Compiler Construction

Lecture 9: Syntax Analysis V ($LR(0)$ and $SLR(1)$ Parsing)

Summer Semester 2016

Thomas Noll
Software Modeling and Verification Group
RWTH Aachen University

https://moves.rwth-aachen.de/teaching/ss-16/cc/

Software Modeling and Verification Chair

RWTH AACHEN UNIVERSITY

$LR(0)$ Grammars

The case $k = 0$ is relevant (in contrast to $LL(0)$): here the decision is just based on the contents of the pushdown, without any lookahead.

Corollary ($LR(0)$ grammar)

$G \in CFG_\Sigma$ has the $LR(0)$ property if for all rightmost derivations of the form

$$S \begin{cases} \Rightarrow_r^* \alpha A w \Rightarrow_r \alpha\beta w \\ \Rightarrow_r^* \gamma B x \Rightarrow_r \alpha\beta y \end{cases}$$

it follows that $\alpha = \gamma$, $A = B$, and $x = y$.

Goal: derive a finite information from the pushdown which suffices to resolve the nondeterminism (similar to abstraction of right context in LL parsing by fo-sets)

$LR(0)$ Items and Sets

Definition ($LR(0)$ items and sets)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ be start separated by $S' \to S$ and $S' \Rightarrow_r^* \alpha A w \Rightarrow_r \alpha\beta_1\beta_2 w$ (i.e., $A \to \beta_1\beta_2 \in P$).

- $[A \to \beta_1 \cdot \beta_2]$ is called an $LR(0)$ item for $\alpha\beta_1$.
- Given $\gamma \in X^*$, $LR(0)(\gamma)$ denotes the set of all $LR(0)$ items for γ, called the $LR(0)$ set (or: $LR(0)$ information) of γ.
- $LR(0)(G) := \{LR(0)(\gamma) \mid \gamma \in X^*\}$.

Corollary

1. *For every $\gamma \in X^*$, $LR(0)(\gamma)$ is finite.*
2. *$LR(0)(G)$ is finite.*
3. *The item $[A \to \beta\cdot] \in LR(0)(\gamma)$ indicates the possible reduction $(w, \alpha\beta, z) \vdash (w, \alpha A, zi)$ where $\pi_i = A \to \beta$ and $\gamma = \alpha\beta$.*
4. *The item $[A \to \beta_1 \cdot Y\beta_2] \in LR(0)(\gamma)$ indicates an incomplete handle β_1 (to be completed by shift operations or ε-reductions).*

$LR(0)$ Conflicts

Definition ($LR(0)$ conflicts)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ and $I \in LR(0)(G)$.

- I has a shift/reduce conflict if there exist $A \to \alpha_1 a \alpha_2, B \to \beta \in P$ such that

$$[A \to \alpha_1 \cdot a\alpha_2], [B \to \beta \cdot] \in I.$$

- I has a reduce/reduce conflict if there exist $A \to \alpha, B \to \beta \in P$ with $A \neq B$ or $\alpha \neq \beta$ such that

$$[A \to \alpha \cdot], [B \to \beta \cdot] \in I.$$

Lemma

$G \in LR(0)$ iff no $I \in LR(0)(G)$ contains conflicting items.

Proof.

omitted □

Computing $LR(0)$ Sets

Theorem (Computing $LR(0)$ sets)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ be start separated by $S' \to S$ and reduced.

1. $LR(0)(\varepsilon)$ *is the least set such that*
   - $[S' \to \cdot S] \in LR(0)(\varepsilon)$ *and*
   - *if* $[A \to \cdot B\gamma] \in LR(0)(\varepsilon)$ *and* $B \to \beta \in P$,
     then $[B \to \cdot\beta] \in LR(0)(\varepsilon)$.
2. $LR(0)(\alpha Y)$ $(\alpha \in X^*, Y \in X)$ *is the least set such that*
   - *if* $[A \to \gamma_1 \cdot Y\gamma_2] \in LR(0)(\alpha)$,
     then $[A \to \gamma_1 Y \cdot \gamma_2] \in LR(0)(\alpha Y)$ *and*
   - *if* $[A \to \gamma_1 \cdot B\gamma_2] \in LR(0)(\alpha Y)$ *and* $B \to \beta \in P$,
     then $[B \to \cdot\beta] \in LR(0)(\alpha Y)$.

