

Compiler Construction

Lecture 9: Syntactic Analysis IV

(*LL*(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/cc10/`

Winter semester 2010/11

- 1 Repetition: $LL(1)$ Grammars
- 2 Deterministic Top-Down Parsing
- 3 Transformation to $LL(1)$
- 4 The Complexity of $LL(1)$ Parsing

Lemma (Characterization of $LL(k)$)

$G \in LL(k)$ iff for all leftmost derivations of the form

$$S \Rightarrow_l^* wA\alpha \begin{cases} \Rightarrow_l w\beta\alpha \\ \Rightarrow_l w\gamma\alpha \end{cases}$$

such that $\beta \neq \gamma$, it follows that $\text{first}_k(\beta\alpha) \cap \text{first}_k(\gamma\alpha) = \emptyset$.

Proof.

omitted □

Remarks:

- If $G \in LL(k)$, then the A -production is determined by the lookahead sets $\text{first}_k(\beta\alpha)$ (for every $A \rightarrow \beta \in P$).
- Problem: still infinitely many rightmost contexts α to be considered (if β/γ “too short”, i.e., $\text{first}_k(\beta\alpha) \neq \text{first}_k(\gamma\alpha)$).

Definition (Lookahead set)

Given $\pi = A \rightarrow \beta \in P$,

$$\text{la}(\pi) := \text{fi}(\beta \cdot \text{fo}(A)) \subseteq \Sigma_\varepsilon$$

is called the **lookahead set** of π (where $\text{fi}(\Gamma) := \bigcup_{\gamma \in \Gamma} \text{fi}(\gamma)$).

Corollary

❶ For all $a \in \Sigma$,

$$a \in \text{la}(A \rightarrow \beta) \text{ iff } a \in \text{fi}(\beta) \text{ or } (\beta \Rightarrow^* \varepsilon \text{ and } a \in \text{fo}(A))$$

❷ $\varepsilon \in \text{la}(A \rightarrow \beta)$ iff $\beta \Rightarrow^* \varepsilon$ and $\varepsilon \in \text{fo}(A)$

Characterization of $LL(1)$

Theorem (Characterization of $LL(1)$)

$G \in LL(1)$ iff for all pairs of rules $A \rightarrow \beta \mid \gamma \in P$ (where $\beta \neq \gamma$):

$$\text{la}(A \rightarrow \beta) \cap \text{la}(A \rightarrow \gamma) = \emptyset.$$

Proof.

on the board



Remark: the above theorem generally does not hold if $k > 1$
(cf. exercises)

Computing Lookahead Sets II

Corollary

- ① $A \rightarrow a\alpha \in P \implies a \in \text{fi}(A)$
- ② $A \rightarrow B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
- ③ $A \rightarrow \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
- ④ $\text{fi}(\varepsilon) = \{\varepsilon\}$
- ⑤ $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
- ⑥ $A \rightarrow \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Computing Lookahead Sets II

Corollary

- ❶ $A \rightarrow a\alpha \in P \implies a \in \text{fi}(A)$
- ❷ $A \rightarrow B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
- ❸ $A \rightarrow \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
- ❹ $\text{fi}(\varepsilon) = \{\varepsilon\}$
- ❺ $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
- ❻ $A \rightarrow \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example

Grammar for
arithmetic
expressions

(cf. Example 7.3):

$$\begin{aligned} G_{AE} : \quad & E \rightarrow E+T \mid T \\ & T \rightarrow T*F \mid F \\ & F \rightarrow (E) \mid a \mid b \end{aligned}$$

Computing Lookahead Sets II

Corollary

- ① $A \rightarrow a\alpha \in P \implies a \in \text{fi}(A)$
- ② $A \rightarrow B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
- ③ $A \rightarrow \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
- ④ $\text{fi}(\varepsilon) = \{\varepsilon\}$
- ⑤ $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
- ⑥ $A \rightarrow \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example

Grammar for
arithmetic
expressions

(cf. Example 7.3):

$$\begin{aligned} G_{AE} : \quad & E \rightarrow E+T \mid T \\ & T \rightarrow T*F \mid F \\ & F \rightarrow (E) \mid a \mid b \end{aligned}$$

$$\bullet F \rightarrow a \in P \implies a \in \text{fi}(F)$$

Computing Lookahead Sets II

Corollary

- ① $A \rightarrow a\alpha \in P \implies a \in \text{fi}(A)$
- ② $A \rightarrow B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
- ③ $A \rightarrow \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
- ④ $\text{fi}(\varepsilon) = \{\varepsilon\}$
- ⑤ $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
- ⑥ $A \rightarrow \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example

Grammar for
arithmetic
expressions

(cf. Example 7.3):

$$\begin{array}{l} G_{AE} : \quad E \rightarrow E+T \mid T \\ \quad \quad T \rightarrow T*F \mid F \\ \quad \quad F \rightarrow (E) \mid a \mid b \end{array}$$

- $F \rightarrow a \in P \implies a \in \text{fi}(F)$
- $T \rightarrow F \in P, a \in \text{fi}(F) \implies a \in \text{fi}(T)$

Computing Lookahead Sets II

Corollary

- ① $A \rightarrow a\alpha \in P \implies a \in \text{fi}(A)$
- ② $A \rightarrow B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
- ③ $A \rightarrow \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
- ④ $\text{fi}(\varepsilon) = \{\varepsilon\}$
- ⑤ $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
- ⑥ $A \rightarrow \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example

Grammar for
arithmetic
expressions

(cf. Example 7.3):

$$\begin{array}{l} G_{AE} : \quad E \rightarrow E+T \mid T \\ \quad \quad T \rightarrow T * F \mid F \\ \quad \quad F \rightarrow (E) \mid a \mid b \end{array}$$

- $F \rightarrow a \in P \implies a \in \text{fi}(F)$
- $T \rightarrow F \in P, a \in \text{fi}(F) \implies a \in \text{fi}(T)$
- $a \in \text{fi}(T) \implies \text{la}(T \rightarrow T * F) = \text{fi}(T * F \cdot \text{fo}(T)) \ni a$

Computing Lookahead Sets II

Corollary

- ① $A \rightarrow a\alpha \in P \implies a \in \text{fi}(A)$
- ② $A \rightarrow B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
- ③ $A \rightarrow \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
- ④ $\text{fi}(\varepsilon) = \{\varepsilon\}$
- ⑤ $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
- ⑥ $A \rightarrow \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example

