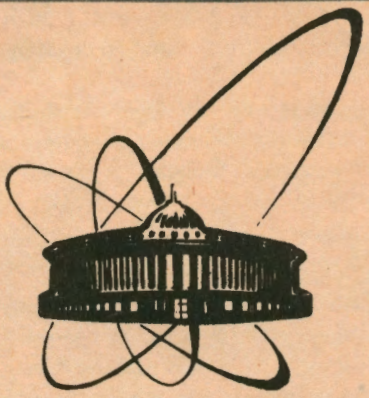

объединенный
институт
ядерных
исследований
дубна

Р11-92-100

В.П.Гердт, А.Ю.Жарков*, Н.В.Хуторной

ASYS: ПАКЕТ ДЛЯ ИССЛЕДОВАНИЯ СИСТЕМ НЕЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ

Направлено в журнал "Программирование"

*Саратовский университет

1992

1. Введение

Нелинейные алгебраические уравнения возникают во многих фундаментальных и прикладных задачах. Их исследование и решение является одним из центральных направлений развития компьютерной алгебры, включая разработку эффективных алгоритмов упрощения систем полиномиальных уравнений с целью нахождения их корней либо аналитически, либо численно, а также создание соответствующего программного обеспечения [1].

Наиболее развитым и универсальным алгоритмическим методом анализа и решения систем нелинейных алгебраических уравнений является техника, основанная на построении базиса Гребнера для полиномиального идеала, генерируемого исходным набором полиномов данной системы [2, 3] (см. также [4, 5]). Вычисление базиса Гребнера позволяет проверить совместность системы, найти размерность пространства решений, а в случае конечного числа корней привести исходную систему к эквивалентному "треугольному" виду, т.е. свести задачу к последовательному решению алгебраических уравнений одной переменной. Все это достижимо чисто алгоритмическим путем и может быть целиком переложено на компьютер. Время, требуемое для построения базиса Гребнера, существенно зависит от выбранного упорядочения переменных. На практике наиболее часто используются лексикографическое упорядочение и упорядочение по полной степени с обратным лексикографическим упорядочением среди мономов одинаковой полной степени [1, 2].

Соответствующие алгоритмы и пакеты встроены во все современные программные системы компьютерной алгебры, такие как Reduce, Maple, Mathematica, Axiom и др.

К сожалению, однако, при решении многих практических задач для построения базиса Гребнера требуются непомерно большие вычислительные ресурсы, обусловленные экспоненциальной сложностью алгоритма Бухбергера [2] по числу полиномиальных переменных. В случае конечного числа решений, т.е. нульмерного идеала, оценки сложности алгоритма дают $O(d^{n^2})$ по степени полиномов d и числу переменных n для лексикографического упорядочения и полиномиальное поведение по d^n для упорядочения по полной степени (см. [4] а также ссылку [6] и библиографию в ней). В этом случае оптимальная схема вычисления состоит в построении базиса Гребнера с упорядочением по полной степени с последующим пересчетом в лексикографическое упорядочение, которое и обеспечивает разделение переменных путем приведения системы к треугольному виду, следуя алгоритму работы [6], имеющего также полиномиальное поведение по d^n.

Для полиномиальных идеалов с положительной размерностью, т.е. для систем с бесконечным числом решений, как теоретический анализ алгоритмов, так и практическое построение базисов Гребнера и основанный на нем поиск решений связаны с гораздо большими трудностями. В этом случае существующие

оценки сложности алгоритма Бухбергера дают субэкспоненциальное поведение вида $O(d^{n*2^n})$. Поэтому многие имеющиеся пакеты и программы настроены именно на нульмерные идеалы, что сильно затрудняет их использование для работы с полиномиальными идеалами ненулевой размерности.

С другой стороны, системы нелинейных алгебраических уравнений с бесконечным числом решений возникают на практике довольно часто, например, при анализе интегрируемости нелинейных эволюционных уравнений [7, 8, 16], конечномерных алгебр Ли [9], исследовании бифуркаций нелинейных динамических систем [10] и многих других задачах.

Описываемый в настоящей работе пакет ASYS, написанный на языке символьной моды Rlisp системы аналитических вычислений Reduce [11],разрабатывался специально для анализа и решения систем с бесконечным числом корней. В отличие от других аналогичных пакетов, основанных на технике базисов Гребнера, данный пакет позволяет эффективно находить всевозможные наборы независимых переменных, которые могут рассматриваться в качестве свободных параметров, а также резко упрощать вычисления в случае, когда исходная система обладает нетривиальными свойствами однородности. Такие системы типичны, в частности, для задач, рассмотренных в работах [7, 8, 9, 10]. Работа пакета ASYS проиллюстрирована в сравнении с другими пакетами на примерах такого рода, взятых из работ [7, 16],а также на хорошо известных тестовых примерах [6, 19].

2. Основные определения и понятия

2.1. Базис Гребнера. Алгоритм Бухбергера

Следуя работе [2],мы будем использовать следующие обозначения и определения:

K – произвольное поле характеристики нуль;
$K[x_1,\dots,x_n]$ – кольцо полиномов от n переменных над полем K;
i,j,k,l,m,n – натуральные числа;
a,b,c,d – элементы поля K;
f,g,h,p,q – полиномы из $K[x_1,\dots,x_n]$;
s,t,u – произведения степеней вида $x_1^{i_1},\dots,x_n^{i_n}$;
$C(f,u)$ – коэффициент при u в f;
F,G – конечные подмножества в $K[x_1,\dots,x_n]$;
$Ideal(F)$ – идеал, порожденный F, т.е. множество
$\{\sum_i h_i f_i \mid h_i \in K[x_1,\dots,x_n], f_i \in F\}$;
$<$ – некоторое допустимое [2] упорядочение мономов, например, уже упоминавшиеся во введении лексикографическое упорядочение для x "младше" y,а y "младше" z и т.д.:

$$x < y < z < \dots < x^2 < xy < y^2 < xz < yz < z^2 < \dots,$$

или упорядочение по полной степени, а затем уже обратное лексикографическое, которое для x "младше" y а y "младше" z и т.д. означает

$$x < y < z < \dots < x^2 < xy < xz < y^2 < yz < z^2 < \dots;$$

$LP(f)$ – старшее (относительно упорядочения $<$) произведение степеней в f;
$LC(f)$ – коэффициент при $LP(f)$ в f.