Examples of $LR(0)$ Conflicts

Shift/Reduce Conflicts

Example 9.1

$G: \; S' \to S$
$\quad\;\; S \to aS \mid a$

$LR(0)(\varepsilon): \quad [S' \to \cdot S] \quad [S \to \cdot aS] \; [S \to \cdot a]$
$LR(0)(S): \quad [S' \to S\cdot]$
$LR(0)(a): \quad [S \to a \cdot S] \; [S \to \cdot aS] \; [S \to \cdot a] \; [S \to a\cdot]$
$LR(0)(aS): \quad [S \to aS\cdot]$

Note: G is unambiguous

Examples of $LR(0)$ Conflicts

Reduce/Reduce Conflicts

Example 9.2

$$
\begin{aligned}
G: \quad & S' \to S \\
& S \to Aa \mid Bb \\
& A \to a \\
& B \to a
\end{aligned}
$$

$$
\begin{aligned}
LR(0)(\varepsilon): \quad & [S' \to \cdot S] \quad [S \to \cdot Aa] \; [S \to \cdot Bb] \; [A \to \cdot a] \; [B \to \cdot a] \\
LR(0)(S): \quad & [S' \to S\cdot] \\
LR(0)(A): \quad & [S \to A \cdot a] \\
LR(0)(B): \quad & [S \to B \cdot b] \\
LR(0)(a): \quad & [A \to a\cdot] \quad [B \to a\cdot] \\
LR(0)(Aa): \quad & [S \to Aa\cdot] \\
LR(0)(Bb): \quad & [S \to Bb\cdot]
\end{aligned}
$$

Note: G is unambiguous

The goto Function I

Observation: if $G \in LR(0)$, then $LR(0)(\gamma)$ yields deterministic shift/reduce decision for $NBA(G)$ in a configuration with pushdown γ

$\Longrightarrow$ new pushdown alphabet: $LR(0)(G)$ in place of X

Moreover $LR(0)(\gamma Y)$ is determined by $LR(0)(\gamma)$ and Y but independent from γ in the following sense:

$$LR(0)(\gamma) = LR(0)(\gamma') \implies LR(0)(\gamma Y) = LR(0)(\gamma' Y)$$

Definition 9.3 ($LR(0)$ goto function)

The function $\mathrm{goto} : LR(0)(G) \times X \to LR(0)(G)$ is determined by

$$\mathrm{goto}(I, Y) = I' \quad \text{iff} \quad \text{there exists } \gamma \in X^* \text{ such that } I = LR(0)(\gamma) \text{ and } I' = LR(0)(\gamma Y).$$

The goto Function II

Example 9.4 (cf. Example 8.16)

$I_0 := LR(0)(\varepsilon)$: $[S' \to \cdot S]$ $[S \to \cdot B]$ $[S \to \cdot C]$
$[B \to \cdot aB]$ $[B \to \cdot b]$
$[C \to \cdot aC]$ $[C \to \cdot c]$

$I_1 := LR(0)(S)$: $[S' \to S\cdot]$

$I_2 := LR(0)(B)$: $[S \to B\cdot]$

$I_3 := LR(0)(C)$: $[S \to C\cdot]$

$I_4 := LR(0)(a)$: $[B \to a \cdot B]$ $[C \to a \cdot C]$ $[B \to \cdot aB]$
$[B \to \cdot b]$ $[C \to \cdot aC]$ $[C \to \cdot c]$

