Langages formels

12. Grammaires LL

7 mai 2007

Exercice 1 – *Calcul des* $\mathrm{First}_k(\alpha)$

Soit $G = \langle \Sigma, V, P, S\rangle$ une grammaire algébrique. On suppose que toutes les variables de G sont productives. Pour $k \in \mathbb{N}$ fixé, on définit la suite d'ensemble suivants :

$$
\begin{array}{rll}
X_m(a) := \{a\} & & \text{si } a \in \Sigma \\
X_0(x) := \emptyset & & \text{si } x \in V \\
X_{m+1}(x) := \bigcup_{(x\to\alpha)\in P} X_m(\alpha) & & \text{si } x \in V \\
X_m(\alpha) := \mathrm{Trunc}_k(X_m(\alpha_1)\dots X_m(\alpha_n)) & & \text{si } \alpha = \alpha_1\dots\alpha_n \text{ avec } \alpha_i \in \Sigma \cup V
\end{array}
$$

Montrer les assertions suivantes :

1. $X_m(\alpha) \subseteq X_{m+1}(\alpha) \subseteq \Sigma^{\leqslant k}$ pour $k \geqslant 1$.
2. Si $\alpha \xrightarrow{m} w \in \Sigma^*$ alors $w[k] \in X_m(\alpha)$.
3. $X_m(\alpha) \subseteq \mathrm{First}_k(\alpha)$.
4. $\mathrm{First}_k(\alpha) = X(\alpha) = \bigcup_{m\geqslant 0} X_m(\alpha)$.

Exercice 2 – *Calcul des* $\mathrm{Follow}_k(x)$

1. Soit G une grammaire algébrique dont toutes les variables sont accessibles. En s'inspirant de l'exercice précédent, montrer que les $\mathrm{Follow}_k(x)$ pour $x \in V$ sont calculables.
2. Calculer les $\mathrm{Follow}_1(x)$ pour la grammaire suivante :

$$E \to TE' \qquad E' \to +TE' \mid \varepsilon \qquad T \to FT' \qquad T' \to *FT' \mid \varepsilon \qquad F \to (E) \mid a \mid b \mid c$$

3. Montrer que la grammaire ci-dessus est fortement LL(1).

Exercice 3 – *Grammaires fortement* LL(1)

Montrer qu'une grammaire est LL(1) si et seulement si elle est fortement LL(1).

Exercice 4 – *Table pour les grammaires* LL

Soit la grammaire définie par :

$$E \to E \vee E \mid E \wedge E \mid \neg E \mid (E) \mid v \mid f$$

Donner une grammaire LL(1) équivalente en supprimant l'ambiguïté et la récursivité gauche. Construire la table de l'analyseur LL(1) de cette grammaire et simuler le fonctionnement de l'analyseur sur le mot $\neg(v \wedge f) \vee v$.

Exercice 5 – *Grammaires* LL(0) *et* LL(1)

1. Montrer que si l'automate expansion/vérification associé à une grammaire est déterministe, alors la grammaire est LL(0).
2. Montrer que si G est en FNPG et que pour toutes règles $x \to a\alpha$ et $x \to b\beta$ avec $a, b \in \Sigma$, on a $a \neq b$ ou $\alpha = \beta$, alors G est LL(1).
3. Montrer que la réciproque est fausse.