Compiler Construction

Lecture 9: Syntax Analysis V ($LR(k)$ Grammars)

Thomas Noll

Lehrstuhl für Informatik 2
(Software Modeling and Verification)

noll@cs.rwth-aachen.de

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

Summer Semester 2014

Outline

Approach:

1. Given $G \in CFG_\Sigma$, construct a nondeterministic bottom-up parsing automaton (NBA) which accepts $L(G)$ and which additionally computes corresponding (reversed) rightmost analyses
   - input alphabet: Σ
   - pushdown alphabet: X
   - output alphabet: $[p]$ (where $p := |P|$)
   - state set: omitted
   - transitions:

     shift: shifting input symbols onto the pushdown
     reduce: replacing the right-hand side of a production by its left-hand side (= inverse expansion steps)
2. Remove nondeterminism by allowing lookahead on the input: $G \in LR(k)$ iff $L(G)$ recognizable by deterministic bottom-up parsing automaton with lookahead of k symbols

Nondeterministic Bottom-Up Automaton I

Definition (Nondeterministic bottom-up parsing automaton)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$. The nondeterministic bottom-up parsing automaton of G, $\mathrm{NBA}(G)$, is defined by the following components.

- Input alphabet: Σ
- Pushdown alphabet: X
- Output alphabet: $[p]$
- Configurations: $\Sigma^* \times X^* \times [p]^*$ (top of pushdown to the right)
- Transitions for $w \in \Sigma^*$, $\alpha \in X^*$, and $z \in [p]^*$:

  shifting steps: $(aw, \alpha, z) \vdash (w, \alpha a, z)$ if $a \in \Sigma$

  reduction steps: $(w, \alpha\beta, z) \vdash (w, \alpha A, zi)$ if $\pi_i = A \to \beta$
- Initial configuration for $w \in \Sigma^*$: $(w, \varepsilon, \varepsilon)$
- Final configurations: $\{\varepsilon\} \times \{S\} \times [p]^*$

Nondeterminisn in $\mathrm{NBA}(G)$

Observation: $\mathrm{NBA}(G)$ is generally nondeterministic

- Shift or reduce? Example:

$$(bw, \alpha a, z) \vdash \begin{cases} (w, \alpha ab, z) \\ (bw, \alpha A, zi) \end{cases} \text{if } \pi_i = A \to a$$

- If reduce: which "handle" β? Example:

$$(w, \alpha ab, z) \vdash \begin{cases} (w, \alpha A, zi) \\ (w, \alpha aB, zj) \end{cases} \text{if } \pi_i = A \to ab \text{ and } \pi_j = B \to b$$

- If reduce β: which left-hand side A? Example:

$$(w, \alpha a, z) \vdash \begin{cases} (w, \alpha A, zi) \\ (w, \alpha B, zj) \end{cases} \text{if } \pi_i = A \to a \text{ and } \pi_j = B \to a$$

- When to terminate parsing? Example:

$$\underbrace{(\varepsilon, S, z)}_{\text{final}} \vdash (\varepsilon, A, zi) \text{ if } \pi_i = A \to S$$

Outline

Resolving Termination Nondeterminism I

General assumption to avoid nondeterminism of last type:
every grammar is start separated

Definition 9.1 (Start separation)

A grammar $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ is called start separated if S only occurs in productions of the form $S \to A$ where $A \neq S$.

Remarks:

- Start separation always possible by adding $S' \to S$ with new start symbol S'
- From now on consider only reduced grammars of this form ($\pi_0 := S' \to S$)

Resolving Termination Nondeterminism II

Start separation removes last form of nondeterminism ("When to terminate parsing?"):

Lemma 9.2

If $G \in CFG_\Sigma$ is start separated, then no successor of a final configuration (ε, S', z) in $\mathrm{NBA}(G)$ is again a final configuration.
(Thus parsing should be stopped in the first final configuration.)

Proof.

