Compiler Construction

Lecture 10: Syntax Analysis VI ($LR(0)$ and $SLR(1)$ Parsing)

Thomas Noll

Lehrstuhl für Informatik 2
(Software Modeling and Verification)

RWTH AACHEN UNIVERSITY

noll@cs.rwth-aachen.de

http://moves.rwth-aachen.de/teaching/ss-14/cc14/

Summer Semester 2014

Outline

1. Recap: $LR(0)$ Grammars
2. The $LR(0)$ Parsing Automaton
3. $SLR(1)$ Parsing
4. $LR(1)$ Parsing

$LR(0)$ 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*-steps).*

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 □

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 ($LR(0)$ 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

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

Outline

1. Recap: $LR(0)$ Grammars
2. The $LR(0)$ Parsing Automaton
3. $SLR(1)$ Parsing
4. $LR(1)$ Parsing

The $LR(0)$ Parsing Table

Example 10.1 (cf. Example 9.15)

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

$LR(0)(G)$	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 $= I_9$)

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

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

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

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

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

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

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

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

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

$I_9 := \emptyset$

The $LR(0)$ Parsing Automaton I

Definition 10.2 ($LR(0)$ 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: $(w, \alpha I I_1 \ldots I_n, z) \vdash (w, \alpha I I', zi)$ if $\mathrm{act}(I_n) = \text{red}\, i$, $\pi_i = A \to Y_1 \ldots Y_n$, and $\mathrm{goto}(I, A) = I'$

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 $\mathrm{act}(I) = \text{error}$

Example 10.3 (cf. Example 10.1)

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

$LR(0)(G)$	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 $= I_9$)

$LR(0)$ parsing of aac:

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

Check by rightmost derivation (on the board)

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

The $LR(0)$ Parsing Automaton III

Theorem 10.4 (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 $z \in \{0, \ldots, p\}^*$:

$$(w, I_0, \varepsilon) \vdash^* (\varepsilon, \varepsilon, z) \quad \textit{iff} \quad \overleftarrow{z} \textit{ is a rightmost analysis of } w$$

Proof.

omitted □

Outline

1. Recap: $LR(0)$ Grammars
2. The $LR(0)$ Parsing Automaton
3. $SLR(1)$ Parsing
4. $LR(1)$ Parsing

Removing Conflicts in $LR(0)$ Parsing

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

Example 10.5

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

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

- $[A \to \beta_1 \cdot a\beta_2] \in LR(0)(\alpha\beta_1)$
  $\implies S' \Rightarrow_r^* \alpha A w \Rightarrow_r \alpha\beta_1 a \beta_2 w$
  (pushdown: $\alpha\beta_1$; next input symbol: a)

  Thus: shift only on lookahead a

- $[A \to \beta\cdot] \in LR(0)(\alpha\beta)$
  $\implies S' \Rightarrow_r^* \alpha A x w \Rightarrow_r \alpha\beta x w$
  (pushdown: $\alpha\beta$; input: xw)
  $\implies x \in \mathrm{fo}(A) \subseteq \Sigma_\varepsilon$ ($x = \varepsilon$ only if $w = \varepsilon$)

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

Adding Lookahead II

Example 10.6 (cf. Example 10.5)

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

$A \in N$	$\mathrm{fo}(A)$
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 +/)/ε
  - shift on lookahead *
- $I_{10} = \{[E \to E{+}T\cdot], [T \to T \cdot {*} F]\}$:
  - red 1 on lookahead +/)/ε
  - shift on lookahead *

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

The $SLR(1)$ Action Function

Definition 10.7 ($SLR(1)$ action function)

The $SLR(1)$ action function

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

is defined by

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

Definition 10.8 ($SLR(1)$ 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 9.14) form the $SLR(1)$ parsing table of G.

Example 10.9 (cf. Example 10.5)

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

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

I_5 : $[F \to a \cdot]$

I_6 : $[F \to b \cdot]$

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

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

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

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

The $SLR(1)$ parsing automaton is defined as in the $LR(0)$ case (see Definition 10.2), except for the transition relation:

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

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

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

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

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

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

Outline

1. Recap: $LR(0)$ Grammars
2. The $LR(0)$ Parsing Automaton
3. $SLR(1)$ Parsing
4. $LR(1)$ Parsing

$SLR(1)$ Conflicts

Problem: not all conflicts can be resolved using fo sets

Example 10.11

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

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

$$
\begin{array}{lllll}
I_0 := LR(0)(\varepsilon): & [S' \to \cdot S] & [S \to \cdot L{=}R] & [S \to \cdot R] & \\
 & [L \to \cdot *R] & [L \to \cdot \mathrm{a}] & [R \to \cdot L] & \\
I_1 := LR(0)(S): & [S' \to S\cdot] & & & \\
I_2 := LR(0)(L): & [S \to L \cdot {=}R] & [R \to L\cdot] & & \\
I_3 := LR(0)(R): & [S \to R\cdot] & & & \\
I_4 := LR(0)(*): & [L \to * \cdot R] & [R \to \cdot L] & [L \to \cdot *R] & [L \to \cdot \mathrm{a}] \\
I_5 := LR(0)(\mathrm{a}): & [L \to \mathrm{a}\cdot] & & & \\
I_6 := LR(0)(L{=}): & [S \to L{=} \cdot R] & [R \to \cdot L] & [L \to \cdot *R] & [L \to \cdot \mathrm{a}] \\
I_7 := LR(0)(*R): & [L \to *R\cdot] & & & \\
I_8 := LR(0)(*L): & [R \to L\cdot] & & & \\
I_9 := LR(0)(L{=}R): & [S \to L{=}R\cdot] & & &
\end{array}
$$

