

## 6. LR(k) Parsing

**LR(k) parsing:**

The most general deterministic parsing method in which the input string is parsed

- (1) in a single Left-to-right scan,
- (2) producing a Right parse, and
- (3) using lookahead of length k.

Generalization of

- (1) nondeterministic shift-reduce parser
- (2) the simple precedence parser

stack symbols:

grammar symbols are divided up into one or more “context dependent” symbols

Two stack strings  $\gamma_1 X$  and  $\gamma_2 X$  are equivalent, if exactly same set of parsing actions are valid in the context of  $\gamma_1 X$  and  $\gamma_2 X$ .

Replacing  $X$  by equivalent class  $[\gamma X]$   
refinement of stack symbol

$$X \in V \quad [\gamma X] \in 2^{V^*}$$

## 6.1 Viable Prefixes

$G_{ab}$ :

$$S \rightarrow aA \mid bB$$

$$A \rightarrow c \mid dAd$$

$$B \rightarrow c \mid dBd$$

$$L(G_{ab}) = \{a, b\}\{d^n cd^n \mid n \geq 0\}.$$

$\$ \alpha c \mid y \$$ , where  $\$ \alpha:1 \in \{a, b, d\}$  and  $1:y\$ \in \{\$, d\}$   
 reduce-reduce conflict for  $A \rightarrow c$  and  $B \rightarrow c$ .  
 $(1:\alpha = a) \quad (1:\alpha = b)$

Extending lookahead and lookback into length  $k$ .

$$\alpha c \mid x \rightarrow \alpha A \mid x, \quad \beta c \mid y \rightarrow \beta B \mid y$$

$$\alpha, \beta \in V^*:k, \quad x, y \in k:\Sigma^* \$.$$

but

$$ad^k c \mid d^k \rightarrow ad^k A \mid d^k, \quad bd^k c \mid d^k \rightarrow bd^k B \mid d^k$$

reduce-reduce conflict for any  $k$ !

A string  $\gamma$  is a **viable stack string** of pda  $M$ , if

$$\$ \gamma_s \mid xy \$ \Rightarrow^* \$ \gamma \mid y \$ \Rightarrow^* \$ \gamma_f \mid \$ \text{ in } M.$$

stack string in some accepting computation  $M$ .

**Not** arbitrary string is a **viable** stack string.

$$2^{V^*} \text{ vs. } 2^{VS} \text{ where } VS \subseteq V^*.$$

*Viable stack strings of  $G_{ab}$ :*

$$\begin{aligned} & \{\varepsilon\} \cup \{ad^n \mid n \geq 0\} \cup \{ad^n c \mid n \geq 0\} \\ & \cup \{ad^n A \mid n \geq 0\} \cup \{ad^n Ad \mid n \geq 1\} \\ & \cup \{bd^n \mid n \geq 0\} \cup \{bd^n c \mid n \geq 0\} \\ & \cup \{bd^n B \mid n \geq 0\} \cup \{bd^n Bd \mid n \geq 1\} \\ & \cup \{S\} \end{aligned}$$

*Not every **action** is valid, for viable stack string*  
 $ad^n c \mid \Rightarrow_{\text{valid}} ad^n A \mid$ ,  $bd^n c \mid \Rightarrow_{\text{valid}} bd^n B \mid$ ; *but*  
 $ad^n c \mid \not\Rightarrow_{\text{valid}} ad^n B \mid$ ,  $bd^n c \mid \not\Rightarrow_{\text{valid}} bd^n A \mid$ .

*An action  $r$  is **valid** for viable stack string  $\gamma$  of  $M$  if*  
 $\$ \gamma \mid y \$ \Rightarrow^r \$ \gamma' \mid y' \$ \Rightarrow^* \$ \gamma_f \mid \$$  *in  $M$*

*The set of viable stack strings are **infinite**. But we can divide the set of viable stack strings in to a **finite** number of **equivalent classes**.*

*Two viable stack string belongs to the **same equivalent class** if they have same set of **valid actions**.*

*Since for any  $G = (N, \Sigma, P, S)$  in shift-reduce parser*  
*number of distinct actions =  $|\Sigma| + |P| \leq |G|$*   
*number of equivalent classes  $\leq 2^{|G|}$ .*  
 *$\therefore$  number of equivalent classes is finite.*

<i>equivalent classes:</i>	<i>valid actions:</i>
$\{\epsilon\}$	<i>shift a, shift b</i>
$\{ad^n \mid n \geq 0\} \cup \{bd^n \mid n \geq 0\}$	<i>shift c, shift d</i>
$\{ad^n c \mid n \geq 0\}$	<i>reduce by <math>A \rightarrow c</math></i>
$\{bd^n c \mid n \geq 0\}$	<i>reduce by <math>B \rightarrow c</math></i>
$\{aA\}$	<i>reduce by <math>S \rightarrow aA</math></i>
$\{bB\}$	<i>reduce by <math>S \rightarrow bB</math></i>
$\{ad^n A \mid n \geq 1\} \cup \{bd^n B \mid n \geq 1\}$	<i>shift d</i>
$\{ad^n Ad \mid n \geq 1\}$	<i>reduce by <math>A \rightarrow dAd</math></i>
$\{bd^n Bd \mid n \geq 1\}$	<i>reduce by <math>B \rightarrow dBd</math></i>
$\{S\}$	—

*stack symbols: equivalent classes (grammar symbol)*

$X \Rightarrow [\delta X]: \delta X: \text{viable stack string}$

$[\delta X]: \text{equivalent class of } \delta X$

***shift a***

$[\delta] \mid a \rightarrow [\delta][\delta a] \mid$

***reduce by  $A \rightarrow X_1 \dots X_n$***

$[\delta][\delta X_1] \dots [\delta X_1 \dots X_n] \mid \rightarrow [\delta][\delta A] \mid$

$\gamma_s = [\epsilon]$  and  $\gamma_f = \{[\epsilon][S]\}$

$\therefore [\epsilon] \mid yz \Rightarrow^* [\epsilon][Y_1] \dots [Y_1 \dots Y_k] \mid z \Rightarrow^* [\epsilon][S] \mid.$

$Y_1 \dots Y_i$  are viable stack string for  $0 \leq \forall i \leq k.$

**Regular expression for valid viable stack strings**

$\varepsilon, ad^* \mid bd^*, ad^*c, aA, ad^+A \mid bd^+B, ad^+Ad, bd^*c, bB, bd^+Bd, S$

For regular expression  $E$ , we define

$$[E] \equiv \cup_{w \in L(E)} [w].$$

$\therefore L(E) \subseteq [E]$ , in fact usually  $L(E) = [E]$ .

equivalent classes:

valid actions:

$[\varepsilon]$

shift  $a$ , shift  $b$

$[S]$

—

$[ad^* \mid bd^*]$

shift  $c$ , shift  $d$

$[ad^*c]$

reduce by  $A \rightarrow c$

$[bd^*c]$

reduce by  $B \rightarrow c$

$[aA]$

reduce by  $S \rightarrow aA$

$[bB]$

reduce by  $S \rightarrow bB$

$[ad^+A \mid bd^+B]$

shift  $d$

$[ad^+Ad]$

reduce by  $A \rightarrow dAd$

$[bd^+Bd]$

reduce by  $B \rightarrow dBd$

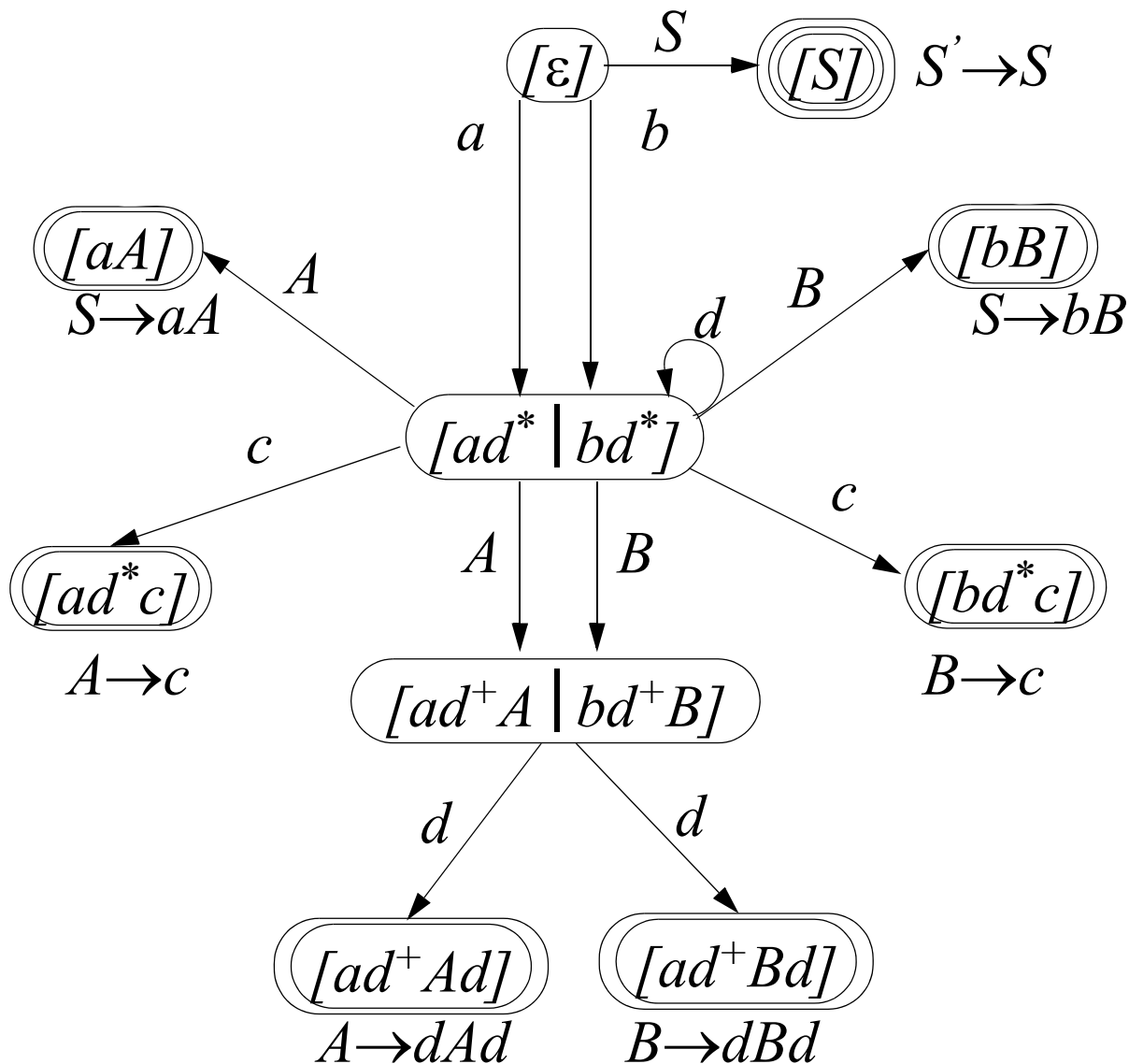
Regular expressions  $\Rightarrow$  finite automata

regular expression over  $N \cup \Sigma$ .

$\therefore$  finite automaton with input alphabet  $N \cup \Sigma$ .

Characteristic finite state machine

Deterministic parsing of context-free languages?



No “reduce-reduce conflicts” by  $A \rightarrow c$  and  $B \rightarrow c$ .

$$[ad^* | bd^*][ad^* c] \mid \rightarrow [ad^* | bd^*][aA] \mid$$

(reduce by  $A \rightarrow c$ ),

$$[ad^* | bd^*][bd^* c] \mid \rightarrow [ad^* | bd^*][bB]$$

(reduce by  $B \rightarrow c$ ).

note that  $[aA] \neq [ad^+ A | bd^+ B] \neq [bB]$ .

But “shift-shift conflict”

$$[ad^*|bd^*] \mid c \rightarrow [ad^*|bd^*][ad^*c] \mid \quad (\text{shift } c),$$

$$[ad^*|bd^*] \mid c \rightarrow [ad^*|bd^*][bd^*c] \mid \quad (\text{shift } c),$$

and

$$[ad^+A|bd^+B] \mid d \rightarrow [ad^+A|bd^+B][ad^+Ad] \mid,$$

$$[ad^+A|bd^+B] \mid d \rightarrow [ad^+A|bd^+B][bd^+Bd] \mid.$$

Consider  $ad^n$ ,  $ad^nA$ , and  $ad^nB$  for  $n \geq 0$ .

$ad^n \in [ad^*|bd^*]$ . But

$$ad^nA \in [aA] \text{ and } [ad^+A|bd^+B].$$

$$bd^nA \in [aB] \text{ and } [ad^+A|bd^+B].$$

$[ad^*|bd^*]$  is split into  $[a]$ ,  $[ad^+]$ ,  $[b]$ , and  $[bd^+]$

Since  $[ad^*c] \neq [bd^*c]$ ,  $[ad^*] \neq [bd^*]$ .

Since  $[aA] \neq [ad^+A]$ ,  $[a] \neq [ad^+]$ .

Since  $[bA] \neq [bd^+A]$ ,  $[b] \neq [bd^+]$ .

$[ad^+A|bd^+B]$  is split into  $[ad^+A]$  and  $[bd^+B]$

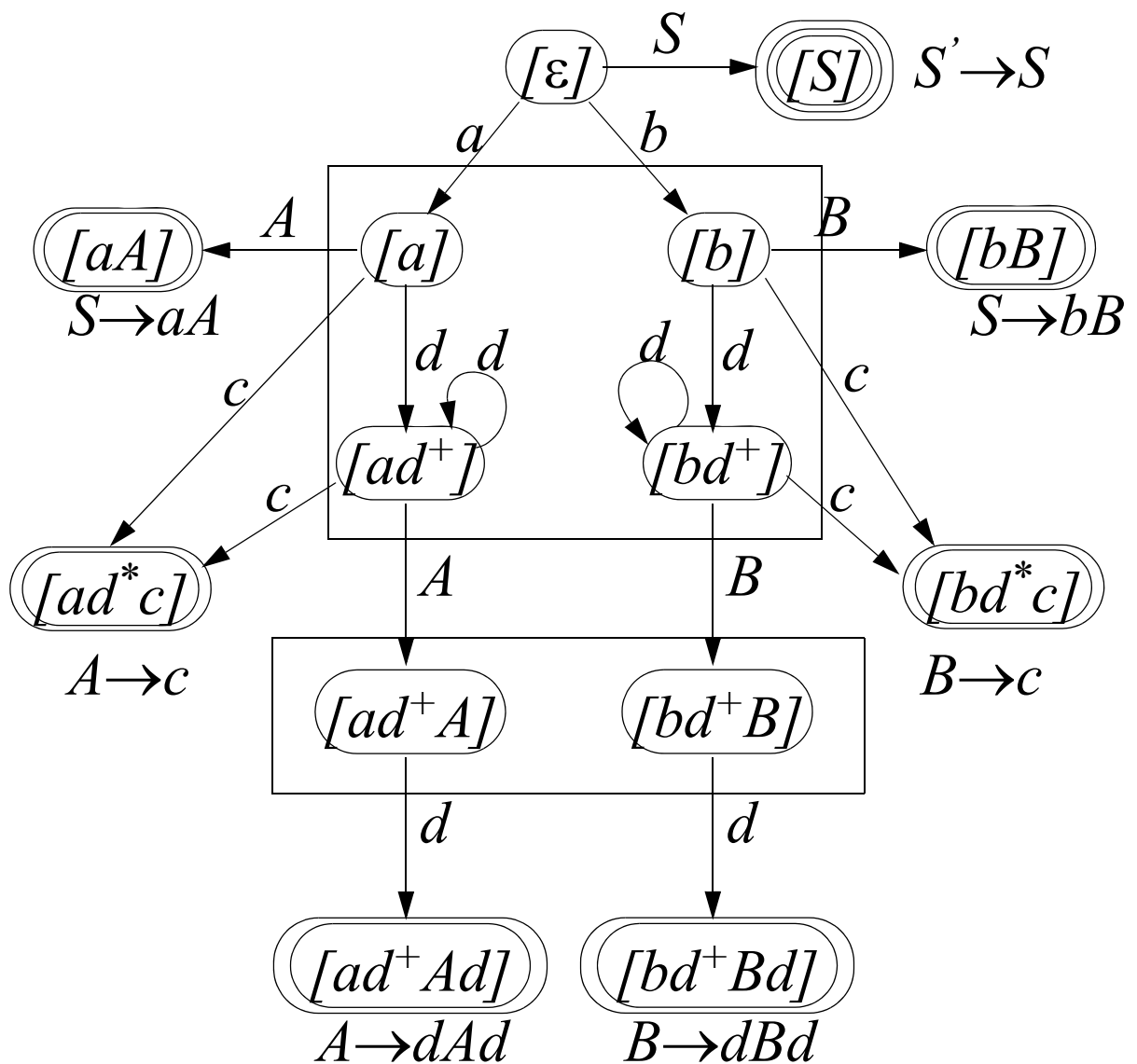
Since  $[ad^+Ad] \neq [bd^+Bd]$ ,  $[ad^+A] \neq [bd^+B]$ .

$[ad^* \mid bd^*]$

$[a], [b], [ad^+], [bd^+]$

$[ad^+A \mid bd^+B]$

$[ad^+A], [bd^+B]$



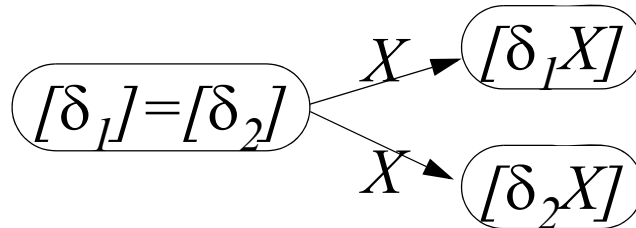


**Right invariant(unique outgoing symbol)**

If two stack string  $\delta_1$  and  $\delta_2$  are equivalent, they remain equivalent when they are lengthened.

If  $[\delta_1] = [\delta_2]$ ,  $[\delta_1 X] = [\delta_2 X]$ .

Otherwise “shift-shift” conflict.



**Unique entry symbol**

Two equivalent stack string should end with same symbols. If  $[\gamma_1] = [\gamma_2]$ ,  $\gamma_1:1 = \gamma_2:1$ .