$I_5 := LR(0)(b)$: $[B \to b\cdot]$

$I_6 := LR(0)(c)$: $[C \to c\cdot]$

$I_7 := LR(0)(aB)$: $[B \to aB\cdot]$

$I_8 := LR(0)(aC)$: $[C \to aC\cdot]$

$I_9 := \emptyset$

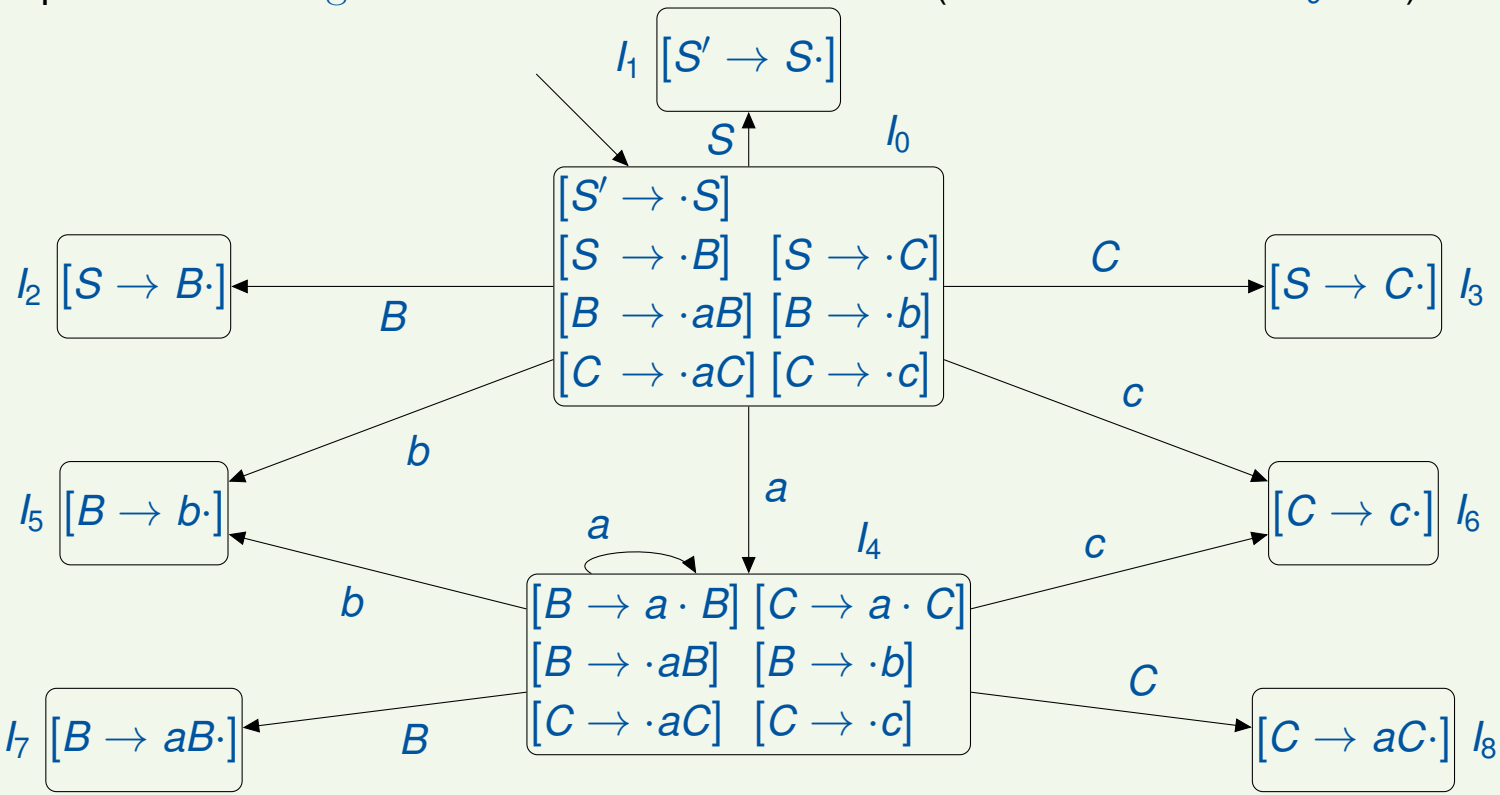
goto	S	B	C	a	b	c
I_0	I_1	I_2	I_3	I_4	I_5	I_6
I_1						
I_2						
I_3						
I_4		I_7	I_8	I_4	I_5	I_6
I_5						
I_6						
I_7						
I_8						
I_9						

(empty = I_9)

The goto Function III

Example 9.4 (continued)

Representation of goto function as finite automaton (omitted: sink state $I_9 = \emptyset$):



The $LR(0)$ Action Function

The parsing automaton will be defined using another table, the action function, which determines the shift/reduce decision (reminder: $\pi_0 = S' \to S$).

