О СИНТАКСИЧЕСКОМ АНАЛИЗЕ И СВОЙСТВАХ ФОРМАЛЬНЫХ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

К. В. Сафонов
Красноярский государственный технический университет, Россия
e-mail: safonov@fivt.krasn.ru

The algorithm of deadlock-free syntactical analysis of context-free languages is suggested. Kronecker criteria on rationality is generalized for power series of algebraic functions, in principle this result provides a possibility to research the interior properties of the context-free languages (programming languages).

Введение

Настоящая статья является продолжением работы [1], посвященной вычислительному распознаванию контекстно-свободных языков (кс-языков) и контекстно-свободных грамматик (кс-грамматик), порождающих эти языки [2–6]. Исследование ведется применительно к языкам программирования, поскольку общим свойством таких языков является их принадлежность классу кс-языков, определяемых следующим образом.

Пусть $\{x_1, \ldots, x_n\}$ — множество терминальных символов языка (используемых, например, при написании программ, в том числе операторов языка программирования, букв, цифр и т. д.), а $\{z_1, \ldots, z_n\}$ — множество нетерминальных символов, необходимых для задания совокупности грамматических правил языка программирования (его грамматики). Грамматика кс-языка (языка программирования) задается в виде совокупности продукций — правил подстановки:

$$z_i \to f_{i1}(x, z), \ldots, z_i \to f_{iq_i}(x, z), i = 1, \ldots, m, \tag{1}$$

где $f_{ij}(x, z)$ — мономы от некоммутативных символьных переменных, как терминальных, так и нетерминальных (левая часть продукций содержит единственный символ и применяется независимо от его окружения — контекста). При этом z_1 — выделенный символ, означающий начало программы (подпрограммы). Грамматике ставится в соответствие система символьных уравнений

$$z_i = p_i(x, z), i = 1, \ldots, m, \tag{2}$$

где многочлен $p_i(x, z)$ равен формальной сумме мономов $f_{i1}(x, z) + \ldots + f_{iq_i}(x, z)$ [5]. Решение системы (1) можно получить методом последовательных приближений:

$$z_i^{(k+1)} = p_i(x, z^{(k)}), k = 0, 1, \ldots,$$

$$z^{(0)} = (0, \ldots, 0), z^{(k)} = (z_1^{(k)}, \ldots, z_m^{(k)}).$$

Неограниченное число итераций дает выражение переменных z_i в виде формальных степенных рядов от $x_1, \ldots, x_n$, в частности, формальный ряд

$$z_1 = \sum_i < z_1, w_i > w_i, \tag{3}$$

выражающий переменную z_1, и является данным кс-языком, а его слагаемые — всевозможные правильные предложения (программы) этого языка.

Изучение характеристических свойств кс-языков дает информацию о внутренних свойствах языков программирования, т. е. совокупности всевозможных "правильных" программ, написанных на этих языках, что представляет интерес для теоретического программирования.

Как отмечено, например, в [7, с. 236], "одной из важных проблем, подлежащих решению при разработке современных систем программирования, является проблема синтаксического анализа программ". Эта проблема состоит, во-первых, в распознавании правильности программы, т. е. ее принадлежности данному алгоритмическому языку (этап синтаксического контроля), во-вторых, в описании синтаксической структуры правильных программ (т. е. определении, какие грамматические продукции использовались при написании данной программы). Причем наибольший интерес представляют алгоритмы беступикового (безостановочного) синтаксического анализа [7, с. 253].

1. Алгоритм беступикового синтаксического анализа

Поставим в соответствие продукциям (1) систему символьных уравнений

$$z_i = P_i(t, z, x), \quad i = 1, \ldots, m, \tag{4}$$

где $P_i(t, z, x) = t_{i1} f_{i1}(x, z) + \ldots + t_{iq_i} f_{iq_i}(x, z)$, $i = 1, \ldots, m$, а t_{ij} — дополнительные символьные множители при мономах — "мономиальные метки". Систему (4) будем решать методом последовательных приближений:

$$z_i^{(k+1)} = P_i(t, z^{(k+1)}, x), \; z^{(0)} = 0, \quad i = 1, \ldots, m.$$

Очевидно, что решение системы (4) при $t_{ij} = e$, где e — пустая цепочка, совпадает с решением системы (2), в частности, если его первая компонента z_1 представлена формальным рядом

$$z_1 = \sum_j < z_1, u_j > u_j,$$

где u_j — мономы от переменных $x = (x_1, \ldots, x_m)$, $t = (t_{ij})$, то при $(t_{ij}) = (e)$ получается ряд (3).