Grammar for
arithmetic
expressions

(cf. Example 7.3):

$$\begin{array}{l} G_{AE} : \quad E \rightarrow E+T \mid T \\ \quad \quad T \rightarrow T * F \mid F \\ \quad \quad F \rightarrow (E) \mid a \mid b \end{array}$$

- $F \rightarrow a \in P \implies a \in \text{fi}(F)$
- $T \rightarrow F \in P, a \in \text{fi}(F) \implies a \in \text{fi}(T)$
- $a \in \text{fi}(T)$
 $\implies \text{la}(T \rightarrow T * F) = \text{fi}(T * F \cdot \text{fo}(T)) \ni a$
- $a \in \text{fi}(F)$
 $\implies \text{la}(T \rightarrow F) = \text{fi}(F \cdot \text{fo}(T)) \ni a$

Computing Lookahead Sets II

Corollary

- ① $A \rightarrow a\alpha \in P \implies a \in \text{fi}(A)$
- ② $A \rightarrow B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
- ③ $A \rightarrow \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
- ④ $\text{fi}(\varepsilon) = \{\varepsilon\}$
- ⑤ $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
- ⑥ $A \rightarrow \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example

Grammar for
arithmetic
expressions

(cf. Example 7.3):

$$\begin{array}{l} G_{AE} : \quad E \rightarrow E+T \mid T \\ \quad \quad T \rightarrow T * F \mid F \\ \quad \quad F \rightarrow (E) \mid a \mid b \end{array}$$

- $F \rightarrow a \in P \implies a \in \text{fi}(F)$
- $T \rightarrow F \in P, a \in \text{fi}(F) \implies a \in \text{fi}(T)$
- $a \in \text{fi}(T) \implies \text{la}(T \rightarrow T * F) = \text{fi}(T * F \cdot \text{fo}(T)) \ni a$
- $a \in \text{fi}(F) \implies \text{la}(T \rightarrow F) = \text{fi}(F \cdot \text{fo}(T)) \ni a$
- $\implies a \in \text{la}(T \rightarrow T * F) \cap \text{la}(T \rightarrow F) \neq \emptyset$

Computing Lookahead Sets II

Corollary

- ① $A \rightarrow a\alpha \in P \implies a \in \text{fi}(A)$
- ② $A \rightarrow B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
- ③ $A \rightarrow \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
- ④ $\text{fi}(\varepsilon) = \{\varepsilon\}$
- ⑤ $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
- ⑥ $A \rightarrow \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example

Grammar for
arithmetic
expressions

(cf. Example 7.3):

$$\begin{array}{l} G_{AE} : \quad E \rightarrow E+T \mid T \\ \quad \quad T \rightarrow T * F \mid F \\ \quad \quad F \rightarrow (E) \mid a \mid b \end{array}$$

- $F \rightarrow a \in P \implies a \in \text{fi}(F)$
- $T \rightarrow F \in P, a \in \text{fi}(F) \implies a \in \text{fi}(T)$
- $a \in \text{fi}(T) \implies \text{la}(T \rightarrow T * F) = \text{fi}(T * F \cdot \text{fo}(T)) \ni a$
- $a \in \text{fi}(F) \implies \text{la}(T \rightarrow F) = \text{fi}(F \cdot \text{fo}(T)) \ni a$
- $\implies a \in \text{la}(T \rightarrow T * F) \cap \text{la}(T \rightarrow F) \neq \emptyset$
- $\implies G_{AE} \notin LL(1)$

Fixing the Problem

(general methods later)

Example 9.1 (continuing Example 8.12)

Restructuring (such that $L(G'_{AE}) = L(G_{AE})$):

$$\begin{aligned} G'_{AE} : \quad & E \rightarrow TE' \\ & E' \rightarrow +TE' \mid \varepsilon \\ & T \rightarrow FT' \\ & T' \rightarrow *FT' \mid \varepsilon \\ & F \rightarrow (E) \mid \mathbf{a} \mid \mathbf{b} \end{aligned}$$

Fixing the Problem

(general methods later)

Example 9.1 (continuing Example 8.12)

Restructuring (such that $L(G'_{AE}) = L(G_{AE})$):

$$\begin{aligned}G'_{AE} : \quad & E \rightarrow TE' \\ & E' \rightarrow +TE' \mid \varepsilon \\ & T \rightarrow FT' \\ & T' \rightarrow *FT' \mid \varepsilon \\ & F \rightarrow (E) \mid \textcolor{blue}{a} \mid \textcolor{blue}{b}\end{aligned}$$

$A \in N$	$\text{fi}(A)$
E	$\{(\textcolor{blue}{,}, \textcolor{blue}{a}, \textcolor{blue}{b})\}$
E'	$\{+, \varepsilon\}$
T	$\{(\textcolor{blue}{,}, \textcolor{blue}{a}, \textcolor{blue}{b})\}$
T'	$\{*, \varepsilon\}$
F	$\{(\textcolor{blue}{,}, \textcolor{blue}{a}, \textcolor{blue}{b})\}$

Fixing the Problem

(general methods later)

Example 9.1 (continuing Example 8.12)

Restructuring (such that $L(G'_{AE}) = L(G_{AE})$):

$$\begin{aligned}G'_{AE} : \quad & E \rightarrow TE' \\ & E' \rightarrow +TE' \mid \varepsilon \\ & T \rightarrow FT' \\ & T' \rightarrow *FT' \mid \varepsilon \\ & F \rightarrow (E) \mid \textcolor{blue}{a} \mid \textcolor{blue}{b}\end{aligned}$$

$A \in N$	$\text{fi}(A)$	$\text{fo}(A)$
E	$\{(\textcolor{blue}{a}, \textcolor{blue}{b})\}$	$\{\varepsilon,)\}$
E'	$\{+, \varepsilon\}$	$\{\varepsilon,)\}$
T	$\{(\textcolor{blue}{a}, \textcolor{blue}{b})\}$	$\{+, \varepsilon,)\}$
T'	$\{*, \varepsilon\}$	$\{+, \varepsilon,)\}$
F	$\{(\textcolor{blue}{a}, \textcolor{blue}{b})\}$	$\{*, +, \varepsilon,)\}$

Fixing the Problem

(general methods later)

Example 9.1 (continuing Example 8.12)