Definition 9.5 ($LR(0)$ action function)

The $LR(0)$ action function $\mathrm{act} : LR(0)(G) \to \{\text{red } i \mid i \in [p]\} \cup \{\text{shift}, \text{accept}, \text{error}\}$ is defined by

$$\mathrm{act}(I) := \begin{cases} \text{red } i & \text{if } i \neq 0, \pi_i = A \to \alpha \text{ and } [A \to \alpha\cdot] \in I \\ \text{shift} & \text{if } [A \to \alpha_1 \cdot a\alpha_2] \in I \\ \text{accept} & \text{if } [S' \to S\cdot] \in I \\ \text{error} & \text{if } I = \emptyset \end{cases}$$

Corollary 9.6

For every $G \in CFG_\Sigma$, $G \in LR(0)$ iff act is well defined.

Together, act and goto form the $LR(0)$ parsing table of G.

The $LR(0)$ Parsing Table

Example 9.7 (cf. Example 9.4)

$$
\begin{array}{lll}
G: & S' \to S & (0) \\
& S \to B \mid C & (1,2) \\
& B \to aB \mid b & (3,4) \\
& C \to aC \mid c & (5,6)
\end{array}
$$

$LR(0)(G)$	act	goto					
		S	B	C	a	b	c
I_0	shift	I_1	I_2	I_3	I_4	I_5	I_6
I_1	accept						
I_2	red 1						
I_3	red 2						
I_4	shift		I_7	I_8	I_4	I_5	I_6
I_5	red 4						
I_6	red 6						
I_7	red 3						
I_8	red 5						
I_9	error						

(empty = I_9)

$$
\begin{array}{lll}
I_0 := LR(0)(\varepsilon): & [S' \to \cdot S] & \\
& [S \to \cdot B] \quad [S \to \cdot C] & \\
& [B \to \cdot aB] \quad [B \to \cdot b] & \\
& [C \to \cdot aC] \quad [C \to \cdot c] & \\
I_1 := LR(0)(S): & [S' \to S\cdot] & \\
I_2 := LR(0)(B): & [S \to B\cdot] & \\
I_3 := LR(0)(C): & [S \to C\cdot] & \\
I_4 := LR(0)(a): & [B \to a \cdot B] & [C \to a \cdot C] \\
& [B \to \cdot aB] \quad [B \to \cdot b] & \\
& [C \to \cdot aC] \quad [C \to \cdot c] & \\
I_5 := LR(0)(b): & [B \to b\cdot] & \\
I_6 := LR(0)(c): & [C \to c\cdot] & \\
I_7 := LR(0)(aB): & [B \to aB\cdot] & \\
I_8 := LR(0)(aC): & [C \to aC\cdot] & \\
I_9 := \emptyset & &
\end{array}
$$

The $LR(0)$ Parsing Automaton I

Definition 9.8 ($LR(0)$ parsing automaton)

Let $G = \langle N, \Sigma, P, S \rangle \in LR(0)$. The (deterministic) $LR(0)$ parsing automaton of G is defined by the following components:

- Input alphabet Σ
- Pushdown alphabet $\Gamma := LR(0)(G)$
- Output alphabet $\Delta := [p] \cup \{0, \text{error}\}$
- Configurations $\Sigma^* \times \Gamma^* \times \Delta^*$
- Initial configuration (w, I_0, ε) where $I_0 := LR(0)(\varepsilon)$
- Final configurations $\{\varepsilon\} \times \{\varepsilon\} \times \Delta^*$
- Transitions:

shift: $(aw, \alpha I, z) \vdash (w, \alpha I I', z)$ if $\mathrm{act}(I) = \text{shift}$ and $\mathrm{goto}(I, a) = I'$

reduce: $(w, \alpha I I_1 \dots I_n, z) \vdash (w, \alpha I I', zi)$ if $\mathrm{act}(I_n) = \text{red } i$, $\pi_i = A \to Y_1 \dots Y_n$, $\mathrm{goto}(I, A) = I'$

accept: $(\varepsilon, I_0 I, z) \vdash (\varepsilon, \varepsilon, z\,0)$ if $\mathrm{act}(I) = \text{accept}$

error: $(w, \alpha I, z) \vdash (\varepsilon, \varepsilon, z\,\text{error})$ if $\mathrm{act}(I) = \text{error}$

