством G, и $f \in P^m$. Прежде всего сформулируем алгоритм частичной редукции f относительно G (определение 3.5)

4.1. АЛГОРИТМ RED(f, s, G)

Арг: f, s, $G = (g_1, \ldots, g_s)$
Рез f

начало
  $q := .\mathrm{false}.$
  цикл пока $q = .\mathrm{false}.$
    $q := .\mathrm{true}.$
    цикл для i **от** 1 **до** s **пока** $q = .\mathrm{true}$
      если $\mathrm{Hterm}(g_i) \le_M \mathrm{Hterm}(f)$
        то $\varphi := \mathrm{Hterm}(f)/\mathrm{Hterm}(g_i)$
          $f := f - \mathrm{Hcoeff}(f)\cdot\varphi\cdot g_i$
          $q := .\mathrm{false}.$
      конец если
    конец цикла
  конец цикла
конец

4-1432

Следующий алгоритм проверяет, является ли данная последовательность элементов G-базисом порождаемого ею подмодуля свободного P-модуля (теорема 3.18.10'). Вспомогательный алгоритм $S(g,h)$ вычисления S-элементов (определение 3.9) приведен после основного алгоритма. В алгоритме используется также обозначение $H(i,j)$ для НОК(g_i, g_j).

4.2. АЛГОРИТМ GROBNER1($s,G,answer$)

Арг: $s,G = (g_1, \ldots, g_s)$
Рез: *answer*

начало
  $B := \{(i,j) : 1 \le i < j \le s \ \ \& \ H(i,j) \ne 0\}$
  $q :=$ .true.
  цикл для каждого (i,j) **из** B **пока** $q =$ .true.
    $S := S(g_i, g_j)$
    RED(S,s,G)
    если $S \ne 0$
      то $q :=$ .false.
    конец если
  конец цикла
  answer q
конец

4.3. АЛГОРИТМ $S(g,h)$

Арг: g,h, Hcoeff(g) = Hcoeff(h) = 1
Рез: S

начало
  $H :=$ НОК(Hterm(g),Hterm(h))
  $\mu := H/\text{Hterm}(g)$
  $\nu := H/\text{Hterm}(h)$
  $S := \mu \cdot g - \nu \cdot h$
конец

Следующий алгоритм представляет собой модификацию алгоритма GROBNER, использующую возможность отбрасывать некоторые пары, не вычисляя соответствующих S-элементов (теорема 3.28.11').

4.4 АЛГОРИТМ GROBNER2(s,G,answer)

Арг: $s,G = (g_1, \ldots, g_s)$
Рез: *answer*

начало
  $B := \{(i,j) : 1 \le i < j \le s \ \& \ H(i,j) \ne 0\}$
  сортировать B по возрастанию $H(i,j)$
  $q :=$ .true.
  цикл для каждого (i,j) **из** B **в порядке возрастания** $H(i,j)$
    пока $q =$ .true.
    если $(\nexists\, l : 1 \le l \le s \ \& \ i \ne l \ne j \ \&$
        $\mathrm{Hterm}(g_l) \le_M H(i,j) \ \& \ (i,l) \in B \ \& \ (l,j) \in B)$
      то $f := S(G_i, G_j)$
        RED(f,s,G)
        если $f \ne 0$
          то $q :=$ .false.
        конец если
    конец если
  конец цикла
  answer $:= q$
конец

Следующие два алгоритма предназначены для пополнения данного множества элементов до G-базиса подмодуля, порожденного этими элементами (простой вариант и оптимизированный).

4.5. АЛГОРИТМ GRBASES1(s,G)

Арг: s,G
Рез: s,G

начало
  $B := \{(i,j) : 1 \le i < j \le s \ \& \ H(i,j) \ne 0\}$
  цикл для каждой пары (i,j) **из** B
    $f := S(g_i, g_j)$
    RED(f,s,G)
    если $f \ne 0$

```
            то s := s+1
               f := f/Hcoeff(f)
               g_s := f
               B := B ∪ {(i,s) : 1 ≤ i < s,   H(i,s) ≠ 0}
            конец если
      конец цикла
конец
```

4.6. АЛГОРИТМ GRBASES2(s,G)

Арг: s, G

Рез: s, G

```
начало.
  B := {(i,j) : 1 ≤ i < j ≤ s  & H(i,j) ≠ 0 }
  сортировать B по возрастанию H(i,j)
  цикл для каждой пары (i,j) из B в порядке возрастания H(i,j)
      если (∄ l : 1≤l≤s & i≠l≠j &
                Hterm(g_l) ≤_M H(i,j) & (i,l) ∈ B & (l,j) ∈ B)
        то f := S(g_i,g_j)
           RED(f,s,G)
           если f ≠ 0
             то s := s+1
                f := f/Hcoeff(f)
                g_s := f
                B := B ∪ {(i,s) : 1 ≤ i < s,   H(i,s) ≠ 0}
                сортировать B по возрастанию H(i,j)
           конец если
      конец если
  конец цикла
конец
```

Прежде чем сформулировать алгоритм построения минимального G-базиса для подмодуля свободного P-модуля, введем тип данных "система", который играет в алгоритме ключевую роль. По определению "система" состоит из конечного множества индексов I, из множества $G = \{g_\alpha : \alpha \in I\}$ элементов рассматриваемого модуля,

упорядоченного по возрастанию ведущих термов, из множества "критических пар" $B = \{(i,j): i,j \in I, i<j, H(i,j) \neq 0\}$, упорядоченных по возрастанию $H(i,j)$, и стека ST для хранения элементов модуля. Предполагаем, что в любой момент множество G квазиавторедуцировано. Кроме того, "система" содержит переменную целого типа t, значение которой совпадает с максимальным значением индекса $\alpha \in I$.

Алгоритм построения минимального G-базиса существенно рекурсивен и его основной вспомогательный алгоритм – "присоединить <элемент> к <системе>"

4.7. АЛГОРИТМ MINBASES(s,G)

Арг: s,G
Рез: s,G

начало
  сформировать пустую систему SYS
  цикл для i **от** 1 **до** s
    присоединить g_i к SYS
    цикл для каждой пары (i,j) **из** B
      в порядке возрастания $H(i,j)$
      если $(\nexists\, l : 1 \le l \le s \;\&\; i \neq l \neq j \;\&$
          $\mathrm{Hterm}(g_l) \le_M H(i,j) \;\&\; (i,l) \in B \;\&\; (l,j) \in B)$
        то $f := S(g_i, g_j)$
          присоединить f к SYS
      все
    конец цикла
  конец цикла
конец

4.8. АЛГОРИТМ присоединить f к SYS

Арг: f,SYS
Рез: SYS

начало
если $f = 0$

Ч²-1432

```
то выход
иначе
  цикл для каждого g_α ∈ G такого, что Hterm (g_α) ≤_т Hterm (f)
      если Hterm (g_α) ≤_н Hterm (f)
        то RED (f,1,g_α)
           присоединить f к SYS
           выход
      все
      t := t+1
      L.добавить t
      G.добавить f
      B.добавить {(α,t), α ∈ L, α < t,   H(α,t) ≠ 0}
      цикл для каждого g_α ∈ G
          такого, что Hterm (g_α) >_т Hterm (f)
          если Hterm (g_α) >_н Hterm (f)
            то RED (g_α,1,f )
               если g_α ≠ 0
                 то ST.добавить g_α
               все
               L удалить α
               G.удалить g_α
               B.удалить пары вида (α,i) и (j,α)
          все
      конец цикла
      цикл для каждого f из ST
          присоединить f к SYS
      конец цикла
  конец цикла
все
конец
```

Предположим теперь, что к – поле частных некоторой области целостности D и образующие модуля нормированы так, что их коэффициенты принадлежат D. В этом случае мы не можем воспользоваться алгоритмом частичной редукции RED в том виде, как он сформулирован выше. Изменим этот алгоритм так, чтобы он мог

быть использован для вычисления G-базиса, коэффициенты элементов которого принадлежат D. Отметим, что модифицированный алгоритм не может быть использован для процесса редукции в общем случае, поскольку он является только нормальным (но не каноническим) симплификатором (см. [4]). Для полиномиальных и дифференциальных модулей алгоритм RED принимает следующий вид:

4.9 АЛГОРИТМ RED(f,s,G)

Арг: f, s, $G = (g_1, \ldots, g_s)$
Рез: f

начало
  q := .false.
  цикл пока q = .false.
    q := .true.
    цикл для i **от** 1 **до** s **пока** q = .true.
      если Hterm(g_1) $\leq_H$ Hterm(f)
        то μ = Hterm(f)/Hterm(g_1)
          f := Hcoeff(g_1)·f − Hcoeff(f)· $\cdot g_1$
          q := .false.
      все
    конец цикла
  конец цикла
конец

В этом случае меняется также алгоритм построния S-элемента.

4.10. АЛГОРИТМ $S(g,h)$

Арг: g,h
Рез: S

начало
  H := НОК(Hterm(g),Hterm(h))
  μ := H/Hterm(g)

```
        := H/Hterm(h)
    S := Hcoeff(h)·μ·g - Hcoeff(g)· ·h
конец
```

Алгоритмы GROBNER1 и GROBNER2 остаются без изменений. В алгоритмах GRBASES1 и GRBASES2 нужно удалить строку

$$f := f/\mathrm{Hcoeff}(f)$$

Если нам нужно вычислить G-базис разностного модуля, то нужно помнить, что умножение в кольце разностных операторов некоммутативно и остается некоммутативным в ассоциированном градуированном кольце. Следовательно, нужно применить соответствующие операторы к $\mathrm{Hcoeff}(g_1)$.

4.11. АЛГОРИТМ RED(f,s,G)

Арг: f, s, $G = (g_1, \ldots, g_s)$
Рез: f

```
начало
    q := .false.
    цикл пока q = .false.
        q := .true.
        цикл для i от 1 до s пока q = .true.
            если Hterm(g_1) ≤_M Hterm(f)
              то    := Hterm(f)/Hterm(g_1)
                 f :=  (Hcoeff(g_1))·f - Hcoeff(f)· ·g_1
                 q := .false.
            все
        конец цикла
    конец цикла
конец
```

Нужно изменить также алгоритм построения S-элемента.