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

$LR(1)$ Items and Sets I

Observation: not every element of $\mathrm{fo}(A)$ can follow every occurrence of A
$\Longrightarrow$ refinement of $LR(0)$ items by adding possible lookahead symbols

Definition 10.12 ($LR(1)$ items and sets)

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

- If $S' \Rightarrow_r^* \alpha A a w \Rightarrow_r \alpha\beta_1\beta_2 a w$, then $[A \to \beta_1 \cdot \beta_2, a]$ is called an $LR(1)$ item for $\alpha\beta_1$.
- If $S' \Rightarrow_r^* \alpha A \Rightarrow_r \alpha\beta_1\beta_2$, then $[A \to \beta_1 \cdot \beta_2, \varepsilon]$ 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.
- $LR(1)(G) := \{LR(1)(\gamma) \mid \gamma \in X^*\}$.

$LR(1)$ Items and Sets II

Corollary 10.13

1. *For every* $\gamma \in X^*$, $LR(1)(\gamma)$ *is finite.*
2. $LR(1)(G)$ *is finite.*
3. *For every* $\gamma \in X^*$, $LR(1)(\gamma)$ *"contains"* $LR(0)(\gamma)$, *i.e.,*

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

4. $[A \to \beta_1 \cdot \beta_2, x] \in I \in LR(1)(G) \implies x \in \mathrm{fo}(A)$

$LR(1)$ Conflicts

Definition 10.14 ($LR(1)$ 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 $A \to \alpha_1 a \alpha_2, B \to \beta \in P$ and $x \in \Sigma_\varepsilon$ such that
$$[A \to \alpha_1 \cdot a\alpha_2, x], [B \to \beta\cdot, a] \in I.$$
- I has a reduce/reduce conflict if there exist $x \in \Sigma_\varepsilon$ and $A \to \alpha, B \to \beta \in P$ with $A \neq B$ or $\alpha \neq \beta$ such that
$$[A \to \alpha\cdot, x], [B \to \beta\cdot, x] \in I.$$

Lemma 10.15

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

Computing $LR(1)$ Sets I

The computation of $LR(0)$ sets (cf. Theorem 9.10) can be extended to cover right contexts:

Theorem 10.16 (Computing $LR(1)$ sets)

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

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

Computing $LR(1)$ Sets II

Example 10.17 (cf. Example 10.11)

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

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

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

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

The $LR(1)$ Action Function

Definition 10.18 ($LR(1)$ action function)

The $LR(1)$ action function

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

is defined by

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

Corollary 10.19

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 9.14).

Definition 10.20 ($LR(1)$ goto function)

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

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

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

The $LR(1)$ Parsing Table

Example 10.21 (cf. Example 10.17)

$LR(1)(G_{LR})$	act/goto$\vert_\Sigma$ *	=	a	ε	goto$\vert_N$ 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		red 4			
I'_6	shift/I'_{11}		shift/I'_{12}			I'_{10}	I'_9
I'_7		red 3		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 10.22 ($LR(1)$ parsing automaton)

The $LR(1)$ parsing automaton is defined as in the $LR(0)$ case (see Definition 10.2), except for the transition relation:

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

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

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

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

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

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

Example 10.23 (cf. Example 10.17)

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

$LR(1)(G_{LR})$	act/goto$\|_\Sigma$ *	=	a	ε	goto$\|_N$ 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		red 4			
I'_6	shift/I'_{11}		shift/I'_{12}			I'_{10}	I'_9
I'_7		red 3		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:

$$
\begin{aligned}
& (\texttt{a=*a},\ I'_0,\ \varepsilon) \\
\vdash\ & (\texttt{=*a},\ I'_0 I'_5,\ \varepsilon) \\
\vdash\ & (\texttt{=*a},\ I'_0 I'_2,\ 4) \\
\vdash\ & (\texttt{*a},\ I'_0 I'_2 I'_6,\ 4) \\
\vdash\ & (\texttt{a},\ I'_0 I'_2 I'_6 I'_{11},\ 4) \\
\vdash\ & (\varepsilon,\ I'_0 I'_2 I'_6 I'_{11} I'_{12},\ 4) \\
\vdash\ & (\varepsilon,\ I'_0 I'_2 I'_6 I'_{11} I'_{10},\ 44) \\
\vdash\ & (\varepsilon,\ I'_0 I'_2 I'_6 I'_{11} I'_{13},\ 445) \\
\vdash\ & (\varepsilon,\ I'_0 I'_2 I'_6 I'_{10},\ 4453) \\
\vdash\ & (\varepsilon,\ I'_0 I'_2 I'_6 I'_9,\ 44535) \\
\vdash\ & (\varepsilon,\ I'_0 I'_1,\ 445351) \\
\vdash\ & (\varepsilon,\ \varepsilon,\ 4453510)
\end{aligned}
$$