The $LR(0)$ Parsing Automaton II

Example 9.9 (cf. Example 9.7)

G: $S' \to S$ (0)
$S \to B \mid C$ (1, 2)
$B \to aB \mid b$ (3, 4)
$C \to aC \mid c$ (5, 6)

$LR(0)(G)$	act	goto					
		S	B	C	a	b	c
I_0	shift	I_1	I_2	I_3	I_4	I_5	I_6
I_1	accept						
I_2	red 1						
I_3	red 2						
I_4	shift			I_7	I_8	I_4	I_5	I_6
I_5	red 4						
I_6	red 6						
I_7	red 3						
I_8	red 5						
I_9	error						

(empty = I_9)

$LR(0)$ parsing of aac:

$$
\begin{aligned}
&(aac, I_0 \quad, \varepsilon \quad)\\
\vdash\ &(ac, I_0 I_4 \quad, \varepsilon \quad)\\
\vdash\ &(c, I_0 I_4 I_4 \quad, \varepsilon \quad)\\
\vdash\ &(\varepsilon, I_0 I_4 I_4 I_6, \varepsilon \quad)\\
\vdash\ &(\varepsilon, I_0 I_4 I_4 I_8, 6 \quad)\\
(*)\\
\vdash\ &(\varepsilon, I_0 I_4 I_8 \quad, 65 \quad)\\
\vdash\ &(\varepsilon, I_0 I_3 \quad, 655 \quad)\\
\vdash\ &(\varepsilon, I_0 I_1 \quad, 6552)\\
\vdash\ &(\varepsilon, \varepsilon \quad, 65520)
\end{aligned}
$$

Check by rightmost derivation (on the board)

Remark: in the corresponding computation of $NBA(G)$, $(*)$ is nondeterministic

The $LR(0)$ Parsing Automaton III

Theorem 9.10 (Correctness of $LR(0)$ Parsing Automaton)

If $G \in LR(0)$, then the $LR(0)$ parsing automaton of G is deterministic, and for every $w \in \Sigma^*$ and $z \in \{0, \ldots, p\}^*$:

$$(w, l_0, \varepsilon) \vdash^* (\varepsilon, \varepsilon, z) \quad \textit{iff} \quad \overleftarrow{z} \textit{ is a rightmost analysis of } w$$

Proof.

omitted □

Removing Conflicts in $LR(0)$ Parsing

In practice: often $G \notin LR(0)$

Example 9.11

G_{AE} : $E' \to E$ $\quad E \to E{+}T \mid T$
$T \to T{*}F \mid F$ $\quad F \to (E) \mid \mathtt{a} \mid \mathtt{b}$

$LR(0)(G_{AE})$ with conflicts:

I_0 : $[E' \to \cdot E]$ $[E \to \cdot E{+}T]$ $[E \to \cdot T]$
$[T \to \cdot T{*}F]$ $[T \to \cdot F]$ $[F \to \cdot(E)]$
$[F \to \cdot \mathtt{a}]$ $[F \to \cdot \mathtt{b}]$

I_1 : $[E' \to E\cdot]$ $[E \to E \cdot {+}T]$

I_2 : $[E \to T\cdot]$ $[T \to T \cdot {*}F]$

I_3 : $[T \to F\cdot]$

I_4 : $[F \to (\cdot E)]$ $[E \to \cdot E{+}T]$ $[E \to \cdot T]$
$[T \to \cdot T{*}F]$ $[T \to \cdot F]$ $[F \to \cdot(E)]$
$[F \to \cdot \mathtt{a}]$ $[F \to \cdot \mathtt{b}]$

I_5 : $[F \to \mathtt{a}\cdot]$

I_6 : $[F \to \mathtt{b}\cdot]$

I_7 : $[E \to E{+}\cdot T]$ $[T \to \cdot T{*}F]$ $[T \to \cdot F]$
$[F \to \cdot(E)]$ $[F \to \cdot \mathtt{a}]$ $[F \to \cdot \mathtt{b}]$

I_8 : $[T \to T{*}\cdot F]$ $[F \to \cdot(E)]$
$[F \to \cdot \mathtt{a}]$ $[F \to \cdot \mathtt{b}]$

