

Compiler Construction

Lecture 11: Syntactic Analysis VII ($LR(1)$ Parsing)

Thomas Noll

Lehrstuhl für Informatik 2
(Software Modeling and Verification)

RWTH Aachen University

`noll@cs.rwth-aachen.de`

`http://www-i2.informatik.rwth-aachen.de/i2/cc08/`

Summer semester 2008

1 Repetition: $LR(0)$ Parsing

2 $SLR(1)$ Parsing

3 $LR(1)$ Parsing

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

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

- $[A \rightarrow \beta_1 \cdot \beta_2]$ 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 .
- $LR(0)(G) := \{LR(0)(\gamma) \mid \gamma \in X^*\}$.

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 $A \rightarrow \alpha_1 a \alpha_2, B \rightarrow \beta \in P$ such that
$$[A \rightarrow \alpha_1 \cdot a \alpha_2], [B \rightarrow \beta \cdot] \in I.$$
- I has a **reduce/reduce conflict** if there exist $A \rightarrow \alpha, B \rightarrow \beta \in P$ with $A \neq B$ or $\alpha \neq \beta$ such that
$$[A \rightarrow \alpha \cdot], [B \rightarrow \beta \cdot] \in I.$$

The $LR(0)$ Action Function

Definition ($LR(0)$ action function)

The $LR(0)$ action function

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

is defined by

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

Corollary

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

The $LR(0)$ Parsing Automaton

Definition ($LR(0)$ 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: $(aw, \alpha I, z) \vdash (w, \alpha II', z)$ if $\text{act}(I) = \text{shift}$ and $\text{goto}(I, a) = I'$

reduce: $(w, \alpha II_1 \dots I_n, z) \vdash (w, \alpha II', zi)$ if $\text{act}(I_n) = \text{red } i$,
 $\pi(i) = A \rightarrow Y_1 \dots Y_n$, and $\text{goto}(I, A) = I'$

accept: $(\varepsilon, I_0 I, z) \vdash (\varepsilon, \varepsilon, z0)$ if $\text{act}(I) = \text{accept}$

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

- 1 Repetition: $LR(0)$ Parsing
- 2 $SLR(1)$ Parsing
- 3 $LR(1)$ Parsing

Conflicts in $LR(0)$ Parsing

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

Example 11.1

$$G_{AE} : \begin{array}{ll} E' \rightarrow E & E \rightarrow E+T \mid T \\ T \rightarrow T*F \mid F & F \rightarrow (E) \mid a \mid b \end{array}$$

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

$$\begin{array}{llll} I_0 : \begin{array}{l} [E' \rightarrow \cdot E] \\ [E \rightarrow \cdot T] \\ [T \rightarrow \cdot F] \\ [F \rightarrow \cdot a] \\ [F \rightarrow \cdot b] \end{array} & \begin{array}{l} [E \rightarrow \cdot E+T] \\ [T \rightarrow \cdot T*F] \\ [F \rightarrow \cdot (E)] \\ [F \rightarrow \cdot b] \end{array} & I_1 : [E' \rightarrow E \cdot] & [E \rightarrow E \cdot +T] \\ & & I_2 : [E \rightarrow T \cdot] & [T \rightarrow T \cdot *F] \\ & & I_3 : [T \rightarrow F \cdot] & \\ I_4 : \begin{array}{l} [F \rightarrow (\cdot E)] \\ [E \rightarrow \cdot T] \\ [T \rightarrow \cdot F] \\ [F \rightarrow \cdot a] \end{array} & \begin{array}{l} [E \rightarrow \cdot E+T] \\ [T \rightarrow \cdot T*F] \\ [F \rightarrow \cdot (E)] \\ [F \rightarrow \cdot b] \end{array} & I_5 : [F \rightarrow a \cdot] & \\ & & I_6 : [F \rightarrow b \cdot] & \\ & & I_7 : \begin{array}{l} [E \rightarrow E+ \cdot T] \\ [T \rightarrow \cdot F] \\ [F \rightarrow \cdot a] \end{array} & \begin{array}{l} [T \rightarrow \cdot T*F] \\ [F \rightarrow \cdot (E)] \\ [F \rightarrow \cdot b] \end{array} \\ I_8 : \begin{array}{l} [T \rightarrow T* \cdot F] \\ [F \rightarrow \cdot a] \end{array} & \begin{array}{l} [F \rightarrow \cdot (E)] \\ [F \rightarrow \cdot b] \end{array} & I_9 : \begin{array}{l} [F \rightarrow (E \cdot)] \\ [E \rightarrow E \cdot +T] \end{array} & [T \rightarrow T \cdot *F] \\ I_{11} : [T \rightarrow T*F \cdot] & & I_{10} : [E \rightarrow E+T \cdot] & \\ & & I_{12} : [F \rightarrow (E) \cdot] & \end{array}$$

