LL(k) LANGUAGES ARE CLOSED UNDER UNION WITH FINITE LANGUAGES

Ileana Streinu
Faculty of Mathematics
University of Bucharest
Str. Academiei 14
Bucuresti 7000 / Romania

1.Introduction

LL(k) grammars were first defined by Lewis and Stearns[3].Algebraic properties, as closure under usual operations (union, intersection, complement, product, etc.) have been first studied by Rosenkrantz and Stearns[4].They have proved that the class of LL languages is not closed under any of these operations.

Nevertheless, closure properties have been investigated. Wood[5] has proved that the LL languages are not closed under pre-product with finite or regular languages. Now we shall prove that they are closed under union and difference with finite languages. Moreover, the union (or difference) of an LL(k) language with a finite one does not modify the initial "k" : the resulting language still remains LL(k).

2.Previous definitions and results

In order to prove the results announced in the introduction, we shall use the following notations, definitions and theorems from Aho and Ullman[1]:

(2.1) Let $G=(N,\Sigma,P,S)$ be a context-free grammar.For $\alpha\in(N\cup\Sigma)^*$ and an integer $k>0$, we define :

$FIRST_k^G(\alpha)=\{w\in\Sigma^* /$ either $|w|<k$ and $\alpha\overset{*}{\underset{G}{\Rightarrow}}w$, or $|w|=k$ and $\alpha\overset{*}{\underset{G}{\Rightarrow}}wx$, for some $x\in\Sigma^*\}$. If $\alpha\notin\Sigma^*$, the definition is independent of G.

For $L\subset\Sigma^*$, $FIRST_k(L)=\{FIRST_k(w) / w\in L\}$

(2.2) A context-free grammar, $G=(N,\Sigma,P,S)$, is LL(k), for some inte-

ger $k>0$, if whenever there are two leftmost derivations:

$$S \overset{*}{\underset{lm}{\Rightarrow}} wA\alpha \underset{lm}{\Rightarrow} w\beta\alpha \overset{*}{\Rightarrow} wx$$

$$S \overset{*}{\underset{lm}{\Rightarrow}} wA\alpha \underset{lm}{\Rightarrow} w\gamma\alpha \overset{*}{\Rightarrow} wy$$

such that $FIRST_k(x)=FIRST_k(y)$, it follows that $\beta=\gamma$.

We say that: a grammar is LL, if it is LL(k) for some integer k; a language is LL(k), if there is an LL(k) grammar G, such that L=L(G).

(2.3) Let $G=(N,\Sigma,P,S)$ be a context-free grammar. Then G is LL(k) if and only if the following condition holds: if $A\to\beta$ and $A\to\gamma$ are distinct productions in P, then $FIRST_k(\beta\alpha)\cap FIRST_k(\gamma\alpha)=\emptyset$, for all $wA\alpha$ such that $S \overset{*}{\underset{lm}{\Rightarrow}} wA\alpha$.

(2.4) Every LL(k) grammar has an equivalent LL(k+1) grammar in Greibach normal form.

(2.5) If a language L has an LL(k) grammar without e-productions, $k\geq 2$, then L has an LL(k-1) grammar.

3. Main results

(3.1) Lemma. Let L_1 and L_2 be LL languages, such that $L_1\cap L_2=\emptyset$. A sufficient condition for $L_1\cup L_2$ to be LL is the existence of an integer $k>0$ such that:

(3.1.1) $FIRST_k(L_1)\cap FIRST_k(L_2)=\emptyset$.

The condition is not necessary.

Proof. Let $G_i=(N_i,\Sigma,P_i,S_i)$ be $LL(k_i)$ grammars, such that $L_i=L(G_i)$ $(i=1,2)$. Without loss of generality, we can assume that $N_1\cap N_2=\emptyset$. We shall define a new grammar $G=(N,\Sigma,P,S)$ such that: $N=N_1\cup N_2\cup\{S\}$, where S is a new symbol, $S\notin N_1\cup N_2$; $P=P_1\cup P_2\cup\{S\to S_1,\ S\to S_2\}$.

It is easy to verify that $L(G)=L_1\cup L_2$.

Now we shall prove that G is an LL(n) grammar, where $n=\max(k,k_1,k_2)$. First of all, we shall remark that the new initial symbol S does not belong to the right side of any production from G (the only productions where S appears are $S\to S_1$ and $S\to S_2$). But, because $N_1\cap N_2=\emptyset$, it is true that:

$$FIRST_k(L_i)=FIRST_k^{G_i}(S_i)=FIRST_k^{G}(S_i),\quad i=1,2.$$

Then, from the hypothesis (3.1.1) it follows that:

(3.1.2) $FIRST_k^G(S_1)\cap FIRST_k^G(S_2)=\emptyset$, and this is also true for n, because $k\leq n$.