I_9 : $[F \to (E \cdot)]$ $[E \to E \cdot {+}T]$

I_{10} : $[E \to E{+}T\cdot]$ $[T \to T \cdot {*}F]$

I_{11} : $[T \to T{*}F\cdot]$

I_{12} : $[F \to (E)\cdot]$

Adding Lookahead I

Goal: resolving conflicts by considering next input symbol

Observations:

- $[A \to \beta_1 \cdot a\beta_2] \in LR(0)(\alpha\beta_1)$
  $\implies \exists \alpha \in X^*, w \in \Sigma^* :$

$$S' \Rightarrow_r^* \alpha A w \Rightarrow_r \alpha\beta_1 : a\beta_2 w$$

pushdown — next input symbol

  Thus: shift only on lookahead a

- $[A \to \beta \cdot] \in LR(0)(\alpha\beta)$
  $\implies \exists \alpha \in X^*, x \in \Sigma_\varepsilon, w \in \Sigma^*$ ($x = \varepsilon$ only if $w = \varepsilon$):

$$S' \Rightarrow_r^* \alpha A x w \Rightarrow_r \alpha\beta : xw$$

pushdown — input

  $\implies x \in \mathrm{fo}(A) \subseteq \Sigma_\varepsilon$

  Thus: reduce with $A \to \beta$ only if lookahead $x \in \mathrm{fo}(A)$

Adding Lookahead II

Example 9.12 (cf. Example 9.11)

$$
\begin{aligned}
G_{AE}: \quad E' &\to E && (0)\\
E &\to E{+}T \mid T && (1,2)\\
T &\to T{*}F \mid F && (3,4)\\
F &\to (E) \mid \mathtt{a} \mid \mathtt{b} && (5,6,7)
\end{aligned}
$$

$A \in N$	$\mathrm{fo}(A)$
E'	$\{\varepsilon\}$
E	$\{+,), \varepsilon\}$

- $I_1 = \{[E' \to E\cdot], [E \to E \cdot +T]\}$:
  - accept on lookahead ε
  - shift on lookahead $+$
- $I_2 = \{[E \to T\cdot], [T \to T \cdot *F]\}$:
  - red 2 on lookahead $+/)/\varepsilon$
  - shift on lookahead $*$
- $I_{10} = \{[E \to E{+}T\cdot], [T \to T \cdot *F]\}$:
  - red 1 on lookahead $+/)/\varepsilon$
  - shift on lookahead $*$

$\Longrightarrow$ $SLR(1)$ parsing (Simple $LR(1)$)

The $SLR(1)$ Action Function

Definition 9.13 ($SLR(1)$ action function)

The $SLR(1)$ action function

$$\mathrm{act} : LR(0)(G) \times \Sigma_\varepsilon \to \{\text{red } i \mid i \in [p]\} \cup \{\text{shift}, \text{accept}, \text{error}\}$$

is defined by

$$\mathrm{act}(I, x) := \begin{cases} \text{red } i & \text{if } i \neq 0, \pi_i = A \to \alpha, [A \to \alpha\cdot] \in I, \\ & \text{and } x \in \mathrm{fo}(A) \\ \text{shift} & \text{if } [A \to \alpha_1 \cdot x\alpha_2] \in I \text{ and } x \in \Sigma \\ \text{accept} & \text{if } [S' \to S\cdot] \in I \text{ and } x = \varepsilon \\ \text{error} & \text{otherwise} \end{cases}$$

Definition 9.14 ($SLR(1)$ grammar)

A grammar $G \in CFG_\Sigma$ has the $SLR(1)$ property (notation: $G \in SLR(1)$) if its $SLR(1)$ action function is well defined.

act and the $LR(0)$ goto function (Definition 9.3) form the $SLR(1)$ parsing table of G.

