УДК 681.3.06 : 51

Л. В. ПЕВЗНЕР

НЕКОТОРЫЕ СВОЙСТВА ГРАФОВО-ПРОГРАММНЫХ ГРАММАТИК

В работе рассматривается сложность алгоритмов на графово-программных грамматиках (ГПГ) [1], предназначенных для описания и обработки графовых иерархических структур (ИГС) [2].

Пусть дана ГПГ $G=<V_T, V_N, P, R, \sigma>$, где V_T, V_N — терминальные и нетерминальные объекты; $P=\{(r)\varphi \rightarrow \psi, s^1, s^2, s^3\}$ — правила грамматики, $\varphi \in V_N$, $\psi \in (V_T \cup V_N)^*$; r, s^1, s^2, s^3 определяют соответственно номер данного правила, на которое осуществляется переход в случае применимости, неприменимости, конкретизации правила r; σ — аксиома грамматики.

Обозначим $B=\{b_i\}$, $i=\overline{1, n}$, $b_i \in V_T$ исходную ИГС, здесь $n=u+m$, u — количество узлов, m — количество связей и отношений иерархического подчинения в ИГС. G — ГПГ, нелеворекурсивная, левовыводимая (т. е. в G не существует правил вида $Q_1 \rightarrow Q_j\varphi_j$, $Q_j \rightarrow Q_1\varphi_{j+1}$, $j \geqslant 1$). Будем считать, что ГПГ не содержит правил вида: $Q_1 \rightarrow q\varphi_1$; $Q_1 \rightarrow q\varphi_2$; $q \in V_T^*$; $Q_1 \in V_N$; φ_1, $\varphi_2 \in (V_T \cup V_N)^*$. Это ограничение не является существенным, т. к. ГПГ G с такими правилами легко преобразовать в ГПГ G' с правилами $Q_1 \rightarrow qQ_2$; $Q_2 \rightarrow \varphi_1$; $Q_2 \rightarrow \varphi_2$, при этом $L(G)=L(G')$.

В [1] приведен алгоритм распознавания A_1, состоящий из следующих шагов: (1) удачное и (2) неудачное сравнение входного элемента с порожденным; (3) разрастание дерева; (4) переходы на правила из множеств s^1, s^2, s^3 на удачное и неудачное завершение разбора.

Шаг 4 выполняется ck раз, здесь c — константа, а k — количество шагов вида 1, 2, 3, применяемых при разборе B. Шаг 1 может быть выполнен не более n раз в искомой ГПГ G для B длиной n. Выходом при выполнении шага 2 могут быть переходы лишь на шаги 1, 3 либо на конфигурацию $B \in L(G)$. Пусть $|R|=t$, тогда для перехода на одно из вышеприведенных состояний необходимо выполнить не более t шагов вида 2; обратное неверно, так как нарушается условие нелеворекурсивности. Следовательно, при распознавании иерархической структуры B может быть выполнено не более nt шагов вида 2. Выходом при выполнении шага 3 могут быть лишь переходы на шаги 1, либо 2, либо на конфигурацию $B \in L(G)$.

Не нарушая общности, можно выделить в P правила двух видов:

(а) $Q_1 \rightarrow qQ_2\varphi$; (б) $Q_1 \rightarrow Q_2\varphi$, где $q \in V_T^* - \{\lambda\}$; $Q_1, Q_2 \in V_N$; $\varphi \in (V_T \cup V_N)^*$.

Правило вида (а) сведется к правилу вида (б) после выполнения шага 1. Правило вида (б) требует не более t шагов вида 3 для каждого нетерминального символа Q. Очевидно, что шаг 3 может применяться не более $t^2n+t^2(n-1)+\ldots+t^2\cdot 1=t^2\frac{(n+1)n}{2}$ раз. Справедливо следующее утверждение.

Теорема 1. Сложность $D(A_1)$ алгоритма A_1 в ГПГ G равна $O(n^2)$.