Moreover, from (2.3) it follows that for any $A\in N_i$ $(i=1,2)$, the following conditions hold, for $i=1,2$: if $A\to\beta$ and $A\to\gamma$ are distinct

productions in P_i (then, also in P), then:

$\mathrm{FIRST}^{G_i}_{k_i}(\beta\alpha) \cap \mathrm{FIRST}^{G_i}_{k_i}(\gamma\alpha) = \phi$, for all $wA\alpha$ such that $S_i \overset{*}{\underset{lm}{\Rightarrow}} wA\alpha$.

But, because $k_i \leq n$, $P_i \subseteq P$, $N_1 \cap N_2 = \phi$ and $S \to S_1$, $S \to S_2$ are the only S-productions in P, it follows that :

(3.1.3) $\mathrm{FIRST}^{G}_{n}(\beta\alpha) \cap \mathrm{FIRST}^{G}_{n}(\gamma\alpha) = \phi$, for all $wA\alpha$ such that $S \overset{*}{\underset{lm}{\Rightarrow}} wA\alpha$.

From (3.1.2) and (3.1.3), using (2.3), we conclude that G is LL(n).

The condition (3.1.1) is not necessary, because $L_1 = \{a^n b^n \ / \ n \text{ odd}\}$ and $L_2 = \{a^n b^n \ / \ n \text{ even}\}$ are LL languages, generated by the LL(1) grammars: G_1: $S \to aA$, $A \to aSbb \ / \ b$; G_2: $S \to aaSbb \ / \ e$. The condition (3.1.1) is not satisfied, but $L_1 \cup L_2 = \{a^n b^n / n \geq 0\}$ is an LL(1) language.

(3.2) Corollary. If L is an LL language and w is a word, then $L \cup \{w\}$ is also LL.

Proof. If $w \in L$, the statement is trivial. If $w \notin L$, the result follows from the preceding lemma, taking $k = |w| + 1$ (where $|w|$ = the length of w).

Remark. The proof of Lemma (3.1) shows that, if L is LL(k) and $|w| = n$, then $L \cup \{w\}$ has an LL(m) grammar, where $m = \max(k, n+1)$. The following theorem will prove a stronger result, namely that $L \cup \{w\}$ is still an LL(k) language, if L is so.

(3.3) Theorem. Let L be an LL(k) language, $w \notin L$. Then there exists an LL(k) grammar which generates $L \cup \{w\}$.

Proof. From (2.4) it follows that we can find an LL(k+1) grammar in Greibach normal form, $G = (N, \Sigma, P, S)$, such that $L = L(G)$. We shall prove that $L \cup \{w\}$ can be generated by an e-free LL(k+1) grammar, and then , using (2.5), we may conclude that $L \cup \{w\}$ can be generated by an LL(k) grammar.

In fact we shall no more complicate the notation and we'll prove that : if $G = (N, \Sigma, P, S)$ is an LL(k) grammar in Greibach normal form ($k \geq 2$), then there exists an e-free LL(k) grammar which generates $L(G) \cup \{w\}$.

Let $w = a_1 a_2 \ldots a_m$. Let $n \geq 0$ be an integer such that for at least one word in L, $a_1 a_2 \ldots a_n$ is a prefix, and for no word in L $a_1 a_2 \ldots a_{n+1}$ is a prefix. The case n=m is when w is a prefix for some word in L(in this case we suppose $a_{n+1} = e$). Note that here we use the fact that $w \notin L$.

If $n < k$, the statement follows from the proof of Lemma (3.1). So we shall suppose that $n \geq k$. Let us consider in G a leftmost derivation of the form:

$$S \overset{p_1}{\Longrightarrow} a_1\alpha_1 \overset{p_2}{\Longrightarrow} a_1 a_2 \alpha_2 \overset{p_3}{\Longrightarrow} \cdots \overset{p_{n-k+1}}{\Longrightarrow} a_1 a_2 \ldots a_{n-k+1} \alpha_{n-k+1}$$

such that: $a_{i+1}\dots a_{i+k}\in FIRST_k(\alpha_i)$, $1\leq i\leq n-k$, $a_{n-k+2}\dots a_{n+1}\notin FIRST_k(\alpha_{n-k+1})$.

Such a leftmost derivation can be found because of the choice of n, and it is even unique, because G is an LL(k) grammar. Additionally, if we set $\alpha_o=S$ and if for a value of i, $0\leq i\leq n-k$, we write the sequence of nonterminals α_i as $\alpha_i=A\beta_i$ ($A\in N, \beta_i\in N^*$), then for any rule $A\to\beta$ which is not the rule p_{i+1}, it follows that:

(3.3.1) $FIRST_k^G(\beta\beta_i)\cap FIRST_k^G(a_{i+1}\alpha_{i+1})=\emptyset$, $0\leq i\leq n-k$.

Let us now modify the initial grammar $G=(N,\Sigma,P,S)$ in order to obtain a new LL(k) grammar, $G'=(N',\Sigma,P',X_o)$ such that $L(G')=L\cup\{w\}$. Let $X_o,X_1,\dots,X_{n-k+1}$ be new symbols not in $N\cup\Sigma$, all different. Let $N'=N\cup\{X_o,X_1,\dots,X_{n-k+1}\}$, where X_o is the new initial symbol. P' is the set obtained by joining to P the following new productions:

(3.3.2) $X_i\to\beta\beta_i$, if $\alpha_i=A\beta_i$ and $A\to\beta$ is a rule in P, but not the rule p_{i+1}.