4.12. АЛГОРИТМ $S(g,h)$

Арг: g,h

Рез: S

начало

H := НОК(Hterm(g),Hterm(h))

μ := H/Hterm(g)

:= H/Hterm(h)

S := (Hcoeff(h))·μ·g – μ(Hcoeff(g))· ·h

конец

4.13. ПРИМЕРЫ

4.13.1. Систему линейных уравнений можно рассматривать как модуль над кольцом многочленов от нулевого количества переменных либо как идеал в кольце многочленов от n переменных, порожденный линейными многочленами. В обоих случаях применим метод базисов Гребнера. Частичное редуцирование заключается в приведении системы к трапецендальному виду, полное – к "диагональному" виду. Алгоритм пополнения не добавляет новых элементов.

4.13.2. В кольце $\kappa[X]$ редуцирование сводится к нахождению остатка от деления многочлена на многочлен, а построение редуцированного базиса Гребнера – к нахождению НОД многочленов. Основные сложности связаны здесь с точной арифметикой в поле (или кольце) коэффициентов. Погрешности округления сильно влияют на результат, а точные вычисления приводят к "разбуханию" коэффициентов. Например, НОД многочленов x^3-8 и $(1/3)x^2-(4/3)$ равен $(1/3)x-(2/3)$, в то время как НОД многочленов x^3-8 и $0.333333x^2-1.33333$ равен 0.000001. При вычислении НОД многочленов $x^8+x^6-3x^4-3x^3+8x^2+2x-5$ и $3x^8+5x^4-4x^2-9x+21$ можно производить вычисления либо в поле $\mathbb{Q}$, либо в кольце $\mathbb{Z}$. В первом случае приходится иметь дело с рациональными числами, числитель и знаменатель которых содержат до 10 десятичных цифр, во втором – с целыми числами, которые содержат до 35 десятичных цифр. Подробно этот пример разобран в монографии [22].

Следует отметить, что окончательный ответ в данном случае очень короткий – исходные многочлены взаимно просты, поэтому базис Гребнера порождаемого ими идеала состоит из единственного элемента 1. Здесь налицо "разбухание" промежуточной информации.

Проверить взаимную простоту выписанных многочленов можно, выполняя вычисления по модулю некоторого простого числа, например, 5, что значительно облегчит вычисления в данном примере.

В общем случае вычисления базисов Гребнера также приходится бороться с разбуханием промежуточных коэффициентов. Один из возможных способов состоит в использовании многомодульной арифметики (см. 4.14)

4.13.3. Кривые Маколея. В предыдущих примерах базисы Гребнера содержали небольшое количество элементов. Следующий пример показывает, что их число может быть как угодно большим. Пусть r – натуральное число. Рассмотрим кольцо $k[x,y,z]$ и идеал, порожденный 2-мя образующими: $I=(xy-z, x^{r-1}-y)$. Применяя упорядочение по степени, затем обратное лексикографическое, получим базис Гребнера

$$\begin{aligned} f_1 &= xy - z, \\ f_i &= x^{r+1-i}z^{i-2} - y^{i-1}, \qquad i \in 2..r-1, \\ f_r &= y^{r-1} - xz^{r-2} \end{aligned}$$

Другие примеры содержатся в параграфе 6 при описании применения комплекса программ.

4.14. Использование многомодульной арифметики для вычисления базисов Гребнера

Рассмотрим более подробно задачу вычисления базисов Гребнера полиномиальных идеалов для многочленов с рациональными коэффициентами. Как отмечалось выше, все вычисления можно при этом производить над многочленами с целыми коэффициентами. Однако нужно учитывать, что эти числа могут быть весьма большими, особенно на промежуточных этапах вычислений. Для ускорения работы алгоритмов Сасаки предложил использовать многомодульную арифметику. При этом можно пользоваться как вероятностными, так и детерминированными алгоритмами. Кратко опишем методы, предлагаемые Сасаки.

4.14.1. Простой вероятностный алгоритм состоит в том, чтобы выбрать некоторое векторное основание счисления $(p_1,\ldots, p_k)$ и выполнять вычисления базиса Гребнера по модулю этого основания. После окончания вычислений перейти от системы вычетов по

векторному основанию к вычетам по модулю $M = p_1 \cdot \ldots \cdot p_k$, пользуясь симметричной (относительно 0) системой вычетов. Если M достаточно велико, то мы получим искомый базис Гребнера. К сожалению, не удается получить приемлимых теоретических оценок величины M, для того чтобы превратить этот алгоритм в детерминированный. Для повышения вероятности того, что полученная система многочленов является искомым базисом Гребнера, можно пользоваться различными тестами.

4.14.2. Прежде всего, вычисляя минимальный базис Гребнера, можно проверить, что исходная система многочленов редуцируется к нулю по модулю полученных соотношений. Далее, можно провести вычисления по модулю еще одного простого числа q и получить базис Гребнера по модулю $M \cdot q$. Если этот базис совпадает с базисом, полученным по модулю M, то вероятность правильного ответа достаточно велика. Наконец, проверка того, что выписанная система многочленов является базисом Гребнера, существенно проще, чем построение базиса Гребнера по некоторой системе образующих идеала. Таким образом можно получить детерминированный алгоритм.

4.15. Задачи

4.15.1. Оценить сложность описанных алгоритмов.

4.15.2. Сформулировать вероятностный и детерминированный алгоритмы вычисления базисов Гребнера полиномиальных идеалов с рациональными коэффициентами на основе многомодульной арифметики поля рациональных чисел.

4.15.3. Реализовать алгоритм вычисления базиса Гребнера на одном из алгоритмических языков, с использованием структур данных, описанных в разделе 1.10.

5. ХАРАКТЕРИСТИЧЕСКИЕ МНОГОЧЛЕНЫ ГИЛЬБЕРТА

В этом параграфе мы будем придерживаться следующих обозначений: $R=R(m)=\mathbb{N}_0^m$, где $\mathbb{N}$ – множество натуральных чисел с нулем, на множестве R вводится отношение частичного порядка $\geq$ так, что если $r=(r_1,\ldots,r_m) \in R$ и $s=(s_1,\ldots,s_m) \in R$, то $r \geq s$ <==> для i выполняется неравенство $r_1 \geq s_1$. Для вектора $r \in R$ через $|r|$ будем обозначать сумму координат этого вектора.

Задача вычисления многочлена Гильберта возникает как в коммутативной алгебре и алгебраической геометрии при описании размерности алгебраического многообразия, так и в дифференциально-разностной алгебре при вычислении дифференциально-разностных размерностных многочленов.

Приведем определение функции Гильберта и многочлена Гильберта для полиномиальных идеалов.

Пусть K – поле, $R=K[x_1, \ldots, x_m]$ – кольцо многочленов, $J \subset R$ – идеал кольца R. На кольце R рассматриваем естественную фильтрацию по степеням многочленов: $R_i = \{f \in R \mid \deg f \leq i\}$. Фильтрация кольца R индуцирует фильтрацию идеала J и фильтрацию фактор-кольца $M=R/J$, рассматриваемого как R-модуль: $M_i = \mathrm{Im}(R_i)$ при естественном гомоморизме $R \longrightarrow M \longrightarrow 0$. Для любого неотрицательного i M_i является векторным пространством над полем K, поэтому на множестве неотрицательных целых чисел определена функция $\varphi_J(s) = \dim_K M_s$.

5.1. Определение. Функция $\varphi_J(s) = \dim_K (R/J)_s$ называется характеристической функцией Гильберта идеала J в кольце R.

Гильберт доказал, что для любого полиномиального идеала существует многочлен $\omega_J(s)$ такой, что для всех достаточно больших натуральных чисел $\omega_J(s)=\varphi_J(s)$. Этот результат будет получен ниже при описании алгоритма вычисления многочлена Гильберта.

5.2. Определение. Многочленом Гильберта идеала J в кольце R называется многочлен $\omega_J(s)$ от одной переменной s такой, что для всех достаточно больших натуральных чисел

$$\omega_J(s) = \dim_K (R/J)_s.$$

Определения 5.1 и 5.2 непосредственно обобщаются на случай подмодулей свободных полиномиальных и дифференциально-разнос-

тных модулей.

Использование техники базисов Гребнера в полиномиальном случае и характеристических множеств в дифференциально-разностном редуцирует задачу вычисления многочлена Гильберта к следующей комбинаторной задаче:

5.3. Для конечного множества точек $E=\{e_1, \ldots, e_n\}$ в R обозначим замыкание E в R через $\bar{E}$, т.е. $\bar{E} = \{r \in R \mid \exists i,\ r \geq e_i\}$. Через V обозначим дополнение к $\bar{E}$ в R. Требуется для любого $s \in \mathbb{N}$ определить количество точек множества V, сумма координат которых не превосходит s.

Функция, описывающая зависимость колнчества точек от s, называется функцией Гильберта множества V. Для почти всех натуральных чисел ее значения совпадают со значениями некоторого целозначного многочлена, называемого многочленом Гильберта. В дальнейшем будем представлять себе множество E в виде строк некоторой матрицы, которую будем обозначать той же буквой, а соответствующий многочлен Гильберта будем обозначать $\omega(E,s)$, либо $\omega(n,m,E,s)$, когда нужно будет подчеркнуть зависнмость многочлена Гильберта от размерности пространства m и мощности n множества E (заметим, что может быть $n=0$). Наряду с этими обозначениями будем пользоваться также обозначением $\omega(V,s)$, если нужно подчеркнуть зависнмость многочлена Гильберта от множества V.

Отметим следующие свойства многочлена Гильберта.

5.4. Упражнения

5.4.1. $\omega(n,m,E,s) = \omega(n,m,E',s)$, если матрица E' получается нз E перестановкой строк или столбцов.

5.4.2. Пусть в матрице E p-я строка мажорирует q-ю, т.е. $e_{pj} \geq e_{qj}$ для всех j. Тогда $\omega(n,m,E,s) = \omega(n-1,m,E',s)$, где матрица E' получается из E удалением p-й строки.

5.4.3. Пусть $n=0$. Показать что

$$\omega(0,m,,s) = \binom{s+m}{m}.$$

5.4.4. Пусть матрица E состоит из одной строки e, $|e|$ –

сумма ее элементов. Показать, что

$$\omega(1,m,e,s) = \binom{s+m}{m} - \binom{s+m-|e|}{m}.$$

5.5. Предложение. Многочлен Гильберта можно вычислять по формуле

$$\omega(n,m,E,s) = \sum_{t\in T}(-1)^{|t|}\binom{s+m-\mu(t,E)}{m}, \qquad (5.1)$$

где $T = \{ t=(t_1, \ldots, t_n) \mid t_i \in \{0;1\} \ \forall i\}$;

$$|t| = \sum_{i=1}^{n} t_i; \quad \mu(t,E) = \sum_{i=1}^{m}\Big(\max_{j=1}^{n} t_j \cdot e_{ji}\Big).$$

Если n=0, то $(n,m,E,s) = \binom{s+m}{m}$

Доказательство может быть получено путем комбинаторых рассуждений, основанных на подстчете кратности, с которой конкретная точка входит в указанную сумму, и оставляется читателю в качестве задачи.