Итерации метода последовательных приближений дают мономы возрастающей длины, и через конечное число шагов их степень (число символов $x_1, \ldots, x_n$) превысит степень заданного монома w (заданное конечное число символов программы). "Считывая" мономы необходимой степени и пропуская при этом метки t_{ij}, можно определить, есть ли среди них моном w. Каждая переменная t_{ij}, содержащаяся в соответствующем мономе, указывает, что при его выводе использована продукция $z_i \to t_{ij} f_{ij}(x, z)$. Действительно, применение

каждой продукции, имеющей в правой части мономиальную метку, добавляет моному эту метку в качестве множителя. Таким образом определяют, какие продукции и сколько раз использованы при выводе монома w (написание программы) с точностью до порядка их применения, что и решает проблему синтаксического анализа.

Итак, имеет место

Теорема 1. *Метод мономиальных меток, заключающийся в решении системы (4) методом последовательных приближений, позволяет провести за конечное число шагов беступиковый (безостановочный) синтаксический анализ любой заданной программы w.*

Пример. Проведем данным методом синтаксический анализ фрагмента арифметической программы $(((a+b)*b)*b)$, написанной на языке программирования, порожденном грамматикой, которая содержит систему продукций:

$$z_1 \to (z_1 * z_2), z_1 \to (a+b), z_1 \to b$$

[7, с. 251]. Здесь $+, *, (,), a, b$ — терминальные символы. Переобозначим их символами $x_1, x_2, x_3, x_4, x_5, x_6$ соответственно, тогда программа примет вид

$$x_3x_3x_3x_5x_1x_6x_4x_2x_6x_4x_2x_6x_4.$$

Запишем систему (4):

$$z_1 = t_{11}x_3z_1x_2z_2x_4 + t_{12}x_3x_5x_1x_6x_4, \quad z_2 = t_{21}x_6,$$

тогда методом последовательных приближений получаем, что

$$z^{(3)} = (t_{11}x_3t_{11}x_3t_{12}x_3x_5x_1x_6x_4x_2t_{21}x_6x_4x_2t_{21}x_6x_4 + t_{12}x_3x_5x_1x_6x_4;\ t_{21}x_6)$$

и первый моном при $t_{11} = t_{12} = t_{21} = e$ совпадает с искомой программой, причем число вхождений переменных t_{11}, t_{12}, t_{21} показывает, сколько раз использованы соответствующие продукции грамматики: продукция $z_1 \to (z_1 * z_2)$ использована дважды, продукция $z_1 \to (a+b)$ — один раз и продукция $z_2 \to b$ — два раза.

2. Обобщенный критерий Кронекера и кс-языки

Формальному степенному ряду поставим в соответствие ряд с комплексными переменными [1], задав отображение терминальных x_i и нетерминальных z_j символов из множества $X \cup Z$ в множество комплексных переменных, за которыми оставляем прежние обозначения — x_i и z_j соответственно, тогда $(x, z) \in \mathbb{C}^{n+m}_{x,z}$. Таким образом, получаем фиксированный гомоморфизм, который ставит в соответствие формальному ряду

$$r = \sum_i < r, w_i > w_i$$

степенной ряд от комплексных переменных

$$ci(r) = \sum_k a_k x^k,$$

называемый его *коммутативным образом*, где $k = (k_1, \ldots, k_n), a_k x^k = a_{k_1,\ldots,k_n} x_1^{k_1} \ldots x_1^{k_n}$, при этом

$$a_k = \sum_{\#x_1(w_i)=k_1,\ldots,\#x_n(w_i)=k_n} < r, w_i >,$$

где $\#\alpha(\beta)$ — число вхождений символа α в моном β. Заметим, что коммутативный образ кс-языка является функцией, голоморфной в непустой окрестности нуля [8].

Свойства кс-языка в значительной мере определяются свойствами его коммутативного образа, который является алгебраической функцией, поскольку удовлетворяет системе полиномиальных уравнений. Поэтому распознавание кс-языка тесно связано с установлением алгебраичности суммы кратного степенного ряда.