Restructuring (such that $L(G'_{AE}) = L(G_{AE})$):

$$\begin{aligned}
 G'_{AE} : \quad & E \rightarrow TE' \\
 & E' \rightarrow +TE' \mid \varepsilon \\
 & T \rightarrow FT' \\
 & T' \rightarrow *FT' \mid \varepsilon \\
 & F \rightarrow (E) \mid a \mid b
 \end{aligned}$$

$A \in N$	$\text{fi}(A)$	$\text{fo}(A)$
E	$\{ (, a, b \}$	$\{ \varepsilon,) \}$
E'	$\{ +, \varepsilon \}$	$\{ \varepsilon,) \}$
T	$\{ (, a, b \}$	$\{ +, \varepsilon,) \}$
T'	$\{ *, \varepsilon \}$	$\{ +, \varepsilon,) \}$
F	$\{ (, a, b \}$	$\{ *, +, \varepsilon,) \}$

$A \rightarrow \beta \in P$	$\text{la}(A \rightarrow \beta) = \text{fi}(\beta \cdot \text{fo}(A))$
$E \rightarrow TE'$	$\{ (, a, b \}$
$E' \rightarrow +TE'$	$\{ + \}$
$E' \rightarrow \varepsilon$	$\{ \varepsilon,) \}$
$T \rightarrow FT'$	$\{ (, a, b \}$
$T' \rightarrow *FT'$	$\{ * \}$
$T' \rightarrow \varepsilon$	$\{ +, \varepsilon,) \}$
$F \rightarrow (E)$	$\{ (\}$
$F \rightarrow a$	$\{ a \}$
$F \rightarrow b$	$\{ b \}$

Fixing the Problem

(general methods later)

Example 9.1 (continuing Example 8.12)

Restructuring (such that $L(G'_{AE}) = L(G_{AE})$):

$$\begin{aligned}
 G'_{AE} : \quad & E \rightarrow TE' \\
 & E' \rightarrow +TE' \mid \varepsilon \\
 & T \rightarrow FT' \\
 & T' \rightarrow *FT' \mid \varepsilon \\
 & F \rightarrow (E) \mid a \mid b
 \end{aligned}$$

$A \in N$	$\text{fi}(A)$	$\text{fo}(A)$
E	$\{ (, a, b \}$	$\{ \varepsilon,) \}$
E'	$\{ +, \varepsilon \}$	$\{ \varepsilon,) \}$
T	$\{ (, a, b \}$	$\{ +, \varepsilon,) \}$
T'	$\{ *, \varepsilon \}$	$\{ +, \varepsilon,) \}$
F	$\{ (, a, b \}$	$\{ *, +, \varepsilon,) \}$

$A \rightarrow \beta \in P$	$\text{la}(A \rightarrow \beta) = \text{fi}(\beta \cdot \text{fo}(A))$
$E \rightarrow TE'$	$\{ (, a, b \}$
$E' \rightarrow +TE'$	$\{ + \}$
$E' \rightarrow \varepsilon$	$\{ \varepsilon,) \}$
$T \rightarrow FT'$	$\{ (, a, b \}$
$T' \rightarrow *FT'$	$\{ * \}$
$T' \rightarrow \varepsilon$	$\{ +, \varepsilon,) \}$
$F \rightarrow (E)$	$\{ (\}$
$F \rightarrow a$	$\{ a \}$
$F \rightarrow b$	$\{ b \}$

$\Rightarrow G'_{AE} \in LL(1)$

- 1 Repetition: $LL(1)$ Grammars
- 2 Deterministic Top-Down Parsing
- 3 Transformation to $LL(1)$
- 4 The Complexity of $LL(1)$ Parsing

Deterministic Top-Down Parsing

Approach: given $G \in CFG_{\Sigma}$,

- 1 Verify that $G \in LL(1)$ by computing the lookahead sets and checking alternatives for disjointness

Deterministic Top-Down Parsing

Approach: given $G \in CFG_{\Sigma}$,

- 1 Verify that $G \in LL(1)$ by computing the lookahead sets and checking alternatives for disjointness
- 2 Start with nondeterministic top-down parsing automaton $NTA(G)$

Deterministic Top-Down Parsing

Approach: given $G \in CFG_{\Sigma}$,

- ① Verify that $G \in LL(1)$ by computing the lookahead sets and checking alternatives for disjointness
 - ② Start with nondeterministic top-down parsing automaton $NTA(G)$
 - ③ Use **1-symbol lookahead** to control the choice of expanding productions:
 - $(aw, A\alpha, z) \vdash (aw, \beta\alpha, zi)$
if $\pi_i = A \rightarrow \beta$ and $a \in la(\pi_i)$
 - $(\varepsilon, A\alpha, z) \vdash (\varepsilon, \beta\alpha, zi)$
if $\pi_i = A \rightarrow \beta$ and $\varepsilon \in la(\pi_i)$
 - [matching steps as before: $(aw, a\alpha, z) \vdash (w, \alpha, z)$]
- $\implies$ **deterministic top-down parsing automaton $DTA(G)$**

Deterministic Top-Down Parsing

Approach: given $G \in CFG_{\Sigma}$,

- ① Verify that $G \in LL(1)$ by computing the lookahead sets and checking alternatives for disjointness
 - ② Start with nondeterministic top-down parsing automaton $NTA(G)$
 - ③ Use **1-symbol lookahead** to control the choice of expanding productions:
 - $(aw, A\alpha, z) \vdash (aw, \beta\alpha, zi)$
if $\pi_i = A \rightarrow \beta$ and $a \in la(\pi_i)$
 - $(\varepsilon, A\alpha, z) \vdash (\varepsilon, \beta\alpha, zi)$
if $\pi_i = A \rightarrow \beta$ and $\varepsilon \in la(\pi_i)$
 - [matching steps as before: $(aw, a\alpha, z) \vdash (w, \alpha, z)$]
- $\implies$ **deterministic top-down parsing automaton $DTA(G)$**

Remarks:

- $DTA(G)$ is actually **not a pushdown automaton** (a is read but not consumed). But: can be simulated using the finite control.
- Advantage of using lookahead is **twofold**:
 - Removal of nondeterminism
 - Earlier detection of syntax errors
(in configurations $(aw, A\alpha, z)$ where $a \notin \bigcup_{A \rightarrow \beta \in P} la(A \rightarrow \beta)$)

The Deterministic Top-Down Automaton I

Definition 9.2 (Deterministic top-down parsing automaton)

Let $G = \langle N, \Sigma, P, S \rangle \in LL(1)$. The **deterministic top-down parsing automaton** of G , $DTA(G)$, is defined by the following components.