Определение. Полином g редуцируется к h по модулю F (обозначается $g \to_F h$), если найдутся такие $f \in F, b, u$, что выполнено $g \to_{f,b,u} h$ и, кроме того, $h = g - buf$; полином g редуцируется с помощью f, b, u, если $C(f,u) \neq 0$, и, кроме того, $b = C(f,u)/LC(f)$.

Определение. Полином h дан в нормальной форме по модулю F, если не существует такого полинома h', что $h \to_F h'$.
Полином h является нормальной формой полинома g по модулю F (обозначается $h = NF(F,g)$), если имеется последовательность редукций $g = q_0 \to_F q_1 \to_F q_2 \to_F \dots \to_F q_m = h$ и, кроме того, h дан в нормальной форме по модулю F.

Определение. Множество F называется базисом Гребнера, если для всяких полиномов g, h_1, h_2, таких, что h_1 и h_2– нормальные формы полинома g по модулю F выполняется равенство $h_1 = h_2$.

Определение. S–полиномом, соответствующим полиномам f_1, f_2,называется полином $SP(f_1,f_2) = u_1 f_1 - (c_1/c_2)u_2 f_2$, где $c_i = LC(f_i)$, произведение степеней u_i таково, что $s_i u_i$ совпадает с наименьшим общим кратным произведения степеней s_1, s_2, где $s_i = LP(f_i)(i = 1,2)$.

Теорема. (Алгоритмически важнейшее свойство базисов Гребнера).
F является базисом Гребнера тогда и только тогда, когда для всех $f,g \in F$: $NF(F,SP(f,g)) = 0$.

Приведем несколько упрощенное описание алгоритма построения базиса Гребнера для полиномиального идеала, генерируемого конечным набором F полиномов из кольца $K[x_1,\dots,x_n]$. Более эффективная форма алгоритма, использованная в пакете ASYS, а также доказательство его корректности содержатся в работах [2, 3] и библиографии к ним.

Алгоритм построения базиса Гребнера

Ввод: F.
Вывод: G, такое, что $Ideal(F) = Ideal(G)$ и G– базис Гребнера.

$G := F$;
$B := \{(f_1,f_2) \mid f_1, f_2 \in G, f_1 \neq f_2\}$;
while $B \neq \emptyset$ *do*
(f_1,f_2)– пара из B; $B := B \setminus \{(f_1,f_2)\}$;
if not Критерий I *and not* Критерий II *then*
$h := SP(f_1,f_2)$;

$h' := NF(G, h)$;
if $h' \neq 0$ *then*
$G := G \cup \{h'\}$;

Использование критериев I и II [2] позволяет резко повысить эффективность алгоритма, поскольку во многих случаях позволяет распознать редуцируемость S - полинома, соответствующего данной паре полиномов f_i, f_j к нулю без вычисления соответствующей нормальной формы.

Критерий I позволяет исключить из рассмотрения каждую пару полиномов f_i, f_j, для которой в процессе работы алгоритма уже оказались рассмотренными две другие пары f_i, f_k и f_k, f_j, такие, что $LP(f_k)$ делит НОК$(LP(f_i), LP(f_j))$. Здесь НОК означает наименьшее общее кратное.

Критерий II исключает из рассмотрения пары такие, что

$$LP(f_i)LP(f_j) = HOK(LP(f_i), LP(f_j)).$$

2.2. Размерность полиномиального идеала и наборы независимых переменных

Размерность многообразия корней данной системы полиномиальных уравнений определяется степенью многочлена Гильберта, который, в свою очередь, может быть вычислен через базис Гребнера (см. работу [5] и библиографию к ней). Именно этот метод определения размерности идеала используется в стандартном пакете GROEBNER системы Reduce [11].

Однако знания размерности идеала недостаточно для анализа структуры многообразия корней полиномиального идеала положительной размерности, что является чрезвычайно важным для поиска решений в алгебраическом виде. Под таким решением мы понимаем явную параметризацию различных неприводимых подмногообразий всего многообразия корней данной системы.

В работе [12] предложен алгоритм нахождения полного множества различных максимальных наборов переменных, (алгебраически) независимых по отношению к рассматриваемому полиномиальному идеалу с заданным упорядочением переменных $<$. Этот алгоритм основан на знании базиса Гребнера и следующем утверждении.

Для минимального, или редуцированного [2], базиса Гребнера G набор

$$S = \{x_{i_1}, \ldots, x_{i_r}\} \subseteq X = \{x_1, \ldots, x_n\}$$

является *независимым* по отношению к $Ideal(G)$ тогда и только тогда, когда

$$T(S) \cap LT(G) = \{\emptyset\}, \tag{1}$$

где $T(S)$ означает множество всевозможных мономов (термов), зависящих только от переменных $x_i \in S$, а $LT(G)$ - множество старших, относительно упорядочения $<$, мономов базиса Гребнера.

Максимальный по числу элементов среди всех независимых наборов соответствует подмногообразию наибольшей размерности и, следовательно, число его элементов и есть размерность идеала. Все другие максимальные (в том смысле, что добавление еще одной переменной к набору нарушает условие (1)) независимые наборы соответствуют [12] простым подидеалам, а число их в наборе есть не что иное, как размерность соответствующего подидеала.

Поэтому, если рассматривать переменные, входящие в любой из максимальных независимых наборов в качестве свободных параметров, то по отношению к остальным переменным исходный полиномиальный идеал является нульмерным, и его базис Гребнера в лексикографическом упорядочении имеет треугольную форму.

2.3. Декомпозиция полиномов

Задача нахождения корней полинома одной переменной может быть существенно облегчена, если этот полином может быть выражен через полиномы более низкой степени. Это обычно достигается путем факторизации или декомпозиции полиномов. В настоящем разделе мы поясним понятие декомпозиции полиномов.