Otherwise, reduce action is not uniquely defined.

Consider  $[\delta][\delta X_1] \dots [\delta X_1 \dots X_n] \mid \rightarrow [\delta A] \mid$ .

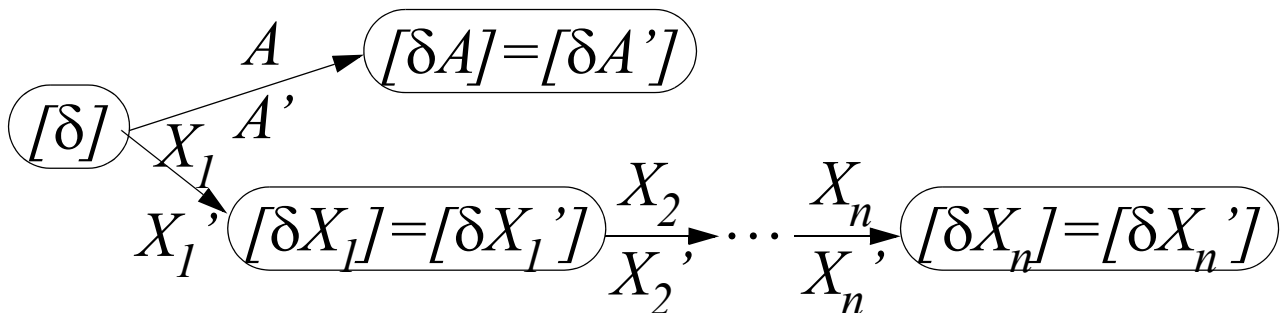
the rule  $A \rightarrow X_1 \dots X_n$  is uniquely defined, if

$[\delta][\delta X_1'] \dots [\delta X_1' \dots X_n'] \mid \rightarrow [\delta A'] \mid$ ,

$\exists A' \rightarrow X_1' \dots X_n' \in P$

$\exists. [\delta A] = [\delta A'], [\delta X_1] = [\delta X_1'], \dots,$

$[\delta X_1 \dots X_n] = [\delta X_1' \dots X_n']$ .



Let  $G = (N, \Sigma, P, S)$  be a grammar. String  $\gamma (= \delta\alpha) \in V^*$  is a **viable prefix** of  $G$ , if

$$S \Rightarrow_{rm}^* \delta A y \Rightarrow_{rm} \delta \alpha \beta y (= \gamma \beta y)$$

where  $\delta \in V^*$ ,  $y \in \Sigma^*$ , and  $A \rightarrow \alpha\beta \in P$ .

$\gamma$  is a **complete viable prefix**, if  $\beta = \varepsilon$ .

**Fact 6.1** Any viable prefix is a **prefix** of some complete viable prefix.

**Lemma 6.4** Any **prefix** of a viable prefix is a **viable prefix**.

**Proof**  $S \Rightarrow_{rm}^n \delta A y \Rightarrow_{rm} \underline{\delta} \alpha \beta y = \underline{\gamma}_1 \underline{\gamma}_2 \beta y$

i)  $\delta$  is a prefix of  $\gamma_1$ .

$\gamma_1 = \delta \alpha'$  where  $\alpha = \alpha' \gamma_2$ .  $\therefore \gamma_1$  is a viable prefix.

ii)  $\gamma_1$  is a prefix of  $\delta$ . ( $\delta \neq \varepsilon$ )

$$\delta A = \gamma_1 \eta. n > 0, \gamma_1 \text{ is a viable prefix. (L6.2)}$$

**Lemma 6.2** Let  $G = (N, \Sigma, P, S)$  be a grammar,  $\pi \in P^+$ ,  $\gamma, \eta, \delta \in V^*$ ,  $A \in N$ , and  $y \in \Sigma^*$ .

If  $S \Rightarrow_{rm}^\pi \gamma \eta y = \delta A y$  in  $G$ , and  $\pi \neq \varepsilon$ . Then

$$S \Rightarrow_{rm}^\pi \delta' A' y'$$

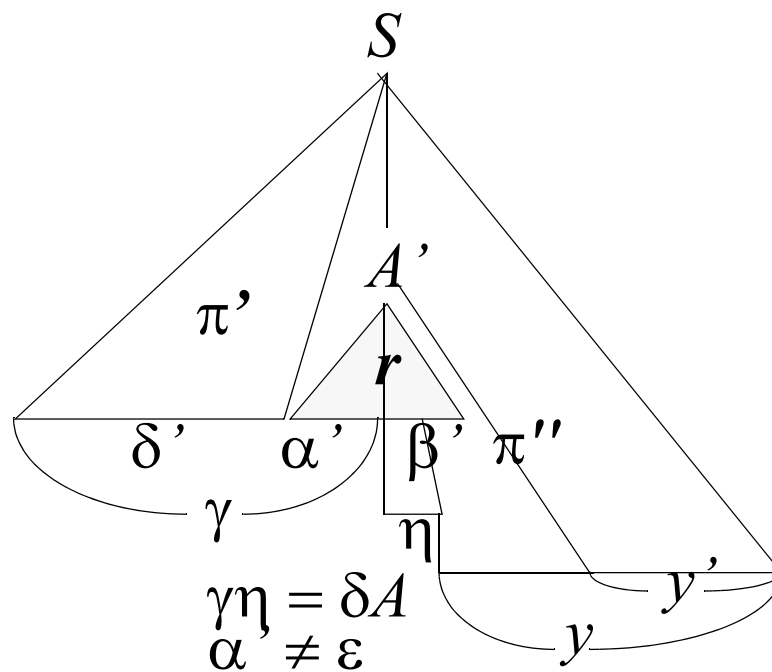
$$\Rightarrow_{rm}^r \underline{\delta'} \alpha' \beta' y' = \underline{\gamma} \beta' y'$$

$$\Rightarrow_{rm}^{\pi''} \gamma \eta y = \delta A y,$$

$\pi' r \pi'' = \pi$ , and  $\alpha' \neq \varepsilon$  ( $\alpha':1 = \gamma:1$ ).

If  $\gamma$  is a **prefix** of some nontrivially derived right sentential form (not extending over the last nonterminal), the derivation contains a segment rule ( $r$ ) that proves  $\gamma$  to be a **viable prefix**, even so that the right-hand side of the rule  $r$  cuts  $\gamma$  properly.

Any prefix of nontrivially derived right sentential form (not extending over the last nonterminal) is a **viable prefix**.



**Proof** induction on the length of  $\pi$ .

i)  $|\pi| = 1$ .  $\pi = S \rightarrow \gamma\eta y = A' \rightarrow \alpha' \beta' y$ .

$$\delta' = y' = \varepsilon, (\gamma = \alpha', \eta y = \beta')$$

ii)  $|\pi| > 1$ . Assume that IH holds for  $\pi_1$  where  $\pi = \pi_1 r_1$ .

$$S \Rightarrow_{rm}^{\pi_1} \gamma_1 \eta_1 y_1 = \delta_1 A_1 y_1$$

$$\Rightarrow_{rm}^{r_1} \delta_1 \omega y_1 = \gamma \eta y = \delta A y, \pi = r \pi_1.$$

$$\text{Then } S \Rightarrow_{rm}^{\pi_1} \delta_1' A_1' y_1'$$

$$\Rightarrow_{rm}^{r_1} \underline{\delta_1'} \underline{\alpha_1'} \underline{\beta_1'} y_1' = \gamma_1 \beta_1' y_1'$$

$$\Rightarrow_{rm}^{\pi_1} \gamma_1 \eta_1 y_1 = \delta_1 A_1 y_1,$$

$$\pi_1' r_1 \pi_1'' = \pi_1, \text{ and } \delta_1' \alpha_1' = \gamma_1.$$

Note that  $\gamma = \delta_1 \alpha''$  where  $\alpha'' \neq \varepsilon$  or  $\gamma \alpha = \delta_1$

a)  $\gamma = \delta_1 \alpha''$ , it is trivial, since

$$\delta' = \delta_1, y' = y_1, \pi' = \pi_1, \pi'' = \varepsilon, r = r_1.$$

b)  $\gamma \alpha = \delta_1$

$$S \Rightarrow_{rm}^{\pi_1} \gamma \eta_1 y_1 = \delta_1 A_1 y_1.$$

**Lemma 6.3**

$S \Rightarrow_{rm}^+ \delta Ay$ . Then  $\delta$  is a *viable prefix*.

**Proof.**  $\eta = \varepsilon$ .

**Lemma 6.5** Let  $A \rightarrow \alpha\beta \in P$ . Then if  $\gamma A$  is a viable prefix of  $G$ , then so is  $\gamma\alpha$ .

**Lemma 6.6** If

$\$ | w\$ \Rightarrow^\pi \$\gamma\eta | y\$$ . Then

$\$ | w\$ \Rightarrow^{\pi'} \$\gamma | z\$ \Rightarrow^{\pi''} \$\gamma\eta | y\$$ , and  $\pi = \pi' \pi''$ .

**Proof** induction on  $|\pi|$ .

i)  $\pi = \varepsilon$ ,  $\gamma = \eta = \varepsilon$ .

ii)  $\pi \neq \varepsilon$  and  $\eta \neq \varepsilon$ ,  $\pi = \pi_1 r_1$ .

(1)  $\$ | w\$ \Rightarrow^{\pi_1} \$\psi | ay\$ \Rightarrow^{r_1} \$\psi a | y\$$ , or

(2)  $\$ | w\$ \Rightarrow^{\pi_1} \$\delta\omega | y\$ \Rightarrow^{r_1} \$\delta A | y\$$ .

$\gamma$  is a prefix of  $\psi$  in (1), and a prefix of  $\delta$  in (2).

**Theorem 6.7**

Let  $G = (N, \Sigma, P, S)$ ,  $M$  be a shift-reduce parser of  $G$ .

Any viable stack string of  $M$  is

either  $S$  or *viable prefix* of  $G$ .

Conversely, any viable prefix of  $G$  is

a *viable stack string* of  $M$ ,

provided that  $G$  is reduced.

**Proof** from lemma 5.17, 5.19.

(shift-reduce parser = right parser)

Given a grammar  $G = (N, \Sigma, P, S)$ ,

Let  $G_{VP} = (N_{VP}, \Sigma_{VP}, P_{VP}, [S])$  where

$$N_{VP} = \{[A] \mid A \in N\},$$

$$\Sigma_{VP} = N \cup \Sigma, \text{ and}$$

$$P_{VP} = \{[A] \rightarrow \alpha \mid A \rightarrow \alpha\beta \in P\} \cup \{[A] \rightarrow \alpha[B] \mid A \rightarrow \alpha B\beta \in P, B \in N\}.$$

Example)

$(G_{ab})_{VP}$ :

$$[S] \rightarrow \varepsilon \mid a \mid aA \mid b \mid bB \mid a[A] \mid b[B]$$

$$[A] \rightarrow \varepsilon \mid c \mid d \mid dA \mid dAd \mid d[A]$$

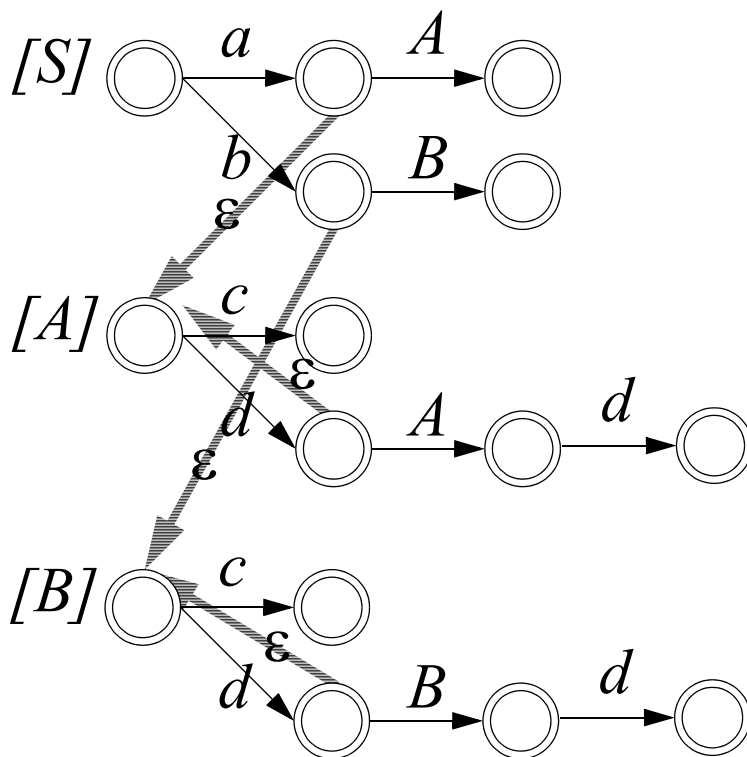
$$[B] \rightarrow \varepsilon \mid c \mid d \mid dB \mid dBd \mid d[B]$$

$G_{ab}$ :

$$S \rightarrow aA \mid bB$$

$$A \rightarrow c \mid dAd$$

$$B \rightarrow c \mid dBd$$



Rule automaton

**Lemma 6.8** Let  $S \Rightarrow_{rm}^n \delta Ay$  in  $G$ . Then

$$[S] \Rightarrow^* \delta[A] \text{ in } G_{VP}.$$

**Proof**

i)  $n=0$ : it is clear ( $A=S$ ,  $\delta=y=\varepsilon$ ).

ii)  $0 \leq \forall m < n$ :

$$S \Rightarrow_{rm}^m \delta' A' y' \Rightarrow_{rm} \delta' \alpha A \beta y' = \delta A \beta y' \text{ in } G. \text{ (L6.2)}$$

$[A'] \rightarrow \alpha[A] \in P_{VP}$ , since  $A' \rightarrow \alpha A \beta \in P$

$$[S] \Rightarrow^* \delta'[A'] \Rightarrow \delta' \alpha[A] = \delta[A] \text{ in } G_{VP}.$$

**Lemma 6.9** Let  $[S] \Rightarrow^n \delta[A]$  in  $G_{VP}$ . Then

$$S \Rightarrow^* \delta Ay \text{ in } G.$$

**Proof**

i)  $n=0$ :  $\delta=\varepsilon$ ,  $A=S$ ,  $y=\varepsilon$ .

ii)  $0 \leq \forall m < n$ :

$$S \Rightarrow^m \delta'[A'] \Rightarrow \delta' \alpha[A] = \delta[A] \text{ in } G_{VP},$$

$A' \rightarrow \alpha A \beta \in P$ ,  $\beta \Rightarrow^* x \in \Sigma^*$ , since  $[A'] \rightarrow \alpha[A] \in P_{VP}$ .

$$S \Rightarrow^* \delta' A' y' \Rightarrow \delta' \alpha A \beta \Rightarrow^* \delta' \alpha A x y' = \delta Ay \text{ in } G.$$

**Theorem 6.10** *The grammar  $G_{VP}$  generates the set of viable prefixes of  $G$ . And  $G_{VP}$  is right linear.*

*Proof.*

If  $S \Rightarrow_{rm}^* \delta Ay \Rightarrow_{rm} \delta \alpha \beta y (= \gamma \beta y)$  in  $G$  ( $A \rightarrow \alpha \beta \in P$ ),  
 $[S] \Rightarrow^* \delta[A] \Rightarrow \delta \alpha (= \gamma)$ .

If  $[S] \Rightarrow^* \delta[A] \Rightarrow \delta \alpha \in V^*$  and  $A \rightarrow \alpha \beta \in P$ , then  
 $S \Rightarrow_{rm}^* \delta Ay \Rightarrow_{rm} \delta \alpha \beta y$ .

**Theorem 6.11** *For any grammar  $G = (N, \Sigma, P, S)$ , the set of all viable prefixes is a **regular expression** over  $V$ .*

*viable prefixes = valid stack strings  
 = regular expression*

$G_{VP}$  is a regular grammar generating the set of viable prefixes of  $G$ .

$C_0$  in  $G$  is the **dfa** for  $G_{VP}$ .



## 6.2 Valid LR(k) Items

Let  $A \rightarrow \alpha\beta \in P$ . Then  $[A \rightarrow \alpha\bullet\beta, y]$  is a **k-item**, if

$A \rightarrow \alpha\bullet\beta$  is a position of  $G$  and  $y \in \Sigma^k$ .

0-item  $[A \rightarrow \alpha\bullet\beta, \varepsilon] \equiv [A \rightarrow \alpha\bullet\beta]$

$A \rightarrow \alpha\bullet\beta$  is **core** of the item,

$y$  is the **lookahead** of the item.

A k-item  $[A \rightarrow \alpha\bullet\beta, y]$  is **LR(k)-valid** (or **valid**) for string  $\gamma(=\delta\alpha) \in V^*$  if

$S \Rightarrow_{rm}^* \delta Az \Rightarrow_{rm} \delta\alpha\beta z (= \gamma\beta z)$  and  $y = k:z\$^k$ .

Let  $R_k$  denotes the set of whole valid LR(k) items.

**Fact 6.12** If  $[A \rightarrow \alpha\bullet\beta, y]$  is a LR(k) valid item for string  $\gamma(=\delta\alpha)$ , then  $\gamma$  is a **viable prefix** and

$y \in \text{Follow}_k(\delta\alpha\beta) = \text{Follow}_k(\delta A) \subseteq \text{Follow}_k(A)$ .

Conversely, if a string  $\gamma$  is a viable prefix, then some item is LR(k)-valid for  $\gamma$ .

**Define**  $Valid_{LR(k)}^G: V^* \rightarrow 2^{R_k}$ .

Let  $\gamma \in V^*$ . Then

$$Valid_k(\gamma)_{LR(k)}^G =$$

$$\{[A \rightarrow \alpha \bullet \beta, x] \mid S \xrightarrow{rm}^* \delta Az \xrightarrow{rm} \delta \alpha \beta z = \gamma \beta z, x = k:z\$^k\}$$

Valid LR(k) items for the **viable prefix**  $\gamma$

$$Valid_{LR(k)}^G \equiv Valid_{LR(k)} \equiv Valid_k \equiv Valid$$

$$Valid_k: V^* \rightarrow 2^{R_k}.$$

**Define**  $\rho_{LR(k)} \subseteq V^* \times V^*$

$\gamma_1$  is **LR(k)-equivalent** to  $\gamma_2$ ,

$$\text{written } \gamma_1 \rho_{LR(k)} \gamma_2 \text{ (or } \gamma_1 \rho_k \gamma_2),$$

$$\text{if } Valid_k(\gamma_1) = Valid_k(\gamma_2).$$

The relation  $\rho_k$  is called the **LR(k)-equivalence** for  $G$ .

