文章编号:1007-6492(1999)03-0013-03

算法与一类等价文法的制约关系

王文义,张行进,张 影

(郑州工业大学电气信息工程学院,河南 郑州 450002)

摘 要:存在着一类文法,虽然它们接受的是相同的语言,但却不能被算符优先分析算法所识别.针对问题产生的原因提出改进方法,该方法要求对文法中的产生式顺序加以限制,即文法除了必需满足算符文法的既定条件以外,还应保持文法中各产生式左部的文法变量必需是可区分的.

关键词:等价文法;优先关系矩阵;算符优先分析算法

中图分类号:TP 314 **文献标识码**:A

0 引言

在编译理论中[1~3],对算符文法的特征是这样定义的:

若 $P,Q,R\in V_n$,'…'$\in(V_n\cup V_t)^*$,则有

(1) 文法中不存在形如 $P\to\varepsilon$ 的产生式,

(2) 文法中任一产生式右部不存在两个相继的非终结符号,如 $P\to\cdots RQ\cdots$.

为了进行语法分析的需要,还要对算符文法中的任一终结符号对(a,b)确定其优先关系,如'a'$\lessdot$'b','a'$\gtrdot$'b','a'$\doteq$'b'.构造这些优先关系的传统算法为:

(3) iff 文法中有形如 $P\to\cdots ab\cdots$或 $P\to\cdots aQb\cdots$的产生式,让'a'$\doteq$'b'

(4) iff 文法中有形如 $P\to\cdots aR\cdots$的产生式并且 $R\overset{+}{\Rightarrow}b\cdots$或 $R\overset{+}{\Rightarrow}Qb\cdots$,让'$a$'$\lessdot$'$b$'

(5) iff 文法中有形如 $P\to\cdots Rb\cdots$的产生式并且 $R\overset{+}{\Rightarrow}\cdots a$ 或 $R\overset{+}{\Rightarrow}\cdots aQ$,让'$a$'$\gtrdot$'$b$'

在文法的优先关系确定之后,当文法中任一终结符号对(a,b)之间至多只满足上述 3 种优先关系之一时(也允许 a,b 间无优先关系),则称文法为算符优先文法 OPG(Operator Precedence Grammar).

实际上,上述方法在有些情况下是失效的,因为存在着描述相同语言(即正规集)的不同文法形式,这些文法也叫等价文法.在这些文法中,有些文法的句子能够被算符优先分析算法所识别,而有些则会失败,因此必须对文法的产生式形式加以改变(或加上某些限制),才能使用既定的算法进行分析.

1 描述相同语言的两个等价文法

1.1 典型的简单算术表达式文法

几乎在设计每一种程序设计语言的分析器时,都要用到算术表达式文法的设计,该文法的一种无二义性的简单形式为

$$G(E):\ ①\ E\to E+T\mid T$$
$$②\ T\to T*F\mid F$$
$$③\ F\to P\uparrow F\mid P$$
$$④\ P\to(\ E\ )\mid i$$

文法 $G(E)$所描述的语言为由变量、常数以及由'+','*','↑'和'('和')'组成的任何有意义的算术表达式.由于在文法中存在着如下形式

'(E)'

'$E+$' '$+T$'

'$T*$' '$*F$'

'$P\uparrow$' '$\uparrow F$'

'$E)$' '$(E$'

故根据算法可以得到

Lastvt$(E)\gtrdot\{+,)\}$,即

$\{+,*,\uparrow,)\},i\}\gtrdot\{+,)\}$(表示第一个集合中的任一元素的优先级都大于后一集合中的任

收稿日期:1999-03-30;**修订日期**:1999-05-19

基金项目:国家社会科学基金资助项目(97BTQ004)

作者简介:王文义(1947-),男,河南省洛阳市人,郑州工业大学副教授,主要从事软件工程及数据库应用方面的研究.

元素.以下解释类同).