Определение. Пусть $P(x)$– некоторый полином степени n из $K[x]$, где $K[x]$– кольцо полиномов от одной переменной над некоторым полем K характеристики нуль. Тогда, если существуют такие $F(x), h(x) \in K[x]$ степени больше, чем 1 и

$$P(x) = F(h(x)), \tag{2}$$

то представление (2) называется декомпозицией полинома $P(x)$. В противном случае мы будем называть $P(x)$ простым.

Определение. Если $P(x)$ может быть представлен в виде

$$P(x) = P_1(P_2 \ldots (P_m(x)) \ldots)), \tag{3}$$

где $i = 1, \ldots, m$, степень $P_i(x)$ больше 1 и $P_i(x)$ простой, тогда мы называем (3) простой декомпозицией полинома $P(x)$.

В настоящей версии пакета ASYS реализован простой алгоритм декомпозиции полиномов, предложенный в [13]. Алгоритм основан на следующем немедленном следствии из (2):

$$P'(x) = F'((h(x)) * h'(x).$$

При этом для работы алгоритма требуется однократная факторизация исходного полинома, после чего представление (3) строится рекурсивным образом посредством операций дифференцирования и полиномиального деления.

2.4. Однородность

Рассмотрим систему полиномиальных уравнений вида

$$f_m = \sum_{(i)} a_{m,(i)} x^{(i)} = 0, \quad m = 1, 2, \ldots, M, \tag{4}$$

где использованы мультииндексные обозначения $(i) = (i_1, \ldots, i_n)$, $a_{m,(i)} = a^{(m)}_{i_1,\ldots,i_n}$, $x^{(i)} = x_1^{i_1} \cdots x_n^{i_n}$.

Определение. Систему полиномиальных уравнений (4) будем называть однородной, если под действием масштабных преобразований

$$x_i \longrightarrow \alpha_i x_i, \quad x^{(i)} \longrightarrow \alpha^{(i)} x^{(i)}, \quad \alpha_i > 0, \tag{5}$$

таких, что, по меньшей мере, один из факторов $\alpha_i \neq 1$, каждый из мономов любого отдельно взятого полинома $f_m(x_1, \ldots, x_n)$ в (4) приобретает один и тот же масштабный множитель. Другими словами, для любых двух мономов (i) и (j) полинома f_m выполняется равенство

$$\alpha^{(i)} = \alpha^{(j)} \quad \longleftrightarrow \quad \sum_{k=1}^{n} (i_k - j_k)\tilde{\alpha}_k = 0, \quad \tilde{\alpha}_k = \log \alpha_k.$$

Следует отметить, что общем случае масштабные факторы $\alpha^{(i)}$, возникающие в разных полиномах f_m, могут быть различными.

Приравнивая масштабные факторы для различных термов каждого из полиномов системы, получаем следующую систему линейных уравнений с целыми коэффициентами:

$$\sum_{k=1}^{n} z_{ik}\tilde{\alpha}_k = 0, \quad z_{ij} \in Z \ . \tag{6}$$

Ясно, что система (6) всегда имеет тривиальное решение $\tilde{\alpha}_k = 0$ или $\alpha_k = 1$, $k = 1, 2, \ldots, n$.

С другой стороны, при наличии нетривиальных решений система (6) имеет бесконечно много решений. При этом часть переменных $\tilde{\alpha}_i$ может рассматриваться в качестве свободных параметров. Максимально возможный набор таких свободных переменных есть не что иное, как максимально независимый набор (см. раздел 2.2) по отношению к идеалу, генерируемому левыми частями системы (6).

Определение. Переменные x_i, соответствующие выбранным в качестве независимых, т.е. свободных, переменных $\tilde{\alpha}_i$ системы (6), мы будем называть однородными переменными исходной полиномиальной системы (4), а их число ее степенью однородности.

Ясно, что степень однородности алгебраической системы уравнений меньше либо равна размерности соответствующего полиномиального идеала. Это означает, что для однородных систем либо все переменные, входящие в максимальный по числу элементов независимый набор (раздел 2.2), либо определенная их часть могут быть рассмотрены в качестве однородных.

Ниже показано как наличие независимых и/или однородных переменных, позволяет свести исходную систему к эквивалентному набору подсистем с меньшим числом неизвестных и соответствующим нульмерным идеалам. Соответствующие алгоритмы встроены в пакет ASYS, описанный в разделе 4.

3 Редукция полиномиальных систем с бесконечным числом решений

3.1 Редукция по максимальным наборам независимых переменных

В разделе 2.2 уже отмечалось, что в случае, если все переменные, входящие в какой либо из максимально независимых наборов переменных, рассматривать в качестве свободных параметров, полиномиальный идеал становится нульмерным по отношению к остальным переменным.

Это обстоятельство позволяет полностью автоматически, в рамках техники базисов Гребнера, проверить разрешимость исходной системы полиномов

$$f_i \in F \subset K[x_1, \ldots, x_n],$$

установить, конечно или бесконечно множество их общих корней и осуществить, в последнем случае, редукцию к эквивалентному конечному набору подсистем по следующей схеме, реализованной в пакете ASYS:

- вычисление базиса Гребнера для исходной системы по алгоритму Бухбергера (раздел 2.1) при некотором заранее выбранном упорядочении переменных. Если при этом оказывается, что

  $$1 \in \text{Базис Гребнера } \{F\},$$

  то это означает несовместность системы [2] и процесс решения оканчивается;

- вычисление всех максимально независимых наборов по лидирующим термам базиса Гребнера, основанное на соотношении (1). В пакете ASYS реализован несколько иной вариант алгоритма построения независимых наборов, чем в работе [12], более эффективный по числу полиномиальных переменных;

- перебор всех максимально независимых наборов и, при рассмотрении переменных каждого из наборов в качестве свободных параметров, построение базиса Гребнера в лексикографическом упорядочении по остальным переменным.

Выполнение данной редукции в пакете ASYS осуществляется при поднятии флагов SETDIM и SETGB, описанных в разделе 4.3. Подъем только первого флага позволяет находить размерность и все максимально независимые наборы без последующего выполнения редукции исходной системы.