$\rho_k$  is an **equivalent** relation.

$[\gamma]_{\rho_k}$  denotes an **equivalent class** of  $\gamma$  under  $\rho_k$

$$[\gamma]_{\rho_k} = \{\delta \mid \gamma \rho_k \delta\}$$

$$[\gamma]_{\rho_k} \equiv [\gamma]_k \equiv [\gamma].$$

We denote  $[\gamma]_{\rho_k}$  by  $[\gamma]_k$  (or even  $[\gamma]$ ).

We extend the domain of  $Valid_k$  from  $V^*$  to  $2^{V^*}$ :

$$Valid_k(L) = \{I \in R_k \mid I \in Valid_k(\alpha), \alpha \in L \subseteq V^*\}$$

$$Valid_k([\gamma]_k) = \{I \mid I \in Valid_k(\delta), \delta \in [\gamma]_k\}$$

Since  $Valid_k(\gamma_1) = Valid_k(\gamma_2)$ , if  $\gamma_1, \gamma_2 \in [\gamma]_k$  or  $\gamma_1 \rho_k \gamma_2$

We may write  $Valid_k(\gamma)$  to denote  $Valid_k([\gamma]_k)$ .

$$\begin{aligned} Valid_k(\gamma) &= Valid_k([\gamma]_k) \\ &= \{I \in R_k \mid I \in Valid_k(\delta), \delta \in [\gamma]_k\} \end{aligned}$$

$[\gamma]_k$ : denotes an **equivalent class** of

$\gamma$  (viable prefixes) under  $\rho_k$ .

may be **infinite** ( $[\gamma]_k \subseteq V^*$ )

$Valid_k(\gamma)$ : denotes a set of

$[A \rightarrow \alpha.\beta, x]$  (LR(k) items) under  $\rho_k$ .

always be **finite** ( $Valid_k(\gamma) \subseteq R_k$ )

$[\gamma_1]_k = [\gamma_2]_k$  iff  $Valid_k(\gamma_1) = Valid_k(\gamma_2)$ ,

bijection correspondence between

$[\gamma]_k$  and  $Valid_k(\gamma)$ .

We may write  $\langle \gamma \rangle_k$  instead of  $Valid_k(\gamma)$

Is it possible that  $\gamma = \delta$  implies  $[\gamma, x]_k = [\delta, y]_k$ ?

$$\text{Valid}_k(\varepsilon) = \{[S' \rightarrow \bullet S, \$^k], \\ [S \rightarrow \bullet aA, \$^k], [S \rightarrow \bullet bB, \$^k]\}$$

$$\text{Valid}_k(a) = \{[S \rightarrow a \bullet A, \$^k], \\ [A \rightarrow \bullet c, \$^k], [A \rightarrow \bullet dAd, \$^k]\}$$

$$\text{Valid}_k(aA) = \{[S \rightarrow aA \bullet, \$^k]\}$$

$$\text{Valid}_k(ad^{n+1}) = \{[A \rightarrow d \bullet Ad, k:d^n \$^k], \\ [A \rightarrow \bullet c, k:d^{n+1} \$^k], [A \rightarrow \bullet dAd, k:d^{n+1} \$^k]\}$$

$$\text{Valid}_k(ad^n c) = \{[A \rightarrow c \bullet, k:d^n \$^k]\}$$

$$\text{Valid}_k(ad^{n+1} A) = \{[A \rightarrow dA \bullet d, k:d^n \$^k]\}$$

$$\text{Valid}_k(ad^{n+1} Ad) = \{[A \rightarrow dAd \bullet, k:d^n \$^k]\}$$

...

$$1 + 2 \cdot (2 + 4(k+1)) = 8k + 13 \text{ (LR(k) states)}$$

$$[ad^+]_0 = [ad]_k \cup \dots \cup [ad^k]_k \cup [ad^{k+1} d^*]_k$$

$$[ad^+]_0 = [add^*]_0$$

$$= [ad]_1 \cup [add^*]_1$$

$$= [ad]_2 \cup [add]_2 \cup [adddd^*]_2$$

$$= [ad]_3 \cup [add]_3 \cup [addd]_3 \cup [addddd^*]_3$$

$$[ad^* c]_0 = [ac]_k \cup \dots \cup [ad^{k-1} c]_k \cup [ad^k cd^*]_k$$

$$[ad^+ A]_0 = [adA]_k \cup \dots \cup [ad^k A]_k \cup [ad^{k+1} d^* A]_k$$

$$[ad^+ Ad]_0 = [adAd]_k \cup \dots \cup [ad^k d]_k \cup [ad^{k+1} d^* d]_k$$

...

$[\gamma, \varepsilon]_{\rho_0}$  denotes the an equivalent class of  $\gamma$ ,  
under  $\rho_0$ .

$[\gamma]_{\rho_0} \equiv [\gamma]_0 \equiv [\gamma]$  denotes an equivalent class of  
valid prefixes under LR(0) equivalence.

$Valid_k(\langle \gamma, x \rangle)$  denotes an equivalent class of  
 $[A \rightarrow \alpha \cdot \beta, x]$  under  $\rho_k$ .

$Valid_k(\gamma)$  can denotes an equivalent class of  
valid LR(k)-items under LR(k) equivalence.

$\rho_k: V^* \times V^*, \iota_k: I \times I.$       equivalent relation

$[\gamma]_{\rho_k}: 2^{V^*}, [I]_{\iota_k}: 2^I.$       equivalent class

$Valid_k: V^* \rightarrow 2^I, \text{ or } 2^{V^*} \rightarrow 2^I.$

$Valid_k^{-1}: I \rightarrow 2^{V^*}, \text{ or } 2^I \rightarrow 2^{V^*}.$

$Valid_k(\{[\gamma]_{\rho_k}\}) = Valid_k^{-1}(\{[I]_{\iota_k}\})$

iff  $I \in Valid_k(\gamma)$  and/or  $\gamma \in Valid_k^{-1}(I)$

**Theorem 6.13** *The LR(k)-equivalence  $\rho_k$  for  $G$  is an equivalence relation on  $V^*$ ,  $\rho_k$  is a finite index, and the index of  $\rho_k$  is at most  $2^{|G| \cdot (|\Sigma| + 1)^k}$ .*

*One of the equivalent class under  $\rho_k$  is*

*$\{\gamma \mid \gamma \text{ is not a viable prefix of } G\}$ .*

**Proof.**

*As  $[\gamma_1]_{\rho_k} = [\gamma_2]_{\rho_k}$  iff  $\langle \gamma_1 \rangle_{\rho_k} = \langle \gamma_2 \rangle_{\rho_k}$*

*bijjective correspondence:  $[\gamma]_{\rho_k}$  and  $\langle \gamma \rangle_{\rho_k}$ .*

*$\therefore$  index of  $\rho_k$  = number of distinct sets  $\langle \gamma \rangle_{\rho_k}$*

*At most distinct  $|G|$  item cores in  $G$  and*

$$|\Sigma|^k + |\Sigma|^{k-1} + \dots + |\Sigma|^1 + 1 \leq (|\Sigma| + 1)^k.$$

*A string  $\gamma$  is a viable prefix iff  $[\gamma]_{\rho_k} \neq \emptyset$*

*$\therefore$  set of non viable prefixes forms a single equivalent class under  $\rho_k$ .*

**Lemma 6.14** *Let  $k \leq l$ . Then*

$$\langle \gamma \rangle_k = \{[A \rightarrow \alpha \bullet \beta, k:y] \mid [A \rightarrow \alpha \bullet \beta, y] \in \langle \gamma \rangle_l\}.$$

**Lemma 6.15** *Let  $k \leq l$ . Then LR(l)-equivalence is a refinement of LR(k)-equivalence. More specifically*

$$[\gamma]_k = \cup [\delta]_l.$$

$[\gamma]_k$  are **bijective** correspondence with  $\langle \gamma \rangle_k$ .

$\langle \gamma \rangle_k$ : finite representation of the class  $[\gamma]_k$

**collection** of all sets  $[\gamma]_k$

finite representation of the entire LR(k) equivalence

**canonical collection** of set of LR(k)-valid items for  $G$   
**canonical LR(k) collection** for  $G$ :  $C_k$

**canonical LR(k) machine**  $M$

(or **deterministic LR(k) machine**)

$$M = (C_k, V, \{[\gamma]_k \cdot X \rightarrow [\gamma \cdot X]_k\}, [\varepsilon]_k, \emptyset)$$

$\varepsilon$ -free, normal-form, completely specified, and deterministic fa

(1) **right-invariance** of the LR(k)-equivalence.

Since dfa, if  $[\gamma_1]_k = [\gamma_2]_k$ ,  $[\gamma_1 \cdot X]_k = [\gamma_2 \cdot X]_k$ .

(2)  $[\gamma]_k$  has a **unique entry symbol**.

Since  $[\gamma]_k \cdot X \rightarrow [\gamma \cdot X]_k \in P$ ,

if  $[\gamma_1] = [\gamma_2]$ ,  $\gamma_1 \cdot 1 = \gamma_2 \cdot 1$

$$[A \rightarrow \alpha \bullet B \beta, y] \partial_{LR(k)} [B \rightarrow \bullet \omega, z], z \in First_k(\beta y)$$

$$\partial_{LR(k)} \equiv \partial_k \equiv \partial.$$

$I_2$  is an **immediate LR(k)-descendant** of  $I_1$ , if  $I_1 \partial I_2$ .

$I_2$  is an **LR(k)-descendant** of  $I_1$ , if  $I_1 \partial^* I_2$ .

$I_1$  is an **(immediate) LR(k)-ancestor** of  $I_2$ ,

if  $I_2$  is an **(immediate) LR(k)-descendant** of  $I_1$ .

$[B \rightarrow \cdot \omega, z]$  is **immediate LR(k)-descendant** of  
 $[A \rightarrow \alpha \bullet B \beta, y]$ , if  $z \in First_k(\beta y)$

$$\langle \gamma \rangle_k^n = \{[A \rightarrow \alpha \cdot \beta, y] |$$

$$S \Rightarrow_{rm}^n \delta A z \Rightarrow_{rm} \delta \alpha \beta z (= \gamma \beta z), y = k:z\}$$

**Fact 6.16**  $\langle \gamma \rangle_k = \bigcup_{n=0}^{\infty} \langle \gamma \rangle_k^n = \langle \gamma \rangle_k^*$

**Lemma 6.17** If

$[A \rightarrow \alpha \bullet B \beta, y] \in \langle \gamma \rangle_k^n$  and  $\beta \Rightarrow^m v \in \Sigma^*$ . Then

$$[B \rightarrow \cdot \omega, k:vy] \in \langle \gamma \rangle_k^{n+m+1}(\gamma).$$

**Lemma 6.18**  $\langle \gamma \rangle_k$  is closed under  $\partial_k$ , i.e.,

$$\partial_k^*(\langle \gamma \rangle_k) = \langle \gamma \rangle_k.$$



**Lemma 6.19** *If*

$[B \rightarrow \bullet\omega, z] \in \langle \gamma \rangle_k^n$  *and*  $n > 0$ . *Then*

$$[A \rightarrow \alpha \bullet B \beta, y] \in \langle \gamma \rangle_k^m, \beta \Rightarrow^{n-m-1} \nu, k: \nu y = z.$$

**Fact 6.20**  $\langle \gamma \rangle_k^0 = \{[S \rightarrow \gamma \bullet \omega, \varepsilon] \mid S \rightarrow \gamma \omega \in P\}$

$[A \rightarrow \alpha \bullet \beta]$  *is LR-essential (or essential), if*  $\alpha \neq \varepsilon$  *inessential, otherwise.*

$$Ess_{LR}(q) = \{I \in q \mid I \text{ is LR-essential}\}.$$

**Lemma 6.21** *Let*  $I \in \langle \gamma \rangle_k^n$ ,  $k, n \geq 0$ .

(1)  $n = 0$ ,  $\gamma = \varepsilon$ ,  $I = [S \rightarrow \bullet\omega, \varepsilon]$ .

(2)  $\gamma \neq \varepsilon$  *and*  $I$  *is essential.*

(3)  $n > 0$ ,  $I$  *is inessential*

$$\text{and } \exists J, J \partial_k I, J \in \langle \gamma \rangle_k^m, m < n.$$

**Lemma 6.22**

$$\langle \varepsilon \rangle_k^n \subseteq \partial_k^* (\{[S \rightarrow \bullet\omega, \varepsilon] \mid S \rightarrow \omega \in P\})$$

$$\langle \gamma \rangle_k^n \subseteq \partial_k^* (Ess(\langle \gamma \rangle_k)), \text{ if } \gamma \neq \varepsilon.$$

**Lemma 6.23** (F.6.16, L6.18, and L6.22)

$$\langle \varepsilon \rangle_k = \partial_k^* (\{[S \rightarrow \bullet\omega, \varepsilon] \mid S \rightarrow \omega \in P\})$$

$$\langle \gamma \rangle_k = \partial_k^* (Ess(\langle \gamma \rangle_k)), \text{ if } \gamma \neq \varepsilon.$$

$\chi_k^X$ : relation on set of LR(k) items.

$$[A \rightarrow \alpha \bullet X \beta, y] \chi_k^X [A \rightarrow \alpha X \bullet \beta, y],$$

*pass-X, or  $\chi^X$  for short*

$$\begin{aligned} \text{Basis}_{LR}(q, X) &= \{[A \rightarrow \alpha X \bullet \beta, y] \mid [A \rightarrow \alpha \bullet X \beta, y] \in q\} \\ &\equiv \chi_k^X(q). \end{aligned}$$

$\delta_k^X$ : relation on set of LR(k) items.

$$\begin{aligned} \text{Goto}_{LR}(q, X) &= \partial_k^*(\text{Basis}_{LR}(q, X)) = \partial_k^*(\chi_k^X(q)) \\ &= \chi_k^X \cdot \partial_k^* \equiv \delta_k^X(q). \end{aligned}$$

*X-successor,  $\delta_k^X$  for short*

### Fact 6.24

If  $[A \rightarrow \alpha \bullet \omega \beta, y] \in \langle \gamma \rangle_k^n$ ,  $[A \rightarrow \alpha \omega \bullet \beta, y] \in \langle \gamma \omega \rangle_k^n$ .

If  $[A \rightarrow \alpha \omega \bullet \beta, y] \in \langle \gamma \rangle_k^n$ ,  $[A \rightarrow \alpha \bullet \omega \beta, y] \in \langle \delta \rangle_k^n$ ,  $\gamma = \delta \omega$ .

**Lemma 6.25**  $\text{Ess}(\langle \gamma X \rangle_k) = \text{Basis}(\langle \gamma \rangle_k, X)$

$$\text{Ess}(\langle \gamma X \rangle_k) = \chi_k^X(\langle \gamma \rangle_k).$$

**Lemma 6.26**  $\langle \gamma X \rangle_k = \text{Goto}(\langle \gamma \rangle_k, X)$

$$\langle \gamma X \rangle_k = \partial_k^*(\chi_k^X(\langle \gamma \rangle_k)) = \delta_k^X(\langle \gamma \rangle_k).$$

$$\delta_k^\varepsilon(q) = \partial_k^*(q)$$

$$\delta_k^{\gamma \cdot X}(q) = \partial_k^*(\chi_k^X(\delta_k^\gamma(q))), \quad \gamma \neq \varepsilon.$$

$$\begin{aligned} \therefore \delta_k^\gamma(q_s) &= \delta_k^{X_1}(\delta_k^{X_2}(\dots(\delta_k^{X_n}(\delta_k^\varepsilon(q_s)))\dots)) \\ &= \partial_k^*(\chi_k^{X_1}(\partial_k^*(\chi_k^{X_2}(\dots(\partial_k^*(\chi_k^{X_n}(\partial_k^*(q_s)))\dots))))). \end{aligned}$$

$$[\varepsilon]_k = \delta_k^\varepsilon(q_s) = \partial_k^*({S' \rightarrow \bullet S, \$^k})$$

$$[X]_k = \partial_k^*(\chi_k^X([\varepsilon]_k))$$

$$[\gamma X]_k = \partial_k^*(\chi_k^X([\gamma]_k))$$

**Algorithm** Compute  $M = (C_k, V, P, q_s, \emptyset)$

$$q_s := \partial_k^*([S' \rightarrow \bullet S, \$^k]);$$

$$C_k := \{q_s\};$$

$$P := \emptyset;$$

**repeat**

**for**  $q \in C_k$  **and**  $X \in V$  **do**

$$p := \partial_k^*(\chi_k^X(q));$$

$$C_k := C_k \cup \{p\};$$

$$P := P \cup \{q \cdot X \rightarrow p\}$$

**od**

**until** nothing is added to  $C_k$ .

**Lemma 6.27** Let  $M = (Q_M, V, P_M, q_s, F)$  be a canonical LR(k) machine for  $G = (V, \Sigma, P, S)$ . Then

(a)  $M$  is deterministic.

(b)  $q \in Q_M$ ,  $\text{Goto}(p, X) = q$ , unique  $X$ .

(c)  $q_s \mid \gamma \Rightarrow^* \Phi q \mid$ , iff  $q = \langle \gamma \rangle_k$

(d) If  $F = \{\langle \gamma \rangle_k\}$  for some  $\gamma$ ,  $L_M = [\gamma]_k$

(e) If  $F = \{\langle \gamma \rangle_k \mid \langle \gamma \rangle_k \neq \emptyset\}$ ,

$L_M = \text{Set of viable prefixes of } X.$

(f) If  $F = \{\langle \gamma \rangle_k\}$  for all  $\gamma$ ,  $L_M = V^*$ .