Adding Lookahead I

Goal: resolving conflicts by considering first input symbol

Observations:

- $[A \rightarrow \beta_1 \cdot a\beta_2] \in LR(0)(\alpha\beta_1)$

$$\implies S' \Rightarrow_r^* \alpha A w \Rightarrow_r$$

$\alpha\beta_1 a\beta_2 w$
 $\nearrow$ $\nwarrow$
pushdown next input symbol

Thus: shift only on lookahead a

- $[A \rightarrow \beta \cdot] \in LR(0)(\alpha\beta)$

$$\implies S' \Rightarrow_r^* \alpha A x w \Rightarrow_r$$

$\alpha\beta x w$
 $\nearrow$ $\nwarrow$
pushdown input

$$\implies x \in \text{fo}(A) \subseteq \Sigma_\varepsilon \text{ (} x = \varepsilon \text{ only if } w = \varepsilon \text{)}$$

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

Adding Lookahead II

Example 11.2 (cf. Example 11.1)

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

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

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

$\Rightarrow$ **SLR(1) parsing** (Simple LR(1))

The $SLR(1)$ Action Function

Definition 11.3 ($SLR(1)$ action function)

The $SLR(1)$ action function

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

is defined by

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

Definition 11.4 ($SLR(1)$ 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.

Together, act and the $LR(0)$ goto function (cf. Definition ??) form the $SLR(1)$ parsing table of G .

The $SLR(1)$ Parsing Table

Example 11.5 (cf. Example 11.1)

I_0 :	$[E' \rightarrow \cdot E]$ $[E \rightarrow \cdot T]$ $[T \rightarrow \cdot F]$ $[F \rightarrow \cdot a]$	$[E \rightarrow \cdot E+T]$ $[T \rightarrow \cdot T*F]$ $[F \rightarrow \cdot (E)]$ $[F \rightarrow \cdot b]$	I_1 :	$[E' \rightarrow E \cdot]$ $[E \rightarrow T \cdot]$ $[T \rightarrow F \cdot]$	$[E \rightarrow E \cdot +T]$ $[T \rightarrow T \cdot *F]$
I_4 :	$[F \rightarrow (\cdot E)]$ $[E \rightarrow \cdot T]$ $[T \rightarrow \cdot F]$ $[F \rightarrow \cdot a]$	$[E \rightarrow \cdot E+T]$ $[T \rightarrow \cdot T*F]$ $[F \rightarrow \cdot (E)]$ $[F \rightarrow \cdot b]$	I_5 :	$[F \rightarrow a \cdot]$ $[F \rightarrow b \cdot]$ $[E \rightarrow E+ \cdot T]$ $[T \rightarrow \cdot F]$ $[F \rightarrow \cdot a]$	$[T \rightarrow \cdot T*F]$ $[F \rightarrow \cdot (E)]$ $[F \rightarrow \cdot b]$
I_8 :	$[T \rightarrow T* \cdot F]$ $[F \rightarrow \cdot a]$	$[F \rightarrow \cdot (E)]$ $[F \rightarrow \cdot b]$	I_9 :	$[F \rightarrow (E \cdot)]$ $[E \rightarrow E+T \cdot]$ $[F \rightarrow (E) \cdot]$	$[E \rightarrow E \cdot +T]$ $[T \rightarrow T \cdot *F]$
I_{11} :	$[T \rightarrow T*F \cdot]$		I_{12} :		

$A \in N$	$fo(A)$
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 11.6 ($SLR(1)$ parsing automaton)

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

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

reduce_a: $(aw, \alpha II_1 \dots I_n, z) \vdash (aw, \alpha II', zi)$ if $\text{act}(I_n, a) = \text{red } i$, $\pi(i) = A \rightarrow Y_1 \dots Y_n$, and $\text{goto}(I, A) = I'$

reduce _{ϵ} : $(\epsilon, \alpha II_1 \dots I_n, z) \vdash (\epsilon, \alpha II', zi)$ if $\text{act}(I_n, \epsilon) = \text{red } i$, $\pi(i) = A \rightarrow Y_1 \dots Y_n$, and $\text{goto}(I, A) = I'$