Переформулируем полученный результат в виде алгоритма.

5.6. Алгоритм Hpol1(m,n,E,w)

Арг: $m \in \mathbb{N}$, $n \in \mathbb{Z}+$, E – матрица размера $m \cdot n$ элементов типа $\mathbb{Z}+$

Рез: w – целозначный многочлен Гильберта матрицы E

Переменные:

v – вектор типа $\mathbb{Z}+$ с индексом $1..m$

S – вектор типа "да/нет" с индексом $1..m$

w' – целозначный многочлен

начало

если n=0

то $w := \binom{s+m}{m}$

иначе

$w := 0$

цикл для каждого подмножества X **множества** $1..n$

v:=0; S:=0

цикл для j **от** 1 **до** n

```
            если j ∈ X
              то S_j := -S_j
                 цикл для i от 1 до n
                        v_1 := max(v_1, e_j1)
                 конец цикла
            все
        конец цикла
    конец цикла
все
конец
```

Достоинством этого алгоритма является простота и малая пространственная сложность, недостатком – экспоненциальная временная сложность по n. Непосредственное вычисление многочлена Гильберта таким образом требует не менее $\Theta(2^n - m)$ операций и может выполняться только при небольших n.

Многочлен Гильберта принято записывать в виде

$$\omega(s) = \sum_{i=0}^{m} a_i \binom{s+i}{i},$$

который мы будем в дальнейшем называть стандартным. Приведение многочленов $\binom{s+m-t}{m}$ к стандартному виду основано на формуле

$$\binom{s+r-1}{q} = \binom{s+r}{q} - \binom{s+r-1}{q}. \qquad (5.2)$$

Формулу (5.2) можно применять к каждому слагаемому формулы (5.1), а можно сгруппировать слагаемые с одинаковым значением t и переписать (5.1) в виде

$$\omega(s) = \sum_{q \in Z} \sum_{k=0}^{m} (-1)^k \sum_{\sigma \in A(k,n) \mid t_\sigma = q} \binom{s+m-q}{m} =$$

$$= \sum_{q \in Z} b_q \binom{s+m-q}{m},$$

где $b_q = \sum_{k=0}^{m} \sum_{\sigma \in A(k,n) \mid t_\sigma = q} (-1)^k$

Обозначим через $\sigma(E)$ вектор e^{σ}, где $e^{\sigma}(i) = \max\limits_{j \in \sigma} e_{j1}$, и будем группировать слагаемые с одинаковыми значениями $\sigma(E)$.

Тогда $t_{\sigma(E)} = t_{\sigma} = \sum\limits_{j=1}^{m} e^{\sigma}(j)$ и

$$\omega(s) = \sum_{q \in \mathbb{N}_0^m} \sum_{k=0}^{m} (-1)^k \sum_{\sigma \in A(k,n) \mid \sigma(E)=q} \binom{s+m-t_q}{m} =$$

$$= \sum_{q \in \mathbb{N}_0^m} \mu_q \binom{s+m-t_q}{m},$$

где $$\mu_q = \sum_{k=0}^{m} (-1)^k \sum_{\sigma \in A(k,n) \mid \sigma(E)=q} (-1)^k.$$

При этом суммирование требуется производить только по "допустимым" векторам q, т.е. таким, которые могут быть представлены в виде $\sigma(E)$. Необходимым условием этого является наличие значения q_1 в l-м столбце матрицы E. Следовательно, количество допустимых векторов не превосходит n^m. Поскольку в формуле (5.1) фигурирует 2^n слагаемых, при достаточно больших n различные сочетания σ будут давать одинаковые векторы $\sigma(E)$.

Пусть τ' и τ'' – два подмножества множества $1..n$ такие, что $\tau'(E)=\tau''(E)$ и τ'' получается из τ' добавлением одного элемента, не принадлежащего τ'. Тогда слагаемые, соответствующие сочетаниям τ' и τ'' в формуле (5.1) взаимно уничтожатся, то же самое произойдет со всеми слагаемыми, соответствующими подмножествам σ' и σ'', которые получаются из τ' и τ'' добавлением одних и тех же элементов, не принадлежащих τ''. На этом замечании основан следующий алгоритм вычисления многочлена Гильберта.

5.7. Алгоритм Hpol2(m,n,E,w)

Арг: $m \in \mathbb{N}$, $n \in \mathbb{Z}+$, E – матрица размера $m \cdot n$ элементов типа $\mathbb{Z}+$

Рез: T – множество векторов типа $\mathbb{Z}+$ с индексом $1..m$

μ – вектор типа $\mathbb{Z}$ с индексом из T

Переменные:

T' – множество векторов типа $\mathbb{Z}+$ с индексом $1..m$

μ' – вектор типа **Z** с индексом из T
τ – вектор типа **Z+** с индексом $1..m$

```
начало
  T := {(0, ..., 0)}
  μ((0, ..., 0)) := 1
  цикл для i от 1 до n
      T' := T
      цикл для каждого σ из T'
          μ'(σ) := μ(σ)
      конец цикла
      цикл для каждого σ из T'
          цикл для j от 1 до m
              τ(j) := max_{i=1}^{n} (σ(j), e_{ij})
          конец цикла
          если τ ∈ T
            то μ(τ) := μ(τ) - μ'(σ)
            иначе
                T.добавить τ
                μ(τ) := - μ'(τ)
          все
      конец цикла
  конец цикла
конец
```

Асимптотическая (по n) сложность этого алгоритма составляет $n^{m+1} \log n$. Данный алгоритм, а также следующий, предложены в работе [33].

Следующий алгоритм вычисления функции Гильберта также основан на вычислении коэффициентов μ_τ и использует рекуррентную формулу $\mu_\tau(K,m) = \mu_\tau(K\backslash k,m) - \mu_\tau(K\backslash k,J)$, в которой K – множество строк, мажорирующихся вектором τ, k – очередная строка, J – множество столбцов, в которых координаты векторов τ и k совпадают.

5-1439

5.8. Алгоритм Hpol3(m,n,E,w)

Арг: $m \in \mathbb{N}$, $n \in \mathbb{Z}+$, E – матрица размера $m \cdot n$ элементов типа $\mathbb{Z}+$

Рез: T – множество "допустимых" векторов

μ – вектор типа $\mathbb{Z}$ с индексом из T

Переменные:

J – множество элементов типа $1..m$

K – множество векторов типа $\mathbb{Z}+$ с индексом $1..m$

w – вектор типа ± 1 с индексом из T

начало

сформировать множество T

цикл для каждого τ **из** T

$\mu(\tau) := 0$

$w(\tau) := 1$

$J := 1..m$

$K := \{e_1 \mid e_1 \leq \}$

NEXTINDEX (J,K,τ,w,μ)

конец цикла

конец

5.9. Алгоритм NEXTINDEX(J,K,τ,w,μ)

Арг: J – множество элементов типа $1..m$

K – множество векторов типа $\mathbb{Z}+$ с индексом $1..m$

τ – вектор типа $\mathbb{Z}+$ с индексом $1..m$

w – вектор типа ± 1 с индексом из T

μ – вектор типа $\mathbb{Z}$ с индексом из T

Рез: J – множество элементов типа $1..m$

w – вектор типа ± 1 с индексом из T

μ – вектор типа $\mathbb{Z}$ с индексом из T

Переменные:

J' – множество элементов типа $1..m$

начало

цикл для каждого k **из** K **такого, что** $k(j_1) = \tau(j_1)$, **где**

j_1 – первый элемент множества J

$w(\tau) := -w(\tau)$

```
        J' := {j∈J | k_j < τ_j}
        J.удалить J'
        выбор
            при K = 0  и J = ∅
                μ(τ) := μ(τ) + ω(τ)
            - - -
            при K = 0  и J ≠ ∅
                NEXTINDEX(J, K, τ, ω, μ)
            - - -
        все
        J.добавить J'
    конец цикла
конец
```

Время работы полученного алгоритма составляет $O(n^{m+1})$ [33].

Далее в данном параграфе выводится алгоритм вычисления многочлена Гильберта, основанный на добавлении к множеству E векторов по некоторому правилу, предложенный в работе [8].

Рассмотрим, как меняется многочлен Гильберта при добавлении к множеству E некоторого вектора v. Положим $R_v = \{s \in R \mid s \geq v\}$ и $V_v = \{w \in R \mid w+v \in V\}$.

5.10. Предложение. Многочлен Гильберта $\omega(V,s)$ множества V равен сумме многочлена Гильберта $\omega(V \setminus R_v, s)$ множества $V \setminus R_v$ и многочлена Гильберта $\omega(V_v, s-|v|)$ множества V_v, аргумент которого сдвинут на сумму координат $|v|$ вектора v.

Доказательство. Точки множества V_v, сумма координат которых не превосходит $s-|v|$, находятся во взаимно однозначном соответствии с точками множества $V \cap R_v$, сумма координат которых не превосходит s.

5.11. Следствие. Пусть E – матрица размера $n-m$ и v – вектор. Тогда $\omega(n,m,E,s)=\omega(n+1,m,E',s)+\omega(n,m,E'',s-|v|)$, где матрица E' получается из E добавлением еще одной строки v, а матрица E'' получается вычитанием вектора v из каждой строки матрицы E и заменой отрицательных элементов нулями.

Заметим, что в матрице E' обычно появляются "лишние" строки, которые можно отбросить, и количество строк в ней не увели-

чивается, а уменьшается. В частности, выбирая вектор v, координаты которого равны минимальным значениям элементов в соответствующем столбце матрицы E, получаем в матрице E' строку, которая мажорируется всеми остальными, значит для вычисления многочлена Гильберта достаточно оставить в матрице E' только строку v. Формула для вычисления многочлена Гильберта в этом случае хорошо известна. Выбирая в качестве v строку (1, 0,...,0), можем при формировании матрицы E' отбросить все строки матрицы E, которые содержат в первом столбце ненулевой элемент. Точно так же их можно отбросить, выбирая v=(k, 0,...,0), где k – минимальное ненулевое значение в первом столбце матрицы E.