- To (ε, S', z), only reductions by ε-productions can be applied:
$$(\varepsilon, S', z) \vdash (\varepsilon, S'A, zi) \quad \text{if } \pi_i = A \to \varepsilon$$
- Thereafter, only reductions by productions of the form $A_0 \to A_1 \dots A_n$ $(n \geq 0)$ can be applied
- Every resulting configuration is of the (non-final) form
$$(\varepsilon, S'B_1 \dots B_k, z) \quad \text{where } k \geq 1$$

$\square$

Outline

Goal: resolve remaining nondeterminism of $\mathrm{NBA}(G)$ by supporting lookahead of $k \in \mathbb{N}$ symbols on the input

$\Longrightarrow$ $LR(k)$: reading of input from left to right with k-lookahead, computing a rightmost analysis

Definition 9.3 ($LR(k)$ grammar)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ be start separated and $k \in \mathbb{N}$. Then G has the $LR(k)$ property (notation: $G \in LR(k)$) if for all rightmost derivations of the form

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

such that $\mathrm{first}_k(w) = \mathrm{first}_k(y)$, it follows that $\alpha = \gamma$, $A = B$, and $x = y$.

Remarks:

- If $G \in LR(k)$, then the reduction of $\alpha\beta w$ to $\alpha A w$ is already determined by $\mathrm{first}_k(w)$.
- Therefore $\mathrm{NBA}(G)$ in configuration $(w, \alpha\beta, z)$ can decide to reduce and how to reduce.
- Computation of $\mathrm{NBA}(G)$ for $S \Rightarrow_r^* \alpha A w \Rightarrow_r \alpha\beta w$:

$$(w'w, \varepsilon, \varepsilon) \vdash^* (w, \alpha\beta, z) \overset{\text{red } i}{\vdash} (w, \alpha A, zi) \vdash \ldots$$

  where $\pi_i = A \to \beta$
- Computation of $\mathrm{NBA}(G)$ for $S \Rightarrow_r^* \gamma B x \Rightarrow_r \alpha\beta y$:
  - with direct reduction ($y = x$, $\alpha\beta = \gamma\delta$, $\pi_j = B \to \delta$):

$$(y'y, \varepsilon, \varepsilon) \vdash^* (y, \alpha\beta, z') = (x, \gamma\delta, z') \overset{\text{red } j}{\vdash} (x, \gamma B, z'j) \vdash \ldots$$

  - with previous shifts ($y = x'x$, $\alpha\beta x' = \gamma\delta$, $\pi_j = B \to \delta$):

$$\begin{aligned}(y'y, \varepsilon, \varepsilon) &\vdash^* (y, \alpha\beta, z') = (x'x, \alpha\beta, z') \\ &\overset{\text{shift}^*}{\vdash} (x, \alpha\beta x', z') = (x, \gamma\delta, z') \\ &\overset{\text{red } j}{\vdash} (x, \gamma B, z'j) \vdash \ldots\end{aligned}$$

Outline

$LR(0)$ 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 9.4 ($LR(0)$ 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 $x = y$.

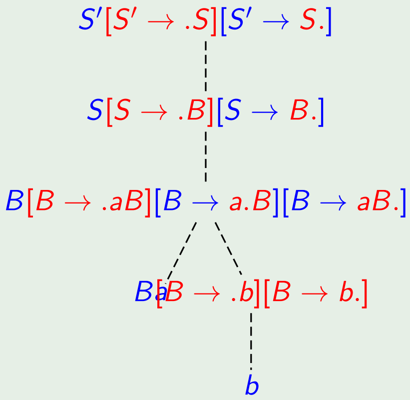
Goal: derive a finite information from the pushdown which suffices to resolve the nondeterminism (similar to abstraction of right context in LL parsing by fo-sets)

Example 9.5

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

$\mathrm{NBA}(G)$:

$$
\begin{array}{ll}
 & (ab, \varepsilon, \varepsilon) \\
\vdash & (b, a, \varepsilon) \\
\vdash & (\varepsilon, ab, \varepsilon) \\
\vdash & (\varepsilon, aB, 4) \\
\vdash & (\varepsilon, B, 43) \\
\vdash & (\varepsilon, S, 431) \\
\vdash & (\varepsilon, S', 4310)
\end{array}
$$



Definition 9.6 ($LR(0)$ 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 9.7

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 9.8 ($LR(0)$ 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 9.9

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

Proof.

omitted $\square$

Computing $LR(0)$ Sets I

Theorem 9.10 (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* $[B \to \cdot\beta] \in LR(0)(\alpha Y)$.