- **Input alphabet** Σ , **pushdown alphabet** X , **output alphabet** $[p]$
- **Configurations** $\Sigma^* \times X^* \times [p]^*$, **initial configuration** (w, S, ε) , **final configurations** $\{\varepsilon\} \times \{\varepsilon\} \times [p]^*$ (as $NTA(G)$)
- **Action function**
$$\text{act} : \Sigma_\varepsilon \times X_\varepsilon \rightarrow \{(\alpha, i) \mid \pi_i = A \rightarrow \alpha\} \cup \{\text{pop}, \text{accept}, \text{error}\}$$

with $\text{act}(x, A) := (\alpha, i)$ if $\pi_i = A \rightarrow \alpha$ and $x \in \text{la}(\pi_i)$
$$\begin{aligned}\text{act}(a, a) &:= \text{pop} \\ \text{act}(\varepsilon, \varepsilon) &:= \text{accept} \\ \text{act}(x, y) &:= \text{error} \quad \text{otherwise}\end{aligned}$$
- **Transitions** for $x \in \Sigma_\varepsilon$, $w \in \Sigma^*$, $Y \in X$, $\beta \in X^*$, and $z \in [p]^*$:
$$(xw, Y\beta, z) \vdash \begin{cases} (xw, \alpha\beta, zi) & \text{if } \text{act}(x, Y) = (\alpha, i) \\ (w, \beta, z) & \text{if } \text{act}(x, Y) = \text{pop} \end{cases}$$

The Deterministic Top-Down Automaton II

Example 9.3 (cf. Example 9.1)

$$\begin{aligned} G'_{AE} : \quad & E \rightarrow TE' && (1) \\ & E' \rightarrow +TE' \mid \varepsilon && (2, 3) \\ & T \rightarrow FT' && (4) \\ & T' \rightarrow *FT' \mid \varepsilon && (5, 6) \\ & F \rightarrow (E) \mid a \mid b && (7, 8, 9) \end{aligned}$$

$A \rightarrow \beta \in P$	$\text{la}(A \rightarrow \beta)$
$E \rightarrow TE'$	$\{ (, a, b \}$
$E' \rightarrow +TE'$	$\{ + \}$
$E' \rightarrow \varepsilon$	$\{ \varepsilon,) \}$
$T \rightarrow FT'$	$\{ (, a, b \}$
$T' \rightarrow *FT'$	$\{ * \}$
$T' \rightarrow \varepsilon$	$\{ +, \varepsilon,) \}$
$F \rightarrow (E)$	$\{ (\}$
$F \rightarrow a$	$\{ a \}$
$F \rightarrow b$	$\{ b \}$

The Deterministic Top-Down Automaton II

Example 9.3 (cf. Example 9.1)

$$\begin{aligned}
 G'_{AE} : \quad & E \rightarrow TE' & (1) \\
 & E' \rightarrow +TE' \mid \varepsilon & (2, 3) \\
 & T \rightarrow FT' & (4) \\
 & T' \rightarrow *FT' \mid \varepsilon & (5, 6) \\
 & F \rightarrow (E) \mid a \mid b & (7, 8, 9)
 \end{aligned}$$

$A \rightarrow \beta \in P$	$\text{la}(A \rightarrow \beta)$
$E \rightarrow TE'$	$\{ (, a, b) \}$
$E' \rightarrow +TE'$	$\{ + \}$
$E' \rightarrow \varepsilon$	$\{ \varepsilon,) \}$
$T \rightarrow FT'$	$\{ (, a, b) \}$
$T' \rightarrow *FT'$	$\{ * \}$
$T' \rightarrow \varepsilon$	$\{ +, \varepsilon,) \}$
$F \rightarrow (E)$	$\{ (\}$
$F \rightarrow a$	$\{ a \}$
$F \rightarrow b$	$\{ b \}$

$\text{act} : \Sigma_\varepsilon \times X_\varepsilon \rightarrow \{(\alpha, i) \mid \pi_i = A \rightarrow \alpha\} \cup \{\text{pop}, \text{accept}, \text{error}\}$ (empty = error)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
$($	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
$)$		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
$*$				$(*FT', 5)$						pop		
$+$		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
$($	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
$)$		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
$*$				$(*FT', 5)$						pop		
$+$		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
$($	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
$)$		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
$*$				$(*FT', 5)$						pop		
$+$		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$((a)*b, E, \varepsilon)$

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
$($	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
$)$		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
$*$				$(*FT', 5)$						pop		
$+$		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$$\begin{array}{l}
 ((a)*b, E, \varepsilon) \\
 \vdash ((a)*b, TE', 1)
 \end{array}$$

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	(	)	*	+	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$((a)*b, E, \varepsilon)$
 $\vdash ((a)*b, TE', 1)$
 $\vdash ((a)*b, FT'E', 14)$

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

	$((a)*b, E$	$, \varepsilon$	)
$\vdash$	$((a)*b, TE'$	$, 1$	)
$\vdash$	$((a)*b, FT'E'$	$, 14$	)
$\vdash$	$((a)*b, (E)T'E'$	$, 147$	)

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

	$(a)*b, E$	, ε	)
$\vdash$	$(a)*b, TE'$	, 1	)
$\vdash$	$(a)*b, FT'E'$	, 14	)
$\vdash$	$(a)*b, (E)T'E'$	, 147	)
$\vdash$	$(a)*b, E)T'E'$	, 147	)

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

	$((a)*b, E$	$, \varepsilon$	)
$\vdash$	$((a)*b, TE'$	$, 1$	)
$\vdash$	$((a)*b, FT'E'$	$, 14$	)
$\vdash$	$((a)*b, (E)T'E'$	$, 147$	)
$\vdash$	$((a)*b, E)T'E'$	$, 147$	)
$\vdash$	$((a)*b, TE')T'E'$	$, 1471$	)

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		(a, 8)	pop						
b	$(TE', 1)$		$(FT', 4)$		(b, 9)		pop					
(	$(TE', 1)$		$(FT', 4)$		((E), 7)			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				(*FT', 5)						pop		
+		(+TE', 2)		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of **(a)*b**:

$((a)*b, E, \varepsilon)$
 $\vdash ((a)*b, TE', 1)$
 $\vdash ((a)*b, FT'E', 14)$
 $\vdash ((a)*b, (E)T'E', 147)$
 $\vdash (a)*b, E)T'E', 147)$
 $\vdash (a)*b, TE')T'E', 1471)$
 $\vdash (a)*b, FT'E')T'E', 14714)$

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	(	)	*	+	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$((a)*b, E, \varepsilon)$
 $\vdash ((a)*b, TE', 1)$
 $\vdash ((a)*b, FT'E', 14)$
 $\vdash ((a)*b, (E)T'E', 147)$
 $\vdash (a)*b, E)T'E', 147)$
 $\vdash (a)*b, TE')T'E', 1471)$
 $\vdash (a)*b, FT'E')T'E', 14714)$
 $\vdash (a)*b, aT'E')T'E', 147148)$

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$((a)*b, E, \varepsilon)$
 $\vdash ((a)*b, TE', 1)$
 $\vdash ((a)*b, FT'E', 14)$
 $\vdash ((a)*b, (E)T'E', 147)$
 $\vdash (a)*b, E)T'E', 147)$
 $\vdash (a)*b, TE')T'E', 1471)$
 $\vdash (a)*b, FT'E')T'E', 14714)$
 $\vdash (a)*b, aT'E')T'E', 147148)$
 $\vdash ()*b, T'E')T'E', 147148)$

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$((a)*b, E, \varepsilon) \vdash (b, E')T'E', 1471486)$
 $\vdash (a)*b, TE', 1)$
 $\vdash (a)*b, FT'E', 14)$
 $\vdash (a)*b, (E)T'E', 147)$
 $\vdash (a)*b, E)T'E', 147)$
 $\vdash (a)*b, TE')T'E', 1471)$
 $\vdash (a)*b, FT'E')T'E', 14714)$
 $\vdash (a)*b, aT'E')T'E', 147148)$
 $\vdash (b)*b, T'E')T'E', 147148)$

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$(a)*b, E$	$, \varepsilon$	)	$\vdash (b, E')T'E', 1471486$	)
$\vdash (a)*b, TE'$	$, 1$	)	$\vdash (b,)T'E', 14714863$	)
$\vdash (a)*b, FT'E'$	$, 14$	)		
$\vdash (a)*b, (E)T'E'$	$, 147$	)		
$\vdash (a)*b, E)T'E'$	$, 147$	)		
$\vdash (a)*b, TE')T'E'$	$, 1471$	)		
$\vdash (a)*b, FT'E')T'E'$	$, 14714$	)		
$\vdash (a)*b, aT'E')T'E'$	$, 147148$	)		
$\vdash ()*b, T'E')T'E'$	$, 147148$	)		

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$(a)*b, E$	$, \varepsilon$	)	$\vdash (b, E')T'E', 1471486$	)
$\vdash (a)*b, TE'$	$, 1$	)	$\vdash (b,)T'E', 14714863$	)
$\vdash (a)*b, FT'E'$	$, 14$	)	$\vdash (*b, T'E', 14714863$	)
$\vdash (a)*b, (E)T'E'$	$, 147$	)		
$\vdash (a)*b, E)T'E'$	$, 147$	)		
$\vdash (a)*b, TE')T'E'$	$, 1471$	)		
$\vdash (a)*b, FT'E')T'E', 14714$		)		
$\vdash (a)*b, aT'E')T'E', 147148$		)		
$\vdash ()*b, T'E')T'E', 147148$		)		

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(\text{a}, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(\text{b}, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$(a)*b, E, \varepsilon$	$\vdash () * b, E') T' E', 1471486$
$\vdash (a)*b, TE', 1$	$\vdash () * b,) T' E', 14714863$
$\vdash (a)*b, FT' E', 14$	$\vdash (* b, T' E', 14714863$
$\vdash (a)*b, (E) T' E', 147$	$\vdash (* b, * FT' E', 147148635$
$\vdash (a)*b, E) T' E', 147$	
$\vdash (a)*b, TE') T' E', 1471$	
$\vdash (a)*b, FT' E') T' E', 14714$	
$\vdash (a)*b, a T' E') T' E', 147148$	
$\vdash () * b, T' E') T' E', 147148$	

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$(a)*b, E, \varepsilon$	$\vdash () * b, E') T' E', 1471486$
$\vdash (a)*b, TE', 1$	$\vdash () * b,) T' E', 14714863$
$\vdash (a)*b, FT' E', 14$	$\vdash (* b, T' E', 14714863$
$\vdash (a)*b, (E) T' E', 147$	$\vdash (* b, * FT' E', 147148635$
$\vdash (a)*b, E) T' E', 147$	$\vdash (b, FT' E', 147148635$
$\vdash (a)*b, TE') T' E', 1471$	
$\vdash (a)*b, FT' E') T' E', 14714$	
$\vdash (a)*b, a T' E') T' E', 147148$	
$\vdash () * b, T' E') T' E', 147148$	

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$(a)*b, E, \varepsilon$	$\vdash () * b, E') T' E', 1471486$
$\vdash (a)*b, TE', 1$	$\vdash () * b,) T' E', 14714863$
$\vdash (a)*b, FT' E', 14$	$\vdash (* b, T' E', 14714863$
$\vdash (a)*b, (E) T' E', 147$	$\vdash (* b, * FT' E', 147148635$
$\vdash (a)*b, E) T' E', 147$	$\vdash (b, FT' E', 147148635$
$\vdash (a)*b, TE') T' E', 1471$	$\vdash (b, b T' E', 1471486359$
$\vdash (a)*b, FT' E') T' E', 14714$	
$\vdash (a)*b, a T' E') T' E', 147148$	
$\vdash () * b, T' E') T' E', 147148$	

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$(a)*b, E, \varepsilon$	$\vdash () * b, E') T' E', 1471486$
$\vdash (a)*b, TE', 1$	$\vdash () * b,) T' E', 14714863$
$\vdash (a)*b, FT' E', 14$	$\vdash (* b, T' E', 14714863$
$\vdash (a)*b, (E) T' E', 147$	$\vdash (* b, * FT' E', 147148635$
$\vdash (a)*b, E) T' E', 147$	$\vdash (b, FT' E', 147148635$
$\vdash (a)*b, TE') T' E', 1471$	$\vdash (b, b T' E', 1471486359$
$\vdash (a)*b, FT' E') T' E', 14714$	$\vdash (\varepsilon, T' E', 1471486359$
$\vdash (a)*b, a T' E') T' E', 147148$	
$\vdash () * b, T' E') T' E', 147148$	