Вопрос о рациональности суммы степенного ряда с одной переменной

$$a(z) = \sum_{k\geq 0} a_k z^k \tag{5}$$

решается с использованием хорошо известного критерия Кронекера [9, с. 173], согласно которому рациональность равносильна тому, что при $j \geq j_0, l \geq l_0$ равны нулю определители Ганкеля

$$H_j^{(l)} = \begin{vmatrix} a_j & a_{j+1} & \dots & a_{j+l} \\ a_{j+1} & a_{j+2} & \dots & a_{j+l+1} \\ \vdots & \vdots & \dots & \vdots \\ a_{j+l} & a_{j+l+1} & \dots & a_{j+2l} \end{vmatrix}.$$

Эти равенства легко получаются, если умножить ряд (5) на многочлен, который является знаменателем его суммы — рациональной функции. Если сказать более точно, критерий Кронекера связывает рациональность с обращением в нуль определителей $H_0^{(l)}$, что равносильно исходной формулировке в силу свойств определителей Ганкеля.

Вопрос о существовании критерия, позволяющего определить алгебраичность суммы степенного ряда по его коэффициентам, оставался открытым, хотя класс алгебраических функций всегда вызывал большой интерес. Очевидно, что этот вопрос сводится к вопросу о том, как адекватно сформулировать условия на коэффициенты степенного ряда, которые получаются в результате его подстановки в полиномиальное уравнение, определяющее его сумму.

Решение последнего вопроса получается следующим образом. Обозначим

$$a_k^{(1)} = \sum_{i=0}^{k} a_i a_{k-i},\ a_k^{(j)} = \sum_{i=0}^{k} a_i a_{k-i}^{(j-1)},\quad j = 2, 3, \dots, k = 0, 1, \dots,$$

тогда имеет место

Теорема 2. *Для того чтобы функция (5) была алгебраической, необходимо и достаточно существование чисел d, j_0, l_0 таких, что при всех $j \geq j_0, l \geq l_0$ выполнены равенства $H_j^{(l)} = 0$, где*

$$H_j^{(l)}(d) = \begin{vmatrix} a_j & \dots & a_{j+l} & a_j^{(1)} & \dots & a_{j+l}^{(1)} & \dots & a_j^{(d-1)} & \dots & a_{j+l}^{(d-1)} \\ a_{j+1} & \dots & a_{j+l+l} & a_{j+1}^{(1)} & \dots & a_{j+l+1}^{(1)} & \dots & a_{j+1}^{(d-1)} & \dots & a_{j+l+1}^{(d-1)} \\ \vdots & \dots & \vdots & \vdots & \dots & \vdots & \dots & \vdots & \dots & \vdots \\ a_{j+q} & \dots & a_{j+q+l} & a_{j+q+l}^{(1)} & \dots & a_{j+q+l}^{(1)} & \dots & a_{j+q}^{(d-1)} & \dots & a_{j+q+l}^{(d-1)} \end{vmatrix}$$

— обобщенный определитель Ганкеля степени d, а число $q = (l+1)d - 1$.

Теорема 2 обобщает критерий Кронекера о рациональности: если функция — рациональная, то степень определяющего многочлена $d = 1$ и $H_j^{(l)}(d) = H_j^{(l)}$, т. е. в этом случае получается равенство нулю определителя Ганкеля.

Доказательство. Пусть $a(z)$ — голоморфная в окрестности нуля ветвь алгебраической функции $w = a(z)$, представленная рядом (5) и заданная неприводимым многочленом

$$S(w, z) = s_0(z) + s_1(z)w + \ldots + s_d(z)w^d, \tag{6}$$

где $s_j(z),\ j = 1, \ldots;\ d$ — многочлены; $S(a(z), z) = 0$.

Очевидно, что $a_k^{(j)}$ — тейлоровские коэффициенты функции $(a(z))^{j+1}$. Подставляя функцию $w = a(x)$ в тождество (6), получим тождество

$$s_1(z)(a(z)) + \ldots + s_d(z)(a(z))^d = -s_0(z). \tag{7}$$

Возьмем $l = \max_{0\le j\le d}\{l_j\}$ и, дополняя при необходимости коэффициенты многочленов $s_j(z)$ нулевыми значениями, будем считать, что $s_1(z) = q_0 + q_1 z + \ldots + q_l z^l,\ s_j(z) = q_0^{(j-1)} + q_1^{(j-1)} z + \ldots + q_1^{(j-1)} z^l,\ j = 2, \ldots, d$. Подставляя в тождество (7) ряды функций $(a(z)), (a(z))^2, \ldots, (a(z))^d$, получаем равенство

$$(q_0 + q_1 z + \ldots + q_l z^l) \sum_{k\ge 0} a_k z^k +$$

$$+(q_0^{(1)} + q_1^{(1)} z + \ldots + q_l^{(l)} z^l) \sum_{k\ge 0} a_k^{(1)} z^k + \ldots$$

$$\ldots + (q_0^{(d-1)} + q_1^{(d-1)} z + \ldots + q_l^{(d-1)} z^l) \sum_{k\ge 0} a_k^{(d-1)} z^k = -s_0(z).$$