Lastvt(T)$\cdot>$|*|,即|*,↑,),i|$\cdot>$|*|;

Lastvt(P)$\cdot>$|↑|,即|),i|$\cdot>$|↑|

和 +$\lessdot$Firstvt(T),即|+|$\lessdot$|*,↑,(,i|;

*,↑|$\lessdot$Firstvt(F),即

*,↑|$\lessdot$|↑,(,i|;

($\lessdot$Firstvt(E),即

($\lessdot$|+,*,↑,(,i|;

于是可得到文法 $G(E)$的优先关系矩阵见表1.

表1 文法 $G(E)$的优先关系矩阵

左 V_t \ 右 V_t	+	*	↑	i	(	)
+	$\cdot>$	$\lessdot$	$\lessdot$	$\lessdot$	$\lessdot$	$\cdot>$
*	$\cdot>$	$\cdot>$	$\lessdot$	$\lessdot$	$\lessdot$	$\cdot>$
↑	$\cdot>$	$\cdot>$	$\lessdot$	$\lessdot$	$\lessdot$	$\cdot>$
i	$\cdot>$	$\cdot>$				$\cdot>$
(	$\lessdot$	$\lessdot$	$\lessdot$	$\lessdot$	$\lessdot$	$=$
)	$\cdot>$	$\cdot>$	$\cdot>$			

1.2 一个等价文法特例[3,4]

观察文法 $G'(E)$

① $E\to E+E$

② $E\to E*E$

③ $E\to E↑E$

④ $E\to(E)|i$

由形式语言理论知道,该文法虽然与上述文法 $G(E)$是描述同一语言的完全等价的两个不同文法,但从下面的分析可知,文法 $G'(E)$却并不适合于算符优先分析算法.

由文法的产生式形状.

'(E)'

'$E+$' '$+E$'

'$E*$' '$*E$'

'$E↑$' '$↑E$'

'$E)$' '$(E$'

可得到

'('$=$')';

+,*,↑,),i|$\cdot>$|+,*,↑,)|

和|+,*,↑,(|$\lessdot$|+,*,↑,(,i|,

即表2所示的文法 $G'(E)$的优先关系矩阵.

观察表2,由于矩阵中出现了互相矛盾的优先关系,因此可以认定文法 $G'(E)$不是算符优先文法,故算符优先分析算法对它是无效的.通过对文法 $G(E)$和 $G'(E)$优先关系的分析,可以得出下述结论:对描述相同语言的不同文法,即是使用同样的算法,但对同一合法的句子,也可能会得到迥然不同的分析结果.

表2 文法 $G'(E)$的优先关系矩阵

左 V_t \ 右 V_t	+	*	↑	i	(	)
+	$\cdot>$ $\lessdot$	$\cdot>$ $\lessdot$	$\cdot>$ $\lessdot$	$\lessdot$	$\lessdot$	$\cdot>$
*	$\cdot>$ $\lessdot$	$\cdot>$ $\lessdot$	$\cdot>$ $\lessdot$	$\lessdot$	$\lessdot$	$\cdot>$
↑	$\cdot>$ $\lessdot$	$\cdot>$ $\lessdot$	$\cdot>$ $\lessdot$	$\lessdot$	$\lessdot$	$\cdot>$
i	$\cdot>$	$\cdot>$	$\cdot>$			$\cdot>$
(	$\lessdot$	$\lessdot$	$\lessdot$	$\lessdot$	$\lessdot$	$=$
)	$\cdot>$	$\cdot>$	$\cdot>$			

2 影响优先关系的因素[5]

尽管文法 $G(E)$和 $G'(E)$描述的语言是相同的,但它们在产生式形式上却存在着根本的区别,其实质是体现出了在推导(分析或识别)过程中对产生式使用顺序的选择上,也就是产生式自身所体现出的"优先级"问题.现就表2中出现的与'+','*'和'↑'3个运算符出现矛盾优先关系有关的产生式形式进行分析.在下面的分析中,设 $P,R\in V_n,\theta\in V_t$.