Настоящую оценку легко сделать линейной, если на правила грамматики $Q_1 \rightarrow Q_2\varphi$ наложить ограничение: в φ должно находиться не более f нетерминальных символов, где $1 \leqslant f \leqslant n$. Тогда оценка $t^2\frac{(n+1)n}{2}$ заменится на t^2fn, где ft^2 — константы; это ограничение несущественно, так как для ГПГ с правилами $Q \rightarrow q_1Q_1q_2Q_2\ldots q_fQ_fq_{f+1}Q_{f+1}q_{f+2}$, где $q_j \in V_T^*$, $j=\overline{1, f}$ легко построить ГПГ G' с правилами $Q' \rightarrow q_1'Q_1'q_2'\ldots q_{f+1}'$, что $L(G)=L(G')$. Для этого достаточно правило $Q \rightarrow q_1Q_1\ldots Q_{f+1}q_{f+2}$ заменить на правила: $Q \rightarrow q_1Q_1q_2\ldots q_{f+1}Q'$; $Q' \rightarrow Q_{f+1}q_{f+2}$. В [1, 2] предлагается набор операций над ИГС и алгоритм A_2 выполнения их с помощью ГПГ, для чего необходимо: (1) модифицировать правила ГПГ согласно [1], т. е. найти в таблице правило r и заменить его на соответствующее правило, реализующее данную операцию; (2) построить в преобразованной грамматике вывод, определяющий результат выполнения операции над ИГС.

Модификация правил осуществляется за c_1k и c_2u операций, соответственно; здесь c_1 и c_2 — константы; k — количество элементов в таблице операций; u — количество узлов в искомой ИГС. Вывод в ГПГ требует не более $c'n^2(c''n)$ операций (см. теорему 1). Отсюда следует

Теорема 2. Сложность $D(A_2)$ алгоритма A_2 выполнения операций над ИГС с помощью ГПГ определяется следующей формулой:

$$D(A_2) \leqslant \begin{cases} c'n^2, & c'\text{-константа} \\ c''n, & c''\text{-константа (если длина правой части правила ограничена)} \end{cases}$$

Для построения ГПГ G, описывающей ИГС Г, необходим размер памяти, не превышающий $c(u+m)$.

Оцениваемый здесь алгоритм A_1 может быть применен для синтаксического анализа и анализа контекстных условий языков программирования. Сложность хорошо известных алгоритмов синтаксического анализа — $O(n^3)$, $O(n^2)$, $O(n)$ (алгоритмы Янгера, Касами, Эрли, Кнута для $LR(k)$-грамматик) [3]. Следовательно, алгоритм A_1 при использовании его для проверки контекстных условий согласуется с лучшими алгоритмами синтаксического анализа.

Сравним ГПГ с лучшими другими представлениями графовых структур [4]: матрица смежности, список ребер, структура смежности. Память, требуемая для размещения подобных структур, пропорциональна u^2, $m[\lg u]$, $u+m$ соответственно. Причем время вычисления, требуемое для перехода от одного представления к другому, пропорционально u^2. Если принять к сведению, что память, занимаемая ГПГ, пропорциональна $u+m$, а время вычисления пропорционально $u+m$ (если длина правой части ограничена), то предлагаемое представление графовых структур в виде ГПГ не только согласуется с лучшими известными представлениями, но и превосходит их в случае ИГС, так как ИГС являются обобщением обычных графовых структур.

Рассмотрим связь ГПГ с другими представлениями графовых грамматик на примере WEB-грамматик [5], так как основное отличие этих представлений — в описании правил погружения подграфа β в граф φ, если к графу φ применяется правило $\alpha \to \beta$. Структура правила WEB-грамматики следующая: $\alpha \to \beta$, E, где $\alpha, \beta \in (V_T \cup V_N)^*$, а E — описание правила погружения (обычно E состоит из логических функций, которые устанавливают связь вершин β с вершинами $\varphi - \alpha$). Для того чтобы по WEB-грамматике построить ГПГ с соответствующими правилами погружения $E = \{f_j \mid j \in [1, K], f_j$ — логические функции$\}$, необходимо выполнить следующие действия: (1) выделить в функции f_j ее аргументы и значение; (2) сформировать правило $\alpha_j \to \beta_j$, s_j^1, s_j^2, s_j^3, описывающее правилами ГПГ функцию f_j; (3) дополнить правило $\alpha \to \beta$ множествами s^1, s^2, s^3. Процесс продолжается до тех пор, пока не будут построены правила ГПГ для всех $f_j \in E$. Аналогичным образом можно осуществить преобразование алгоритмов погружения других представлений графовых грамматик, сложность таких преобразований будет $O(n)$, где n — количество операций, необходимых для полного перебора E. Можно сформулировать:

Теорема 3. Существует алгоритм A_3 сложности $D(A_3) = O(n)$, осуществляющий преобразование правил погружения E в ГПГ.