В результате такой редукции получается набор подсистем треугольного вида, коэффициенты которых являются рациональными функциями некоторого числа параметров. Решение таких систем представляет собой отдельную проблему, для которой пока не известен сколь-нибудь общий алгоритм. На практике весьма полезной может быть попытка факторизации и/или декомпозиции входящих в эти системы полиномов одной переменной.

3.2. Редукция по однородным переменным

Покажем, что в случае, когда рассматриваемая система алгебраических уравнений является однородной (раздел 2.4), она может быть сведена к системам меньшего числа переменных совсем другим образом, отличным от рассмотренного в предыдущем разделе.

Опишем такую редукции в виде легко алгоритмизуемой последовательности шагов, встроенной в пакет ASYS в виде рекурсивной процедуры, выполнение которой контролируется флагом SCALE (раздел 4.3):

1. **Генерация и решение линейной системы уравнений (6)**
   Для решения этой системы можно использовать любой подходящий метод, например, метод Гаусса.
   В пакете ASYS и на этом шаге используется техника базисов Гребнера. В данном линейном случае алгоритм Бухбергера превращается в алгоритм Гаусса [2]. Кроме того, в качестве однородных можно выбрать те переменные x_i, которые соответвуют любому из максимальных независимых наборов идеала, генерируемого левыми частями (6) при упорядочении переменных, заданном для исходной системы (4).
   Если на этом шаге выясняется что система неоднородна, т.е. имеет только тривиальное решение $\tilde{\alpha}_i = 0$, процесс редукции оканчивается, оставляя исследуемую систему в первоначальном виде (4).
   В противном случае на этом шаге получается набор из $l > 0$ величин $\tilde{\alpha}_i$, в качестве которого, без ограничения общности, будем брать $\tilde{\alpha}_1, \ldots, \tilde{\alpha}_l$. Соответственно, однородными переменными системы (4) будут $x_1, \ldots, x_l$. Решение системы получается в виде

$$\tilde{\alpha}_j = \sum_{k=1}^{l} q_{jk}\tilde{\alpha}_k, \quad q_{ij} \in Q, \quad j = l+1, \ldots, n. \tag{7}$$

2. **Преобразование переменных**
   Из уравнений (7) с учетом того, что $\alpha_i = e^{\tilde{\alpha}_i}$ получаются выражения для масштабных факторов α_j, $j > l$ преобразования (5)

$$\alpha_j = \prod_{k=1}^{l} \alpha_k^{q_{ik}}$$

   в терминах произвольных $\alpha_1, \ldots, \alpha_l$.
   Под действием преобразования неоднородных переменных x_j, $j > l$

$$x_j = (\prod_{k=1}^{l} x_k^{q_{jk}})\tilde{x}_j \tag{8}$$

   каждый моном m-го уравнения системы (4) преобразуется мультипликативно

$$x^{(i)} = K_m \tilde{x}^{(i)}$$

   с одним и тем же множителем

$$K_m(x_1, \ldots, x_l) = \prod_{k=1}^{l} x_k^{i_k + \sum_{j=l+1}^{n} q_{jk} i_j},$$

   зависящим только от однородных переменных. Соответственно, система (4) переписывается в виде

$$f_m(x_1, \ldots, x_n) = K_m(x_1, \ldots, x_l)\tilde{f}_m(\tilde{x}_{l+1}, \ldots, \tilde{x}_n). \tag{9}$$

3. **Редукция при ненулевых однородных переменных**
   Пусть все $x_i \neq 0$, $i = 1, \ldots, l$. Тогда мультипликативные факторы K_m в (9) можно опустить и система уравнений (4) с n неизвестными сводится к системе

$$\tilde{f}_m(\tilde{x}_{l+1}, \ldots, \tilde{x}_n) = 0, \ m = 1, \ldots, M) \tag{10}$$

   с $n - l$ неизвестными. Эту систему можно исследовать на совместность и решать построением ее базиса Гребнера. Возврат к старым переменным осуществляется преобразованием, обратным к (8). При этом однородные переменные следует рассматривать как свободные параметры со специальным анализом их нулевых значений в соответствии с предписанием следующего шага.

4. **Редукция по различным наборам нулевых значений однородных переменных**
   Рассмотрение по очереди всех различных наборов однородных переменных,в которых, по меньшей мере, одна обращается в нуль. При этом все нулевые значения однородных переменных рассматриваемого набора подставляются в исходную систему (4) и процесс редукции полученной новой

системы с уменьшенным числом переменных осуществляется с самого первого шага.

Все несовместные наборы с нулевыми однородными переменными отсеиваются на шаге 3. Те же из них, которые зануляют все полиномы исходной системы,немедленно помещаются в список решений, после чего осуществляется переход к следующему набору с нулями.

В общем случае зануление части переменных в однородной алгебраической системе может привести к появлению дополнительных однородных переменных, что, в свою очередь, приводит к дальнейшей редукции.

Отметим, что редукция по однородным переменным может осуществляться совместно с редукцией по максимально независимым наборам, что часто оказывается полезным для решения систем, получающихся на шаге 3.

4. Описание пакета ASYS

4.1. Общая структура пакета

Разработанный авторами пакет программ реализован в системе аналитических вычислений REDUCE 3.4 [11] для IBM PC AT/386 в среде MS-DOS и включает следующие основные блоки, функциональное назначение которых соответствует алгоритмам и вычислительным схемам, рассмотренным в разделах 2 и 3:

- построение базиса Гребнера;
- определение размерности полиномиального идеала, нахождение всех наборов независимых переменных и редукция по этим наборам;
- анализ системы на однородность и редукция по однородным переменным;
- декомпозиция полиномов.

Пакет программ написан на языке RLISP и содержит около 1500 строк текста.

4.2. Дистрибутивное представление полиномов

При реализации алгоритма существенным является выбор такой структуры данных, которая бы обеспечивала его максимальную эффективность с точки зрения вычислительных ресурсов. Используемое в системе REDUCE рекурсивное представление полиномов не обеспечивает достаточную эффективность. Поэтому в настоящее время общепринятым является следующий подход, которому следовали и авторы этой статьи. Исходная система полиномов, заданная в префиксной форме, перед началом работы основного алгоритма переводится в некоторое внутреннее представление, которое называется дистрибутивным или рас-

пределенным [1]. По окончании работы основного алгоритма осуществляется обратное преобразование.