The $SLR(1)$ Parsing Table

Example 9.15 (cf. Example 9.11)

I_0 : $[E' \to \cdot E]$ $[E \to \cdot E{+}T]$ $[E \to \cdot T]$
$[T \to \cdot T{*}F]$ $[T \to \cdot F]$ $[F \to \cdot(E)]$
$[F \to \cdot \mathtt{a}]$ $[F \to \cdot \mathtt{b}]$

I_1 : $[E' \to E\cdot]$ $[E \to E \cdot {+}T]$

I_2 : $[E \to T\cdot]$ $[T \to T \cdot {*}F]$

I_3 : $[T \to F\cdot]$

I_4 : $[F \to (\cdot E)]$ $[E \to \cdot E{+}T]$ $[E \to \cdot T]$
$[T \to \cdot T{*}F]$ $[T \to \cdot F]$ $[F \to \cdot(E)]$
$[F \to \cdot \mathtt{a}]$ $[F \to \cdot \mathtt{b}]$

I_5 : $[F \to \mathtt{a}\cdot]$

I_6 : $[F \to \mathtt{b}\cdot]$

I_7 : $[E \to E{+}\cdot T]$ $[T \to \cdot T{*}F]$ $[T \to \cdot F]$
$[F \to \cdot(E)]$ $[F \to \cdot \mathtt{a}]$ $[F \to \cdot \mathtt{b}]$

I_8 : $[T \to T{*}\cdot F]$ $[F \to \cdot(E)]$
$[F \to \cdot \mathtt{a}]$ $[F \to \cdot \mathtt{b}]$

I_9 : $[F \to (E \cdot)]$ $[E \to E \cdot {+}T]$

I_{10} : $[E \to E{+}T\cdot]$ $[T \to T \cdot {*}F]$

I_{11} : $[T \to T{*}F\cdot]$

I_{12} : $[F \to (E)\cdot]$

$A \in N$	$\mathrm{fo}(A)$
E'	$\{\varepsilon\}$
E	$\{+,), \varepsilon\}$
T	$\{+, *,), \varepsilon\}$
F	$\{+, *,), \varepsilon\}$

$LR(0)(G_{AE})$	act							goto							
	+	*	(	)	a	b	ε	E	T	F	+	*	(	)	a	b
I_0			shift		shift	shift		I_1	I_2	I_3			I_4		I_5	I_6
I_1	shift						accept				I_7					
I_2	red 2	shift		red 2			red 2					I_8				
I_3	red 4	red 4		red 4			red 4									
I_4			shift		shift	shift		I_9	I_2	I_3			I_4		I_5	I_6
I_5	red 6	red 6		red 6			red 6									
I_6	red 7	red 7		red 7			red 7									
I_7			shift		shift	shift			I_{10}	I_3			I_4		I_5	I_6
I_8			shift		shift	shift				I_{11}			I_4		I_5	I_6
I_9	shift			shift							I_7			I_{12}		
I_{10}	red 1	shift		red 1			red 1					I_8				
I_{11}	red 3	red 3		red 3			red 3									
I_{12}	red 5	red 5		red 5			red 5									

The $SLR(1)$ Parsing Automaton

Definition 9.16 ($SLR(1)$ parsing automaton)

The $SLR(1)$ parsing automaton is defined as in the $LR(0)$ case (see Definition 9.8), except for the transition relation:

shift: $(aw, \alpha I, z) \vdash (w, \alpha I I', z)$ if $\mathrm{act}(I, a) = \text{shift}$ and $\mathrm{goto}(I, a) = I'$

reduce_a: $(aw, \alpha I I_1 \dots I_n, z) \vdash (aw, \alpha I I', zi)$ if $\mathrm{act}(I_n, a) = \text{red } i$, $\pi_i = A \to Y_1 \dots Y_n$, and $\mathrm{goto}(I, A) = I'$

$\text{reduce}_\varepsilon$: $(\varepsilon, \alpha I I_1 \dots I_n, z) \vdash (\varepsilon, \alpha I I', zi)$ if $\mathrm{act}(I_n, \varepsilon) = \text{red } i$, $\pi_i = A \to Y_1 \dots Y_n$, and $\mathrm{goto}(I, A) = I'$

accept: $(\varepsilon, I_0 I, z) \vdash (\varepsilon, \varepsilon, z\,0)$ if $\mathrm{act}(I, \varepsilon) = \text{accept}$

error_a: $(aw, \alpha I, z) \vdash (\varepsilon, \varepsilon, z\,\text{error})$ if $\mathrm{act}(I, a) = \text{error}$

error_ε: $(\varepsilon, \alpha I, z) \vdash (\varepsilon, \varepsilon, z\,\text{error})$ if $\mathrm{act}(I, \varepsilon) = \text{error}$