LL(2)文法から強LL(2)文法への書き換えアルゴリズムの正当性について

3C－5

広瀬卓也† 竹内淑子‡ 吉田敬一†
静岡大学大学院理工学研究科†
浜松職業能力開発短期大学校‡

1 はじめに

文法の大小と言語の大小は異なる。つまり文法クラスが異なるものでも同じ言語を生成する場合がありうることはよく知られている。同じ言語に対しては、文法クラスの小さい方が解析表の大きさや、解析速度、実装の容易さの点で優れている。

本研究では、LL(2)文法から強LL(2)文法へ書き換え、その書き換えアルゴリズム[5]の正当性を書き換え前と書き換え後の文法から生成される2つの言語が等価であることを利用して証明する。

2 基本定義と記法

[定義1]文脈自由文法(以下CFG) Gを

$G=(N, \Sigma, P, S)$

とする。ここに、N、Σはそれぞれ文法Gの非終端記号の集合ならびに終端記号の集合、Pは生成規則の集合であり、Sは出発記号である。

[記法1]Nの要素をA、B、C...、Σの要素をa、b、c...、$N \cup \Sigma$の要素をX、Y、Zで表す。また、Σ^*の要素をs、t、u...で表し、$(N \cup \Sigma)^*$の要素をα、β、γ...で表す。特にε、ϕはそれぞれ空列、空集合を表す。ε、ϕ以外のこれらの記号は添字をつけて用いることもある。またp、qは生成規則番号である。

[定義2]集合$FIRST_k(\alpha)$は以下で定義される。

$FIRST_k(\alpha)=\{u \mid (\alpha \overset{*}{\Rightarrow} u\beta, |u|=k)$
または$(\alpha \overset{*}{\Rightarrow} u, |u|<k)\}$

ただし、α、$\beta \in (N \cup \Sigma)^*$、$u \in \Sigma^*$、$|u|$は$u$の長さを表す。また、$\overset{*}{\Rightarrow}$は、生成規則を0回以上使用する最左導出である。

[定義3]集合$FOLLOW_k(A)$は以下で定義される。

CFG $G=(N, \Sigma, P, S)$において

$S \overset{*}{\Rightarrow} u_i A \xi_i$なる導出に対し

$FOLLOW_k(A)=\cup_i FIRST_k(\xi_i)$

[定義4]L_1、L_2をΣ^*の部分集合とするとき、演算子$\oplus_k$は以下のように定義される。

$L_1 \oplus_k L_2$
$=\{w \mid$ ある$x \in L_1$、$y \in L_2$に対して、
$\| xy \| \leqq k$ならば$w=xy$、$\| xy \| > k$
ならば$w=u$、ただし$xy=uv$、$\| u \| =k\}$

[定義5]CFG $G=(N, \Sigma, P, S)$において、$A \rightarrow \alpha$、$A \rightarrow \beta$を相異なる生成規則とするとき

$S \overset{*}{\Rightarrow} uAv$

に対して

$(FIRST_k(\alpha) \oplus_k FIRST_k(v)) \cap (FIRST_k(\beta) \oplus_k FIRST_k(v))=\phi$

が成り立つとき、GはLL(k)文法であるという。

[定義6]CFG $G=(N, \Sigma, P, S)$において、$A \rightarrow \alpha$、$A \rightarrow \beta$を相異なる生成規則とするとき

$(FIRST_k(\alpha) \oplus_k FOLLOW_k(A)) \cap (FIRST_k(\beta) \oplus_k FOLLOW_k(A))=\phi$

が成り立つとき、Gは強LL(2)文法であるという。

3 文法の書き換え

与えられた文法がLL(2)文法であり、その文法が以下の定理を満たさないとき、[定義6]にあてはめ、その左辺が$\neq \phi$となった場合、その文法は強LL(2)でないLL(2)文法である。このとき、書き換えアルゴリズム[5]に従い書き換えを行う。

[定理] LL(2)文法$G=(N, \Sigma, P, S)$において、

$A \rightarrow \alpha$、$A \rightarrow \beta$、$\alpha \neq \beta$、$\alpha \overset{*}{\Rightarrow} w$、$\beta \overset{*}{\Rightarrow} v$

とするとき、

(1) $|w| \geqq 2$かつ$|v| \geqq 2$

(2) $|w|=1$かつ$|v|=1$

となる場合、文法Gは強LL(2)文法となる。 [終]

[例]

$S \rightarrow bAb \mid aAab$

A Proof of Correctness for Translating Algorithm from LL(2)Grammar to Strong-LL(2)Grammar.