Пусть полином задан в виде $f = \sum_{i=1}^{m} c_i u_i = \sum T_i$, где u_i– произведение степеней вида $x_1^{i_1}, \ldots, x_n^{i_n}$, c_i– коэффициенты при соответствующих степенях. Тогда в дистрибутивном представлении этот полином будет иметь вид $((T_1)(T_2)\ldots(T_m))$, где $T_i = (D_i.c_i)$– точечная пара, $D_i = (i_1 i_2 \ldots i_n)$– список степеней u_i, $c_i =< s.q. >$ – коэффициент при u_i в форме стандартного отношения.

Пример

Пусть задан полином $f = x^2y + 1/2xy^2$. Тогда в дистрибутивном представлении он будет иметь вид

$$f = (((2\ 1)\ 1\ .\ 1)\ (1\ 2)\ 1\ .\ 2)).$$

4.3. Загрузка пакета, вызов основной процедуры и управляющие флаги

Загрузка пакета в компилированном виде осуществляется по команде
LOAD ASYS;

Вызов основной процедуры осуществляется командой
GROEBNER(POLYS,VARS);
первый аргумент которой является списком полиномов системы (4)
POLYS:=$\{f_1, \ldots, f_M\}$;, а второй - списком переменных, расположенных в порядке убывания старшинства по упорядочению, задаваемому пользователем с помощью описанного ниже флага LEXORD. Так при $x_1 > x_2 > \cdots > x_n$ аргумент VARS следует задавать в виде
VARS:=$\{X_1, \ldots, X_n\}$;

В пакете используются следующие флаги: LEXORD, SETORD, SETDIM, SETGB, SCALE, SCALETST, UNIBAS. Каждому из флагов соответствует своя внутренняя переменная,значение которой до вызова программы должно быть определено. Если пользователем не было установлено новое значение, в программе используется значение переменной, заданное по умолчанию. Вызов процедуры LISP DEFSW(); позволяет вернуться к заданному по умолчанию стандартному значению флагов.

LEXORD. Флаг LEXORD определяет используемое в программе линейное упорядочение < произведений степеней $x_1^{i_1} \ldots x_n^{i_n}$. Если флаг по команде ON LEXORD; установлен, используется лексикографическое упорядочение. При опускании флага LEXORD по команде OFF LEXORD; используется упорядочение сначала по полной степени,а затем уже обратное лексикографическое. По умолчанию используется лексикографическое упорядочение.

SETORD. Время вычисления базиса Гребнера существенно зависит от выбранного упорядочения переменных. Флаг SETORD позволяет выбрать эвристически оптимальное упорядочение переменных в соответствии с алгоритмом, изло-

женным в [14]. В 85-90% рассмотренных нами случаев выбранное таким образом упорядочение действительно является оптимальным. Если флаг установлен (по команде ON SETORD;), используется автоматический выбор упорядочения переменных. В этом случае значение соответсвующей переменной !*SETORD равно T. По умолчанию (OFF SETORD;) сохраняется первоначально заданное упорядочение переменных.

SETDIM. Флаг SETDIM позволяет определить размерность полиномиального идеала и найти все наборы независимых переменных. По умолчанию - OFF SETDIM; .

SETGB. Флаг SETGB, если он установлен, последовательно фиксирует каждый набор независимых переменных в качестве свободных параметров и строит базис Гребнера, оставляя порядок остальных переменных неизменным.В результате мы получаем набор базисов Гребнера, соответствующих нульмерным идеалам. По умолчанию - OFF SETGB;.Следует отметить, что если флаг SETGB установлен, то нет необходимости устанавливать флаг SETDIM - это делается автоматически.

SCALE. В случае, если исходная система полиномов является однородной, можно существенно (иногда на несколько порядков) сократить время счета, если использовать флаг SCALE. Если флаг SCALE установлен, в результате мы получаем набор подсистем, эквивалентных исходной. По умолчанию - OFF SCALE;.

SCALETST. В случае, если требуется только проверка системы на однородность, без построения базиса Гребнера и поиска решений, следует установить флаг SCALETST. Если флаг SCALETST установлен, результатом работы пакета является нахождение набора однородных переменных (если они есть, NIL - в противном случае). По умолчанию - OFF SCALETST; .

UNIBAS. Если флаг SCALE установлен, в результате мы получаем набор подсистем, эквивалентных исходной. Если мы хотим получить результат в виде одного базиса Гребнера, необходимо установить флаг UNIBAS. При этом объединенный базис Гребнера строится методом, предложенным в [15]. По умолчанию - OFF UNIBAS; .

5. Примеры полиномиальных систем и сравнение с другими пакетами

В настоящем разделе приведены результаты сравнительного счета различных примеров с помощью пакета ASYS, стандартного пакета GROEBNER системы Reduce 3.4 [11, 18] и специализированных систем FELIX [17] и AlP*I*[19], предназначенных для решения различных алгебраических задач.

Примеры I–III взяты из работ [7, 16] и возникают при исследовании интегрируемости нелинейных эволюционных уравнений, играющих важную роль в математической физике и прикладной математике. Хорошо известные в литературе [6, 19] примеры нульмерных идеалов IV-V являются общепризнанными тестами эффективности пакетов вычисления базисов Гребнера. Заметим, что последние два примера, отличающиеся всего лишь одним слагаемым, приводят к резко отличающимся по объему вычислениям.

Все расчеты с использованием пакетов ASYS и GROEBNER проводились на IBM PC AT/386 с тактовой частотой 25 МГц и памятью 8 Мб. Результаты сравнения приведены в Табл. 1 и 2. Вычисления с системой FELIX выполнялись на IBM PC AT/386 с тактовой частотой 33 МГц и памятью 8 Мб; соответствующие времена счета, приведенные в Табл. 1 и 2, пересчитаны с учетом разницы в быстродействии компьютеров. При этом с помощью системы FELIX на IBM PC AT/486 с тактовой частотой 33 МГц и памятью 64 Мб все-таки удалось построить лексикографический базис Гребнера для примера III при упорядочении переменных: $a_0 > a_1 > \ldots > a_4 > b_0 > \ldots > b_4$. Время счета составило около 70 часов, объем базиса Гребнера – 3 Мб. Для системы AlP*I* время счета было взято из работы [19] и пересчитано с учетом разницы в быстродействии компьютеров.