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

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

error _{ϵ} : $(\epsilon, \alpha I, z) \vdash (\epsilon, \epsilon, z \text{error})$ if $\text{act}(I, \epsilon) = \text{error}$

- 1 Repetition: $LR(0)$ Parsing
- 2 $SLR(1)$ Parsing
- 3 $LR(1)$ Parsing

SLR(1) Conflicts

Problem: not all conflicts can be resolved using fo sets

Example 11.7

$G_{LR} : S' \rightarrow S \quad S \rightarrow L=R \mid R \quad L \rightarrow *R \mid a \quad R \rightarrow L$

$LR(0)(G_{LR})$:

$I_0 := LR(0)(\varepsilon) :$	$[S' \rightarrow \cdot S]$	$[S \rightarrow \cdot L=R]$	$[S \rightarrow \cdot R]$
	$[L \rightarrow \cdot *R]$	$[L \rightarrow \cdot a]$	$[R \rightarrow \cdot L]$
$I_1 := LR(0)(S) :$	$[S' \rightarrow S \cdot]$		
$I_2 := LR(0)(L) :$	$[S \rightarrow L \cdot =R]$	$[R \rightarrow L \cdot]$	
$I_3 := LR(0)(R) :$	$[S \rightarrow R \cdot]$		
$I_4 := LR(0)(*) :$	$[L \rightarrow * \cdot R]$	$[R \rightarrow \cdot L]$	$[L \rightarrow \cdot *R] \quad [L \rightarrow \cdot a]$
$I_5 := LR(0)(a) :$	$[L \rightarrow a \cdot]$		
$I_6 := LR(0)(L=) :$	$[S \rightarrow L= \cdot R]$	$[R \rightarrow \cdot L]$	$[L \rightarrow \cdot *R] \quad [L \rightarrow \cdot a]$
$I_7 := LR(0)(*R) :$	$[L \rightarrow *R \cdot]$		
$I_8 := LR(0)(*L) :$	$[R \rightarrow L \cdot]$		
$I_9 := LR(0)(L=R) :$	$[S \rightarrow L=R \cdot]$		

But: conflict in I_2 not $SLR(1)$ -solvable since $= \in \text{fo}(R)$

Observation: not every element of $\text{fo}(A)$ can follow every occurrence of A

$\implies$ refinement of $LR(0)$ items by adding possible lookahead symbols

Definition 11.8 ($LR(1)$ items and sets)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_{\Sigma}$ be start separated by $S' \rightarrow S$.

- If $S' \Rightarrow_r^* \alpha A a w \Rightarrow_r \alpha \beta_1 \beta_2 a w$, then $[A \rightarrow \beta_1 \cdot \beta_2, a]$ is called an **$LR(1)$ item** for $\alpha \beta_1$.
- If $S' \Rightarrow_r^* \alpha A \Rightarrow_r \alpha \beta_1 \beta_2$, then $[A \rightarrow \beta_1 \cdot \beta_2, \varepsilon]$ is called an **$LR(1)$ item** for $\alpha \beta_1$.
- Given $\gamma \in X^*$, $LR(1)(\gamma)$ denotes the set of all $LR(1)$ items for γ , called the **$LR(1)$ set** (or: **$LR(1)$ information**) of γ .
- $LR(1)(G) := \{LR(1)(\gamma) \mid \gamma \in X^*\}$.

Corollary 11.9

- ❶ For every $\gamma \in X^*$, $LR(1)(\gamma)$ is finite.
- ❷ $LR(1)(G)$ is finite.
- ❸ For every $\gamma \in X^*$, $LR(1)(\gamma)$ “contains” $LR(0)(\gamma)$, i.e.,

$$\{[A \rightarrow \beta_1 \cdot \beta_2] \mid [A \rightarrow \beta_1 \cdot \beta_2, x] \in LR(1)(\gamma)\} = LR(0)(\gamma).$$

- ❹ $[A \rightarrow \beta_1 \cdot \beta_2, x] \in LR(1)(G) \implies x \in \text{fo}(A)$

Definition 11.10 ($LR(1)$ conflicts)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ and $I \in LR(1)(G)$.

- I has a **shift/reduce conflict** if there exist $A \rightarrow \alpha_1 a \alpha_2, B \rightarrow \beta \in P$ and $x \in \Sigma_\epsilon$ such that

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

- I has a **reduce/reduce conflict** if there exist $x \in \Sigma_\epsilon$ and $A \rightarrow \alpha, B \rightarrow \beta \in P$ with $A \neq B$ or $\alpha \neq \beta$ such that

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

Lemma 11.11

$G \in LR(1)$ iff no $I \in LR(1)(G)$ contains conflicting items.