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(\text{a}, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(\text{b}, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$((a)*b, E, \varepsilon)$	$\vdash ((a)*b, E')T'E', 1471486)$
$\vdash ((a)*b, TE', 1)$	$\vdash ((a)*b,)T'E', 14714863)$
$\vdash ((a)*b, FT'E', 14)$	$\vdash (*b, T'E', 14714863)$
$\vdash ((a)*b, (E)T'E', 147)$	$\vdash (*b, *FT'E', 147148635)$
$\vdash (a)*b, E)T'E', 147)$	$\vdash (b, FT'E', 147148635)$
$\vdash (a)*b, TE')T'E', 1471)$	$\vdash (b, bT'E', 1471486359)$
$\vdash (a)*b, FT'E')T'E', 14714)$	$\vdash (\varepsilon, T'E', 1471486359)$
$\vdash (a)*b, aT'E')T'E', 147148)$	$\vdash (\varepsilon, E', 14714863596)$
$\vdash ()*b, T'E')T'E', 147148)$	

The Deterministic Top-Down Automaton III

Example 9.3 (continued)

act	E	E'	T	T'	F	a	b	$($	$)$	$*$	$+$	ε
a	$(TE', 1)$		$(FT', 4)$		$(a, 8)$	pop						
b	$(TE', 1)$		$(FT', 4)$		$(b, 9)$		pop					
(	$(TE', 1)$		$(FT', 4)$		$((E), 7)$			pop				
)		$(\varepsilon, 3)$		$(\varepsilon, 6)$					pop			
*				$(*FT', 5)$						pop		
+		$(+TE', 2)$		$(\varepsilon, 6)$							pop	
ε		$(\varepsilon, 3)$		$(\varepsilon, 6)$								accept

Leftmost analysis of $(a)*b$:

$(a)*b, E, \varepsilon$	$\vdash () * b, E') T' E', 1471486$
$\vdash (a)*b, TE', 1$	$\vdash () * b,) T' E', 14714863$
$\vdash (a)*b, FT' E', 14$	$\vdash (* b, T' E', 14714863$
$\vdash (a)*b, (E) T' E', 147$	$\vdash (* b, * FT' E', 147148635$
$\vdash (a)*b, E) T' E', 147$	$\vdash (b, FT' E', 147148635$
$\vdash (a)*b, TE') T' E', 1471$	$\vdash (b, b T' E', 1471486359$
$\vdash (a)*b, FT' E') T' E', 14714$	$\vdash (\varepsilon, T' E', 1471486359$
$\vdash (a)*b, a T' E') T' E', 147148$	$\vdash (\varepsilon, E', 14714863596$
$\vdash () * b, T' E') T' E', 147148$	$\vdash (\varepsilon, \varepsilon, 147148635963$

- 1 Repetition: $LL(1)$ Grammars
- 2 Deterministic Top-Down Parsing
- 3 Transformation to $LL(1)$
- 4 The Complexity of $LL(1)$ Parsing

Transformation to $LL(1)$

Assume that $G = \langle N, \Sigma, P, S \rangle \in CFG_{\Sigma} \setminus LL(1)$

(i.e., there exist $A \rightarrow \beta \mid \gamma \in P$ such that $la(A \rightarrow \beta) \cap la(A \rightarrow \gamma) \neq \emptyset$)

Transformation to $LL(1)$

Assume that $G = \langle N, \Sigma, P, S \rangle \in CFG_{\Sigma} \setminus LL(1)$

(i.e., there exist $A \rightarrow \beta \mid \gamma \in P$ such that $la(A \rightarrow \beta) \cap la(A \rightarrow \gamma) \neq \emptyset$)

Two **heuristics** for transforming G into $G' \in LL(1)$:

- 1 Removal of left recursion
- 2 Left factorization

(used in parser-generating systems such as ANTLR)

Transformation to $LL(1)$

Assume that $G = \langle N, \Sigma, P, S \rangle \in CFG_{\Sigma} \setminus LL(1)$

(i.e., there exist $A \rightarrow \beta \mid \gamma \in P$ such that $la(A \rightarrow \beta) \cap la(A \rightarrow \gamma) \neq \emptyset$)

Two **heuristics** for transforming G into $G' \in LL(1)$:

- 1 Removal of left recursion
- 2 Left factorization

(used in parser-generating systems such as ANTLR)

Remarks:

- Transformations generally **preserve the semantics** (= generated language) of CFGs but **not the syntactic structure** of words (different syntax trees).
- Transformations **cannot always yield an $LL(1)$ grammar** (since not every context-free language is generated by an LL grammar; details later).

Definition 9.4 (Left recursion)

A grammar $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ is called **left recursive** if there exist $A \in N$ and $\alpha \in X^*$ such that $A \Rightarrow^+ A\alpha$.

Definition 9.4 (Left recursion)

A grammar $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ is called **left recursive** if there exist $A \in N$ and $\alpha \in X^*$ such that $A \Rightarrow^+ A\alpha$.

Corollary 9.5

If $G \in CFG_\Sigma$ is left recursive with $A \Rightarrow^+ A\alpha$, then there exists $\beta \in X^$ such that $A \Rightarrow_l^+ A\beta$.*

Definition 9.4 (Left recursion)

A grammar $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ is called **left recursive** if there exist $A \in N$ and $\alpha \in X^*$ such that $A \Rightarrow^+ A\alpha$.

Corollary 9.5

If $G \in CFG_\Sigma$ is left recursive with $A \Rightarrow^+ A\alpha$, then there exists $\beta \in X^$ such that $A \Rightarrow_l^+ A\beta$.*

Example 9.6

The grammar (cf. Example 7.3)

$$\begin{aligned} G_{AE} : \quad E &\rightarrow E+T \mid T \\ T &\rightarrow T*F \mid F \\ F &\rightarrow (E) \mid a \mid b \end{aligned}$$

is left recursive, and in Example 8.12 it was shown that $G_{AE} \notin LL(1)$

Lemma 9.7

If $G \in CFG_\Sigma$ is left recursive, then $G \notin \bigcup_{k \in \mathbb{N}} LL(k)$.