Сравнение проводилось при двух различных типах упорядочения переменных, показанных в Табл. 1 и 2. Ключ **Lex** означает использование чисто лексикографического упорядочения, ключ **Lex+Scale** означает использование этого же упорядочения с редукцией по однородным переменным, ключ **DegRevLex** означает использование упорядочения по полной степени, а затем обратного лексикографического. Вместе с явным видом алгебраической системы при каждом из рассмотренных примеров указаны: выбранный порядок старшинства переменных; размерность полиномиального идеала, а для нульмерных идеалов - число их корней; степень однородности.

Для примера III мы приводим явный вид одной из подсистем, полученных редукцией по однородным переменным.

Пример I

Используемое упорядочение переменных: $l_7 > l_6 > l_5 > l_4 > l_3 > l_2 > l_1$.
Размерность полиномиального идеала: 3.
Степень однородности: 1.

$$
\begin{aligned}
&l_1(l_4 - l_5/2 + l_6) = (2/7l_1^2 - l_4)(-10l_1 + 5l_2 - l_3) = 0,\\
&(2/7l_1^2 - l_4)(3l_4 - l_5 + l_6) = 0,\\
&a_1(-3l_1 + 2l_2) + 21a_2 = a_1(2l_4 - 2l_5) + a_2(-45l_1 + 15l_2 - 3l_3) = 0,\\
&2a_1l_7 + a_2(12l_4 - 3l_5 + 2l_6) = b_1(2l_2 - l_1) + 7b_2 = b_1l_3 + 7b_2 = 0,\\
&b_1(-2l_4 - 2l_5) + b_2(2l_2 - 8l_1) + 84b_3 = 0,\\
&b_1(8/3l_5 + 6l_6) + b_2(11l_1 - 17/3l_2 + 5/3l_3) - 168b_3 = 0,\\
&15b_1l_7 + b_2(5l_4 - 2l_5) + b_3(-120l_1 + 30l_2 - 6l_3) = 0,
\end{aligned}
$$

$$-3b_1l_7 + b_2(-l_4/2 + l_5/4 - l_6/2) + b_3(24l_1 - 6l_2) = 0,$$
$$3b_2l_7 + b_3(40l_4 - 8l_5 + 4l_6) = 0,$$

где

$$a_1 = -2l_1^2 + l_1l_2 + 2l_1l_3 - l_2^2 - 7l_5 + 21l_6,\ \ a_2 = 7l_7 - 2l_1l_4 + 3/7l_1^3,$$
$$b_1 = l_1(5l_1 - 3l_2 + l_3),\ \ b_2 = l_1(2l_6 - 4l_4),\ \ b_3 = l_1l_7/2.$$

Пример II

Используемое упорядочение переменных: $l_3 > l_4 > l_1 > l_5$.
Размерность полиномиального идеала: 2.
Степень однородности: 1.

$$-2l_4^3l_1 + (3l_4^2l_1 - 2l_4^2 - 6l_4l_3l_1 + 6l_4l_3 + 6l_3^2l_1 - 6l_3^2)l_5 - l_4l_1l_5^2 = 0,$$
$$18l_4^3l_1^2 - 9l_4^3l_1 - 18l_4^2l_3l_1^2 + 18l_4^2l_3l_1 + 18l_4l_3^2l_1^2 - 18l_4l_3^2l_1 +$$
$$(-27l_4^2l_1^2 + 24l_4^2l_1 - 5l_4^2 + 63l_4l_3l_1^2 - 78l_4l_3l_1 + 15l_4l_3 - 63l_3^2l_1^2 +$$
$$78l_3^2l_1 - 15l_3^2)l_5 + 9l_4l_1^2l_5^2 = 0,$$
$$-8l_4^4l_1 + (6l_4^3l_1 - 6l_4^3 - 12l_4^2l_3l_1 + 12l_4^2l_3 + 12l_4l_3^2l_1 - 12l_4l_3^2)l_5 +$$
$$(5l_4^2l_1 - 4l_4^2 - 18l_4l_3l_1 + 18l_4l_3 + 18l_3^2l_1 - 18l_3^2)l_5^2 - 3l_4l_1l_5^3 = 0,$$
$$(3l_1 - 5)l_4^2l_3 - 15(l_1 - 1)l_4l_3^2 + 10(l_1 - 1)l_3^3 +$$
$$(l_4l_3 + 3l_3^2l_1 - 3l_3^2)l_5 - l_3l_1l_5^2 = 0.$$

Пример III

Используемое упорядочение переменных:
$a_2 > b_2 > a_4 > b_4 > a_1 > b_1 > a_3 > b_3 > a_0 > b_0$.
Размерность полиномиального идеала: 6.
Степень однородности: 3.