Различные способы выбора векторов v дают различные алгоритмы вычисления многочленов Гильберта.

Алгоритм, основанный на выборе вектора v = (1, 0, ..., 0), если матрица E содержит ненулевые элементы в первом столбце

Тогда матрицы E' и E'' вычисляются следующим образом: E' состоит из строк матрицы E, содержащих 0 в первом столбце, и строки (1,0,...,0), а E'' получается вычитанием 1 из ненулевых элементов первого столбца матрицы E.

5.12. Лемма. Пусть

$$E = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & E_\alpha & \\ 0 & & & \end{bmatrix}$$

матрица, где e_{11}=1, а все остальные элементы первого столбца и E_α получается из E вычеркиванием первого столбца и первой строки.

Доказательство. Отображение $(0, e_2, \ldots, e_m) \dashrightarrow (e_2,$

$\ldots, e_m$) устанавливает взаимно однозначное соответствие между m–мерными векторами, не мажорирующими ни одной строки матрицы E, и $(m-1)$–мерными векторами, не мажорирующими ни одной строки матрицы E_α. Сумма координат при этом отображении не меняется. □

Введем на кольце $\mathbb{Q}[x]$ операторы Δ_t и Δ^{-1}.

$\Delta_t\ (p(x)) = p(x) - p(x-t)$.

Δ^{-1} определим на биномиальных коэффициентах следующим образом:

$$\Delta^{-1}\ \binom{x+i}{i} = \binom{x+i+1}{i+1}.$$

На все кольцо $\mathbb{Q}[x]$ распространим оператор Δ^{-1} по линейности.

Легко видеть, что $\Delta_t \cdot \Delta^{-1}(p(x)) = p(x) + p(x-1) + \ldots + p(x-t+1)$. В частности, $\Delta_1 \cdot \Delta^{-1}(p(x)) = p(x)$.

k–кратным применением леммы 5.12, получаем следующий результат:

5.13. Лемма. Пусть

$$E = \begin{bmatrix} k & 0 & & 0 \\ 0 & & & \\ \vdots & & E_\alpha & \\ 0 & & & \end{bmatrix}$$

матрица, где e_{11}=k, а все остальные элементы первого столбца и первой строки равны нулю. Тогда

$$\omega(n,m,E,s) = \Delta_K \cdot \Delta^{-1}\ \omega(n-1,m-1,E_\alpha,s).$$

Прежде чем перейти к общему случаю, рассмотрим случай, когда матрица E содержит не более двух столбцов, т.е. m=1 или m=2.

Если m=1, то по предложению 1 можно считать, что матрица E состоит из единственного элемента e, тогда

5 1- 1432

$$\omega(E,s) = \binom{s+1}{1} - \binom{s+1-e}{1} = e.$$

Пусть $m=2$. По упражнениям 5.4.1 и 5.4.2 можно считать, что элементы первого столбца возрастают, а второго – убывают, т.е. $e_{11} < e_{21} < \dots < e_{n1}$ и $e_{12} > e_{22} > \dots > e_{n2}$. Предположим, что $e_{11} = e_{n2} = 0$.

5.14. Предложение. Пусть

$$E = \begin{bmatrix} e_{11} & e_{12} \\ \cdot & \cdot \\ \cdot & \cdot \\ e_{n1} & e_{n2} \end{bmatrix}$$

причем $0 = e_{11} < e_{21} < \dots < e_{n1}$ и $e_{12} > e_{22} > \dots > e_{n2} = 0$.

Тогда $\deg \omega(E,s) = 0$ и $\omega(E,s) = \omega(E) = \sum_{i=1}^{n-1} (e_{i+1,1} - e_{11}) \cdot e_{12}$.

Доказательство. Случай $n=1$ тривиален.

Пусть $n > 1$. Тогда $\omega(E,s) = \omega(E',s) + \omega(E'',s-e_{21}) = \Delta_{(e_{21}-e_{11})}\, \Delta^{-1} \omega(E'_1,s) + \omega(E'',s-e_{21})$, где $E'_1 = (e_{12})$ и

$$E'' = \begin{bmatrix} 0 & e_{12} \\ 0 & e_{22} \\ e_{31}-e_{21} & e_{32} \\ \cdot & \\ e_{n1}-e_{21} & e_{n2} \end{bmatrix}$$

Выше показано, что $\omega(E'_1) = e_{12}$, значит,

$$\Delta_{(e_{21}-e_{11})}\, \Delta\ \omega(E'_1,s) = (e_{21} - e_{11}) \cdot e_{12}.$$

По упражнению 5.4.2 первую строку матрицы E'' можно удалить. Остается воспользоваться предположением индукции.

Таким образом получаем алгоритм вычисления многочлена Гильберта при m=2.

5.15. Алгоритм Hilbpol2(N, E, w)

Арг: N – натуральное число

E(N,2) – матрица из неотрицательных целых элементов.

Рез: $w(x)=w(N,2,E,x)$ – многочлен Гильберта матрицы E.

Начало

E.упорядочить строки по возрастанию первого элемента, удаляя лишние

n := количество строк получившейся матрицы E

$v_1 := e_{11}$; $v_2 := e_{n2}$

цикл для i **от** 1 **до** n

$e_{11} := e_{11} - v_1$; $e_{12} := e_{12} - v_2$

конец цикла

$$w(x) := \binom{x+2}{2} - \binom{x+2-v_1-v_2}{2} + \sum_{i=1}^{n-1} (e_{i+1,1} - e_{11}) \cdot e_{12}$$

конец

Сложность предложенного алгоритма не превосходит $\mathbb{O}(N \log N)$, поскольку такого количества операций достаточно для сортировки, отбрасывание лишних строк можно производить одновременно с сортировкой. Остальные действия требуют $\mathbb{O}(N)$ вычислений.

5.16. Замечание. Предложенный алгоритм без увеличения асимптотической сложности обобщается на случай, когда $m > 2$, но матрица E содержит только два ненулевых столбца.

Предлагаемый алгоритм вычисления многочлена Гильберта подразделяется на два этапа. Первым шагом выбираем минимальное значение в каждом столбце матрицы E и осуществляем сдвиг на получившийся вектор. Ниже будет показано, что многочлен Гильберта матрицы, содержащей в каждом столбце по нулевому элементу (назовем такую матрицу нормированой), имеет степень не выше m–2. При вычислении многочлена Гильберта нормированой матрицы в первом ненулевом столбце выбирается минимальный ненулевой элемент k и осуществляется сдвиг на вектор (k, 0, ..., 0).

5.17. Алгоритм Hilbpol($n,m,E,$)

Арг: n – неотрицательное целое;
m – натуральное число;
E – матрица размера $n \cdot m$ элементов типа Z+

Рез: $w(x)$ – многочлен Гильберта матрицы E.

Начало
. **выбор**
. . **при** n=0 ==> $w(x) := \binom{x+m}{m}$
при n=1 ==> $w(x) := \binom{x+m}{m} - \binom{x+m-|e|}{m}$
. . **иначе** ==>
. . $v := (v_1, \ldots, v_m)$; $v_1 := \min_{j=1}^{n} e_{j1}$
цикл для i **от** 1 **до** m; **цикл для** j **от** 1 **до** n
. . $e_{j1} := e_{j1} - v_1$ **конец цикла**
конец цикла
. Hilbpoln($m,n,E,$)
. $(x) := (x-|v|) + \binom{x+m}{m} - \binom{x+m-|v|}{m}$
. **конец выбора**
конец

Основной командой этого алгоритма является обращение к программе нахождения многочлена Гильберта нормированной матрицы Hilbpoln. Ниже описывается алгоритм Hilbpoln.

5.18. Алгоритм Hilbpoln(n,m,E,w)

Арг: n – неотрицательное целое;
m – натуральное число, $m \geq 2$;
E – матрица размера $n \cdot m$ элементов типа Z+, первый столбец которой содержит как нулевые, так и ненулевые элементы.

Рез: $w(x)$ – многочлен Гильберта матрицы E.

Переменные: E_0 – матрица, размеры которой не превосходят $n \cdot m$.
w_0 – многочлен
N_s – целое, текущее значение первой координаты
N_r – следующее значение первой координаты

Начало
если m=2
то Hilbpol2 (n, E, w)
иначе
$w(x)$:= 0
N_s := 0
E_0 := 0
цикл для каждого ненулевого значения e_{11} **в порядке возрастания**
N_r := e_{11}
E_0.добавить последовательность строк {(e_{j2}, ..., e_{jm})| e_{j1}= N_s}
N_0 := количество заполненных строк в матрице E_0.
Hilbpol(m-1,N_0,E_0,w_0)
$w_0(x)$:= $\Delta_{(Nr-Ns)}\Delta^{-1}$ $w_0(x)$
$w(x)$:= $w(x)$ + $w_0(x$-Ns)
N_s := N_r
конец цикла
$E0$.добавить последовательность строк {(e_{j2}, ..., e_{jm})| e_{j1}= N_s}
E_0.добавить нулевую координату в конце
Hilbpol(m,n,E_0,w_0)
$w(x)$:= $w(x)$ + $w(x$-N_s)
конец если
конец

Предложенный алгоритм редуцирует задачу с матрицей размера $n \cdot m$ не более чем к n задачам, каждая из которых содержит m–1 ненулевой столбец и не более n строк. Таким образом получаем не более n^{m-2} двумерных задач. При этом задачи с малым количеством строк могут обрабатываться по другим алгоритмам.

При реализации алгоритма на ЭВМ в предписании:

E_0. добавить последовательность строк $\{(e_{j2}, \ldots, e_{jm}) \mid e_{j1} = N_s\}$ целесообразно предусмотреть упорядочение строк получающейся матрицы по возрастанию элементов первого столбца и удаление "лишних" строк. Упорядочение не меняет оценку теоретической сложности алгоритма, с удалением лишних строк теоретическая сложность алгоритма получается не хуже, чем $O(n^m)$, а в практических задачах этот алгоритм работает, как правило, быстрее.

Сформулированный алгоритм проиллюстрируем на примере кривых Маколея (пример 4.13.3), которые использовались для оценки скорости вычисления функции Гильберта в работе [33].