### Proof

Assume  $\langle \gamma_1 \rangle_k \cdot X \rightarrow \langle \gamma_1 X \rangle_k$  and

$\langle \gamma_2 \rangle_k \cdot X \rightarrow \langle \gamma_2 X \rangle_k$

where  $\langle \gamma_1 \rangle_k = \langle \gamma_2 \rangle_k$

Then  $\langle \gamma_1 X \rangle_k = \text{Goto}(\langle \gamma_1 \rangle_k, X)$

$= \text{Goto}(\langle \gamma_1 \rangle_k, X) = \langle \gamma_2 X \rangle_k$

$\therefore \langle \gamma_1 X \rangle_k = \langle \gamma_2 X \rangle_k \therefore M$  is deterministic. (a)

Assume  $\langle \gamma_1 \rangle_k \cdot X_1 \rightarrow \langle \gamma \rangle_k$  and  $\langle \gamma_2 \rangle_k \cdot X_1 \rightarrow \langle \gamma \rangle_k$ . Then

$\langle \gamma \rangle_k = \langle \gamma_1 X_1 \rangle_k = \partial_k^*(\text{Basis}(\langle \gamma_1 \rangle_k, X_1)),$

$= \langle \gamma_2 X_2 \rangle_k = \partial_k^*(\text{Basis}(\langle \gamma_2 \rangle_k, X_2)).$

$\therefore \text{Basis}(\langle \gamma_1 \rangle_k, X_1) = \text{Basis}(\langle \gamma_2 \rangle_k, X_2) \neq \emptyset.$

$\therefore X_1 = X_2.$  (b)

$[\varepsilon]_k \cdot \gamma_1 \gamma_2 \Rightarrow^* [\gamma_1]_k \gamma_2$  in  $M$ .

Since  $M$  is deterministic,  $[\gamma]_k$  is the only state. (c)

### **Theorem 6.28**

(a) The LR(k) equivalence of  $G$  is the equivalence induced by the canonical LR(k) machine of  $G$ .

$$[\gamma_1]_{\rho_k} = [\gamma_2]_{\rho_k},$$

iff  $q_s \mid \gamma_1 \Rightarrow^* q_s \dots q \mid$  and  $q_s \mid \gamma_2 \Rightarrow^* q_s \dots q \mid$ .

(b) The LR(k) equivalence of  $G$  is **right invariance**.

$$\text{If } [\gamma_1]_{\rho_k} = [\gamma_2]_{\rho_k}, [\gamma_1 \cdot X]_{\rho_k} = [\gamma_2 \cdot X]_{\rho_k}.$$

(c) The LR(k) equivalence of  $G$  is **ends with same symbols**.

$$\text{If } [\gamma_1]_{\rho_k} = [\gamma_2]_{\rho_k}, \gamma_1 \cdot 1 = \gamma_2 \cdot 1.$$

$$[\gamma]_k = \delta_M^\gamma(q_s) = \delta_M^\gamma([\varepsilon]_k) \stackrel{?}{=} (\partial_k^* \chi_k^\gamma)^* (\partial_k^*([\varepsilon]_k)).$$

$$[\varepsilon]_k = \partial_k^*([S' \rightarrow \bullet S, \$^k]).$$

$$[\gamma \cdot X]_k = \partial_k^*(\chi_k^X([\gamma]_k)).$$

$$\equiv \delta_k^X([\gamma]_k).$$

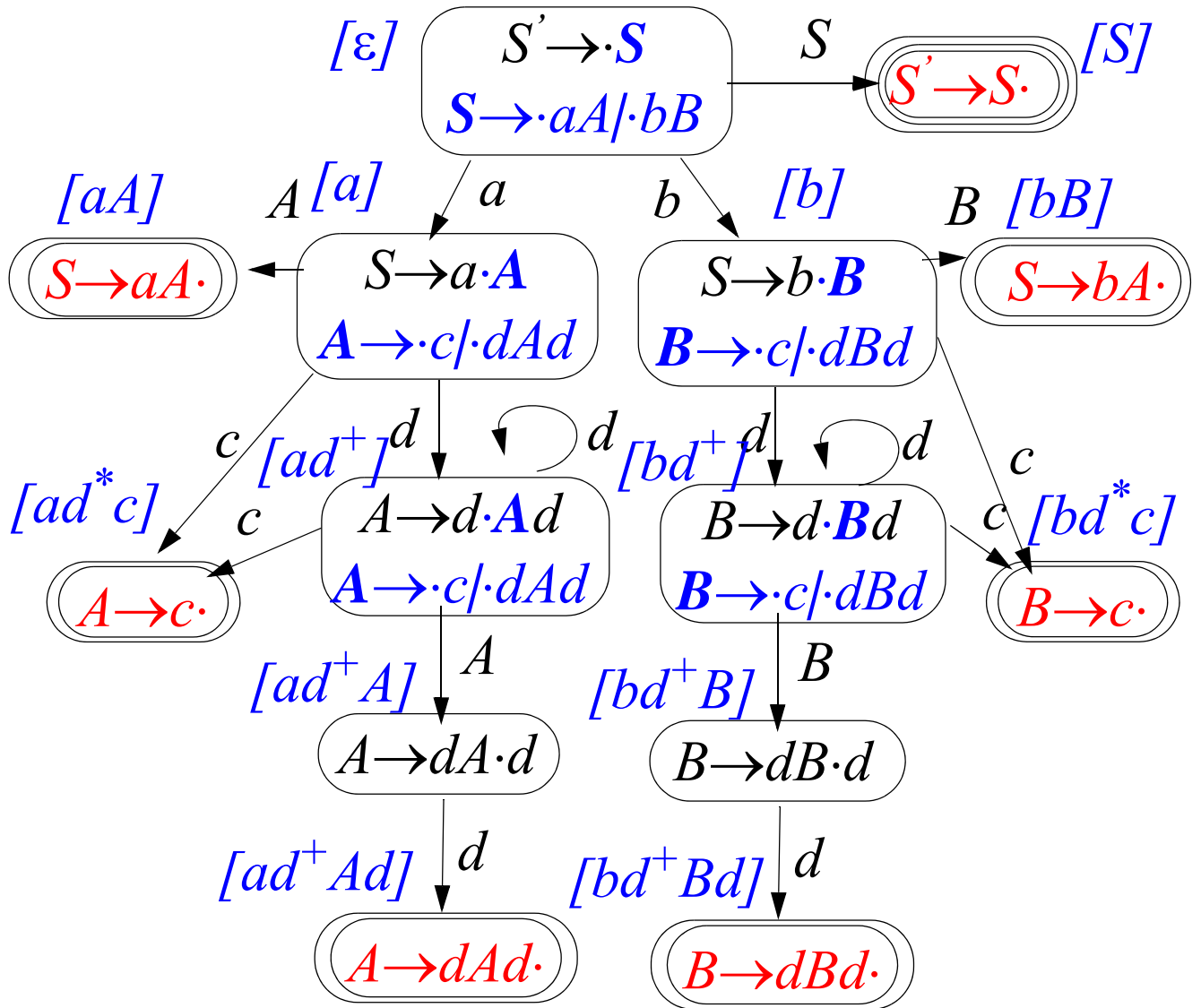
$$\therefore [\gamma]_k = \delta_k^\gamma([\varepsilon]_k).$$

$$\delta_k^\varepsilon(q) = \{q\}.$$

$$\delta_k^{\gamma \cdot X}(q) = \partial_k^*(\chi_k^X(\delta_k^\gamma(q))).$$

$$[\varepsilon]_k = \partial_k^*([S' \rightarrow \bullet S, \$^k]).$$

$$[\gamma \cdot X]_k = \delta_k^X(\delta_k^\gamma(q)).$$



$\$[\epsilon] \mid addcdd\$$

LR(0) Parser

$\Rightarrow^a \$[\epsilon][a] \mid ddcdd\$$

$\Rightarrow^d \$[\epsilon][a][ad] \mid dcdd\$$

$\Rightarrow^d \$[\epsilon][a][ad][add] \mid cdd\$$

$\Rightarrow^c \$[\epsilon][a][ad][add][addc] \mid dd\$$

$\Rightarrow^{A \rightarrow c} \$[\epsilon][a][ad][add][addA] \mid dd\$$

$\Rightarrow^d \$[\epsilon][a][ad][add][addA][addAd] \mid d\$$

$\Rightarrow^{A \rightarrow dAd} \$[\epsilon][a][ad][addA] \mid d\$$

$\Rightarrow^d \$[\epsilon][a][ad][addA][addAd] \mid \$$

$\Rightarrow^{A \rightarrow dAd} \$[\epsilon][a][aA] \mid \$$

$\Rightarrow^{S \rightarrow aA} \$[\epsilon][S] \mid \$$  accept "addcdd"!

$\$ \mid addcdd\$$  Right Parser

$\Rightarrow^a \$a \mid ddcdd\$$

$\Rightarrow^d \$ad \mid dcdd\$$

$\Rightarrow^d \$add \mid cdd\$$

$\Rightarrow^c \$addc \mid dd\$$

$\Rightarrow^{A \rightarrow c} \$addA \mid dd\$$

$\Rightarrow^d \$addAd \mid d\$$

$\Rightarrow^{A \rightarrow dAd} \$adA \mid d\$$

$\Rightarrow^d \$adAd \mid \$$

$\Rightarrow^{A \rightarrow dAd} \$aA \mid \$$

$\Rightarrow^{S \rightarrow aA} \$S \mid \$$

### 6.3 Canonical LR(k) Parser

Let  $G = (N, \Sigma, P, S)$ . The **canonical LR(k) parser** for  $G$  is a pushdown transducer  $M = ([G]_k, \Sigma, \Gamma, P, \tau, [\varepsilon]_k, \{[\varepsilon]_k[S]_k\}, \$, |)$  where

$$[G]_k = \{[\delta]_k \mid \delta \in V^*\}$$

$$\Gamma = \{[\delta]_k[\delta X_1]_k \dots [\delta X_1 \dots X_n]_k \mid y \rightarrow [\delta]_k[\delta A]_k \mid y \\ \mid [A \rightarrow X_1 \dots X_n \bullet, y] \in \langle \delta X_1 \dots X_n \rangle_k\} \quad (ra)$$

$$\cup \{[\delta]_k \mid ay \rightarrow [\delta]_k[\delta a]_k \mid y \\ \mid a \in \Sigma, [A \rightarrow \alpha \bullet a \beta, z] \in \langle \delta \rangle_k \\ y \in First_{\max\{k-1, 0\}}(\beta z)\} \quad (sa)$$

$$\tau([\delta]_k[\delta X_1]_k \dots [\delta X_1 \dots X_n]_k \mid y \rightarrow [\delta]_k[\delta A]_k \mid y) \\ = A \rightarrow X_1 \dots X_n, \\ \tau([\delta]_k \mid ay \rightarrow [\delta]_k[\delta a]_k \mid y) = \varepsilon.$$

$$[B \rightarrow \alpha \bullet A \beta, x] \in \langle \delta \rangle_k \quad [B \rightarrow \alpha A \bullet \beta, x] \in \langle \delta A \rangle_k$$

$$[A \rightarrow \bullet X_1 \dots X_n, y] \in \langle \delta \rangle_k$$

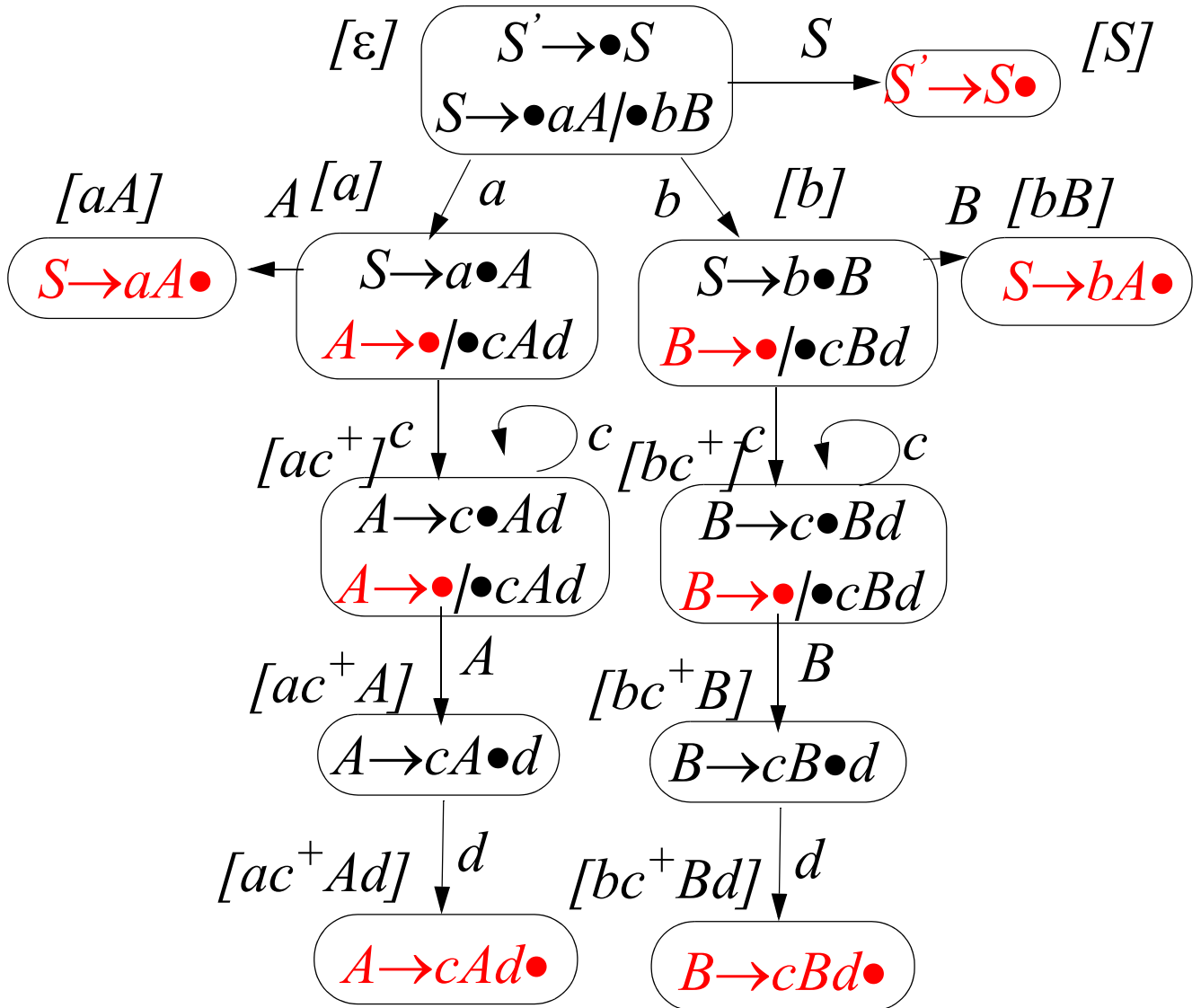
$$[A \rightarrow X_1 \bullet X_2 \dots X_n, y] \in \langle \delta X_1 \rangle_k$$

...

$$[A \rightarrow X_1 \dots X_n \bullet, y] \in \langle \delta X_1 \dots X_n \rangle_k$$



$G_{ab\epsilon}$ :  $S \rightarrow aA \mid bB \quad A \rightarrow \epsilon \mid cAd \quad B \rightarrow \epsilon \mid cBd$



$G_{ab\epsilon}$  is not LR(0): shift-reduce conflicts

in states  $[a]$  and  $[ac^+]$ ,

**shift** to  $[ac^+]$  for  $c \in \Sigma$  or **reduce**  $A \rightarrow \epsilon$  for  $c \in \Sigma$

in states  $[b]$  and  $[bc^+]$

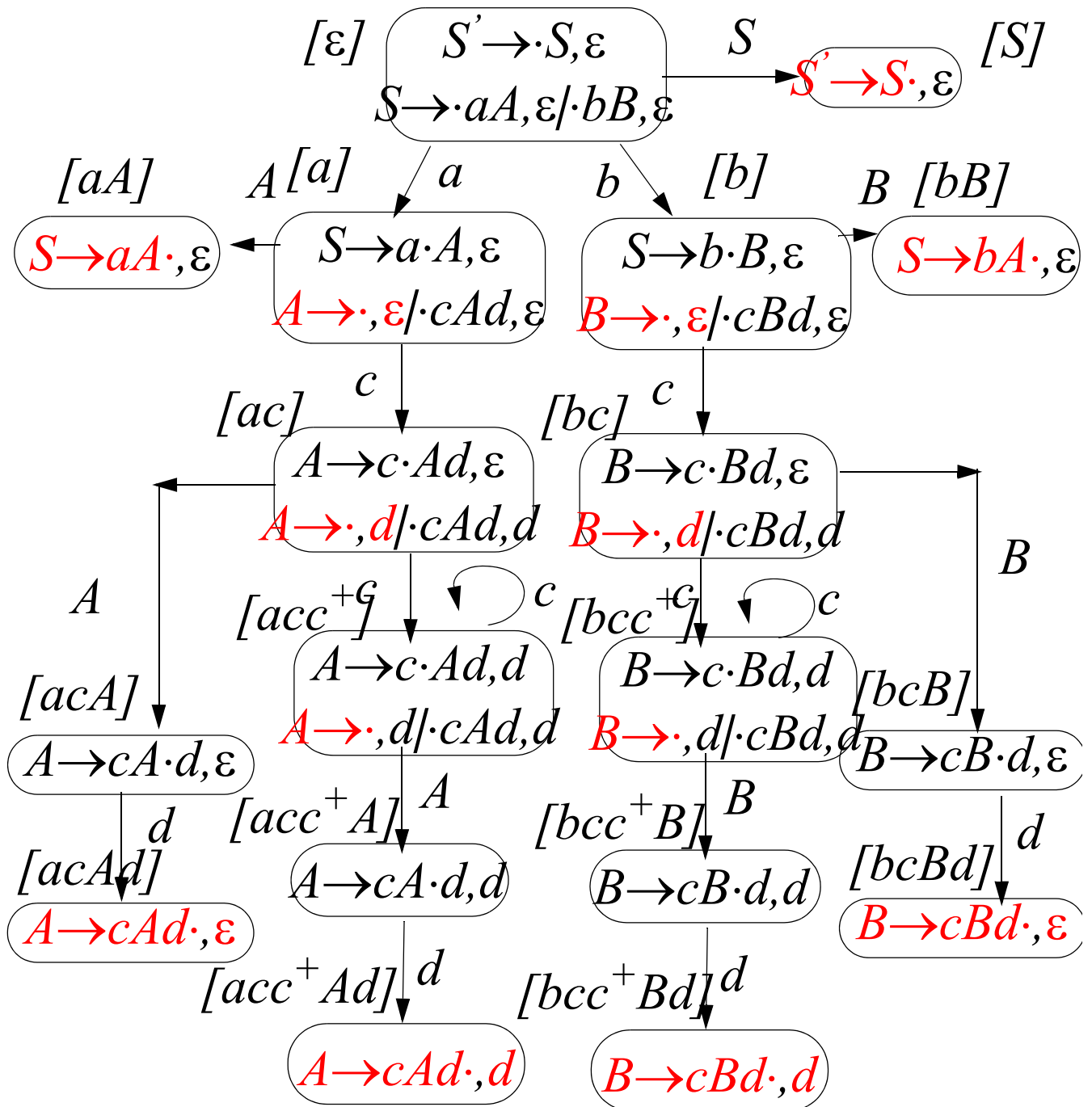
**shift** to  $[bc^+]$  for  $c \in \Sigma$  or **reduce**  $B \rightarrow \epsilon$  for  $c \in \Sigma$