$$e_k = \hat{e}_k = 0, \quad (k = 1 \div 6),$$

где $\hat{e}_k = e_k\,|_{a_i \Leftrightarrow b_i}$ и

$$e_1 = a_1\ (a_3 - a_4) - a_4\ (b_3 - b_4)\ ,$$
$$e_2 = (2a_3 - a_4)\ y_1 - b_2y_2\ , \qquad y_1 = 6a_0a_3b_2 + (a_0 - b_0)\ (a_1^2 + a_4b_2)\ ,$$
$$e_3 = a_2y_1 - (2b_3 - b_4)\ y_2\ , \qquad y_2 = 6a_0a_2b_3 + (a_0 - b_0)\ (a_1a_2 + a_4b_1)\ ,$$
$$e_4 = 3a_0\ (a_2b_2 + a_3b_3) + (a_0 - b_0)\ (a_1 + b_3)\ a_4\ ,$$
$$e_5 = 2\ (2a_0^2 + 8a_0b_0 - b_0^2)\ a_3b_3 + 2\ (a_0 - b_0)\ (4a_0 - b_0)\ a_3b_4 -$$
$$6a_0\ (a_0 + 2b_0)\ a_2b_2 + (a_0 - b_0)^2\ (5a_1a_3 - 5a_1a_4 + a_4b_4) -$$
$$(a_0 - b_0)\ (7a_0 - b_0)\ a_4b_3\ ,$$
$$e_6 = 3a_0\ [\ (a_0 - b_0)^3 - 3a_0\ (a_0 + 2b_0)^2\ ]\ (a_2b_2 + a_3b_3) +$$
$$(a_0 - b_0)^3\ [\ 3a_0a_1a_3 - 2\ (2a_0 + b_0)a_1a_4\ ] + 9a_0^2\ (a_0 - b_0)$$
$$[\ (a_0 - b_0)\ a_4 - (a_0 + 2b_0)\ a_3\ ]\ b_4 - (a_0 - b_0)\ (2a_0^3 - 30a_0^2b_0 + b_0^3)\ a_4b_3\ .$$

Результат счета с использованием пакета ASYS и ключа **Lex+Scale** выдается в виде набора 76 подсистем, каждая из которых, после отделения ненулевого мультипликативного фактора K_m в соответствии с (9), т.е. преобразованная к виду (10), является базисом Гребнера в новых переменных $\tilde{x}_j$. Приведем явный вид одной из таких подсистем, соответствующей 3-мерному подидеалу с максимально независимым набором (b_0, a_3, b_3), совпадающим с набором однородных переменных для исходной системы:

$$\{a_2b_0b_3 - 1/3a_0a_3^2 - 5/3a_3^2b_0,$$
$$b_2a_3b_0 + 1/3a_0b_3^2 + 2/3b_0b_3^2,$$
$$a_4b_0 + 1/3a_0a_3 - 1/3a_3b_0,$$
$$b_4b_0 - 1/3a_0b_3 - 8/3b_0b_3,$$
$$a_1 + b_3,$$
$$b_1 + a_3,$$
$$a_0^2 + 7a_0b_0 + b_0^2\}\ .$$

Пример IV

Используемое упорядочение переменных: $x_1 > x_2 > x_3 > x_4 > x_5$.
Размерность идеала: 0 (число корней - 70).

$$x_1 + x_2 + x_3 + x_4 + x_5 = 0,$$
$$x_1x_2 + x_2x_3 + x_3x_4 + x_4x_5 + x_5x_1 = 0,$$
$$x_1x_2x_3 + x_2x_3x_4 + x_3x_4x_5 + x_4x_5x_1 + x_5x_1x_2 = 0,$$
$$x_1x_2x_3x_4 + x_2x_3x_4x_5 + x_3x_4x_5x_1 + x_4x_5x_1x_2 + x_5x_1x_2x_3 = 0,$$
$$x_1x_2x_3x_4x_5 - 1 = 0.$$

Пример V

Используемое упорядочение переменных: $x_4 > x_1 > x_2 > x_5 > x_3$.
Размерность идеала: 0 (число корней - 64).

$$x_1 + x_2 + x_3 + x_4 + x_5 = 0,$$
$$x_1x_2 + x_2x_3 + x_3x_4 + x_4x_5 + x_5x_1 = 0,$$
$$x_1x_2x_3 + x_2x_3x_4 + x_3x_4x_5 + x_4x_5x_1 + x_5x_1x_2 = 0,$$
$$x_2x_3x_4 + x_2x_3x_4x_5 + x_3x_4x_5x_1 + x_4x_5x_1x_2 + x_5x_1x_2x_3 = 0,$$
$$x_1x_2x_3x_4x_5 - 1 = 0.$$

Таблица 1
Сравнительное время счета примеров I–III с идеалами положительной размерности

Пакет	Ключи	I	II	III
ASYS	Lex	155.5 с	34.8 с	переполнение памяти
ASYS	Lex+Scale	25.2 с	2.9 с	147.6 с
GROEBNER	Lex	10.8 с	7.3 с	переполнение памяти
FELIX	Lex	21.5 с	17.7 с	переполнение памяти
AlP*I*	Lex	71.7 с	21.8 с	–

Таблица 2
Сравнительное время счета примеров IV-V с идеалами нулевой размерности

Пакет	Ключи	IV	V
ASYS	DegRevLex	41.5 с	839.9 с
GROEBNER	DegRevLex	10.3 с	1180.6 с
FELIX	DegRevLex	19.8 с	397.3 с
AlP*I*	DegRevLex	95.5 с	–

6. Заключение

Описанные выше и встроенные в пакет ASYS редукционные процедуры по максимальным независимым наборам и однородным переменным на практике оказываются весьма полезными и позволяют резко упростить исходную задачу при ее обработке по методу базисов Гребнера. Такая эффективность обусловлена уменьшением числа переменных, по которому, как уже отмечалось во введении, алгоритм Бухбергера имеет субэкспоненциальное поведение.

Однако, если в случае редукции по максимальным независимым наборам получаются подсистемы, коэффициенты которых принадлежат расширению исходного поля коэффициентов K рациональными функциями независимых переменных, то редукция по однородным переменным не выводит из начального поля K. Кроме того, для последней не требуется построения базиса Гребнера для исходной системы. Поэтому редукция по однородным переменным является особенно эффективным средством упрощения и решения алгебраических систем, обладающих нетривиальными свойствами однородности. Яркой иллюстрацией сказанному служит пример III раздела 5, решение которого в рамках техники базисов Гребнера и без учета его свойств однородности требует несоизмеримо большего объема вычислений.

В целом же использование различных редукционных процедур и специальных свойств рассматриваемой алгебраической системы в рамках техники базисов Гребнера представляется весьма перспективным для повышения ее практической значимости. В качестве важных примеров отметим использование полиномиальной факторизации, встроенной в стандартный пакет GROEBNER системы Reduce [18] и применимой как для нульмерных идеалов, так и для идеалов положительной размерности, а также учет дискретных симметрий свойств алгебраических систем [20].