Lemma 9.7

If $G \in CFG_\Sigma$ is left recursive, then $G \notin \bigcup_{k \in \mathbb{N}} LL(k)$.

Proof.

(for $k = 1$) Assume that $G \in LL(1)$ is left recursive with $A \Rightarrow_l^+ A\beta$. Together with the reducedness of G this implies that $S \Rightarrow_l^* vA\alpha \Rightarrow_l^+ vA\beta\alpha \Rightarrow_l^+ vw$ for some $v, w \in \Sigma^*$ and $\alpha \in X^*$.

Lemma 9.7

If $G \in CFG_\Sigma$ is left recursive, then $G \notin \bigcup_{k \in \mathbb{N}} LL(k)$.

Proof.

(for $k = 1$) Assume that $G \in LL(1)$ is left recursive with $A \Rightarrow_l^+ A\beta$. Together with the reducedness of G this implies that $S \Rightarrow_l^* vA\alpha \Rightarrow_l^+ vA\beta\alpha \Rightarrow_l^+ vw$ for some $v, w \in \Sigma^*$ and $\alpha \in X^*$. The corresponding computation of $DTA(G)$ (Def. 9.2) starts with $(vw, S, \varepsilon) \vdash^* (w, A\alpha, \dots) \vdash^+ (w, A\beta\alpha, \dots)$.

Lemma 9.7

If $G \in CFG_\Sigma$ is left recursive, then $G \notin \bigcup_{k \in \mathbb{N}} LL(k)$.

Proof.

(for $k = 1$) Assume that $G \in LL(1)$ is left recursive with $A \Rightarrow_l^+ A\beta$.

Together with the reducedness of G this implies that

$S \Rightarrow_l^* vA\alpha \Rightarrow_l^+ vA\beta\alpha \Rightarrow_l^+ vw$ for some $v, w \in \Sigma^*$ and $\alpha \in X^*$.

The corresponding computation of $DTA(G)$ (Def. 9.2) starts with $(vw, S, \varepsilon) \vdash^* (w, A\alpha, \dots) \vdash^+ (w, A\beta\alpha, \dots)$.

But in the last state the behaviour of $DTA(G)$ is determined by the same input ($\text{fi}(w)$) and stack symbol (A). Thus it enters a loop of the form $(w, A\alpha, \dots) \vdash^+ (w, A\beta\alpha, \dots) \vdash^+ (w, A\beta\beta\alpha, \dots) \vdash^+ \dots$ and will never recognize w . Contradiction □

Removing Direct Left Recursion

Direct left recursion occurs in productions of the form

$$A \rightarrow A\alpha_1 \mid \dots \mid A\alpha_m \mid \beta_1 \mid \dots \mid \beta_n \quad \text{where } \alpha_i \neq \varepsilon \text{ and } \beta_j \neq A\dots$$

Removing Direct Left Recursion

Direct left recursion occurs in productions of the form

$$A \rightarrow A\alpha_1 \mid \dots \mid A\alpha_m \mid \beta_1 \mid \dots \mid \beta_n \quad \text{where } \alpha_i \neq \varepsilon \text{ and } \beta_j \neq A\dots$$

Transformation: replacement by **right recursion**

$$\begin{aligned} A &\rightarrow \beta_1 A' \mid \dots \mid \beta_n A' \\ A' &\rightarrow \alpha_1 A' \mid \dots \mid \alpha_m A' \mid \varepsilon \end{aligned}$$

(with a new $A' \in N$) which preserves $L(G)$.

Removing Direct Left Recursion

Direct left recursion occurs in productions of the form

$$A \rightarrow A\alpha_1 \mid \dots \mid A\alpha_m \mid \beta_1 \mid \dots \mid \beta_n \quad \text{where } \alpha_i \neq \varepsilon \text{ and } \beta_j \neq A\dots$$

Transformation: replacement by **right recursion**

$$\begin{aligned} A &\rightarrow \beta_1 A' \mid \dots \mid \beta_n A' \\ A' &\rightarrow \alpha_1 A' \mid \dots \mid \alpha_m A' \mid \varepsilon \end{aligned}$$

(with a new $A' \in N$) which preserves $L(G)$.

Example 9.8

$$\begin{aligned} G_{AE} : \quad & E \rightarrow E+T \mid T \\ & T \rightarrow T*F \mid F \\ & F \rightarrow (E) \mid a \mid b \end{aligned} \quad \text{is transformed into}$$

$$\begin{aligned} G'_{AE} : \quad & E \rightarrow TE' \\ & E' \rightarrow +TE' \mid \varepsilon \\ & T \rightarrow FT' \\ & T' \rightarrow *FT' \mid \varepsilon \\ & F \rightarrow (E) \mid a \mid b \end{aligned} \quad \text{with } G'_{AE} \in LL(1) \text{ (see Example 9.1).}$$

Removing Indirect Left Recursion

Indirect left recursion occurs in productions of the form ($n \geq 1$)

$$\begin{array}{lcl} A & \rightarrow & A_1\alpha_1 \mid \dots \\ A_1 & \rightarrow & A_2\alpha_2 \mid \dots \\ & \vdots & \\ A_{n-1} & \rightarrow & A_n\alpha_n \mid \dots \\ A_n & \rightarrow & A\beta \mid \dots \end{array}$$

Removing Indirect Left Recursion

Indirect left recursion occurs in productions of the form ($n \geq 1$)

$$\begin{array}{lcl} A & \rightarrow & A_1\alpha_1 \mid \dots \\ A_1 & \rightarrow & A_2\alpha_2 \mid \dots \\ & \vdots & \\ A_{n-1} & \rightarrow & A_n\alpha_n \mid \dots \\ A_n & \rightarrow & A\beta \mid \dots \end{array}$$

Transformation: into **Greibach Normal Form** with productions of the form

$$\begin{array}{l} A \rightarrow aB_1 \dots B_n \quad (\text{where } n \in \mathbb{N} \text{ and each } B_i \neq S) \text{ or} \\ S \rightarrow \varepsilon \end{array}$$

(cf. *Formale Systeme, Automaten, Prozesse*)

Left Factorization

Applies to productions of the form

$$A \rightarrow \alpha\beta \mid \alpha\gamma$$

which are problematic if α “at least as long as lookahead”.

Left Factorization

Applies to productions of the form

$$A \rightarrow \alpha\beta \mid \alpha\gamma$$

which are problematic if α “at least as long as lookahead”.