Since they have both shiftable item  $A \rightarrow \bullet cAd$  and

reducible item  $A \rightarrow \epsilon \bullet$  and

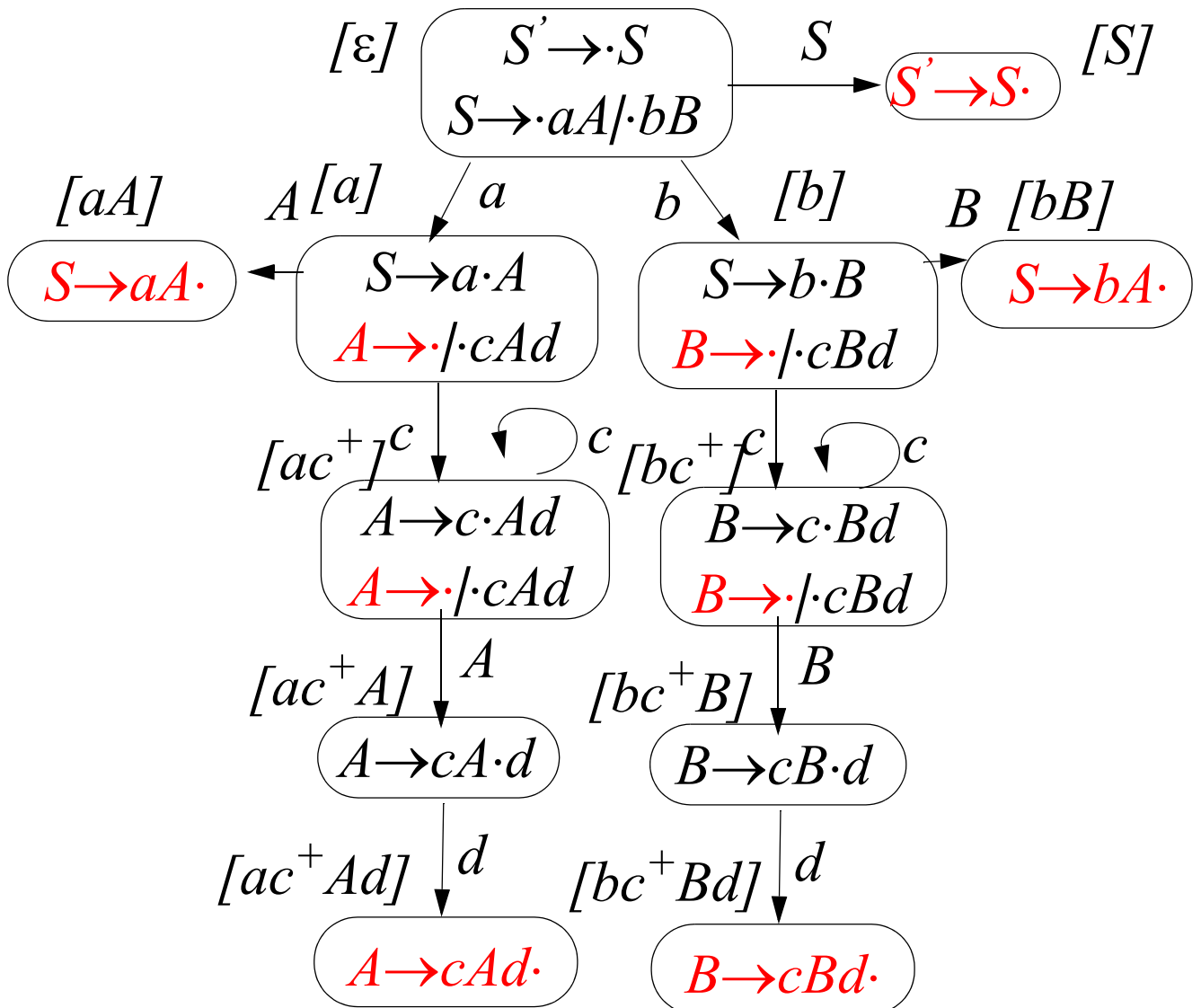
in the state  $[a]$  and  $[ac^+]$  and ...

LR(1) states for  $G_{ab\varepsilon}$ .



$[a]$ : reduce  $A \rightarrow \varepsilon$  for  $\varepsilon \in \Sigma^{\leq 1}$  shift to  $[ac]$  for  $c \in \Sigma$ .  
 $[ac]$ : reduce  $A \rightarrow \varepsilon$  for  $d \in \Sigma^{\leq 1}$  shift to  $[acc^+]$  for  $c \in \Sigma$ .  
 $[acc^+]$ : reduce  $A \rightarrow \varepsilon$  for  $d \in \Sigma^{\leq 1}$  shift  $[acc^+]$  for  $c \in \Sigma$ .  
 $\therefore G_{ab\varepsilon}$  is LR(1).

LR(0) states for  $G_{ab\varepsilon}$ .



SLR(1) Parser  $\equiv$  LR(0) states Follow(A) Lookahead

[a] reduce  $A \rightarrow \varepsilon$  for  $\varepsilon, d \in \Sigma^{\leq 1}$  shift to  $[ac^+]$  for  $c \in \Sigma$

$[ac^+]$  reduce  $A \rightarrow \varepsilon$  for  $\varepsilon, d \in \Sigma^{\leq 1}$  shift to  $[ac^+]$  for  $c \in \Sigma$

LALR(1) Parser  $\equiv$  LR(0) states LR(1) Lookahead

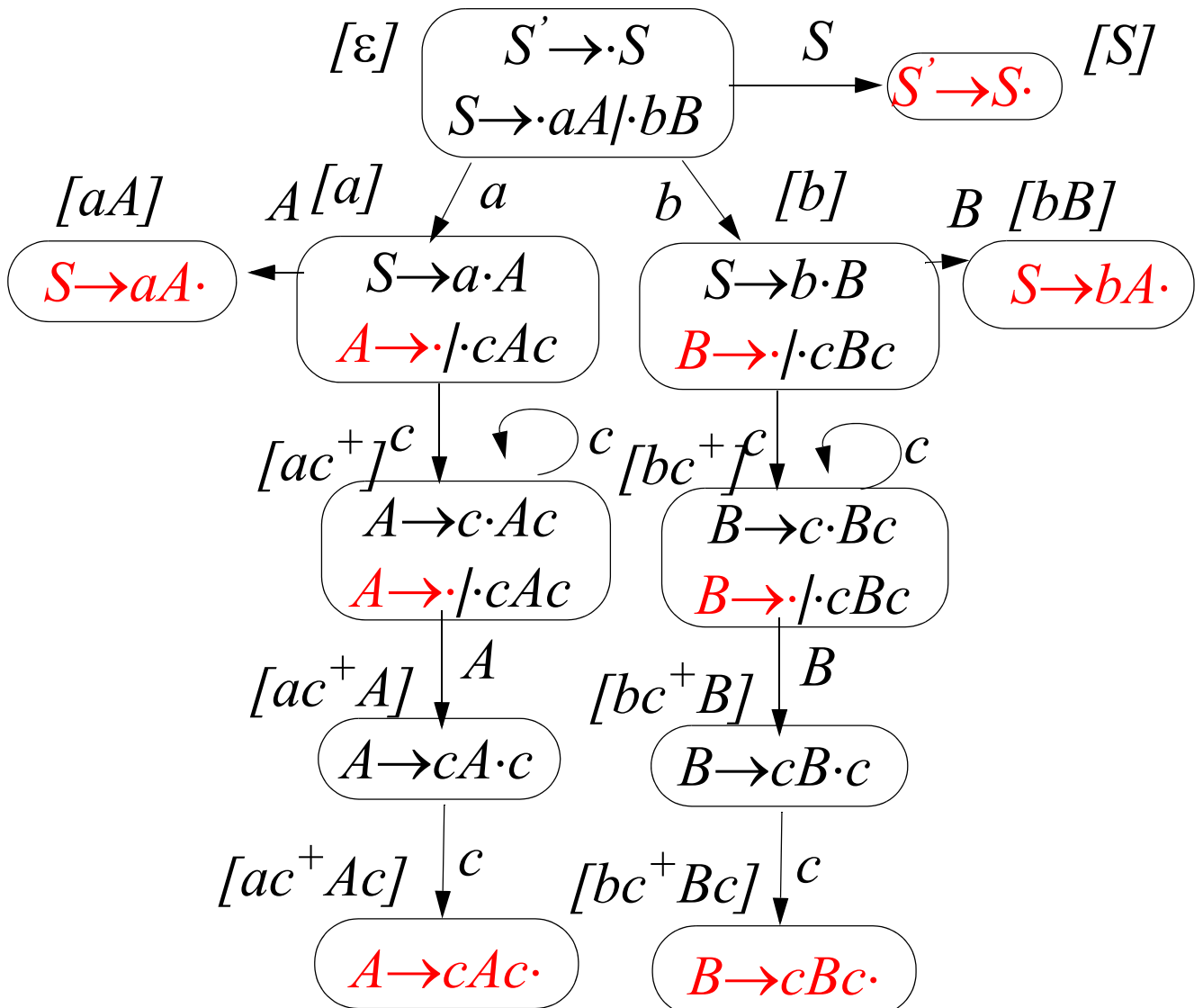
[a] reduce  $A \rightarrow \varepsilon$  for  $\varepsilon \in \Sigma^{\leq 1}$  shift to  $[ac^+]$  for  $c \in \Sigma$

$[ac^+]$  reduce  $A \rightarrow \varepsilon$  for  $d \in \Sigma^{\leq 1}$  shift to  $[ac^+]$  for  $c \in \Sigma$

$G_{ab\varepsilon}$  is both SLR(1) and LALR(1).

Consider grammar that is **not** SLR(1) but LALR(1)?

$G_{abc\varepsilon}: S \rightarrow aA \mid bB \quad A \rightarrow \varepsilon \mid cAc \quad B \rightarrow \varepsilon \mid cBc$



SLR(1) Parser  $\equiv$  LR(0) states Follow<sub>1</sub>(A) Lookahead

[a] reduce  $A \rightarrow \varepsilon$  for  $\varepsilon, c \in \Sigma^{\leq 1}$  shift to  $[ac^+]$  for  $c \in \Sigma$

$[ac^+]$  reduce  $A \rightarrow \varepsilon$  for  $\varepsilon, c \in \Sigma^{\leq 1}$  shift to  $[ac^+]$  for  $c \in \Sigma$

LALR(1) Parser  $\equiv$  LR(0) states LR(1) Lookahead

[a] reduce  $A \rightarrow \varepsilon$  for  $\varepsilon \in \Sigma^{\leq 1}$  shift to  $[ac^+]$  for  $c \in \Sigma$

$[ac^+]$  reduce  $A \rightarrow \varepsilon$  for  $\varepsilon, c \in \Sigma^{\leq 1}$  shift to  $[ac^+]$  for  $c \in \Sigma$

$G_{abc\varepsilon}$  is **neither** SLR(1) **nor** LALR(1).

Consider grammar that is **not** SLR(1) **but** LALR(1)!

## Canonical LR(k) Parser right parser

LR(k) parser  $\Leftrightarrow$  right parser

**T6.34** (**T5.21**: s/r par., **T5.65**: simple-prec. par)

LR(k) parser  $\Rightarrow$  right parser

**L6.29, L6.30** (**L5.17, 5.18** and **L5.60, L5.61**)

homomorphism  $h$ :

action in LR(k) parser

$\rightarrow$  action in shift-reduce parser

$$h([\delta]_k[\delta X_1]_k \dots [\delta X_1 \dots X_n]_k \mid y \rightarrow [\delta]_k[\delta A]_k \mid y)$$

$$= X_1 \dots X_n \mid \rightarrow A \mid,$$

$$h([\delta]_k \mid a \rightarrow [\delta]_k[\delta a]_k \mid) = \mid a \rightarrow a \mid.$$

Furthermore,  $h$ :

configuration in LR(k) parser

$\rightarrow$  configuration in shift-reduce parser

$$h([\varepsilon]_k[X_1]_k \dots [X_1 \dots X_n]_k \mid w\$^k) = \$X_1 \dots X_n \mid w\$^k.$$

$$\therefore [\varepsilon]_k[X_1]_k \dots [X_1 \dots X_n]_k \mid x\$^k$$

$$\Rightarrow^\theta [\varepsilon]_k[Y_1]_k \dots [Y_1 \dots Y_m]_k \mid y\$^k \text{ implies}$$

$$\$X_1 \dots X_n \mid x\$^k \Rightarrow^{h(\theta)} \$Y_1 \dots Y_m \mid y\$^k.$$

LR(k) parser  $\Leftarrow$  right parser

**L6.31, L6.32, L6.33**

(**L5.19, 5.20** and **L5.63, L5.64**)

**Lemma 6.29** Let  $M$  be a LR(k) parser for  $G$ . If

$$[\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_n]_k \mid xy \$^k$$

$$\Rightarrow^\theta [\varepsilon]_k [X_1]_k \dots [X_1 \dots X_m]_k \mid y \$, \theta \in \Gamma^* \text{ in } M,$$

then

$$X_1 \dots X_m \Rightarrow_{rm}^{\tau(\theta)^R} Y_1 \dots Y_n x \text{ in } G,$$

$$\text{and } |\theta| = |\tau(\theta)| + |x|.$$

**Proof** Induction on the length of action string  $\theta$ .

i)  $\theta = \varepsilon$ .  $x = \varepsilon$ ,  $Y_1 \dots Y_n = X_1 \dots X_m$ , and  $\tau(\varepsilon) = \varepsilon$ .

ii)  $\theta = r\theta'$ .

ii.1)  $r$  is reduce by  $A \rightarrow Y_p \dots Y_n$ ,  $1 \leq p \leq n$ .

$$[\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_n]_k \mid xy \$^k$$

$$\Rightarrow^r [\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_{p-1}]_k [Y_1 \dots Y_{p-1} A]_k \mid xy \$^k$$

$$\Rightarrow^{\theta'} [\varepsilon]_k [X_1]_k \dots [X_1 \dots X_m]_k \mid y \$^k.$$

$$X_1 \dots X_m \Rightarrow_{rm}^{\tau(\theta')^R} Y_1 \dots Y_{p-1} A x \Rightarrow_{rm}^{A \rightarrow \omega} Y_1 \dots Y_n x,$$

and

$$|\theta'| = |\tau(\theta')| + |x|.$$

$$\therefore X_1 \dots X_m \Rightarrow_{rm}^{(\tau(\theta') \cdot A \rightarrow \omega)^R} Y_1 \dots Y_n x, \text{ and}$$

$$|\theta| = 1 + |\theta'| = 1 + |\tau(\theta')| + |x| = |\tau(\theta)| + |x|.$$

ii.2)  $r = [\delta]_k \mid a \rightarrow [\delta]_k [\delta a]_k \mid \in \Gamma$ ,

$$[\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_n]_k \mid xy \$$$

$$= [\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_n]_k \mid ax' y \$$$

$$\Rightarrow^r [\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_n]_k [Y_1 \dots Y_n a]_k \mid x' y \$$$

$$\Rightarrow^{\theta'} [\varepsilon]_k [X_1]_k \dots [X_1 \dots X_m]_k \mid y \$$$

and  $|\theta'| = |\tau(\theta')| + |x'|.$

$$X_1 \dots X_m ax'y \Rightarrow_{rm}^{\tau(\theta')R} Y_1 \dots Y_n ax'y$$

$$\therefore \delta \Rightarrow_{rm}^{\tau(\theta') \cdot \varepsilon} \gamma x \text{ in } G, \text{ and}$$

$$|\theta| = 1 + |\theta'| = |\tau(\theta')| + 1 + |x'| = |\tau(\theta)| + |x|$$

**Lemma 6.30** *Let  $M$  be a canonical LR(k) parser for  $G$ . Then*

(1)  $L(M) \subseteq L(G),$

(2)  $\forall \theta$ : actions in  $M$ ,  $\tau(\theta)$  is a **right parse** of  $w$ ,

(3)  $Time_G(w) \leq Time_M(w) - |w|.$

**Lemma 6.31** Let  $M$  be a LR(k) parser.

If  $[A \rightarrow \alpha \cdot \beta, z] \in \langle \gamma a_1 \dots a_n \rangle_k$  and  $k:y\$^k \in \text{First}_k(\beta z)$ ,

then  $[\varepsilon]_k \dots [\gamma]_k \mid a_1 \dots a_n y \$^k$

$\Rightarrow^\theta [\varepsilon]_k \dots [\gamma]_k [\gamma a_1]_k \dots [\gamma a_1 \dots a_n]_k \mid y \$^k$

where  $\theta$  is a sequence of shift actions.

valid  $k$ -item  $\Rightarrow$  valid stack string

**Proof**

$S' \Rightarrow^* \delta A z' \Rightarrow \delta \alpha \beta z' = \gamma a_1 \dots a_n \beta z'$ , and

$k:z' = z. \quad \forall i, 1 \leq i \leq n,$

(i) if  $\alpha = \alpha' a_i \dots a_n$ ,

$[A \rightarrow \alpha' a_i \dots a_n \cdot \beta, z] \in [\gamma a_1 \dots a_{i-1}]_k$

(ii) if  $\delta = \gamma a_1 \dots a_{j-1} A z'$ . By lemma 6.2,

$S' \Rightarrow^+ \delta' A' y' \Rightarrow \delta' \alpha'' a_i \beta' y' = \gamma a_1 \dots a_i \beta' y'$ , and

$\beta' y' \Rightarrow^* a_{i+1} \dots a_{j-1} A z'$ .