Следует также отметить, что отдельные редуцированные подсистемы, хотя и эквивалентные, в совокупности, исходной системе, могут совпадать друг с другом либо давать одни и те же решения, по-разному параметризованные независимыми переменными. Отбор минимального набора подсистем, эквивалентного исходной системе, представляет собой отдельную проблему, алгоритмически пока не решенную.

После редукции исходной системы к системам треугольного вида задача сводится к последовательному нахождению корней полиномов одной переменной. Заметим, что для примеров I–III раздела 5 редуцированные системы имеют достаточно простую структуру (см., например, одну из подсистем для примера III, приведенную в разделе 5) и позволяют найти явные алгебраические выражения для всего пространства корней. Такое замечательное свойство этих систем является, по-видимому, следствием интегрируемости соответствующих нелинейных эволюционных уравнений [7, 8].

В самом общем случае, как уже отмечалось выше, на этом шаге полезно использовать полиномиальную факторизацию и/или декомпозицию. Используемый в настоящей версии пакета ASYS алгоритм полиномиальной декомпозиции [13] применим для полиномов над полем Q. В будущих версиях пакета ASYS предполагается использовать недавно предложенный алгоритм [21], в отличие от всех других известных алгоритмов, не требующий факторизации и применимый к полиномам, заданным над различными полями и кольцами. В частности, его можно использовать после редукции по максимальным независимым наборам и триангуляции получающихся подсистем построением их лексикографических базисов Гребнера. Кроме того, алгоритм работы [21] обладает полиномиальной сложностью, что означает его высокую эффективность для полиномов больших степеней.

Авторы выражают благодарность В.В.Менькову за улучшение алгоритма вычисления максимальных независимых наборов, В.Лассперу за многочисленные полезные обсуждения, Й.Апелю и У.Клаусу за вычисления примеров раздела 5 с помощью системы FELIX, а также М.Джюсти, Д.Дюваль, Х.-Г.Гребе, Г.Меленку и Т.Мора за ряд важных замечаний и рекомендаций.

Литература

[1] Дэвенпорт Дж., Сире И., Турнье Э. Компьютерная алгебра. Системы и алгоритмы алгебраических вычислений. М., "Мир", 1991.

[2] Бухбергер Б. Базисы Гребнера. Алгоритмический метод в теории полиномиальных идеалов. В кн.: Компьютерная алгебра. Символьные и алгебраические вычисления. М., Мир, с.331-372, 1986.

[3] Gebauer R., Möller H.M. On an Installation of Buchberger's Algorithm, J. Symb. Comp. 6, 275-286, 1988.

[4] Латышев В.Н. Комбинаторная теория колец, стандартные базисы. Из-во МГУ, Москва, 1988.

[5] Михалев А.В., Панкратьев Е.В. Компьютерная алгебра. вычисления в дифференциальной и разностной алгебре. Из-во МГУ, Москва, 1989.

[6] Faugère J.C., Gianni P., Lazard D., Mora T. Efficient Computation of Zero-Dimensional Gröbner Bases by Change of Ordering, Technical Report LITP 89-52, 1989.

[7] Gerdt V.P., Khutornoy N.V., Zharkov A.Yu. Solving Algebraic Systems Which Arise as Necessary Integrability Conditions for Polynomial-Nonlinear Evolution Equations. In: Computer Algebra in Physical Research (Shirkov D.V.,Rostovtsev V.A.,Gerdt V.P., eds.), World Scientific, Publ. Co., Singapore, 1991, pp.321-328.

[8] Gerdt V.P. Computer Algebra Tools for Higher Symmetry Analysis of Nonlinear Evolution Equations. JINR E5-91-402, Dubna, 1991.

[9] Lassner W. Symmetrien von Differentialgleichungen und computergestützte Identifikation und Klassifikation von Lie-Algebren. To appear in Proceedings of V. Internationales Kolloquium über Aktuelle Problem der Rechentechnik (19-22 March 1991, Dresden).

[10] Lloyd N.G., Pearson J.M. REDUCE and the Bifurcation of Limit Cycles, J. Symb. Comp. 9, 215-234, 1990.

[11] Hearn A.C. REDUCE User's Manual. Version 3.4. The Rand Corporation, Santa Monica, 1991.

[12] Kredel H., Weispfenning V. Computing Dimension and Independent Sets for Polynomial Ideals, J. Symb. Comp. 6, 231-247, 1988.

[13] Liu Zhoujun Algorithm of Decomposing High Degree Polynomials, Mathematics - Mechanization Research Preprints 2, 62-67, Institute of Systems Science, Beijing, 1987.

[14] Boege W., Gebauer R., Kredel H. Some Examples for Solving Systems of Algebraic Equations by Calculating Groebner Bases, J. Symb. Comp. 2, 83-98, 1986.

[15] Gianni P., Trager B., Zacharias G. Gröbner Bases and Primary Decomposition of Polynomial Ideals, J. Symb. Comp. 6, 149-167, 1988.

[16] Gerdt V.P., Zharkov A.Yu. Computer Classification of Integrable Coupled KDV-Like Systems, J. Symb. Comp., 10, 203-207, 1990.

[17] Apel J., Klaus U. FELIX: an assistant for algebraists, Proceedings of ISSAC'91 (Watt S.M., ed.), ACM Press, 1991, pp.382-389.

[18] Melenk H., Möller H.M., Neun W. Symbolic Solution of Large Stationary Chemical Kinetics Problems, Impact of Computing in Science and Engineering 1, 138-167, 1989.

[19] Giovini A., Mora T., Niesi G., Robbiano L., Traverso C. "One sugar cube, please" OR Selection strategies in the Buchberger algorithm, Proceedings of ISSAC'91 (Watt S.M., ed.), ACM Press, 1991, pp.49-54.

[20] Gatermann K. Symbolic Solution of Polynomial Equation Systems with Symmetry. Preprint SC 90-3, Konrad-Zuse-Zentrum für Informationstechnik, Berlin, 1990.

[21] Kozen D., Landau S. Polynomial Decomposition Algorithms. J. Symb. Comp., 7, 445-456, 1989.

Рукопись поступила в издательский отдел
6 марта 1992 года.