5.19. Пример. Пусть идеал J порожден многочленами

$$f_1 = xy - z,$$
$$f_i = x^{r+1-i}z^{i-2} - y^{i-1}, \qquad i \in 2..r-1,$$
$$f_r = y^{r-1} - xz^{r-2}.$$

Выписанные многочлены образуют базис Гребнера идеала J для упорядочения с учетом степени и лексикографического относительно переменных $y > x > z$, либо обратного лексикографического для переменных $x < y < z$ (доказательство этого утверждения оставляется читателю в качестве упражнения). Функция Hterm выделяет у многочленов f_i, $i \in 1..r$, первое слагаемое. Матрица E будет иметь вид

$$E = \begin{bmatrix} 1 & 1 & 0 \\ 0 & r-1 & 0 \\ 0 & r-2 & 0 \\ \cdot & \cdot & \\ 0 & 2 & r-3 \\ r-1 & 0 & 0 \end{bmatrix}$$

Алгоритм 5.18 утверждает, что

$$\omega(E,s) = \omega'(s) + (\Delta_{r-2}\,\Delta^{-1}\omega'')(s-1) + \hat{\omega}(s-r+1),$$

где

$\omega'(s)$, $\omega''(s)$ и $\hat{\omega}(s)$ – многочлены Гильберта соответственно матриц

$$E' = \begin{bmatrix} r-1 & 0 \\ r-2 & 0 \\ . & . \\ 2 & r-3 \\ 0 & 0 \end{bmatrix}, \quad E'' = \begin{bmatrix} 1 & 0 \\ r-1 & 0 \\ . & . \\ . & . \\ 2 & r-3 \end{bmatrix},$$

$$\hat{E} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & r-1 & 0 \\ 0 & r-2 & 0 \\ . & . & \\ 0 & 2 & r-3 \\ 0 & 0 & 0 \end{bmatrix}$$

Многочлен $\omega'(s)$ вычисляем по алгоритму 5.15:

$$\omega'(s) = \binom{s+2}{2} - \binom{s+1-2}{2} + \sum_{i=0}^{r-1} 1\cdot(r-s-i) =$$

$$= \frac{(s+2)(s+1)}{2} - \frac{s(s-1)}{2} - \sum_{i=1}^{r-3} i = 2s + 1 + \frac{(r-s)(r-2)}{2}.$$

В матрице E'' первая строка мажорируется всеми остальными, следовательно, их можно удалить и

$$\omega''(s) = \binom{s+2}{2} - \binom{s+1}{2} = \binom{s+1}{1}.$$

Применяя оператор $\Delta_{r-2}\Delta^{-1}$, получим

$$(\Delta_{r-2}\Delta^{-1}\omega'')(s-1) = \binom{s-1+2}{2} - \binom{s-1+2-r}{2} =$$

$$= \frac{(s+1)\cdot s}{2} - \frac{(s+3-r)(s+2-r)}{2} = (r-2)\cdot s - \frac{(r-3)(r-2)}{2}.$$

Наконец, матрица $\hat{E}$ содержит нулевую строку, следовательно, $\hat{\omega}(s) = 0$.

Итак,

$$\omega(s) = 2s + 1 + \frac{(r-3)(r-2)}{2} - (r-2)\cdot s - \frac{(r-3)(r-2)}{2} = rs + 1.$$

Приведем пример еще одной стратегии выбора очередного вектора, которая часто дает хорошие результаты, особенно при ручном счете.

5.20. Пример. Пусть p – натуральное число и матрица E имеет вид

$$E = \begin{bmatrix} p & p & 0 \\ p-1 & p-1 & 2 \\ \cdot & \cdot & \\ \cdot & \cdot & \\ 1 & 1 & 2p-2 \\ 0 & 0 & 2p \end{bmatrix}$$

Будем выбирать в качестве вектора v строку матрицы E с максимальным значением 1-й координаты и пользоваться следствием 5.11.

Тогда $\omega(s) = \omega'(s) - \omega''(s-2p)$, где

$$E' = \begin{bmatrix} p-1 & p-1 & 2 \\ \cdot & & \\ \cdot & & \\ 1 & 1 & 2p-2 \\ 0 & 0 & 2 \end{bmatrix} \qquad E'' = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 4 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ 0 & 0 & 2p \end{bmatrix}$$

При этом $\omega(E'',s) = \omega(\hat{E},s)$, где $\hat{E} = (0, 0, 2)$.

Применяя этот же алгоритм к матрице E', по индукции получим

$\omega(E,s) = \omega(E_p,s) - p\cdot\omega(\hat{E},s-2p)$, где $E_p = (0, 0, 2p)$.

Следовательно,

$$\omega(E,s) = \binom{s+3}{3} - \binom{s+3-2p}{3} - p\cdot\binom{s+3}{3} + p\cdot\binom{s+3-2p}{3} =$$

$$= (2p^2 + 2p)(s+1) - p\,\frac{10p^2 + 3p - 1}{3}.$$

Широко используемый в дифференциальной алгебре дифференциальный размерностный многочлен не является дифференциальным бирациональным инвариантом расширения. Такими инвариантами явля-

ются, однако, степень этого многочлена и его старший коэффициент, так называемые дифференциальный тип расширения и типовая дифференциальная размерность. Следующее предложение дает алгоритм вычисления степени многочлена Гильберта, не требующий вычисления самого многочлена.

5.21. Предложение.

a) $\deg\ \omega(n,m,E,x) \le m$;

b) $\deg\ \omega(n,m,E,x) = m$ тогда и только тогда, когда $n=0$; в этом случае

$$\omega(n,m,E,x) = \binom{x+m}{m};$$

c) $\deg\ \omega(n,m,E,x) < \tau$ тогда и только тогда, когда для любого подмножества I, состоящего из $m-\tau$ элементов множества $(1,\ldots,\ m)$, существует строка e_I матрицы E такая, что все элементы этой строки, стоящие в столбцах с индексами из I, равны 0. В частности, $\deg\ \omega(n,m,E,x) < m-1$ тогда и только тогда, когда каждый столбец матрицы содержит нулевой элемент; $\omega(n,m,E,x) = \text{const}$ тогда и только тогда, когда E содержит диагональную подматрицу.

d) Пусть $\deg\ \omega(n,m,E,x) \le \tau$. Тогда

$$a_\tau = \sum_{\sigma\in A(\tau,m)} (n,m-\tau,E_\sigma),$$

где $A(\tau,m)$ – множество сочетаний из m по τ и, если $\sigma=(i_1,\ \ldots,\ i_\tau) \in A(\tau,m)$, то матрица E_σ получается из E вычеркиванием столбцов с номерами $i_1,\ \ldots,\ i_\tau$.

Доказательство.

a) Формула 5.1. представляет $\omega(n,m,E,x)$ в виде суммы многочленов, степень которых не превышает n.

b) Если $n=0$, то $\omega(n,m,E,x)$ равно числу мономов от m переменных, степень которых не больше x, т.е. $\omega(n,m,E,x) = \binom{x+m}{m}$ – многочлен степени m. Если $n > 0$, т.е. E содержит некоторую строку e, то $\omega(n,m,E,x) \le \omega(1,m,e,x) = \binom{x+m}{m} - \binom{x+m-|e|}{m}$.

Неравенство выполняется для всех $x > x_0$. Следовательно, $\deg\ \omega(n,m,E,x) \le \deg\ \omega(1,m,e,x) \le m-1$ (отметим, что $\deg\ \omega(1,m,e,x) = m-1$ для любого ненулевого вектора e).

c) Покажем, что $\deg \omega(n,m,E,x) < \tau$, если для любого подмножества I, состоящего из $m-\tau$ элементов множества $1..m$, существует строка e_1 матрицы E такая, что все элементы этой строки, стоящие в столбцах с индексами из I, равны нулю. Обратное утверждение будет следовать из пункта d.

Воспользуемся индукцией по сумме Σ элементов матрицы E. Если $\Sigma=0$, то E – нулевая матрица, и можно полагать $n=1$ и $E = e = (0, \ldots, 0)$. Тогда $\omega(n,m,E,x) = 0$ и $\deg \omega(n,m,E,x) \le \tau$ для любого $\tau \ge 0$.

Пусть $\Sigma>0$. Без потери общности можно предполагать, что первый столбец матрицы E содержит ненулевой элемент. Полагая в следствии 5.11 $v=(1, 0, \ldots, 0)$, представляем $\omega(n,m,E,x)$ как сумму многочленов $\omega(n+1,m,E',x)$ и $\omega(n,m,E'',x-1)$. К E'' можно применить индуктивное предположение; степень многочлена не меняется при сдвиге аргумента, следовательно, $\deg \omega(n,m,E'',x-|v|) < \tau$.

Относительно степени первого многочлена заметим, что можно оставить в E' только одну строку с ненулевым элементом в первом столбце и применить лемму 5.13. К E можно применить индуктивное предположение, заменив в нем m на $m-1$ и τ – на $\tau-1$.

d) Применим индукцию по $k = m-\tau$.

Пусть $m-\tau = 1$. Положим $v = (\min\limits_{1=}^{n} e_{11}, \ldots, \min\limits_{1=1}^{n} e_{1m})$. Из следствия 5.11 и только что полученного результата следует, что

$$a_{m-1} = \sum_{j=1}^{m} \min_{1=1}^{n} e_{1j}.$$

Отметим, что другим способом этот результат получен в [13].

Предположим, что $k=m-\tau > 1$. Доказательство пункта d в значительной мере аналогично доказательству пункта c и основано на индукции по сумме Σ элементов матрицы E. Многочлен $\omega(n,m,E,x)$ является суммой многочленов $\omega(n+1,m,E',x)$ и $\omega(n,m,E'',x-1)$. К матрицам E' и E'' можно применить индуктивное предположение. Сдвиг аргумента во втором многочлене не влияет на его старший коэффициент. Заметим, что $\omega(n+1,k,E'_\sigma,x)$ и $\omega(n,k,E''_\sigma,x-1)$ кон-

станты. Если сочетание σ не содержит первый столбец, то $\omega(n+1,k,E'_\sigma) = 0$, поскольку в матрице E'_σ все элементы первого столбца, кроме первого, нулевые и должна существовать строка, содержащая k–1 нулевой элемент в столбцах со второго по k-й, т.е. нулевая строка. Значит, $\omega(n,k,E''_\sigma) = \omega(n,k,E_\sigma)$. Если σ содержит первый столбец, то $\omega(n,k,E_\sigma) = \omega(n+1,\tau,E'_\sigma) + \omega(n,k,E''_\sigma)$ по свойствам определителя.