†Takuya Hirose
‡Yoshiko Takeuchi
†Keiichi Yoshida
†Graduate School of Science and Engineering, Shizuoka University
‡Hamamatsu Polytechnic college

$A \rightarrow a \mid \varepsilon$

において、A-生成規則に関して$|w|=1$、$|v|=0$なので、[定理]の(1)(2)のいずれにも該当していないため、強LL(2)文法でない可能性がある。そこで[定義6]を適用すると、左辺$\neq \phi$となるので強LL(2)文法でないことが判明する。そこでAを非終端記号A_1とA_2に分割し、A-生成規則の右辺を持たせる。

$S \rightarrow b\,A_1\,b \mid a\,A_2\,a\,b$

$A_1 \rightarrow a \mid \varepsilon$

$A_2 \rightarrow a \mid \varepsilon$

ここで$S \rightarrow b\,A_2\,b \mid a\,A_1\,a\,b$となる規則を加えないのは、書き換え前の文法における2つのS-生成規則の右辺にAが現れ、そのことが強LL(2)文法でないことの原因となっているので、これを区別する必要があるためである。ここで書き換え後の文法に[定義6]を適用するとその左辺はϕとなり強LL(2)文法になっていることが分かる。

4 書き換えの正当性

書き換えの正当性を示すため、文法Gとそれを書き換えた文法G'において$L(G)=L(G')$となることを証明する。

[証明] 書き換え対象となる非終端記号Aにおいて、その書き換え法から$A \rightarrow \alpha$、$A \rightarrow \beta$が$A_1 \rightarrow \alpha \mid \beta, \ldots, A_n \rightarrow \alpha \mid \beta$と書き換えられるので、書き換え前の文法$G=(N, \Sigma, P, S)$に対して書き換え後の文法$G'$は

$N'=N_0 \cup \{A_1, \ldots, A_n\}$

ただし$N_0=N-\{A\}$

を満足するように構成したN'と

$P'=P_0 \cup \{A_1 \rightarrow \alpha \mid \beta, \ldots, A_n \rightarrow \alpha \mid \beta\}$

ただし$P_0=P-\{A$-生成規則$\}$

を満足するように構成したP'をもつ$G'=(N', \Sigma, P', S)$と表すことができる。

まず$L(G) \subseteq L(G')$を示す。

PにもP'にも含まれている生成規則を用いている限り問題はない。そこで

$A \rightarrow \gamma \in P$ かつ $A \rightarrow \gamma \notin P'$

の場合について考える。これはAが書き換え対象となる非終端記号であり、$A_i \rightarrow \gamma \in P'$が存在することを意味する。従って$G$における導出

$S \overset{*}{\Rightarrow} u\,A\,v \overset{G}{\Rightarrow} u\,\gamma\,v$

においてAは書き換え対象となった最初の非終端記号とする。このときG'においては

$S \overset{*}{\Rightarrow} u\,A_i\,v \overset{G'}{\Rightarrow} u\,\gamma\,v$

なる導出が対応する。また$\gamma\,v$の中に書き換えられた非終端記号が現れることがあっても同様のことが言える。さらに、SがGの右辺に現れ、かつ書き換え対象となったときは、拡張文法を考えればよい。故に$L(G) \subseteq L(G')$が成り立つ。

$L(G) \supseteq L(G')$の場合も同様にして成り立つ。

よって$L(G)=L(G')$が成り立つ。

以上により書き換えの正当性が示された。

[証明終]

5 評価

LL(2)文法にはその生成規則において、既に強LL(2)の条件を満たすものとそうでないものがある。前者において書き換えの必要がないことは明らかである。しかしAhoらの手法では、既に強LL(2)文法の条件を満たす規則についても余分な書き換えを行う可能性がある。本研究の手法では、強LL(2)文法の条件を満足しない部分のみを書き換えるため、こうした余分な書き換えは行われない。本研究での手法においてLL(2)文法から強LL(2)文法への書き換えの正当性の証明を両文法から生成される言語クラスの等価性を示すことで行った。

6 参考文献

[1]吉田・竹内：準LL(2)文法に対する構文解析表の作成アルゴリズム,「情報処理学会論文誌」第31号，第6月号

[2]吉田・竹内：準LL(2)文法に対する解析表の構造と解析アルゴリズム,「情報処理学会論文誌」第31号，第9月号

[3]井上謙蔵：コンパイラ「プログラム言語処理の基礎」pp77-117 丸善

[4]Aho, A. V. and Ullman, J. D.: *The Theory of Parsing, Translation and Compiling, Vol. 1pp334-361*Prentice-Hall(1972)

[5] 広瀬・吉田：LL(2)文法から強LL(2)文法への書き換えアルゴリズム,「情報処理学会第55回全国大会講演論文集」(1997)