$\therefore [A' \rightarrow \alpha'' \cdot a_i \beta, k:y'] \in \langle \gamma a_1 \dots a_{i-1} \rangle_k$ ,

and  $\beta' y' \Rightarrow^* a_{i+1} \dots a_{j-1} A z'$

$\Rightarrow^* a_{i+1} \dots a_n \beta z'$ .



**Lemma 6.32** Let  $M$  be a LR(k) parser.

If  $X_1 \dots X_m \Rightarrow_{rm}^{\pi R} Y_1 \dots Y_n x$  in  $G$ ,

$[A \rightarrow \alpha \bullet \beta, z] \in \langle X_1 \dots X_m \rangle_k$

$k: y \$^k \in \text{First}_k(\beta z)$ , and

either  $Y_1 \dots Y_n = \varepsilon$  or  $Y_n \in N$ .

Then  $\tau(\theta) = \pi$ ,  $|\theta| = |\pi| + |x|$ , and

$[\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_n]_k \mid xy \$^k$

$\Rightarrow^\theta [\varepsilon]_k [X_1]_k \dots [X_1 \dots X_m]_k \mid y \$^k$ ,  $\theta \in \Gamma^*$  in  $M$ .

**Proof** Induction on  $|\pi|$ .

i)  $\pi = \varepsilon$ .  $X_1 \dots X_m = Y_1 \dots Y_n x$ .

$[\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_n]_k \mid xy \$^k$

$\Rightarrow^\theta [\varepsilon]_k [Y_1]_k \dots [Y_1 \dots Y_n]_k [Y_1 \dots Y_n 1 : x]_k \dots$   
 $[Y_1 \dots Y_n x]_k \mid y \$$ ,

$\theta$  is a  $|x|$ -length shift action string (**L6.31**)

$|\theta| = |x|$ ,  $\tau(\theta) = \pi = \varepsilon$ .

ii)  $\pi = B \rightarrow \omega \cdot \pi_1$ .

$X_1 \dots X_m \Rightarrow_{rm}^{\pi_1 R} Z_1 \dots Z_p B x_1$

$\Rightarrow_{rm}^r Z_1 \dots Z_p \omega x_1 = Y_1 \dots Y_n x$  in  $G$ .

where  $x = vx_1$ .

$\exists \theta_1 . \exists . \tau(\theta_1) = \varepsilon_1$ ,  $|\theta_1| = |\pi_1| + |x_1|$ , and

$[\varepsilon] [Z_1] \dots [Z_1 \dots Z_p] [Z_1 \dots Z_p B] \mid x_1 y \$$

$\Rightarrow^{\theta_1} [\varepsilon] [X_1] \dots [X_1 \dots X_m] \mid y \$$ .

And because  $[A \rightarrow \alpha \bullet \beta, z] \in \langle X_1 \dots X_m \rangle_k$

$\exists \delta, z' . \exists$ .

$$S' \xrightarrow{rm}^* \delta A z' \xrightarrow{rm} \delta \alpha \beta z' = X_1 \dots X_m \beta z' \\ \xrightarrow{rm} X_1 \dots X_m u z' \text{ in } G',$$

and  $k:z' = z, k:uz = k:y\$$ .

$$\therefore S' \xrightarrow{rm}^* Z_1 \dots Z_p B_1 u z' \xrightarrow{rm} Z_1 \dots Z_p \omega x_1 u z' \text{ in } G'.$$

Here  $k:x_1 u z' = k:x_1 u z = k:x_1 y \$$ , so

$$[B \rightarrow \omega \cdot, k:x_1 y \$] \in \langle Z_1 \dots Z_p \omega \rangle_k$$

Then  $[\varepsilon][Y_1] \dots [Y_1 \dots Y_n] \mid xy \$$

$$= [\varepsilon][Y_1] \dots [Y_1 \dots Y_n] \mid vx_1 y \$$$

$$\xrightarrow{rm}^{\theta_2} [\varepsilon][Y_1] \dots [Y_1 \dots Y_n v] \mid x_1 y \$$$

$$= [\varepsilon][Z_1] \dots [Z_1 \dots Z_p] \dots [Z_1 \dots Z_p \omega] \mid x_1 y \$ \text{ in } M,$$

for some  $|v|$ -length shift action string  $\theta_2$ .

Then  $\exists$  an action  $r' . \exists$ .

$$r' = [Z_1 \dots Z_p] \dots [Z_1 \dots Z_p \omega] \mid y' \rightarrow$$

$$[Z_1 \dots Z_p] \dots [Z_1 \dots Z_p B] \mid y',$$

where  $y' = k:x_1 y \$$ .

$$\therefore [\varepsilon][Y_1] \dots [Y_1 \dots Y_n] \mid xy \$$$

$$\xrightarrow{rm}^{\theta_1} [\varepsilon][X_1] \dots [X_1 \dots X_m] \mid y \$ \text{ in } M,$$

where  $\theta = \theta_2 r' \theta_1$ .

And  $\tau(\theta) = \tau(\theta_2) \tau(r') \tau(\theta_1) = r \pi_1 = \pi$ ,

$$|\theta| = |\pi| + |x|.$$

**Lemma 6.33** *Let  $M$  be a canonical LR(k) parser for  $G$ . Then*

- (1)  $L(G) \subseteq L(M)$ ,
- (2)  $\forall \pi$ : *right parse* of  $w$  in  $G$ ,  $\tau(\theta) = \pi$  in  $M$ ,
- (3)  $\text{Time}_G(w) \leq \text{Time}_M(w) + |w|$ .

**Proof**

$$Y_1 \dots Y_m = \varepsilon, X_1 \dots X_m = S,$$

$$[A \rightarrow \alpha \bullet \beta, z] = [S' \rightarrow \bullet S, \$^l], y = \varepsilon.$$

**Theorem 6.34** *Let  $M$  be a canonical LR(k) parser for  $G$ . Then*

- (1)  $M$  is a *right parser* for  $G$ .
- (2)  $\forall w \in L(G)$ ,  $M$  produces *all right parses* of  $w$ .
- (3)  $\text{Time}_M(w) = \text{Time}_G(w) + |w|$ .

## 6.4 LR(k) Grammar

$G$  is **LR(k) grammar** if its canonical LR(k) parser is **deterministic** and  $S \Rightarrow^+ S$  in  $G$ .

**Theorem 6.35** Any LR(k) grammar is **unambiguous**.

“**reduce-reduce conflicts**”

$$[A_1 \rightarrow \omega_1 \bullet, y_1], [A_2 \rightarrow \omega_2 \bullet, y_2],$$

if  $y_1 = y_2$ ,  $A_1 \rightarrow \omega_1 \neq A_2 \rightarrow \omega_2$ .

“**shift-reduce conflicts**”

$$[A \rightarrow \alpha \bullet a \beta, z], [B \rightarrow \omega \bullet, y]$$

if  $y \in \text{First}_k(a\beta z)$ .

**Lemma 6.36** Let  $M$  be the CLR(k) parser for  $G$ .

Then  $M$  is nondeterministic iff

$\exists$  a state which contains a pair of items exhibiting a reduce-reduce or shift-reduce conflict.

**Proof**

( $\Leftarrow$ )  $I_1, I_2 \in \langle \gamma \rangle_k$  and either

(i)  $I_1 = [A \rightarrow X_1 \dots X_n \bullet, y]$ ,  $I_2 = [B \rightarrow Y_1 \dots Y_m \bullet, y]$ , or

(ii)  $I_1 = [A \rightarrow X_1 \dots X_n \bullet a \beta, z]$ ,  $I_2 = [B \rightarrow Y_1 \dots Y_m \bullet, y]$ ,  
 $y \in \text{First}_k(a\beta z)$ .

Assume  $m \leq n$ , then for  $[\delta][\delta X_1] \dots [\delta X_1 \dots X_n] | y$ ,

(i)  $[\delta][\delta X_1] \dots [\delta X_1 \dots X_n] | y \rightarrow [\delta][\delta A] | y$  and

$$[\delta X_1 \dots X_{i-1}] [\delta X_1 \dots X_i] \dots [\delta X_1 \dots X_n] | y$$

$$\rightarrow [\delta X_1 \dots X_{i-1}] [\delta X_1 \dots X_{i-1} B] | y \text{ in } M,$$

$$Y_1 \dots Y_m = X_i \dots X_n (i = n - m + 1; \mathbf{L6.24})$$

(ii)  $[\delta X_1 \dots X_n] | ay' \rightarrow [\delta X_1 \dots X_n] [\delta X_1 \dots X_n a] | y'$

$$[\delta X_1 \dots X_{i-1}] [\delta X_1 \dots X_i] \dots [\delta X_1 \dots X_n] | y$$

$$\rightarrow [\delta X_1 \dots X_{i-1}] [\delta X_1 \dots X_{i-1} B] | y, k: ay' = y \text{ in } M.$$

$\therefore$  parser is nondeterministic.

( $\Rightarrow$ ) let  $r_1, r_2$  are conflicting actions in  $[\gamma]$  of  $M$ , then

(i)  $(r_1) [\gamma] | ay \rightarrow [\gamma] [\gamma a] | y,$

$(r_2) [\gamma] | ayv \rightarrow [\gamma] [\gamma a] | yv,$

where  $[A_1 \rightarrow \alpha_1 \bullet a \beta_1, z_1], [A_2 \rightarrow \alpha_2 \bullet a \beta_2, z_2] \in \langle \gamma \rangle_k$

$y \in \text{First}_{\max\{k-1, 0\}}(\beta_1 z_1), yv \in \text{First}_{\max\{k-1, 0\}}(\beta_2 z_2).$

Here  $\beta_1, \beta_2 \not\Rightarrow * x \$$ , and if  $y \neq yv$ , then  $y:1 = \$$ ,

$\therefore v = \varepsilon. \therefore$  no shift-shift conflict.

(ii)  $(r_1) [\delta] \dots [\delta X_1 \dots X_n] | y \rightarrow [\delta] [\delta A] | y,$

$(r_2) [\gamma] \dots [\gamma Y_1 \dots Y_m] | y \rightarrow [\gamma] [\gamma B] | y,$

then  $[\delta X_1 \dots X_n] = [\gamma Y_1 \dots Y_m]$ , and

$[A \rightarrow X_1 \dots X_n \bullet, y] \in \langle \delta X_1 \dots X_n \rangle_k$  and

$[B \rightarrow Y_1 \dots Y_m \bullet, y] \in \langle \gamma Y_1 \dots Y_m \rangle_k$

$\therefore$  reduce-reduce conflict.

**Lemma 6.37**  $\langle \gamma \rangle_k$  contains a pair of items exhibiting a reduce-reduce conflict iff

- (a)  $S' \Rightarrow^* \delta_1 A_1 y_1 \Rightarrow \underline{\delta}_1 \underline{\omega}_1 y_1 = \gamma y_1$ ,
- (b)  $S' \Rightarrow^* \delta_2 A_2 y_2 \Rightarrow \underline{\delta}_2 \underline{\omega}_2 y_2 = \gamma y_2$ ,
- (c)  $k:y_1 = k:y_2$ , and
- (d)  $A_1 \rightarrow \omega_1 \neq A_2 \rightarrow \omega_2$

hold in  $G'$ .

**Proof**  $[A \rightarrow \omega_1 \bullet, k:y_1], [A \rightarrow \omega_2 \bullet, k:y_2] \in \langle \gamma \rangle_k$

**Lemma 6.38**  $\langle \gamma \rangle_k$  contains a pair of items exhibiting a shift-reduce conflict iff

- (a)  $S' \Rightarrow^* \delta_1 A_1 y_1 \Rightarrow \underline{\delta}_1 \underline{\omega}_1 y_1 = \gamma y_1$ ,
- (b)  $S' \Rightarrow^* \delta_2 A_2 y_2 \Rightarrow \underline{\delta}_2 \underline{\omega}_2 y_2 = \gamma v y_2$ ,
- (c)  $k:y_1 = k:vy_2$ ,  $v \neq \varepsilon$

hold in  $G'$ .

**Proof**

$(\Leftarrow)$   $[A \rightarrow \alpha \bullet a \beta, z]$  and  $[B \rightarrow \omega \bullet, y]$  are in  $\langle \gamma \rangle_k$

$(\Rightarrow)$   $\omega_2 = \alpha v$ ,  $[A_2 \rightarrow \alpha \cdot v, k:y_2]$ , or

$$v = az\omega_2, S' \Rightarrow^* \delta_2 A_2 y_2 = \gamma az A_2 y_2.$$

By lemma 6.2,  $\exists A' \rightarrow \alpha'' \cdot a \beta' \in P$ ,

$[A' \rightarrow \alpha'' \bullet a \beta', k:y'] \in \langle \gamma \rangle_k$

$$a\beta'y' \Rightarrow^* az A_2 y_2 \Rightarrow az\omega_2 y_2 = v y_2.$$

By (c), there exists a shift reduce conflict.

**Lemma 6.39** *The following statements are logically equivalent for all  $G$  and  $k \geq 0$ .*

(a) *The canonical LR(k) parser of  $G$  is **deterministic**.*

(b) *In the canonical LR(k) machine of the  $\$$ -augmented grammar  $G'$  no states contains a pair of items exhibiting a **reduce-reduce** or **shift-reduce** conflicts.*

(c) *The conditions*

$$S' \Rightarrow^* \delta_1 A_1 y_1 \Rightarrow \underline{\delta}_1 \underline{\omega}_1 y_1 = \forall y_1,$$

$$S' \Rightarrow^* \delta_2 A_2 y_2 \Rightarrow \underline{\delta}_2 \underline{\omega}_2 y_2 = \forall v y_2,$$

$$\text{and } k:y_1 = k:vy_2, v \neq \varepsilon$$

*always implies that*

$$\delta_1 = \delta_2, A_1 = A_2, \text{ and } \omega_1 = \omega_2.$$

**Theorem 6.40** *For all  $k \geq 0$ , the class of LR(k) grammars is **properly contained** in the class of LR(k+1) grammars.*

**Proposition 6.41** *Any pushdown automaton  $M$  with input alphabet  $\Sigma$  can be transformed into an equivalent grammar  $G$  with terminal alphabet  $\Sigma$  such that  $M$  is **deterministic** if and only if  $G$  is **LR(k)** for some  $k \geq 0$ .*

*LR(k) languages = LR(1) languages*

*deterministic languages = LR(1) languages*

**Lemma 6.42** Let  $G$  be LR(k) grammar and  $M$  be a LR(k) parser for  $G$ .

Further let  $x, y \in \Sigma^*$  and

$$\psi \in [G']^* .\exists. [\varepsilon] | xy\$ \Rightarrow^* \psi | y\$.$$

If  $\forall y' .\exists. \text{ the condition } xy' \in L(G), k:y \neq k:y',$   
then  $\psi|y\$$  is an error configuration.

**Proof by contradiction**

$$\psi = [\varepsilon][X_1] \dots [X_1 \dots X_n], X_1 \dots X_n \Rightarrow^* x.$$

If  $\psi|y\$$  were not an error configuration, then

$$[A \rightarrow \alpha \cdot \beta, z] \in \langle X_1 \dots X_n \rangle_k$$

$$k:y\$ \in \text{First}_k(\beta z).$$

$$S' \Rightarrow^* \delta A z' \$ \Rightarrow \delta \alpha \beta z' \$ = X_1 \dots X_n \beta z' \$,$$

$k:z' \$ = z, \beta \Rightarrow^* v,$  then  $k:y\$ = k:vz = k:vz' = k:y',$   
and  $xvz' \in L(G),$  a contradiction.



**Lemma 6.43** Let  $G$  be a LR(k) grammar and  $M$  be a LR(k) parser for  $G$ ,  $k \geq 1$ .

Then  $M$  detects an error in any input string in  $\Sigma^* \setminus L(G)$ .

**Proof**

(i)  $k:w \neq k:w'$ , for all  $w' \in L(G)$ , by lemma 6.42.

(ii)  $k:w = k:w'$  for some  $w' \in L(G)$ .

Then  $\exists x, y, y' . \exists$ .

(a)  $w = xy$ ,

(b)  $k:y = k:y'$  and  $xy' \in L(G)$ .

(c)  $\forall y'' \in \Sigma^*$ ,  $xy'' \in L(G)$  implies  $k+1:y \neq k+1:y''$ .

Let  $y = ay_1$ ,  $y' = ay_1'$ , and there exist  $\psi, \psi' . \exists$ .

$[\epsilon] | xay_1 \$ \Rightarrow \psi | ay_1 \$ \Rightarrow \psi' | y_1 \$$  in  $M$ .

Then

$[A \rightarrow \alpha \bullet a \beta, z] \in \langle X_1 \dots X_n \rangle_k$

if  $\psi = [\epsilon] \dots [X_1 \dots X_n]$ , where  $k:ay_1' \in \text{First}(a\beta z)$ .

By lemma 6.32,

$$\begin{aligned} [\epsilon] | xay_1 \$ &\Rightarrow^* [\epsilon] [X_1] \dots [X_1 \dots X_n] | ay_1 \$ \\ &= \psi | ay_1 \$ \Rightarrow^* \psi' | y_1 \$ \end{aligned}$$

By (c),  $xay_1'' \in L(G)$  always implies  $k:y_1 \neq k:y_1''$ .

$\therefore$  By lemma 6.42,  $\psi | y_1 \$$  is an error configuration.

The parser loops forever when

(1)  $S \rightarrow S|a$  : LR(0) parser is deterministic,

(2)  $S \rightarrow a^{k+1}|ASb^k$ ,  $A \rightarrow \varepsilon$  : not LR(k).

**Theorem 6.44** Let  $k \geq 0$ . Then  $M$  does not loop forever on any input string.

**Proof** Assume that  $M$  loops forever for  $w = xy \in \Sigma^*$ .

Then  $\exists \psi_i, r_i \cdot \exists$ .

$$[\varepsilon]|w\$ \Rightarrow^* \psi_1|y$,$$

$$\psi_i|y\$ \Rightarrow^{r_i} \psi_{i+1}|y\$ \text{ in } M \forall i \geq 1.$$

Let  $\psi_i = [\varepsilon] \dots [\gamma_i]$ ,  $\gamma_1 \Rightarrow^* x$ ,  $\gamma_{i+1} \Rightarrow_{rm} \gamma_i$ .