The computation of $LR(0)$ sets (cf. Theorem 9.14) can be extended to cover right contexts:

Theorem 11.12 (Computing $LR(1)$ sets)

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

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

Computing $LR(1)$ Sets II

Example 11.13 (cf. Example 11.7)

$G_{LR} : S' \rightarrow S \quad S \rightarrow L=R \mid R \quad L \rightarrow *R \mid a \quad R \rightarrow L$

$LR(1)(G_{LR}) : [S' \rightarrow \cdot S, \varepsilon] \in LR(1)(\varepsilon) \quad [A \rightarrow \cdot B\gamma, x] \in LR(1)(\varepsilon), B \rightarrow \beta \in P, y \in \text{fi}(\gamma x)$
 $\implies [B \rightarrow \cdot \beta, y] \in LR(1)(\varepsilon)$

$I'_0 := LR(1)(\varepsilon) :$	$[S' \rightarrow \cdot S, \varepsilon]$	$[S \rightarrow \cdot L=R, \varepsilon]$	$[S \rightarrow \cdot R, \varepsilon]$	$[L \rightarrow \cdot *R, =]$
	$[L \rightarrow \cdot a, =]$	$[R \rightarrow \cdot L, \varepsilon]$	$[L \rightarrow \cdot *R, \varepsilon]$	$[L \rightarrow \cdot a, \varepsilon]$
$I'_1 := LR(1)(S) :$	$[S' \rightarrow S \cdot, \varepsilon]$			
$I'_2 := LR(1)(L) :$	$[S \rightarrow L \cdot =R, \varepsilon]$	$[R \rightarrow L \cdot, \varepsilon]$		
$I'_3 := LR(1)(R) :$	$[S \rightarrow R \cdot, \varepsilon]$			
$I'_4 := LR(1)(*) :$	$[L \rightarrow * \cdot R, =]$	$[L \rightarrow * \cdot R, \varepsilon]$	$[R \rightarrow \cdot L, =]$	$[R \rightarrow \cdot L, \varepsilon]$
	$[L \rightarrow * \cdot R, =]$	$[L \rightarrow \cdot a, =]$	$[L \rightarrow \cdot *R, \varepsilon]$	$[L \rightarrow \cdot a, \varepsilon]$
$I'_5 := LR(1)(a) :$	$[L \rightarrow a \cdot, =]$	$[L \rightarrow a \cdot, \varepsilon]$		
$I'_6 := LR(1)(L=) :$	$[S \rightarrow L= \cdot R, \varepsilon]$	$[R \rightarrow \cdot L, \varepsilon]$	$[L \rightarrow \cdot *R, \varepsilon]$	$[L \rightarrow \cdot a, \varepsilon]$
$I'_7 := LR(1)(*R) :$	$[L \rightarrow *R \cdot, =]$	$[L \rightarrow *R \cdot, \varepsilon]$		
$I'_8 := LR(1)(*L) :$	$[R \rightarrow L \cdot, =]$	$[R \rightarrow L \cdot, \varepsilon]$		
$I'_9 := LR(1)(L=R) :$	$[S \rightarrow L=R \cdot, \varepsilon]$			
$I'_{10} := LR(1)(L=L) :$	$[R \rightarrow L \cdot, \varepsilon]$			
$I'_{11} := LR(1)(L=*) :$	$[L \rightarrow * \cdot R, \varepsilon]$	$[R \rightarrow \cdot L, \varepsilon]$	$[L \rightarrow \cdot *R, \varepsilon]$	$[L \rightarrow \cdot a, \varepsilon]$
$I'_{12} := LR(1)(L=a) :$	$[L \rightarrow a \cdot, \varepsilon]$			
$I'_{13} := LR(1)(L=*R) :$	$[L \rightarrow *R \cdot, \varepsilon]$			
$I'_{14} := \emptyset$				

In I'_2 : shift on $=$ /reduce on $\varepsilon \implies G_{LR} \in LR(1)$

The $LR(1)$ Action Function

Definition 11.14 ($LR(1)$ action function)

The $LR(1)$ action function

$$\text{act} : LR(1)(G) \times \Sigma_\varepsilon \rightarrow \{\text{red } i \mid i \in [p]\} \cup \{\text{shift, accept, error}\}$$

is defined by

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

Corollary 11.15

For every $G \in CFG_\Sigma$, $G \in LR(1)$ iff its $LR(1)$ action function is well defined.