Computing $LR(0)$ Sets II

Example 9.11 (cf. Example 9.5)

$$G: \quad \begin{aligned} S' &\to S \\ S &\to B \mid C \\ B &\to aB \mid b \\ C &\to aC \mid c \end{aligned} \qquad [S' \to \cdot S] \in$$

$$LR(0)(\varepsilon) \quad \begin{array}{l} [A \to \cdot B\gamma] \in LR(0)(\varepsilon), B \to \beta \in P \\ \Longrightarrow \ [B \to \cdot\beta] \in LR(0)(\varepsilon) \end{array} \quad \begin{array}{l} [A \to \gamma_1 \cdot Y\gamma_2] \in LR(0)(\alpha) \\ \Longrightarrow \ [A \to \gamma_1 Y \cdot \gamma_2] \in LR(0)(\alpha Y) \end{array}$$

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

($LR(0)(aa) = LR(0)(a) = I_4$, $LR(0)(ab) = LR(0)(b) = I_5$,
$LR(0)(ac) = LR(0)(c) = I_6$, $I_9 := LR(0)(\gamma) = \emptyset$ in all remaining cases)

Outline

Reduce/Reduce Conflicts

Example 9.12

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

$$
\begin{array}{lllllll}
LR(0)(\varepsilon): & [S' \to \cdot S] & [S \to \cdot Aa] & [S \to \cdot Bb] & [A \to \cdot a] & [B \to \cdot a] \\
LR(0)(S): & [S' \to S\cdot] \\
LR(0)(A): & [S \to A \cdot a] \\
LR(0)(B): & [S \to B \cdot a] \\
LR(0)(a): & [A \to a\cdot] & [B \to a\cdot] \\
LR(0)(Aa): & [S \to Aa\cdot] \\
LR(0)(Ba): & [S \to Ba\cdot]
\end{array}
$$

Note: G is unambiguous

Shift/Reduce Conflicts

Example 9.13

$$G: \quad \begin{array}{lll} S' & \to & S \\ S & \to & aS \mid a \end{array}$$

$$\begin{array}{lllll}
LR(0)(\varepsilon): & [S' \to \cdot S] & [S \to \cdot aS] & [S \to \cdot a] & \\
LR(0)(S): & [S' \to S \cdot] & & & \\
LR(0)(a): & [S \to a \cdot S] & [S \to \cdot aS] & [S \to \cdot a] & [S \to a \cdot] \\
LR(0)(aS): & [S \to aS \cdot] & & &
\end{array}$$

Note: G is unambiguous

Outline

The goto Function I

Observation: if $G \in LR(0)$, then $LR(0)(\gamma)$ yields deterministic shift/reduce decision for $\mathrm{NBA}(G)$ 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 9.14 ($LR(0)$ goto function)

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

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

Example 9.15 (cf. Example 9.11)

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

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

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

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

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

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

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

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

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

$I_9 := \emptyset$

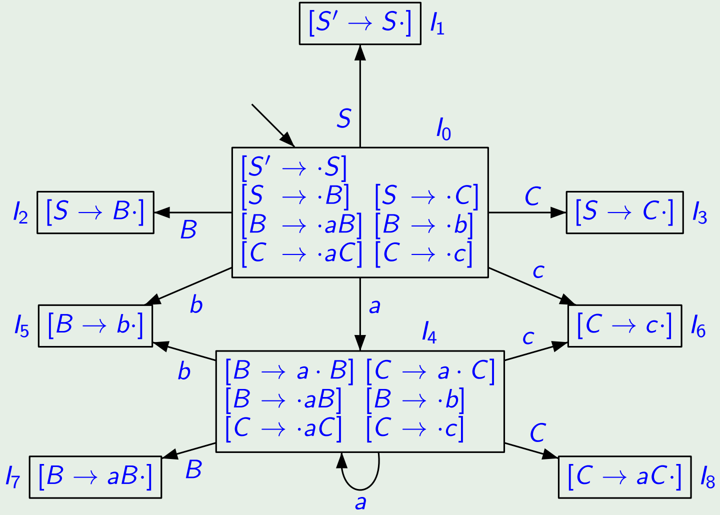
goto	S	B	C	a	b	c
I_0	I_1	I_2	I_3	I_4	I_5	I_6
I_1						
I_2						
I_3						
I_4		I_7	I_8	I_4	I_5	I_6
I_5						
I_6						
I_7						
I_8						
I_9						

(empty $= I_9$)

The goto Function III

Example 9.15 (continued)

Representation of goto funtion as finite automaton:



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 9.16 ($LR(0)$ 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 9.17

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