Let  $r_i$  be reduce action by  $A_i \rightarrow \omega_i$  and

$$[A_i \rightarrow \omega_i, k:z_i\$] \in \langle \gamma_i \rangle \forall i.$$

$$[\varepsilon]|xz_i\$ \Rightarrow^* \psi_1|z_i\$ \Rightarrow \psi_2|z_i$,$$

and more generally,

$$\psi_i|z_i\$ \Rightarrow^{r_n} \psi_{n+1}|z_i$.$$

$M$  loops forever on all  $xz_i$ ,  $i \geq 1$ .

But  $z_i = \varepsilon$  because  $xz_i \in L(M) \forall i$  and

$M$  is deterministic. Then

$$S \Rightarrow^* \delta_{n+1} A_{n+1} \Rightarrow \delta_{n+1} w_{n+1} = \gamma_{n+1} \Rightarrow^n \gamma_1 \Rightarrow^* x.$$

$G$  is ambiguous, a contradiction.

## 6.5 LALR(k) parsing

**Theorem 6.45** *The size of the canonical LR(k) parser for grammar  $G$  is  $O(2^{(|\Sigma|+1)^k|G|} + k \log|\Sigma| + \log|G|)$ .*

**Proof**

$2^{(|\Sigma|+1)^k|G|}$  : # of distinct LR(k)-equivalent classes.

$2^{(|\Sigma|+1)^k|G|} \cdot |G| \cdot (|\Sigma|+1)^k$

: sum of the lengths of all reduce actions

*Whether does the grammar exist with this upper bound?*

$k=0$

**Proposition 6.46** *For each  $n \geq 0$ , let  $G_n = (\{A_0, A_1, \dots, A_n\}, \{0, 1, a, a_0, a_1, \dots, a_n\}, P, A_0)$  where  $P$  is*

$$A_i \rightarrow 1A_{i+1}a_i, \quad 0 \leq i \leq n-1$$

$$A_n \rightarrow 1A_0a_n$$

$$A_i \rightarrow 0A_i a_i, \quad 1 \leq i \leq n$$

$$A_i \rightarrow 0A_0 a_i, \quad 1 \leq i \leq n$$

$$A_0 \rightarrow a.$$

*Then the size of the canonical LR(0) collection for  $G_n$  is at least  $2^{c|G_n|}$  for all  $n \geq 0$ ,  $c > 0$ .*

Let  $G = (N, \Sigma, P, S)$  be a cfg. Then a rule automaton,

$$M = (Q, N \cup \Sigma, R, \langle S', \varepsilon \rangle, Q)$$

$$Q = \{ \langle A, \alpha \rangle \mid A \rightarrow \alpha\beta \in P \} \cup \{ \langle S', \varepsilon \rangle, \langle S', S \rangle \}$$

$$R = \{ \langle S', \varepsilon \rangle S \rightarrow \langle S', S \rangle \}$$

$$\cup \{ \langle S', \varepsilon \rangle \rightarrow \langle B, \varepsilon \rangle \mid S \rightarrow B\beta \in P \}$$

$$\cup \{ \langle A, \alpha \rangle X \rightarrow \langle A, \alpha X \rangle \mid A \rightarrow \alpha X\beta \in P \}$$

$$\cup \{ \langle A, \alpha \rangle \rightarrow \langle B, \varepsilon \rangle \mid A \rightarrow \alpha B\beta \in P \}$$

$M$  is a dfa but  $\varepsilon$ -moves.

$$M' = (K, N \cup \Sigma, R, \langle S', \varepsilon \rangle, K)$$

$$K = \{ \langle A, \alpha \rangle \mid A \rightarrow \alpha\beta \in P, \alpha \neq \varepsilon \} \cup \{ \langle S', \varepsilon \rangle, \langle S', S \rangle \}$$

$$R' = \{ \langle S', \varepsilon \rangle S \rightarrow \langle S', S \rangle \}$$

$$\cup \{ \langle S', \varepsilon \rangle \rightarrow \langle B, X \rangle \mid S \rightarrow B\beta \in P, B \rightarrow X\gamma \in P \}$$

$$\cup \{ \langle A, \alpha \rangle X \rightarrow \langle A, \alpha X \rangle \mid A \rightarrow \alpha X\beta \in P, \alpha \neq \varepsilon \}$$

$$\cup \{ \langle A, \alpha \rangle X \rightarrow \langle B, X \rangle \mid A \rightarrow \alpha B\beta \in P, \alpha \neq \varepsilon, \\ B \rightarrow X\gamma \in P \}$$

$$|K| = |Q| - |N|$$

$$R' = \{ \langle S', \varepsilon \rangle S \rightarrow \langle S', S \rangle \}$$

$$\cup \{ \langle S', \varepsilon \rangle \rightarrow \langle B, X \rangle \mid S \rightarrow B\beta \in P, B \rightarrow X\gamma \in P \}$$

$$\cup \{ \langle A, \alpha \rangle a \rightarrow \langle A, \alpha a \rangle \mid A \rightarrow \alpha a\beta \in P, \alpha \neq \varepsilon \}$$

$$\cup \{ \langle A, \alpha \rangle B \rightarrow \langle A, \alpha B \rangle, \langle A, \alpha \rangle X \rightarrow \langle B, X \rangle \\ \mid A \rightarrow \alpha B\beta \in P, \alpha \neq \varepsilon, B \rightarrow X\gamma \in P \}$$

Let  $G = (N, \Sigma, P, S)$  be a cfg. Then LR(k) automaton,

$$M_k = (Q, N \cup \Sigma, R, \langle S', \varepsilon, \$^k \rangle, Q)$$

$$Q = \{ \langle A, \alpha, x \rangle \mid A \rightarrow \alpha\beta \in P, x \in \text{Follow}_k(A) \}$$

$$\cup \{ \langle S', \varepsilon, \$^k \rangle, \langle S', S, \$^k \rangle \}$$

$$R = \{ \langle S', \varepsilon, \$^k \rangle S \rightarrow \langle S', S, \$^k \rangle \}$$

$$\cup \{ \langle S', \varepsilon, \$^k \rangle \rightarrow \langle B, \varepsilon, y \rangle \mid$$

$$S \rightarrow B\beta \in P, y \in \text{First}_k(\beta \$^k) \}$$

$$\cup \{ \langle A, \alpha, x \rangle X \rightarrow \langle A, \alpha X, x \rangle \mid A \rightarrow \alpha X\beta \in P \}$$

$$\cup \{ \langle A, \alpha, x \rangle \rightarrow \langle B, \varepsilon, y \rangle \mid$$

$$A \rightarrow \alpha B\beta \in P, y \in \text{First}_k(\beta x) \}$$

$M$  is a dfa but  $\varepsilon$ -moves.

$$M' = (K, N \cup \Sigma, R, \langle S', \varepsilon, \$^k \rangle, K)$$

$$K = \{ \langle A, \alpha, x \rangle \mid A \rightarrow \alpha\beta \in P, \alpha \neq \varepsilon, x \in \text{Follow}_k(A) \}$$

$$\cup \{ \langle S', \varepsilon, \$^k \rangle, \langle S', S, \$^k \rangle \}$$

$$R' = \{ \langle S', \varepsilon, \$^k \rangle S \rightarrow \langle S', S, \$^k \rangle \}$$

$$\cup \{ \langle S', \varepsilon, \$^k \rangle \rightarrow \langle B, X, y \rangle \mid$$

$$S \rightarrow B\beta \in P, B \rightarrow X\gamma \in P, y \in \text{First}_k(\beta \$^k) \}$$

$$\cup \{ \langle A, \alpha, x \rangle X \rightarrow \langle A, \alpha X, x \rangle \mid A \rightarrow \alpha X\beta \in P, \alpha \neq \varepsilon \}$$

$$\cup \{ \langle A, \alpha, x \rangle X \rightarrow \langle B, X, y \rangle \mid A \rightarrow \alpha B\beta \in P, \alpha \neq \varepsilon,$$

$$B \rightarrow X\gamma \in P, y \in \text{First}_k(\beta x) \}$$

$$|K| = |Q| - |N|$$

$$\begin{aligned}
R' = & \{ \langle S', \varepsilon \rangle S \rightarrow \langle S', S \rangle \} \\
& \cup \{ \langle S', \varepsilon \rangle \rightarrow \langle B, X \rangle \mid S \rightarrow B\beta \in P, B \rightarrow X\gamma \in P \} \\
& \cup \{ \langle A, \alpha \rangle a \rightarrow \langle A, \alpha a \rangle \mid A \rightarrow \alpha a\beta \in P, \alpha \neq \varepsilon \} \\
& \cup \{ \langle A, \alpha \rangle B \rightarrow \langle A, \alpha B \rangle, \langle A, \alpha \rangle X \rightarrow \langle B, X \rangle \\
& \quad \mid A \rightarrow \alpha B\beta \in P, \alpha \neq \varepsilon, B \rightarrow X\gamma \in P \}
\end{aligned}$$

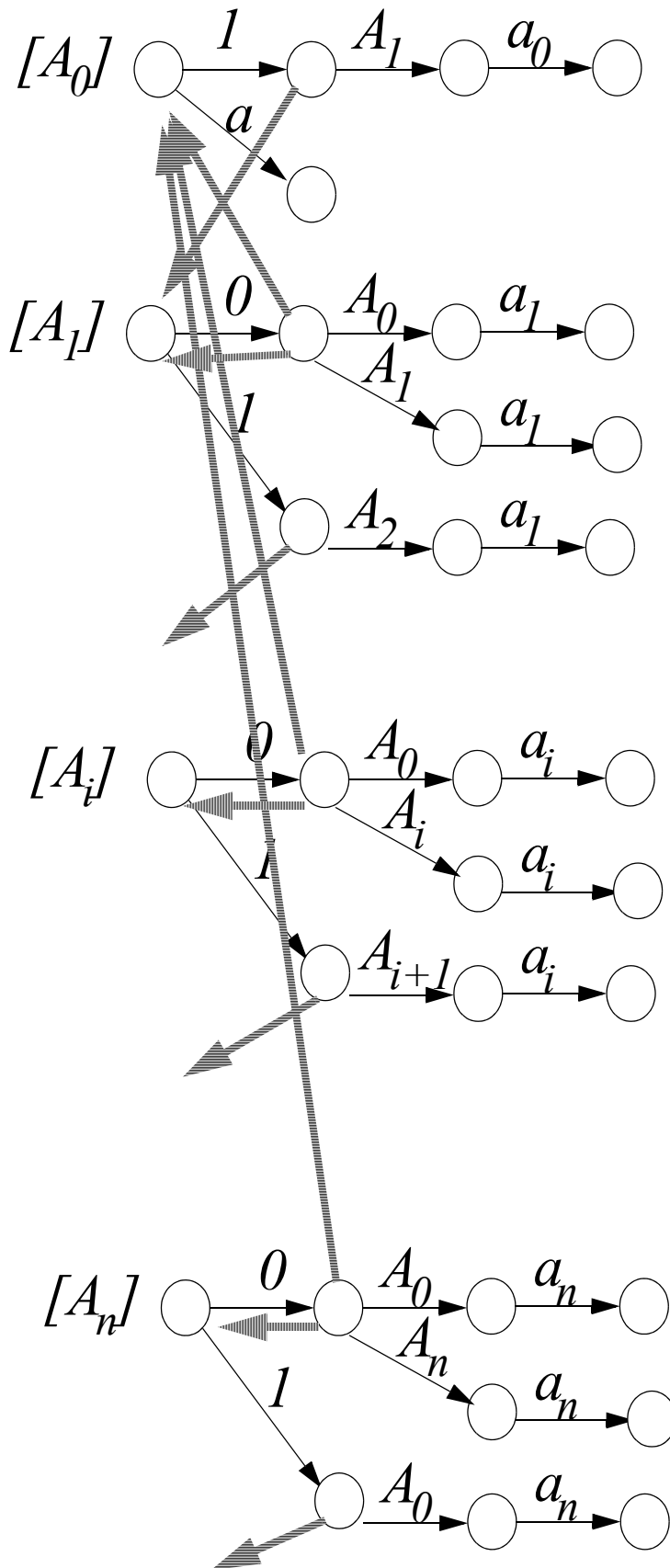
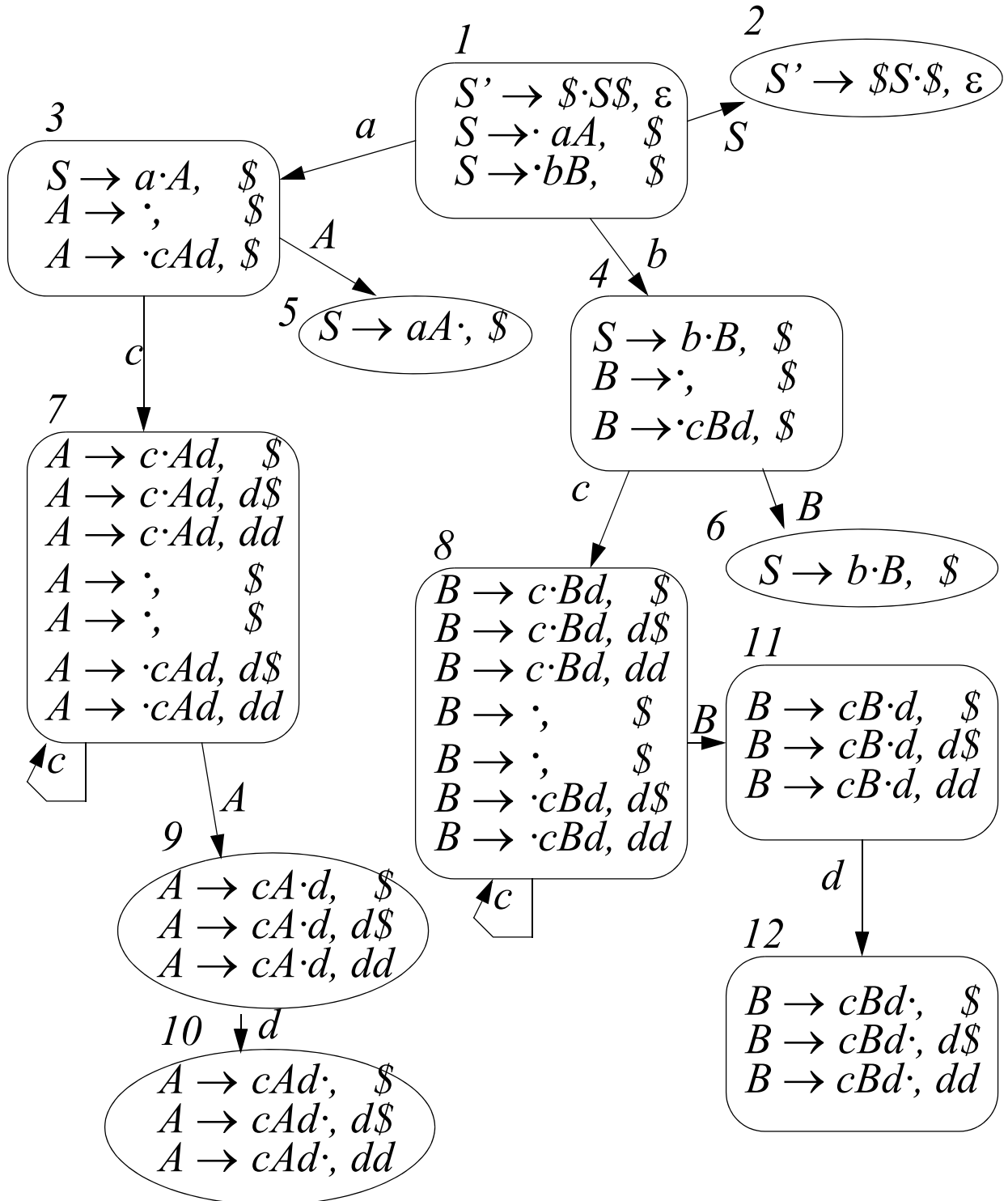


Fig 6.15  $G_{ab\varepsilon}: S \rightarrow aA|bB, A \rightarrow \varepsilon|cAd, B \rightarrow \varepsilon|cBd$





the partition of the LR(0)-equivalent classes into LR( $k$ ) equivalent classes are:

$$[]_0 = []_k$$

$$[S]_0 = [S]_k$$

$$[a]_0 = [a]_k$$

$$[a]_0 = [a]_k$$

$$[aA]_0 = [aA]_k$$

$$[ac^+]_0 = [ac]_k \cup \dots \cup [ac^k]_k \cup [ac^k c^+]_k$$

$$[ac^+ A]_0 = [acA]_k \cup \dots \cup [ac^k A]_k \cup [ac^k c^+ A]_k$$

$$[ac^+ Ad]_0 = [acAd]_k \cup \dots \cup [ac^k Ad]_k \cup [ac^k c^+ Ad]_k$$

$$[b]_0 = [b]_k$$

$$[bB]_0 = [bB]_k$$

$$[bc^+]_0 = [bc]_k \cup \dots \cup [bc^k]_k \cup [bdc^k c^+]_k$$

$$[bc^+ B]_0 = [bc]_k \cup \dots \cup [bc^k]_k \cup [bdc^k c^+]_k$$

$$[bc^+ Bd]_0 = [bcd]_k \cup \dots \cup [bc^k d]_k \cup [bdc^k c^+ d]_k$$

$$6k + 12 \quad \text{linear in } k$$

**Theorem 6.47**

Let  $q$  be the state of LALR( $k$ ) machine for  $G$ , and  $q$  is accessible upon reading string  $\delta$ .

Then  $\delta$  is a viable prefix of  $G$ , and

$\exists \gamma \in V^* . \exists . \forall I \in q, I \in \langle \gamma \rangle_k$  where  $\gamma \rho_0 \delta$ .

Conversely,

If  $I \in \langle \gamma \rangle_k$

Then  $\exists q$  s.t.  $I \in q$  and

$q$  is accessible upon reading any viable prefix  $\delta$

$. \exists . \delta \rho_0 \gamma$ .