The $LR(1)$ goto Function

The goto function is defined in analogy to the $LR(0)$ case (Definition ??). Likewise, it can be obtained using a **powerset construction**.

Definition 11.16 ($LR(1)$ goto function)

The function $\text{goto} : LR(1)(G) \times X \rightarrow LR(1)(G)$ is determined by

$$\text{goto}(I, Y) = I' \quad \text{iff} \quad \begin{array}{l} \text{there exists } \gamma \in X^* \text{ such that} \\ I = LR(1)(\gamma) \text{ and } I' = LR(1)(\gamma Y). \end{array}$$

Again, act and goto form the **$LR(1)$ parsing table** of G .

The $LR(1)$ Parsing Table

Example 11.17 (cf. Example 11.13)

$LR(1)(G_{LR})$	act/goto Σ_ϵ				goto N		
	*	=	a	ϵ	S	L	R
I'_0	shift/ I'_4		shift/ I'_5		I'_1	I'_2	I'_3
I'_1				accept			
I'_2		shift/ I'_6		red 5			
I'_3				red 2			
I'_4	shift/ I'_4		shift/ I'_5			I'_8	I'_7
I'_5		red 4					
I'_6	shift/ I'_{11}		shift/ I'_{12}		I'_{10}	I'_9	
I'_7		red 3					
I'_8		red 5					
I'_9				red 1			
I'_{10}				red 5			
I'_{11}	shift/ I'_{11}		shift/ I'_{12}		I'_{10}	I'_{13}	
I'_{12}				red 4			
I'_{13}				red 3			

(empty = error/ $\emptyset$)

The $LR(1)$ Parsing Automaton I

Definition 11.18 ($LR(1)$ parsing automaton)

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

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

reduce_a: $(aw, \alpha II_1 \dots I_n, z) \vdash (aw, \alpha II', zi)$ if $\text{act}(I_n, a) = \text{red } i$, $\pi(i) = A \rightarrow Y_1 \dots Y_n$, and $\text{goto}(I, A) = I'$

reduce_ε: $(\varepsilon, \alpha II_1 \dots I_n, z) \vdash (\varepsilon, \alpha II', zi)$ if $\text{act}(I_n, \varepsilon) = \text{red } i$, $\pi(i) = A \rightarrow Y_1 \dots Y_n$, and $\text{goto}(I, A) = I'$

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

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

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

The $LR(1)$ Parsing Automaton II

Example 11.19 (cf. Example 11.13)

$G_{LR} : S' \rightarrow S$ (0) $S \rightarrow L=R$ | R (1,2) $L \rightarrow *R$ | a (3,4) $R \rightarrow L$ (5)

$LR(1)(G_{LR})$	act/goto Σ_ϵ				goto N		
	*	=	a	ϵ	S	L	R
I'_0	shift/ I'_4		shift/ I'_5		I'_1	I'_2	I'_3
I'_1				accept			
I'_2		shift/ I'_6		red 5			
I'_3				red 2			
I'_4	shift/ I'_4		shift/ I'_5		I'_8	I'_7	
I'_5		red 4					
I'_6	shift/ I'_{11}		shift/ I'_{12}		I'_{10}	I'_9	
I'_7		red 3					
I'_8		red 5					
I'_9				red 1			
I'_{10}				red 5			
I'_{11}	shift/ I'_{11}		shift/ I'_{12}		I'_{10}	I'_{13}	
I'_{12}				red 4			
I'_{13}				red 3			

(empty = error/ $\emptyset$)

$LR(1)$ parsing of $a=*a$:

$(a=*a, I'_0, \epsilon)$
 $\vdash (=*a, I'_0 I'_5, \epsilon)$
 $\vdash (=*a, I'_0 I'_2, 4)$
 $\vdash (*a, I'_0 I'_2 I'_6, 4)$
 $\vdash (a, I'_0 I'_2 I'_6 I'_{11}, 4)$
 $\vdash (\epsilon, I'_0 I'_2 I'_6 I'_{11} I'_{12}, 4)$
 $\vdash (\epsilon, I'_0 I'_2 I'_6 I'_{11} I'_{10}, 44)$
 $\vdash (\epsilon, I'_0 I'_2 I'_6 I'_{11} I'_{13}, 445)$
 $\vdash (\epsilon, I'_0 I'_2 I'_6 I'_{10}, 4453)$
 $\vdash (\epsilon, I'_0 I'_2 I'_6 I'_9, 44535)$
 $\vdash (\epsilon, I'_0 I'_{11}, 445351)$
 $\vdash (\epsilon, \epsilon, 4453510)$