5.22. Пример

Вычисление разностного размерностного многочлена для кольца инверсных разностных многочленов сводится к вычислению многочлена Гильберта для идеала, порожденного элементами $x_1 \cdot x_{1+m} - 1$, $i=1,\ldots,m$ в кольце $k[x_1, \ldots, x_{2m}]$. Выписанные многочлены образуют базис Гребнера соответствующего идеала, а матрица E для этой системы имеет вид

$$\begin{bmatrix} 1 & 0 & 0 & \ldots & 0 & 1 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 & 0 & 1 & 0 & \ldots & 0 \\ \cdot & \cdot & \cdot & \ldots & \cdot & \cdot & \cdot & \cdot & \ldots & \cdot \\ 0 & 0 & 0 & \ldots & 1 & 0 & 0 & 0 & \ldots & 1 \end{bmatrix}$$

Непосредственное применение формулы (5.1) дает

$$\omega(m,2m,E,x) = \sum_{k=0}^{m} (-1)^k \binom{m}{k} \binom{x+2m-2k}{2m}$$

Из предложения 5.21 следует, что степень этого многочлена равна m, а старший коэффициент равен 2^m. Применение алгоритма Hilbpol приводит к формуле

$$\omega(m,2m,E,x) = \sum_{k=0}^{m} (-1)^{m-k} 2^k \binom{m}{k} \binom{x+k}{k}.$$

В заключение приведем алгоритм вычисления коэффициентов a_m, a_{m-1}, a_{m-2} многочлена Гильберта.

5.23. Алгоритм Maincoef($n,m,E,a_m,a_{m-1},a_{m-2}$)

Арг: n – неотрицательное целое;

m – натуральное число;
E – матрица размера $n \cdot m$ элементов типа $\mathbb{Z}+$
Рез: a_m, a_{m-1}, a_{m-2} – целые числа.

Начало
если $n=0$
. **то** $a_m := 1$; $a_{m-1} := 0$; $a_{m-2} := 0$
иначе
$a_m := 0$
$v := (v_1, \ldots, v_n)$, где $v_1 = \min_{j=1}^{n} e_{j1}$
$a_{m-1} := |v|$
$a_{m-2} := -\binom{|v|}{2}$
$e_{j1} := e_{j1} - v_1$, $i=1,\ldots,m$; $j=1,\ldots,n$
цикл для всех пар $1 \le i < j \le m$
$a_{m-2} := a_{m-2}$+ многочлен Гильберта матрицы, состоящей из i-го и j-го столбца матрицы E
. **конец цикла**
. **конец если**
конец

6. ОПИСАНИЕ КОМПЛЕКСА ПРОГРАММ ДЛЯ ВЫЧИСЛЕНИЯ БАЗИСОВ ГРЕБНЕРА И МНОГОЧЛЕНОВ ГИЛЬБЕРТА

Из особенностей данной реализации следует отметить следующие:

1. Базисы Гребнера вычисляются в свободных полиномиальных, дифференциальных или разностных модулях, а именно:
- в идеалах кольца $\mathbb{Q}[x_1, \ldots, x_n]$ и кольца $\mathbb{Q}[i][x_1, \ldots, x_n]$;
- в подмодулях свободного R-модуля, где $R = \mathbb{Q}[x_1, \ldots, x_n]$ или $R = \mathbb{Q}[i][x_1, \ldots, x_n]$;
- в подмодулях свободного $R[\delta_1, \ldots, \delta_n]$-модуля, где $R = \mathbb{Q}[x_1, \ldots, x_n]$, а $\delta_1 = \frac{\partial}{\partial x_1}$,
- в подмодулях свободного $R[T_1, \ldots, T_n]$-модуля, где $R = \mathbb{Q}[x_1, \ldots, x_n]$, а $T_1(x_1) = x_1 + 1$, и $T_1(x_j) = x_j$ при $i \neq j$.

2. Вычисляются квазиредуцированные базисы Гребнера: после добавления очередного элемента система квазиредуцируется, т.е. новый элемент частично редуцируется относительно всех старых, а старые – относительно нового. Использование неполной редукции сокращает объем вычислений. Упорядочение системы позволяет выполнять частичную редукцию достаточно быстро.

3. Поле коэффициентов колец многочленов, дифференциальных и разностных операторов является полем частных соответствующего кольца. Элементы базиса Гребнера нормируются так, что все их коэффициенты принадлежат кольцу (а не полю) коэффициентов, наибольший общий делитель коэффициентов равен 1. При этом особое внимание нужно уделять вычислению S-элементов в разностных модулях, так как в них существенно влияние некоммутативности операторов в старших степенях.

4. При переборе критических пар множество критических пар упорядочивается по возрастанию НОК старших слагаемых. Очередная пара выбирается минимальной относительно этого порядка. Используется "правило треугольника" отбрасывания "лишних" критических пар (критерий 3 в терминологии работы [20]).

5. Система содержит программу, вычисляющую многочлен Гильберта (по найденному базису Гребнера). Эту программу можно использовать не только в полиномиальных, но и в дифференциальных

6-1432

и разностных модулях (для вычисления дифференциального, соответственно, разностного размерностного многочлена).

6. Дифференциальный (разностный) размерностный многочлен неинвариантен относительно замены образующих модуля, поэтому представляет интерес исследование зависимости размерностного многочлена от множества образующих. С этой целью в систему введен аппарат замены переменных.

Описываемый комплекс программ написан на алгоритмическом языке Рефал и содержит около 20 файлов с исходными текстами, общий объем которых составляет приблизительно 2000 строк. В зависимости от решаемых задач употребляются различные варианты сборки: 3 варианта сборки задачи "базис Гребнера" (для полиномиальных идеалов и модулей, для дифференциальных модулей, для разностных модулей) и два варианта сборки задачи "замена переменных" (для дифференциальных и разностных модулей).

Во всех вариантах задач используется библиотека Рефала и один из стандартных отладчиков.

Ниже приводится изложение основных алгоритмов вычисления базисов Гребнера, ориентируемое на конкретную реализацию. Завершает параграф несколько примеров использования комплекса программ.

6.1. РЕАЛИЗАЦИЯ ОСНОВНЫХ АЛГОРИТМОВ

Ниже сформулированы ключевые алгоритмы, реализованные в описываемой системе. При описании внутренних спецификаций пользуемся терминологией, принятой при программировании на языке Рефал (символ, терм, выражение, ящик, макроцифра и т.д.), поскольку описываемые алгоритмы реализованы на этом языке.

Для задания элементов кольца P и модуля P^m введем следующие типы данных:

```
<кольцевая_образующая> ::= элемент фиксированного подмножества
                            латинского алфавита
<степень>        ::= <кольцевая_образующая>!
                     <кольцевая_образующая><показатель>!
                     D<кольцевая_образующая>!
                     T<кольцевая_образующая>!
```

D<показатель><кольцевая_образующая>!
T<показатель><кольцевая_образующая>!

<мультистепень> ::= <степень>!<степень>*<мультистепень>

<одночлен> ::= (<коэффициент>)*<мультистепень>

<элемент_кольца>::= !<одночлен>!
<одночлен> + <элемент_кольца>!
<одночлен> - <элемент_кольца>

<показатель> ::= <натуральное_число>

<коэффициент> ::= элемент некоторого фиксированного кольца коэффициентов

Если <степень> начинается с буквы *D*, то будем говорить о дифференциальной степени, если с *T* - то о разностной. Предполагаем, что все <степени> имеют один и тот же тип (полиномиальный, дифференциальный или разностный).

Введенные таким образом <элементы> могут представлять собой элементы кольца полиномов, или кольца дифференциальных операторов, или кольца разностных операторов в зависимости от задания операции умножения на множестве <элементов>. Выписанные определения, строго говоря, не задают канонической формы элементов кольца. Для определения канонической формы нужно потребовать еще выполнения условий следующего типа: в каждой <степени> каждая <кольцевая_образующая> появляется не более одного раза (с соответствующим показателем), <кольцевые_образующие> в <степени> упорядочены в соответствии с выбранным порядком <кольцевых_образующих>, каждая <мультистепень> появляется не более чем в одном <одночлене>, которые также упорядочены; для кольца коэффициентов выбрано каноническое представление и нет одночленов с нулевыми коэффициентами.

Во всех трех случаях операция умножения аддитивна и <кольцевые_образующие> коммутируют между собой. Кольцо многочленов является коммутативным, т.е. в нем <кольцевые_образующие> коммутируют и с <коэффициентами>. В кольце дифференциальных операторов каждой <кольцевой_образующей> x соответствует дифференцирование Dx кольца коэффициентов, коммутирование <кольцевой_образующей> x с элементом κ кольца коэффициентов осуществляется по правилу: $x * \kappa = \kappa * x + Dx(\kappa)$.

В кольце разностных операторов каждой <кольцевой_образу-

ющей> x соответствует автоморфизм Tx кольца коэффициентов. Коммутирование <кольцевой_образующей> с элементом $к$ кольца коэффициентов осуществляется по правилу: $x*к = Tx(к)*x$.

Для определения элементов свободного модуля введем дополнительно следующие типы данных:

<модульная_образующая> ::= элемент фиксированного подмножества латинского алфавита (непересекающегося с множеством кольцевых_образующих)

<моном> ::= <мультистепень>*<модульная_образующая>

<слагаемое> ::= (<коэффициент>)*<моном>

<элемент_модуля> ::= !<слагаемое>!
<слагаемое> + <элемент_модуля>!
<слагаемое> – <элемент_модуля>

Замечания, сделанные относительно канонической формы элементов кольца, остаются справедливыми и для элементов модуля.

В дальнейшем, допуская некоторую вольность обозначений, будем обычно опускать угловые скобки.

Спецификации типов данных во внутреннем представлении (используемые при описании скобки являются структурными скобками языка Рефал):

<номер_элемента> ::= <макроцифра>

<степень> ::= <макроцифра>

<образующая> ::= <символ>

<коэффициент> ::= (<выражение>)

<мультистепень> ::= вектор элементов типа <степень> с индексом 1..количество_переменных

<индекс> ::= (<номер_элемента><моном>)

<критическая_пара> ::= (<моном><номер_элемента>
<номер_элемента>)

<слагаемое> ::= (<коэффициент><моном>)

<элемент> ::= <слагаемое>!<элемент><слагаемое>

<система> ::= (множество <критических_пар>)
(дек <индексов>)

Следующее определение зависит от используемых программных модулей (для вычисления базисов Гребнера для идеалов в кольце многочленов или для вычислений в модулях):

кольцевой случай:

<моном> ::= (<мультистепень>)

модульный случай:

<моном> ::= (<мультистепень><образующая>)

Для замены переменных выражение замены хранится в копилке под именем REPL. Формат выражения замены:

<выражение_замены> ::= <образующая>=<элемент>!
<образующая>=<элемент> <выражение_замены>

Большинство используемых типов данных представляет собой записи, при описании алгоритмов мы будем обычным образом ссылаться на их поля, например, критическая_пара.номер_элемента_1, критическая_пара.номер_элемента_2, или просто номер_элемента_1, когда из контекста понятно, о чем идет речь.

