LL(k)-грамматики

SLL(1), $FIRST_k^G$, $FOLLOW_k^G$, LL(1), strong LL(1), LL(k)

В. С. Полозов

Кафедра системого программирования СПбГУ

Теория автоматов и формальных языков

Содержание

Неформально

Цель: построить детерминированный разбор.

BFS

Рассмотрим грамматики $G = (V_N, V_T, P, S)$ с правилами вида $A \to x\alpha, x \in V_T$.

Например:

$$\begin{array}{lcl} S & \to & aB \mid bA \\ A & \to & a \mid aS \mid bAA \\ B & \to & b \mid bS \mid aBB \end{array}$$

И нисходящий разбор поиском в ширину цепочки *aabb*.

Parse table

Построим *таблицу разбора* (управляющую таблицу):

	a	b	#
S	$S_1 : aB$	$S_2 : bA$	
A	$A_1 : a$ $A_2 : aS$	$A_3 : bAA$	
B	$B_3 : aBB$	$B_1 : b$ $B_2 : bS$	

Если в ячейке несколько записей - это поиск.

SLL(1)

Если в ячейке таблицы разбора не более одной записи - получим детерминированный разбор.
Назовем такую грамматику *простой* $LL(1)$ или $SLL(1)$.
Переформулируем:

Определение

Говорят, что контекстно-свободная грамматика G является простой $LL(1)$-грамматикой, если в ней нет ε-правил, и все альтернативы для каждого нетерминала начинаются с терминалов и притом различных.

SLL(1)

Если в ячейке таблицы разбора не более одной записи - получим детерминированный разбор.
Назовем такую грамматику *простой* $LL(1)$ или $SLL(1)$.
Переформулируем:

Определение

Говорят, что контекстно-свободная грамматика G является простой $LL(1)$-грамматикой, если в ней нет ε-правил, и все альтернативы для каждого нетерминала начинаются с терминалов и притом различных.

SLL(1)

Например:

$$\begin{array}{lcl} S & \to & aB \\ B & \to & b \mid aBb \end{array}$$

Таблица разбора:

	a	b	#
S	$S_1 : aB$		
B	$B_2 : aBb$	$B_1 : b$	

И вход: *aabb* (с добавленным символом #)

SLL(1)

Например:

$$\begin{array}{rcl} S & \to & aB \\ B & \to & b \mid aBb \end{array}$$

Таблица разбора:

	a	b	#
S	$S_1 : aB$		
B	$B_2 : aBb$	$B_1 : b$	

И вход: *aabb* (с добавленным символом #)

SLL(1)

Например:

$$\begin{array}{rcl} S & \rightarrow & aB \\ B & \rightarrow & b \mid aBb \end{array}$$

Таблица разбора:

	a	b	#
S	$S_1 : aB$		
B	$B_2 : aBb$	$B_1 : b$	

И вход: *aabb* (с добавленным символом #)

Грамматики без ε-правил

Определим множество $FIRST_1^G$ ($FIRST_1$, $FIRST$):

Определение

$G = (V_N, V_T, P, S)$ - КС-грамматика

$FIRST_1^G(\alpha) = \{w \in V_T \mid \alpha \overset{*}{\underset{G}{\Rightarrow}} wx$ для некоторой цепочки $x \in V_T^*\}$

где $\alpha \in V^*$.

Вычисление $FIRST_1$

Вычисление:

1. Положим $FIRST(A) = \emptyset$ для $\forall A \in V_N$
2. Для каждого правила $A \to x\alpha, x \in V$
   - $x \in V_T$ => $FIRST(A) \leftarrow FIRST(A) \cup x$
   - $x \in V_N$ => $FIRST(A) \leftarrow FIRST(A) \cup FIRST(x)$
3. Повторить шаг 2, если какое-либо множество $FIRST$ изменилось

Грамматики без ε-правил: пример

Пример:

$$
\begin{array}{lcl}
Session & \to & Fact\ Session \\
Session & \to & Question \\
Session & \to & (\ Session\ )\ Session \\
Fact & \to & !\ STRING \\
Question & \to & ?\ STRING
\end{array}
$$

Посчитаем множества $FIRST$ и таблицу анализа

Грамматики с ε-правилами

ε-правила удобны, например, для списков:

$$
\begin{array}{lcl}
Session & \to & Facts\ Question \mid (\ Session\ )\ Session \\
Facts & \to & Fact\ Facts \mid \varepsilon \\
Fact & \to & !\ STRING \\
Question & \to & ?\ STRING
\end{array}
$$

Дополним множество $FIRST$ ε-цепочками

Определение

$$
\begin{aligned}
FIRST_1^G(\alpha) = \{w \in V_T^* \mid & \text{ либо } w = \varepsilon \text{ и } \alpha \overset{*}{\underset{G}{\Rightarrow}} \varepsilon, \\
& \text{ либо } |w| = 1 \text{ и } \alpha \overset{*}{\underset{G}{\Rightarrow}} wx \\
& \text{ для некоторой цепочки } x \in V_T^*\}.
\end{aligned}
$$

Грамматики с ε-правилами

ε-правила удобны, например, для списков:

$$\begin{array}{lcl} Session & \to & Facts\ Question \mid (\ Session\ )\ Session \\ Facts & \to & Fact\ Facts \mid \varepsilon \\ Fact & \to & !\ STRING \\ Question & \to & ?\ STRING \end{array}$$

Дополним множество $FIRST$ ε-цепочками

Определение

$$FIRST_1^G(\alpha) = \{w \in V_T^* \mid \text{либо } w = \varepsilon \text{ и } \alpha \overset{*}{\underset{G}{\Rightarrow}} \varepsilon,$$
$$\text{либо } |w| = 1 \text{ и } \alpha \overset{*}{\underset{G}{\Rightarrow}} wx$$
$$\text{для некоторой цепочки } x \in V_T^*\}.$$

Множество *FOLLOW*

Определим множество $FOLLOW_1^G$ ($FOLLOW_1$, $FOLLOW$) как возможное множество терминальных символов после данной цепочки при выводе из стартового символа.

Определение

Для КС-грамматики
$G' = (V_N' = V_N \cup \{S'\}, V_T' = V_T \cup \{\#\}, P' = P \cup (S' \to S\#), S')$
построенной по $G = (V_N, V_T, P, S)$ и цепочки $\beta \in V^*$

$$FOLLOW_1^G(\beta) = \{w \in V_T \mid S \overset{*}{\Rightarrow} \gamma\beta\alpha, w \in FIRST_1^G(\alpha)\}$$

Вычисление $FOLLOW_1$

Вычисление:

1. Положим $FOLLOW(A) = \emptyset$ для $\forall A \in V_N$
2. Для каждого правила $B \to \beta A \alpha$ где $\alpha, \beta \in V^*$, $A, B \in V'_N$
   - $FOLLOW(A) \leftarrow FOLLOW(A) \cup (FIRST(\alpha) \setminus \{\varepsilon\})$
   - Если $\alpha \overset{*}{\Rightarrow} \varepsilon$ то $FOLLOW(A) \leftarrow FOLLOW(A) \cup FOLLOW(B)$
3. Повторить шаг 2, если какое-либо множество $FOLLOW$ изменилось

Сильный LL(1) анализатор

При построении таблицы анализатора для правила $A \to \alpha$:
$\forall x \in FIRST(\alpha FOLLOW(A))$ *положим ячейку* $T(A,x) \leftarrow A \to \alpha$

Вопросы