Из приведенных рассуждений следует, что ГПГ предоставляют пользователю мощный аппарат для описания и обработки различных ИГС, причем погружение осуществляется с помощью обычного алгоритма вывода.

ЛИТЕРАТУРА

1. Никитин А. С., Певзнер Л. В., Скриган Н. И.— Весці АН БССР. Сер. фіз.-мат. навук, 1981, № 1, с. 5.
2. Никитин А. С., Певзнер Л. В., Скриган Н. И.— Препринт № 17(118) Института математики АН БССР, 1981, вып. 17, 20 с.
3. Эрли Д. Языки и автоматы.— М., 1975, с. 47.
4. Рейнгольд Э., Нивергельт Ю., Део О. Комбинаторные алгоритмы. Теория и практика.— М., 1980, с. 352.
5. Pfaltz J. L., Rozenfeld A.— Proc. Int. Joint Conf. Artifical Intelligence 1st, Washington, D. C., 1969, May, p. 609.

Поступила в редакцию
25.06.82.

Кафедра математического обеспечения ЭВМ

УДК 517.958 : 537.212

В. Т. ЕРОФЕЕНКО, Г. Ч. ШУШКЕВИЧ

РАСЧЕТ ЗАДАЧ ЭЛЕКТРОСТАТИКИ ДЛЯ ПАРАБОЛОИДА СО СФЕРИЧЕСКИМИ СЕГМЕНТАМИ

В настоящее время в печати опубликовано большое количество работ, в которых с помощью теории парных сумматорных и интегральных уравнений решены задачи математической физики для одного тела, представляющего собой неполную координатную поверхность (см. библиографию в [1, 2]). Решение задач для двух тел, одно из которых является полной, а второе неполной координатной поверхностью, представлено в литературе недостаточно полно [3, 4].

В данной статье выведены теоремы сложения, связывающие сферические и параболоидальные гармонические функции, с помощью которых решена задача электростатики для бесконечного параболоида со сферическим сегментом. Полученные результаты могут найти применение при проектировании электронно-оптических устройств [4].

Рассмотрим систему (рис. 1), состоящую из бесконечного параболоида Г, внутри которого помещен тонкий сферический сегмент S, расположенный на поверхности сферы S_1 радиуса a с центром в точке O. В точке O введем сферическую систему координат $\{r, \theta, \varphi\}$ и параболоидальную систему $\{\gamma, \delta, \varphi\}$, определяемую соотношениями $x=c\gamma\delta\cos\varphi$, $y=$

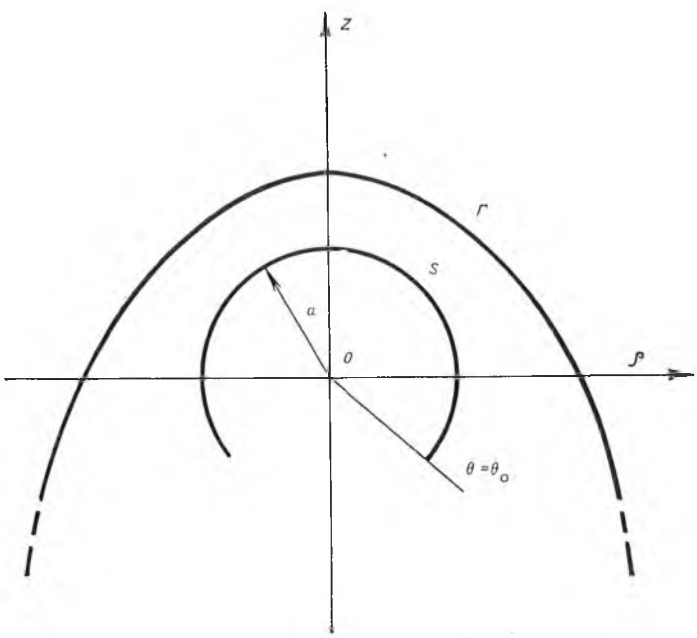

Рис. 1. Геометрия задачи