Назначение ящиков:

IND – содержит максимальный номер_элемента
ORD – содержит данные об отношении порядка образующих (старые образующие для замены переменных)
ORDN – новые образующие для замены переменных
ORDVAR – содержит данные об отношении порядка переменных
STMP – содержит элементы, не включенные в систему
NKRP – содержит критические пары, не включенные в систему
MF – количество переменных в кольце многочленов
MODE – содержит режим (M – модуль, R – кольцо)

Описание алгоритмов

Основные алгоритмы вычисления базисов Гребнера располагаются в соответствии с технологией "сверху-вниз", начиная с головной программы, за которой следуют алгоритмы все более частных подпрограмм. В случае задачи "замена переменных" к ним нужно добавить алгоритм выражения старых переменных через новые.

6.1.1. Головная программа вычисления базиса Гребнера

ВХОД: исходные данные во входном файле
ВЫХОД: результаты расчетов в выходном файле и/или на терминале.

```
начало
   инициализация <вых: SYS: система>
   цикл для каждой KP из SYS.множество_критических_пар
      если  критерий(KP,SYS)
         то
            вычислить S-элемент <вх:KP> <вых: E:элемент>
            добавить элемент к системе <вх: SYS, E> <вых: SYS>
   . конец если
   конец цикла
. завершение работы <вх: SYS.дек_индексов>
конец
```

Ниже приводится алгоритм "инициализация" для зад. и "вычисление базиса Гребнера".

6.1.2. АЛГОРИТМ инициализация (SYS)

Арг: исходные данные во входном файле
Рез: SYS:система

```
начало
   ввести имя файла с исходными данными
   прочитать режим
   прочитать кольцевые образующие
   если режим = 'M'
   . то прочитать модульные образующие
   все
   SYS.начать работу
   цикл пока есть непрочитанные записи во входном файле
      читать очередной элемент <вых: E:элемент>
      привести к каноническому виду <вх: E> <вых: E>
   . добавить элемент к системе <вх: SYS, E> <вых: SYS>
. конец цикла
конец
```

Подпрограмма "завершение работы" работает в интерактивном режиме, по желанию пользователя она записывает найденный базис

Гребнера в файл, также по желанию пользователя она вычисляет многочлен Гильберта. Реализовано несколько версий алгоритмов вычисления многочленов Гильберта, описание которых приводится в следующем параграфе.

6.1.3. АЛГОРИТМ добавить элемент к системе (E,SYS)

Арг: SYS:система, E:элемент
Рез: SYS

```
начало
  если E ≠ 0,
    то  частично редуцировать элемент <вх: E, SYS> <вых: SYS>
    иначе
      цикл  для каждого SE из STMP
      .  добавить элемент к системе <вх: SYS, SE> <вых: SYS>
      конец цикла
  все
конец
```

6.1.4. АЛГОРИТМ частично редуцировать элемент (E,SYS)

Арг: E:элемент
SYS:система
Рез: SYS:система
переменные: ДПМ:дек индексов == дек проверяемых индексов
NI:индекс == (n,Mn)
TI:индекс == (i,Mi)
обозначения: ДПН == SYS.дек индексов

```
начало
  Mn := E.старший_моном
  ДПМ := SYS.дек_индексов
  ДПН.сделать пустым
  цикл для каждого TI из начала ДПМ
    Mi := TI.моном
    сравнить <вх: Mn, Mi> <вых: B:символ>
```

```
    выбор
        B = 'L' ==> вычислить новый_номер
                    Nl := (новый_номер, Mn)
                    ДПН.добавить Nl
                    запомнить E
                    выход из цикла
        B = 'N' ==> ДПН.добавить в конец Tl
        B = 'G' ==> сформировать новую <критическую_пару>
                    NKRP.добавить <критическую_пару>
                    ДПН.добавить в конец Tl
        B = 'E' или B = 'D' ==>
                    NKRP.сделать ящик пустым
                    Ei := Mn/Mi
                    псевдоредуцировать
                              <вх: n, i: номер_элемента,
                              Ei: моном> <вых: E: элемент>
                    ДПН := ДПН ∪ ДПМ
                    ДПМ.сделать пустым
                    выход из цикла
    конец выбора
конец цикла
выбор
    B = 'E' или B = 'D' ==>
            добавить элемент к системе (E, SYS)
    B = 'L' ==>
            квазиредуцировать систему относительно элемента
                                        (SYS, ДПМ, Nl)
конец выбора
конец
```

6.1.5. Замечание. Используемая в данном алгоритме процедура сравнения двух мономов присваивает выходному параметру B типа символ одно из следующих значений:

L – если первый моном старше второго;

N – если мономы зависят от разных модульных образующих и второй моном старше первого;

E – если мономы совпадают;

D – если первый моном делится на второй; частное Ei, являющееся входным параметром в предписании "псевдоредуцировать", в действительности вычисляется при сравнении мономов;

G – мономы от одной модульной образующей, первый моном старше второго и не делится на него.

6.1.6. АЛГОРИТМ частично редуцировать систему относительно элемента (SYS, ДПМ, NI)

Арг: SYS, ДПМ:дек индексов, NI=(n,Mn):индекс
Рез: SYS
обозначения: МКР == SYS.множество критических пар
ДПН == SYS.дек индексов

```
начало
  цикл для всех (i, Mi) из начала ДПМ
    если Mi.образующая = Mn.образующая
      то вычислить НОК(Mi, Mn)
        если Mi = НОК(Mi, Mn)
          то Ei := Mi/Mn
             псевдоредуцировать <вх: n,i:номер_элемента,
                                 Ei:мультистепень> <вых: E:элемент>
             если E ≠ 0
               то STMP.добавить E
             все
             МКР.удалить критические_пары, содержащие
                   номер i
          иначе NKRP.добавить ((НОК(Mi, Mn), n, i)
                  !внешние скобки используются при сортировке
        все
        ДПН.добавить в конец (i,Mi)
    все
  конец цикла
конец
```

6.1.7. Замечание. В алгоритме "псевдоредуцировать E отно-

сительно SE'' элементы E и SE задаются своими номерами. Алгоритм применяется в случае, когда старший моном элемента SE делится на старший моном элемента E, Ei – соответствующее частное. При псевдоредукции из редуцируемого элемента удаляется только старший моном, получающийся в результате элемент не обязан быть частично редуцированным относительно SE.

6.1.8. АЛГОРИТМ псевдоредуцировать E относительно SE(*i*,*n*,Ej,E)

Арг: *i* – номер редуцируемого элемента
n – номер редуцирующего элемента
Ej – дополнительный одночлен
Рез: E
переменные: Ci,Cj: коэффициент

начало
выкопать элемент <вх: *i*> <вых: Ei>
скопировать элемент <вх: *n*> <вых: En>
вычислить НОК старших коэффициентов
вычислить дополнительные множители Ci и Cn
E := Ci*Ei – Cn*Ej*En !S–элемент
конец

Этот алгоритм, так же как и следующий за ним алгоритм вычисления S–элемента, сформулирован для полиномиальных и дифференциальных модулей. Для разностных модулей требуется их модификация.

6.1.9. АЛГОРИТМ вычислить S–элемент (KP, E)

Арг: KP: критическая пара == (M, *i*, *j*)
Рез: E: элемент

начало
читать <вх: *i*> <вых: Ei>
Ci := Ei.старший коэффициент
Mi := Ei.старший моном

```
   читать <вх: j> <вых: Ej>
   Cj := Ej.старший коэффициент
   Mj := Ej.старший моном
   Bi := Cj*(M/Mi)
   Bj := Ci*(M/Mj)
   E  := Bi*Ei - Bj*Ej
конец
```

6.1.10. АЛГОРИТМ критерий (KP,SYS)

Арг: KP:критическая пара == (M, *i*, *j*)
SYS:система
Рез: ответ: 'T' / 'F'

```
начало
   ответ := 'T'
   читать режим
   если режим = 'R'
      то читать моном <вх: i> <вых: Mi>
         читать моном <вх: j> <вых: Mj>
         если M=Mi*Mj
            то ответ := 'F'
         все
   все
   цикл для всех (к, Mk) из начала SYS.дек_индексов
         пока ответ = 'T'
         если Mk < M и ни одна из пар (НОК(Mj,Mk),к,j),
               (НОК(Mi,Mk),i,к),(НОК(Mi,Mk),к,i),(НОК(Mj,Mk),j,к)
               не принадлежит SYS.множество_критических_пар
            то ответ := 'F'
         все
   конец цикла
конец
```

6.1.11. Замечание. В приведенном здесь критерии используется "правило треугольника", получающееся из псевдолокального условия слияния при s=1, а для идеалов в кольце многочленов

также следующий факт, доказанный Б.Бухбергером: если произведение старших мономов двух многочленов равно НОК этих мономов, то соответствующую пару можно исключить из рассмотрения при вычислении базисов Гребнера. Доказательство этого факта оставляется читателю в качестве упражнения.

6.2. ПРИМЕРЫ ПРИМЕНЕНИЯ КОМПЛЕКСА ПРОГРАММ

6.2.1. Рассмотрим дифференциальный модуль над полем $\mathbb{Q}[i](x_1, \ldots, x_4)$, с образующими φ_1, φ_2, φ_3, φ_4, удовлетворяющими уравнениям Дирака

$$\begin{aligned}
d_4\varphi_1 \qquad\qquad\quad - id_3\varphi_3 - (id_1+d_2)\,\varphi_4 &= 0,\\
d_4\varphi_2 - (id_1-d_2)\,\varphi_3 \qquad + id_3\varphi_4 &= 0,\\
id_3\varphi_1 + (id_1+d_2)\,\varphi_2 \quad - d_4\varphi_3 &= 0,\\
(id_1-d_2)\,\varphi_1 \quad - id_3\varphi_2 \qquad\quad - d_4\varphi_4 &= 0,
\end{aligned}$$

где $d_1 = \partial/\partial x_1$.