Transformation: delaying the decision by **left factorization**

$$\begin{aligned} A &\rightarrow \alpha A' \\ A' &\rightarrow \beta \mid \gamma \end{aligned}$$

(with a new $A' \in N$) which preserves $L(G)$.

Left Factorization

Applies to productions of the form

$$A \rightarrow \alpha\beta \mid \alpha\gamma$$

which are problematic if α “at least as long as lookahead”.

Transformation: delaying the decision by **left factorization**

$$\begin{aligned} A &\rightarrow \alpha A' \\ A' &\rightarrow \beta \mid \gamma \end{aligned}$$

(with a new $A' \in N$) which preserves $L(G)$.

Example 9.9

Statement $\rightarrow$ if *Condition* then *Statement* else *Statement* fi
 | if *Condition* then *Statement* fi

is transformed into

Statement $\rightarrow$ if *Condition* then *Statement* S'
 $S' \rightarrow$ else *Statement* fi | fi

- 1 Repetition: $LL(1)$ Grammars
- 2 Deterministic Top-Down Parsing
- 3 Transformation to $LL(1)$
- 4 The Complexity of $LL(1)$ Parsing

The Complexity of $LL(1)$ Parsing I

- $LL(1)$ parsing has time (and hence space) complexity $\mathcal{O}(|w|)$ (where $w \in \Sigma^*$ is the input word)

The Complexity of $LL(1)$ Parsing I

- $LL(1)$ parsing has time (and hence space) **complexity** $\mathcal{O}(|w|)$ (where $w \in \Sigma^*$ is the input word)
- Here: proof for **ε -free grammars** (i.e., $A \rightarrow \alpha \in P \implies \alpha \neq \varepsilon$)

The Complexity of $LL(1)$ Parsing I

- $LL(1)$ parsing has time (and hence space) **complexity** $\mathcal{O}(|w|)$ (where $w \in \Sigma^*$ is the input word)
- Here: proof for **ε -free grammars** (i.e., $A \rightarrow \alpha \in P \implies \alpha \neq \varepsilon$)
- General case: see O. Mayer: *Syntaxanalyse*, p. 211ff

The Complexity of $LL(1)$ Parsing I

- $LL(1)$ parsing has time (and hence space) **complexity** $\mathcal{O}(|w|)$ (where $w \in \Sigma^*$ is the input word)
- Here: proof for **ε -free grammars** (i.e., $A \rightarrow \alpha \in P \implies \alpha \neq \varepsilon$)
- General case: see O. Mayer: *Syntaxanalyse*, p. 211ff

Lemma 9.10

Let $G = \langle N, \Sigma, P, S \rangle \in LL(1)$ be ε -free. If

$$(w, S, \varepsilon) \vdash^n (\varepsilon, \varepsilon, z)$$

in $DTA(G)$, then

$$n \leq (|w| + 1) \cdot (|N| + 1).$$

The Complexity of $LL(1)$ Parsing II

Proof.

Let $(w, S, \varepsilon) \vdash^n (\varepsilon, \varepsilon, z)$ in $DTA(G)$. To show: $n \leq (|w| + 1) \cdot (|N| + 1)$

- ① Clear: the computation involves $|w|$ matching steps.



The Complexity of $LL(1)$ Parsing II

Proof.

Let $(w, S, \varepsilon) \vdash^n (\varepsilon, \varepsilon, z)$ in $DTA(G)$. To show: $n \leq (|w| + 1) \cdot (|N| + 1)$

- 1 Clear: the computation involves $|w|$ matching steps.
- 2 Since G is ε -free, every matching step is preceded (and followed) by k expansion steps of the form

$$(av, A_1\alpha_1, \dots) \vdash (av, A_2\alpha_2\alpha_1, \dots)$$

$$\vdots$$

$$\vdash (av, A_k\alpha_k \dots \alpha_1, \dots)$$

$$\vdash (av, a\alpha_{k+1} \dots \alpha_1, \dots)$$

where $A_i \rightarrow A_{i+1}\alpha_{i+1}$ for each $i \in [k - 1]$ and $A_k \rightarrow a\alpha_{k+1}$.



The Complexity of $LL(1)$ Parsing II

Proof.

Let $(w, S, \varepsilon) \vdash^n (\varepsilon, \varepsilon, z)$ in $DTA(G)$. To show: $n \leq (|w| + 1) \cdot (|N| + 1)$

- ① Clear: the computation involves $|w|$ matching steps.
- ② Since G is ε -free, every matching step is preceded (and followed) by k expansion steps of the form

$$\begin{aligned}(av, A_1\alpha_1, \dots) &\vdash (av, A_2\alpha_2\alpha_1, \dots) \\ &\vdots \\ &\vdash (av, A_k\alpha_k \dots \alpha_1, \dots) \\ &\vdash (av, a\alpha_{k+1} \dots \alpha_1, \dots)\end{aligned}$$

where $A_i \rightarrow A_{i+1}\alpha_{i+1}$ for each $i \in [k-1]$ and $A_k \rightarrow a\alpha_{k+1}$.

- ③ This implies that $A_i \neq A_j$ for $i \neq j$ (by Lemma 9.7, G is not left recursive), and hence $k \leq |N|$.



The Complexity of $LL(1)$ Parsing II

Proof.

Let $(w, S, \varepsilon) \vdash^n (\varepsilon, \varepsilon, z)$ in $DTA(G)$. To show: $n \leq (|w| + 1) \cdot (|N| + 1)$

- ① Clear: the computation involves $|w|$ matching steps.
- ② Since G is ε -free, every matching step is preceded (and followed) by k expansion steps of the form

$$\begin{aligned}(av, A_1\alpha_1, \dots) &\vdash (av, A_2\alpha_2\alpha_1, \dots) \\ &\vdots \\ &\vdash (av, A_k\alpha_k \dots \alpha_1, \dots) \\ &\vdash (av, a\alpha_{k+1} \dots \alpha_1, \dots)\end{aligned}$$

where $A_i \rightarrow A_{i+1}\alpha_{i+1}$ for each $i \in [k - 1]$ and $A_k \rightarrow a\alpha_{k+1}$.

- ③ This implies that $A_i \neq A_j$ for $i \neq j$ (by Lemma 9.7, G is not left recursive), and hence $k \leq |N|$.
- ④ Altogether: $n \leq (|w| + 1) \cdot (|N| + 1)$.