Приравнивая к нулю коэффициент при z^j в левой части последнего равенства при $j \ge j_0 = \deg S_0(z)$, получим бесконечную систему равенств

$$q_l a_j + q_{l-1} a_{j+1} + \ldots + q_0 a_{j+l} + q_l^{(1)} a_j^{(1)} + q_{l-1}^{(1)} a_{j+1}^{(1)} + \ldots + q_0^{(1)} a_{j+l}^{(1)} + \ldots$$

$$\ldots + q_l^{(d-1)} a_j^{(d-1)} + q_{l-1}^{(d-1)} a_{j+1}^{(d-1)} + \ldots + q_0^{(d-1)} a_{j+l}^{(d-1)} = 0. \tag{8}$$

Рассматривая систему равенств (8) как бесконечную систему линейных уравнений относительно неизвестных $q_0, \ldots, q_l, \ldots, q_0^{(d-1)}, \ldots, q_l^{(d-1)}$ и выбирая произвольные $(l+1)d$ уравнений, получаем подсистему, имеющую ненулевое решение $(q_0, \ldots, q_l, \ldots, q_0^{(d-1)}, \ldots, q_l^{(d-1)})$. Последнее равносильно тому, что определитель системы (8) равен нулю, т. е. $H_j^{(l)}(d) = 0$ при всех $j \ge j_0$. Теорема 2 доказана. □

Данная теорема дает возможность сформулировать *алгоритм вычислительного распознавания* степенных рядов алгебраических функций следующим образом. Пусть задан общий вид коэффициентов ряда a_k как функций номера k. Тогда необходимо:

1) вычисляя определители Ганкеля как функции номеров j, l, проверить, равны ли они нулю;

2) вычислить коэффициенты $a_k^{(1)}$ как функции номера и проверить, равны ли нулю значения обобщенных определителей Ганкеля $H_j^{(l)}(2)$;

3) вычислить для следующего значения i коэффициенты $a_k^{(i)}$ и проверить, обращаются ли в нуль обобщенные определители Ганкеля $H_j^{(l)}(i+1)$.

Вычисления продолжаются до тех пор, пока не будет установлено обращение в нуль требуемых определителей либо установлено, что они как функции номеров не равны нулю (последнее будет означать, что функция трансцендентна).

Применение обобщенного критерия Кронекера по каждой переменной кратного ряда (ряда Гартогса) дает возможность сформулировать условие алгебраичности коммутативных образов кс-языков в виде бесконечной системы равенств нулю определителей, составленных из коэффициентов этих рядов.

Список литературы

[1] Сафонов К.В. О возможности вычислительного распознавания контекстно-свободных грамматик // Вычисл. технологии. 2005. Т. 10, № 4. С. 91–98.

[2] Хомский Н. Три модели для описания языка // Кибернетический сборник: Сб. перев. статей. М.: Иностр. лит. 1961. Вып. 2. С. 237–266.

[3] Хомский Н., Щютценберже М.П. Алгебраическая теория контекстно-свободных языков // Кибернетический сборник, нов. серия: Сб. перев. статей. М.: Мир, 1966. Вып. 2. С. 121–230.

[4] Гинзбург С. Математическая теория контекстно-свободных языков. М.: Мир, 1970.

[5] Salomaa A., Soittola M. Automata-Theoretics Aspects of Formal Power Series. N.Y.: Springer-Verlag, 1978.

[6] Гладкий А.В. Формальные грамматики и языки. М.: Наука, 1973.

[7] Глушков В.М., Цейтлин Г.Е., Ющенко Е.Л. Алгебра, языки, программирование. Киев: Наук. думка, 1974.

[8] Семенов А.Л. Алгоритмические проблемы для степенных рядов и контекстно-свободных грамматик // Докл. АН СССР. 1973. Т. 212, № 1. С. 50–52.

[9] Бибербах Л. Аналитическое продолжение. М.: Наука, 1967.

Поступила в редакцию 2 ноября 2005 г.