Для вычисления характеристического множества и дифференциального размерностного многочлена можно в этом случае воспользоваться программой вычисления базисов Гребнера полиномиальных модулей над кольцом многочленов с коэффициентами из $\mathbb{Z}[i]$. Обозначаем неизвестные функции буквами F,G,U,V, а операторы дифференцирования – X,Y,Z,T. Исходная система принимает вид:

```
T*F - i*Z*U - i*X*V - Y*V
T*G - i*X*U + Y*U + i*Z*V
i*Z*F + i*X*G + Y*G - T*U
i*X*F - Y*F - i*Z*G - T*V
```

В результате вычислений получаем базис Гребнера:

```
T*G - i*X*U + Y*U + i*Z*V
T*F - i*Z*U - i*X*V - Y*V
Z*F + X*G - i*Y*G + i*T*U
X*F + i*Y*F - Z*G + i*T*V
X^2*V + Y^2*V + Z^2*V + T^2*V
X^2*U + Y^2*U + Z^2*U + T^2*U
X^2*G + Y^2*G + Z^2*G + i*X*T*U - Y*T*U - i*Z*T*V.
```

Многочлен Гильберта, соответствующий этому базису, равен $4\binom{x+3}{3}$.

6.2.2. Замена переменных

Пусть образующие φ_1, φ_2, φ_3, φ_4 дифференциального модуля удовлетворяют системе дифференциальных уравнений

$$\begin{aligned}
d_4\varphi_1 \qquad\quad - d_3\varphi_3 \; - d_2\varphi_4 \; &= 0,\\
d_4\varphi_2 \; - d_1\varphi_3 \; + d_3\varphi_4 \; &= 0,\\
d_3\varphi_1 \; + d_2\varphi_2 \; - d_4\varphi_3 \qquad\quad &= 0,\\
d_1\varphi_1 \; - d_3\varphi_2 \qquad\quad - d_4\varphi_4 \; &= 0,
\end{aligned}$$

где $d_j = \partial/\partial x_j$. Дифференциальный размерностный многочлен для этого модуля равен $4\binom{s+3}{3}$.

Пусть $\xi = \varphi_1 + x_4\varphi_2$. Тогда ξ, φ_3, φ_4 порождают тот же самый дифференциальный модуль;

$$\varphi_1 = \xi - x_4 d_4\xi + x_4 d_3\varphi_3 + x_4 d_2\varphi_4 + x_4^2 d_1\varphi_3 - x_4^2 d_3\varphi_4,$$

$$\varphi_2 = x_4\xi - d_3\varphi_3 - d_2\varphi_4 - x_4 d_1\varphi_3 + x_4 d_3\varphi_4.$$

Базис Гребнера для новых образующих имеет вид

$$d_4^2\xi - d_1 d_4\varphi_3 - d_3 d_4\varphi_3 + x_4 d_3 d_4\varphi_4 - d_2 d_4\varphi_4 - 2\,d_1\varphi_3 + 2\,d_3\varphi_4;$$

$$d_2 d_4\xi - x_4 d_3 d_4\xi - x_4 d_1 d_2\varphi_3 + x_4^2 d_1 d_3\varphi_3 - d_2 d_3\varphi_3 + x_4 d_3^2\varphi_3 - d_2^2\varphi_4 +$$

$$+ x_4^2 d_2 d_3\varphi_4 - x_4^2 d_3^2\varphi_4 + d_3\xi - d_4\varphi_3;$$

$$-x_4^2 d_1 d_4\xi - d_3 d_4\xi + x_4^2 d_1^2\varphi_3 + 2\,x_4 d_1 d_3\varphi_3 + d_3^2\varphi_3 + x_4 d_1 d_2\varphi_4 -$$

$$- x_4^2 d_1 d_3\varphi_4 + d_2 d_3\varphi_4 - x_4 d_3^2\varphi_4 + d_1\xi - d_4\varphi_4.$$

Дифференциальный размерностный многочлен для этих образующих равен $4\binom{s+3}{3} - 1$.

ЛИТЕРАТУРА

1. Балаба И.Н. Размерностные многочлены расширения разностных полей // Вестн. МГУ. Мат. Мех. 1984, N 2. С. 31–35.

2. Бухбергер Б. Базисы Гребнера. Алгоритмический метод в теории полиномиальных идеалов // Компьютерная алгебра; символьные и алгебраические вычисления. М.: Мир, 1986. С. 331–372.

3. Джавадов Г.А. Характеристический многочлен Гильберта для дифференциально-разностных модулей // Структурные свойства алгебр. систем. Нальчик, 1981. С. 21–30.

4. Компьютерная алгебра: Символьные и алгебраические вычисления // Ред. Б. Бухбергер и др. Перев. с англ. М.: Мир, 1986.– 392 с.

5. Кондратьева М.В. Описание множества минимальных дифференциальных размерностных многочленов // Вестн. МГУ. Мат., мех. 1988, N 1. С. 35–39.

6. Кондратьева М.В., Михалев А.В., Панкратьев Е.В. О границе Якоби для систем дифференциальных многочленов // Алгебра. – М.: Изд-во МГУ, 1982. С. 79–85.

7. Кондратьева М.В., Панкратьев Е.В., Серов Р.Е. Вычисления в дифференциальных и разностных модулях // Тр. Междунар. совещ. по анал. вычисл. на ЭВМ и их применению в теор. физ. Дубна, 17–20 сент. 1985. Дубна, 1985. С. 208–213.

8. Кондратьева М.В., Панкратьев Е.В. Алгоритмы вычисления характеристических многочленов Гильберта // Пакеты прикладных программ. Аналитические преобразования.–М.: Наука, 1988. С. 129–146.

9. Латышев В.Н. Конструктивная теория колец. Стандартные базисы. М.: Изд-во МГУ, 1988.

10. Латышев В.Н., Михалев А.В., Панкратьев Е.В. Построение канонического симплификатора в модулях над кольцами полиномов // Вестн. Киевского университета. Мат. и мех. Вып. 27. Киев: Высшая школа, 1985. С. 65–68.

11. Левин А.Б. Характеристические многочлены фильтрованных разностных модулей и расширений разностных полей // Успехи матем. наук. 1978. Т. 33, N 3. С. 177–178.

12. Левин А.Б., Михалев А.В. Дифференциальный размерностный многочлен и жесткость системы дифференциальных уравнений // Вычислимые инварианты в теории алгебраических систем.- Новосибирск, 1987. С. 58–67.

13. Михалев А.В., Панкратьев Е.В. Дифференциальный размерностный многочлен системы дифференциальных уравнений // Алгебра. М.: Изд-во МГУ. 1980. С. 57–67.

14. Михалев А.В., Панкратьев Е.В. Вычисление дифференциального размерностного многочлена с помощью ЭВМ // Теория и практика автоматизированных систем . Вильнюс, 1984. С. 50–53.

15. Михалев А.В., Панкратьев Е.В. Дифференциальная и разностная алгебра // Итоги науки и техн. ВИНИТИ. Алгебра. Топол. Геом. 1987. Т. 25. С. 67–139.

16. Панкратьев Е.В. Компьютерная алгебра. Факторизация многочленов. М.: Изд-во МГУ, 1988.

17. Эйнштейн А. Собрание научных трудов. Т. 2. Работы по теории относительности. 1921–1955. М.: Наука, 1966. С. 777–786.

18. Bergman G.M. The diamond lemma for ring theory // Advances in mathematics. 1978. V.29. P. 178–218.

19. Buchberger B. Ein Algorithmus zur Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal. Ph. D. thesis., Univ. Innsbruch, Austria, 1965.

20. Buchberger B. A criterion for detecting unnecessary reductions in the construction of Grobner bases // Lect. Notes Comput. Sci. 1979. V. 72. P. 3–21.

21. Cohn R.M. Difference algebra. Interscience. N. Y., 1965.

22. Davenport J., Siret Y., Tournier E. Calcul formel. Systèmes et algorithmes de manipulations algébriques. Masson, Paris, 1987. Англ. перевод: Davenport J., Siret Y., Tournier E. Computer algebra. Systems and algorithms for algebraic computation. L.: Academic Press, 1988.

23. Hermann G. Die Frage der endlichen vielen Schritte in der Theorie der Polynomideale // Math. Ann. 1926. V. 95. P. 736–788.

24. Hironaka H. Resolution of singularities of an algebraic variety over a field of characteristic zero: I, II. // Ann.

Math. 1964. V.79. P. 109–326.

25. Johnson J. Differential dimension polynomials and a fundamental theorem on differential modules // Amer. J. Math. 1969. V. 91, N 1. P. 239–248.

26. Kolchin E.R. Some problems in differential algebra. Тр. Междунар. конгресса математиков, 1966. М.: Мир, 1968. С. 269–276.

27. Kolchin E.R. The notion of dimension in the theory of algebraic differential equations // Bull. Amer. Math. Soc. 1964. V. 70. P. 570–573.

28. Kolchin E.R. Differential algebra and algebraic groups. N. Y.: Acad. Press, 1973.

29. Kolchin E.R. Differential algebraic groups. N. Y.: Acad. Press, 1985.

30. Kondrat'eva M.V., Pankrat'ev E.V. A recursive algorithm for the computation of the Hilbert polynomial // Proc. EUROCAL'87.

31. Lazard D. Gröbner bases, gaussian elimination and resolution of systems of algebraic equations // Lect. Notes Comput. Sci. 1983. V. 162. P. 146–156.

32. Manin Ju. I. Moduli fuchsiani // Ann. Scuola norm. super. Pisa, Sci. fis. et mat. 1965. V. 19, N 1. P. 113–126.

33. Mora F., Möller H.H. The computation of the Hilbert function // Proc. EUROCAL'83. Lect. Notes Comput. Sci. 1983. V. 162, P. 157–167.

34. Möller H.H., Mora F. New constructive methods in classical ideal theory // J. Algebra. 1986. V. 100. P. 138–178.

35. Pankrat'ev E.V. Computations in differential and difference modules // Acta Appl. Math. 1989

36. Ritt J.F. Differential algebra // Amer. Math. Soc. Publ. 1950, N 33.

37. Sit W. Well-ordering of certain numerical polynomials // Trans. Amer. Math. Soc. 1975. V. 212. P. 37–45.