2.1 文法 $G(E)$的产生式左部的文法变量的区别

$G(E)$的产生式右部形状为

$$P\ \theta\ R$$

其中,$P\neq R$,θ是算术运算符.由于'$P\ \theta$'部分与文法 $G'(E)$完全相同,可不予讨论.关键是'$\theta\ R$'部分.由于 P,R是两个不相同的独立文法变量,因此在展开 R时,就必然要用到文法中某一特定的产生式,而不存在任何选择的余地,因此根据算法,必有

$$|\theta|\lessdot\text{Firstvt}(R),$$

而 Firstvt(R)是从 R推导起且满足一定条件的终结符号的集合.例如,对

$$E+T,$$

可得到

$$|+|\lessdot\text{Firstvt}(T).$$

其中:Firstvt(T)=|*,↑,(,i|.

这个优先关系是完全符合通常四则运算的逻辑关系的.

2.2 文法 $G'(E)$中产生式左部的文法变量不可区别性

$G'(E)$的产生式右部形式为

$$\cdot E\ \theta\ E$$

对'θE'部分,根据文法,可知

$$\{\theta\} <\cdot \mathrm{Firstvt}(E)$$

由于文法 $G'(E)$中只有唯一的一个文法变量 E,因此在求解 Firstvt(E)时就产生了不确定性,即究竟文法变量 E 应该使用①,②,③,④4 个产生式左端的哪个'E',而使用不同的 E,将可得到不同的 Firstvt(E)集合,如对'$+E$':

若从产生式①推导起,则有

$$\mathrm{Firstvt}(E)=\{+,*,\uparrow,(,i\};$$

若从产生式②推导起,则有

$$\mathrm{Firstvt}(E)=\{*,\uparrow,(,i\};$$

若从产生式③推导起,则有

$$\mathrm{Firstvt}(E)=\{\uparrow,(,i\};$$

若从产生式④推导起,则有

$$\mathrm{Firstvt}(E)=\{(,i\};$$

但实际上,在上述 4 种情况中只有一种是正确的.

同理,对诸如'$*E$','$\uparrow E$'等,也都存在类似的现象.这就是导致表 2 中矛盾优先关系的真正原因.一旦象文法 $G(E)$那样,对文法的产生式经过必要地改造,使产生式左端的文法变量成为可区别的,即消除其不确定性,则问题就可得到解决.

3　结束语

在算符优先分析算法中,存在着一类等价文法,虽然它们在理论上接受相同的语言,但它们却往往会导致算法失败.究其原因,是因为文法中文法变量的不可区别性所引起的.若要消除这种现象,就必须对文法的产生式顺序(优先级)加以限制,即文法除需满足算符文法的特征之外,还需要使文法中各产生式左端的文法变量是可区分的,只有这样,才能避免出现本来是正确的句子但却被分析为非法句子的现象发生.

参考文献:

[1] 格里斯 D.数字计算机的编译程序构造[M].曹东启,仲萃豪,姚兆炜,译.北京:科学出版社,1982.

[2] 姜文清.计算引论[M].成都:四川科技出版社,1987.

[3] 霍普克罗夫特 J E,厄尔曼 J D.形式语言及其与自动机的关系[M].莫绍揆,段　祥,顾秀芬,译.北京:科学出版社,1979.

[4] AHO A V,SETHI R,ULLMAN J D.Compilers:Principles,Techniques and Tools[M].New York:Addison - weasley Publishing Company,1986.

[5] 王文义,张行进,王若雨.产生式外部形态对算术表达式翻译的影响及其解决办法[J].郑州工业大学学报,1999,20(1):72 - 74.

Interaction Between Algorithm and One Type of Equivalence Grammar

WANG Wen - yi, ZHANG Xing - jin, ZHANG Ying

(College of Electrical & Information Engineering, Zhengzhou University of Technology, Zhengzhou 450002, China)

Abstract: There exists one type of grammar which accepts arithmetic expression language but it can not discerned by operator precedence algorithm. The paper analyzes this phenomenon and gives an improved method. The method is that production's order is limited. Namely, the grammar must not only be arithmetic operator grammar , but also have distinguishable left variables.

Key words: equivalence grammar; precedence relation matrix; operator precedence analysis algorithm