(3.3.3) $X_i\to a_{i+1}X_{i+1}$

for all $i=0,1,\dots,n-k$, and

(3.3.4) $X_{n-k+1}\to\alpha_{n-k+1}$

(3.3.5) $X_{n-k+1}\to a_{n-k+2}\dots a_m$.

Note that for $k\geq 2$ this is an e-free grammar.

Let us now prove that G' is LL(k). We shall use theorem (2.3). Let $A\in N'$, $A\to\beta$, $A\to\gamma$ be two different productions from P', and $\alpha\in N'^*$ such that $X_o\overset{*}{\underset{lm}{\Rightarrow}}xA\alpha$, $x\in\Sigma^*$. We must prove that $FIRST_k^{G'}(\beta\alpha)\cap FIRST_k^{G'}(\gamma\alpha)=\emptyset$. If $A\in N$, it is easy to remark that $\alpha\in N^*$, and $A\to\beta$, $A\to\gamma$ are from P. Then $FIRST_k^{G'}(\beta\alpha)=FIRST_k^G(\beta\alpha)$ and $FIRST_k^{G'}(\gamma\alpha)=FIRST_k^G(\gamma\alpha)$, which are disjoint, because G is LL(k). If $A=X_i$ ($0\leq i\leq n-k+1$), then $\alpha=e$. Then $FIRST_k^{G'}(\beta)\cap FIRST_k^{G'}(\gamma)=\emptyset$, because we have the statement (3.3.1) true, if $A\to\beta$ or $A\to\gamma$ are (3.3.3) or (3.3.5) rules, and because G is LL(k), if both are (3.3.2) rules. So it follows that G' is LL(k).

Let us now prove that $L(G')=L\cup\{w\}$. In order to prove that $L\cup\{w\}\subset L(G')$, we shall remark that any word from L which has no prefix in common with w may be derived in G' by using at first a (3.3.2)-rule, and then rules from P. Any word in L having $a_1a_2\dots a_i$ as a prefix may be derived in G' in the same way as before, if $i<k$, and otherwise by using (i-k+1)-times (3.3.3)-rules, then a (3.3.2) or a (3.3.4)-rule, and finally using rules from P. At last, w may be derived by using all the (3.3.3)-rules and the (3.3.5)-rule. In order to show the other inclusion, it may be proved, by induction on the number of steps used in

the derivation $X_i \overset{*}{\underset{lm}{\Rightarrow}} x$, that $x \in \{y \in \Sigma^* / a_1 a_2 \ldots a_i y \in L \cup \{w\}\}$ $(0 \leq i \leq n-k+1)$. So, for i=0, $X_0 \overset{*}{\Rightarrow} x$ implies that $x \in L \cup \{w\}$. The proof by induction is not difficult and it is left to the reader.So, we may conclude that the LL(k) e-free grammar G' generates $L \cup \{w\}$, and the theorem is thereby proved.

(3.4) Corollary. If L_1 is LL(k) and L_2 is finite, then $L_1 \cup L_2$ is LL(k).

(3.5) Corollary. Lemma (3.1) is valid even if $L_1 \cap L_2$ is finite.

Remark. Using an analogous proof to that of theorem (3.3), one might show that, if L is LL(k) and w is a word in L, then $L-\{w\}$ can be generated by an LL(k) grammar. An important consequence of these facts is the following:

(3.6) Theorem. If L is LL(k) but not LL(k-1), then $L \cup \{w\}$ and $L-\{w\}$ are also LL(k), but not LL(k-1).

(3.7) Corollary. The class of LL(k) languages is closed under union and difference with finite languages.

Remark. Examining the proof of Lemma (3.1), we conclude that $L_1 \cup L_2$ is LL(n), where $n=\max(k,k_1,k_2)$. This value can be improved, remarking that we need k only for expanding S: here we must make a choice between $S \rightarrow S_1$ and $S \rightarrow S_2$. Using a method of"linearization" of a derivation analogous to that from the proof of theorem (3.3), one might obtain an LL(n) grammar to generate $L_1 \cup L_2$, where $n=\max(k_1,k_2)$.

Bibliografy

[1] A.V.Aho,J.D.Ullman: The Theory of Parsing, Translation and Compiling, vol.1(1972),vol.2(1973),Prentice-Hall,Englewood Cliffs.

[2] J.Bordier,H.Saya: A necessary and sufficient condition for a power language to be LL(k),Computer Journal,1973,vol.16,pp.451-456.

[3] P.M.Lewis,R.E.Stearns: Syntax-directed transduction,Journal ACM,1968, vol.15:3,pp.464-488.

[4] D.J.Rosenkrantz,R.E.Stearns: Properties of deterministic top-down grammars, Information and Control,1970, vol.17:3, pp.226-256.

[5] D.Wood: A further note on TD deterministic languages, Computer Journal, 1971, vol.14, pp.396-403.