LR( $k$ ) states:  $\langle [ \gamma ]_k \rangle_k = \langle \gamma \rangle_k \leftrightarrow [ \gamma ]_k$   
 $= \{ [ A \rightarrow \alpha \bullet \beta, x ] \in \langle \delta \rangle_k \mid \delta \in [ \gamma ]_k \}$

LR(0) states:  $\langle [ \gamma ]_0 \rangle_0 = \langle \gamma \rangle_k \leftrightarrow [ \gamma ]_0$   
 $= \{ [ A \rightarrow \alpha \bullet \beta ] \in \langle \delta \rangle_0 \mid \delta \in [ \gamma ]_0 \}$

LALR( $k$ ) states:  $\langle [ \gamma ]_0 \rangle_k = \langle \gamma \rangle_{k, 0} \leftrightarrow [ \gamma ]_0$   
 $= \{ [ A \rightarrow \alpha \bullet \beta, x ] \in \langle \delta \rangle_k \mid \delta \in [ \gamma ]_0 \}$

Since  $[ \gamma ]_k \subseteq [ \gamma ]_0$

$\langle \gamma \rangle_k \subseteq \langle \gamma \rangle_{k, 0}$

$core(\langle \gamma \rangle_k) = core(\langle \gamma \rangle_{k, 0}) = core(\langle \gamma \rangle_0)$

Let  $G = (N, \Sigma, P, S)$ . The **LALR(k) parser** for  $G$  is a pushdown transducer  $M = ([G]_0, \Sigma, \Gamma, P, \tau, [\varepsilon]_0, \{[\varepsilon]_0[S]_0\}, \$, \mid)$  where

$$\Gamma = \{[\delta]_0[\delta X_1]_0 \dots [\delta X_1 \dots X_n]_0 \mid y \rightarrow [\delta]_0[\delta A]_0 \mid y$$

$$\mid [A \rightarrow X_1 \dots X_n \bullet, y] \in \langle [\delta X_1 \dots X_n]_0 \rangle_k\}$$

(ra)

$$\cup \{[\delta]_0 \mid ay \rightarrow [\delta]_0[\delta a]_0 \mid y$$

$$\mid a \in \Sigma, [A \rightarrow \alpha \bullet a \beta, z] \in \langle [\gamma]_0 \rangle_k,$$

$$y \in \text{First}_{\max\{k-1, 0\}}(\beta z)\}$$

(sa)

**Theorem 6.48** The size of LALR(k) parser for  $G$  is  $O(2^{|G|} + k \log |\Sigma| + \log |G|)$ .

*Correctness of LALR(k) parser as a right parser.*

**L6.49:** *right parser  $\Rightarrow$  LALR(k) parser*

*LR(0) parser(L6.29)*

**L6.50:** *LALR(k) parser  $\Rightarrow$  right parser*

*LR(k) parser(L6.32)*

**Theorem 6.51** *For the LALR(k) parser  $M$  for  $G$ ,*

*(1)  $M$  is a right parser for  $G$*

*(2)  $\forall w \in L(G)$ ,  $M$  produces all right parses of  $w$  in  $G$*

*(3)  $TIME_G(w) = TIME_M(w) + |w|$ .*

***making LALR(k) parser***

*from LR(k) parser  $\Rightarrow$  uniting all states with the same set of item cores*

*from LR(0) parser  $\Rightarrow$  add suitable  $k$ -length lookahead strings to 0-items*

*LALR(k) lookahead set is sufficient and **minimal***

**Theorem 6.52**

*Let  $[A \rightarrow \alpha \bullet \beta, z] \in q$ . Then  $\exists x, y$  and  $X_1 \dots X_m \cdot \exists$ .*

*$[\epsilon] | xy \Rightarrow^* [\epsilon] [X_1] \dots [X_1 \dots X_m] | y \$$  in  $M$ ,*

*where the set of cores in  $\langle X_1 \dots X_m \rangle_0$  is same as in  $q$  and  $k:y \$ = \text{First}_k(\beta z)$ .*

*In CLR(k) parser*

*every item  $[A \rightarrow \alpha \bullet \beta, z]$  in any state  $q$*

*can be "used" in the parsing of*

*all terminal strings of the form  $xy$ ,*

*where  $k:y \in \text{First}_k(\beta z)$  and  $q = \langle \$ \gamma \rangle_0$ .*

*sentence            LALR(k) is same as LR(k)*

*no sentence        additional **reduce** actions  
in LALR(k)*

**Immediate Error Detection Property in LR(k)**

*reduce stack for error recovery in LALR(k)*

$G = (N, \Sigma, P, S)$  is **LALR(k)** if

*its LALR(k) parser is deterministic and*

*$S \Rightarrow^+ S$  is impossible in  $G$ .*

*A language over alphabet  $\Sigma$  is LALR(k) if*

*it is generated by an LALR(k) grammar.*

**Theorem 6.53** (Characterization of LALR(k) Grammars) Let  $G'$  be an augmented grammar.

The LALR(k) parser of  $G$  is deterministic iff  
in the LALR(k) machine of  $G'$

no state contains a pair of items  
exhibiting a reduce-reduce or  
a shift-reduce conflict.

**Theorem 6.54** The class of LALR(0) grammars coincides with the class of LR(0) grammars. For  $k \geq 1$  the class of LALR(k) grammars is properly contained in the class of LR(k) grammars.

**Proof**

(i)  $LALR(k) \subseteq LR(k)$ :

uniting of states in CLR(k) machine  
can only increase # of reduce-reduce conflicts.

(ii)  $LALR(k) \neq LR(k)$

counter example:

$$S \rightarrow aAa|bAb|aBb|bBa,$$

$$A \rightarrow c,$$

$$B \rightarrow c.$$

This grammar is LR(1)

but not LALR(k) for any  $k$ .

Generalize the LALR concepts:

**LA(k)LR(l) machine**

→ unite  $q_1$  and  $q_2$

whenever the truncating of the  $k$ -length  
lookahead strings to length  $l \leq k$ ,  
yields the same set of  $l$ -items.

- unite  $q_1, q_2$  if  $\text{Trunc}_l(q_1) = \text{Trunc}_l(q_2)$ ,

$$\text{Trunc}_l(q) = \{[A \rightarrow \alpha \bullet \beta, l:y] \mid [A \rightarrow \alpha \bullet \beta, y] \in q\}$$

**Fact 6.55**

The LA(k)LR(k) machine is same to LR(k) machine.

The LA(k)LR(0) machine

is same to LALR(k) machine.

**Theorem 6.56** Let  $q$  be a state in LA(k)LR(l) machine  
accessible upon reading string  $\delta$ .

Then  $\delta$  is a viable prefix of  $G$ , and

$$\exists \gamma \in V^* . \exists . \forall I \in q, I \in \langle \gamma \rangle_k \text{ where } \delta \rho_l \gamma.$$

Conversely,

If  $I \in \langle \gamma \rangle_k$

then  $\exists q$  s.t.  $I \in q$  and

$q$  is accessible upon reading any viable prefix  $\delta$

$$. \exists . \gamma \rho_l \delta.$$

States in LA(k)LR(l) machine

$$\langle [ \gamma ]_l \rangle_k \leftrightarrow [ \gamma ]_l$$

## 6.6. SLR(k) Parsing

*SLR(k) stands for Simple LR(k).*

*adding k-lookaheads in a crude, simple way.  
seldom minimal lookaheads.*

*SLR(k) parser for G is the pushdown transducer  $M = ([G]_0, \Sigma, \Gamma, P, \tau, [\epsilon]_0, \{[\epsilon]_0[S]_0\}, \$, |)$  where*

$$\Gamma = \{[\delta]_0[\delta X_1]_0 \dots [\delta X_1 \dots X_n]_0 \mid y \rightarrow [\delta]_0[\delta A]_0 \mid y \mid [A \rightarrow X_1 \dots X_n \bullet] \in [\delta X_1 \dots X_n]_0 \text{ and } y \in \text{Follow}_k(A)\} \quad (ra)$$

$$\cup \{[\delta]_0 \mid ay \rightarrow [\delta]_0[\delta a]_0 \mid y \mid a \in \Sigma, [A \rightarrow \alpha \bullet a \beta] \in [\delta]_0,$$

$$y \in \text{First}_{\max\{k-1, 0\}}(\beta \text{Follow}_k(A))\} \quad (sa)$$

### **Theorem 6.57**

*The SLR(k) parser M for G is a right parser for G. Moreover,  $\forall w \in L(G)$ ,*

*M produces all right parses of w in G, and*  

$$\text{TIME}_G(w) = \text{TIME}_M(w) + |w|.$$



**Theorem 6.58** (Characterization of SLR(k) Grammars) *The SLR(k) parser of  $G$  is deterministic iff for all state  $q$  in SLR(k) machine,*

(1) *Whenever  $[A_1 \rightarrow \omega_1 \bullet], [A_2 \rightarrow \omega_2 \bullet] \in q$ , then*

$$\text{Follow}_k(A_1) \cap \text{Follow}_k(A_2) = \emptyset.$$

(2) *Whenever  $[A \rightarrow \alpha \bullet a \beta], [B \rightarrow \omega \bullet] \in q$ , then*

$$\text{First}_k(a\beta \text{Follow}_k(A)) \cap \text{Follow}_k(B) = \emptyset.$$

### **Theorem 6.59**

*The class of SLR(0) grammars*

*coincides with the class of LR(0) grammars.*

*For  $k \geq 1$ ,*

*the class of SLR(k) grammars is properly*

*contained in the class of LALR(k) grammars.*

(eg)  $S \rightarrow Ac|bA|bc,$

$A \rightarrow \varepsilon$

*Time to test SLR(k) property for  $G$ : polynomial to  $|G|$*

## Transformation of $G$ into $T_k(G)$ ,

which is  $SLR(k)$  if and only if  $G$  is  $LR(k)$ .

Idea : replace  $A$  by  $([\gamma]_k A) \dots$

$$G = (N, \Sigma, S, P)$$

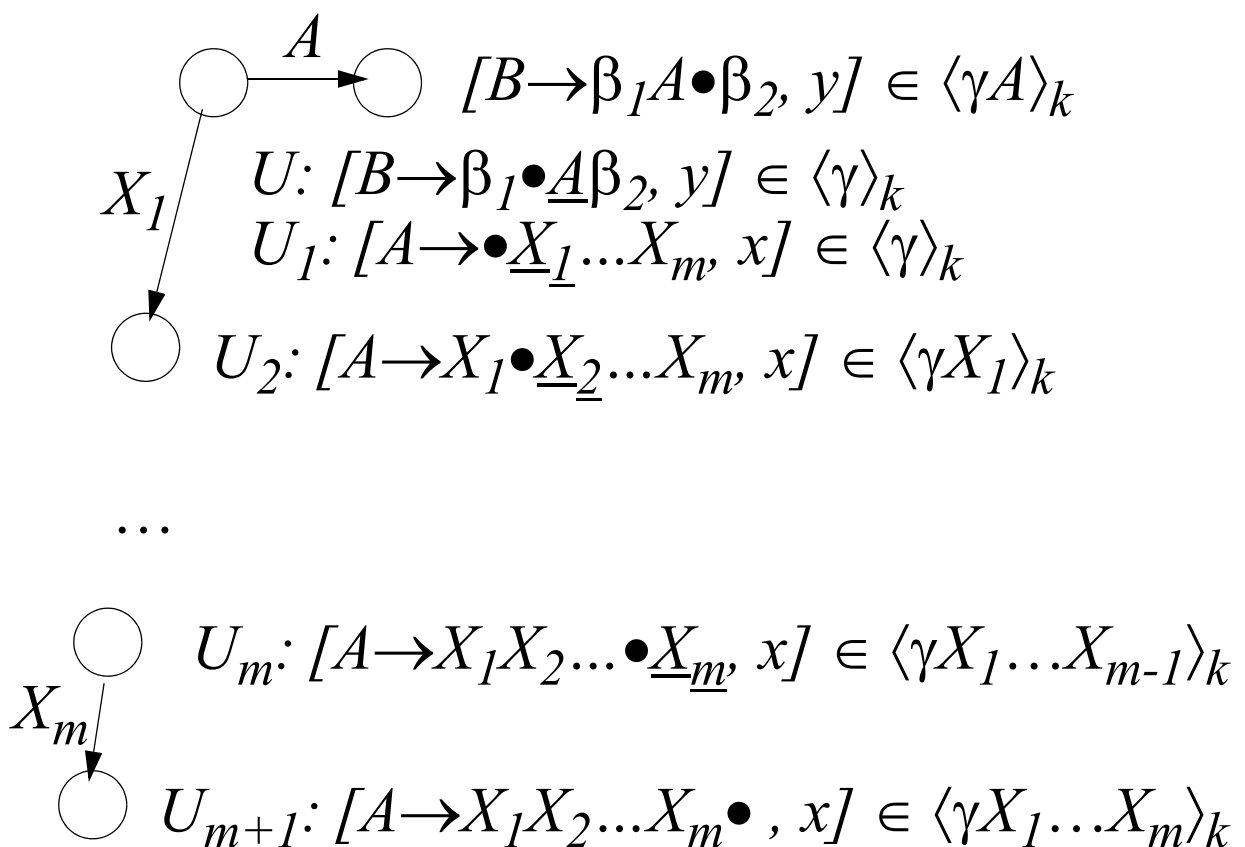
$T_k(G) = ([G]_k \times N, \Sigma, ([\varepsilon]_k S), \hat{P})$ , where

$$\hat{P} = \{([\gamma]_k A) \rightarrow U_1 \dots U_m$$

$$| [B \rightarrow \beta_1 \bullet A \beta_2, y]_k \in \langle \gamma \rangle_k \quad A \rightarrow X_1 \dots X_m \in P,$$

$$1 \leq \forall i \leq m \quad U_i = ([\gamma X_1 \dots X_{i-1}]_k X_i) \text{ if } X_i \in N,$$

$$= X_i \quad \text{if } X_i \in \Sigma \cup \{\$ \}.$$



$([\gamma]_k A)$  is a **useful** nonterminal in  $T_k(G)$ , iff  
 $[\gamma]_k \in [G]_k$ ,  $A \in N$ , and  $[B \rightarrow \alpha.A\beta] \in [\gamma]_k$

$$S \Rightarrow_{rm}^* \gamma Ay$$

$$([\varepsilon]_k S) \Rightarrow_{rm}^* \Phi([\gamma]_k A)y$$

$$\text{Follow}_k([\gamma]_k A) = \{k:y \mid S \Rightarrow_{rm}^* \gamma Ay\}$$

$T_k(G)$  **right-to-right covers**  $G$

$\equiv$  right parses in  $T_k(G)$  are mapped into  
 right parses in  $G$  by a homomorphism  $h$ .

Furthermore,

$T_k(G)$  is **structurally equivalent** to  $G$ .

$\equiv$  parse trees in  $T_k(G)$  and  $G$  have **same structure**  
 parse trees are same except for  
 the labeling of the nonterminal nodes.  
 parse trees are **isomorphic**

### Cover relations between Grammars

Let  $x, y \in \{\text{"left"}, \text{"right"}\}$ . Then

an  **$x$ -to- $y$  cover** of  $G$  is a pair  $(\hat{G}, h)$  where

$$\hat{G} = (\hat{N}, \Sigma, \hat{P}, \hat{S}) \text{ and } h: \hat{P}^* \rightarrow P^* .\exists$$

i)  $\forall w \in L(\hat{G})$  and  $x$ -parses  $\hat{\pi}$  of  $w$  in  $\hat{G}$ ,

$h(\hat{\pi})$  is a  $y$ -parse of  $w$  in  $G$ .

ii)  $\forall w \in L(G)$  and  $y$ -parses  $\pi$  of  $w$  in  $G$ ,

$\exists \hat{\pi} \in \hat{P}^*$ ,  $\hat{\pi}$  is a  $x$ -parse of  $w$  in  $\hat{G}$  and  $h(\hat{\pi}) = \pi$ .

**$h$  maps  $x$ -parses of  $\hat{G}$  into  $y$ -parses of  $G$**

If  $\exists h$ ,  $(\hat{G}, h)$  is  $x$ -to- $y$  covers of  $G$ ,

$\hat{G}$   $x$ -to- $y$  covers  $G$  with respect to  $h$

If  $\hat{G}$   $x$ -to- $y$  covers  $G$  with respect to  $h$ ,

$(\hat{G}, h)$  is  $x$ -to- $y$  covers of  $G$ ,

**Fact 6.60** If  $\hat{G}$   $x$ -to- $y$  covers  $G$ , then  $L(\hat{G}) = L(G)$ .

**Fact 6.61** If  $(M, \tau)$  is an  $x$  parser of  $\hat{G}$  and if  $\hat{G}$   $x$ -to- $y$  covers  $G$  w.r.t.  $h$ , then  $(M, \tau \circ h)$  is a  **$y$  parser** of  $G$ .

$(T_k(G), h_k)$  right-to-right covers  $G$ , if

$h_k(U \rightarrow U_1 \dots U_m) = A \rightarrow X_1 \dots X_m$ , where

$$U = ([\gamma], A), U_i = ([\gamma X_1 \dots X_{i-1}], X_i) \text{ if } X_i \in N, \\ = X_i \text{ if } X_i \in \Sigma.$$

$(M, \tau)$  is a  $x$ -parser of  $G$ , if  $\tau(\theta) = \pi_x$ .

$(\hat{G}, h)$  is a  $x$ -to- $y$  cover of  $G$ , if  $h(\hat{\pi}_x) = \pi_y$ .

$(M, \tau \circ h)$  is a  $x$ -parser of  $G$ .

$G$

$\hat{G}$

$(\hat{M}, \tau)$

$(M, \tau \circ h)$

Consider a function  $M: \Sigma^* \rightarrow \{\Gamma^*\}$

$$M(w) = \theta.$$

$M$  is deterministic if  $M: \Sigma^* \rightarrow \Gamma^*$ .

Consider a function  $G_x: \Sigma^* \rightarrow \{P^*\}$

$$G_x(w) = \pi.$$

$G$  is unambiguous, if  $G_x: \Sigma^* \rightarrow P^*$ .

$M \circ \tau$  is a  $x$ -parser of  $G$ , if  $M \circ \tau: \Sigma^* \rightarrow \{P^*\}$

$$M \circ \tau = G_x.$$

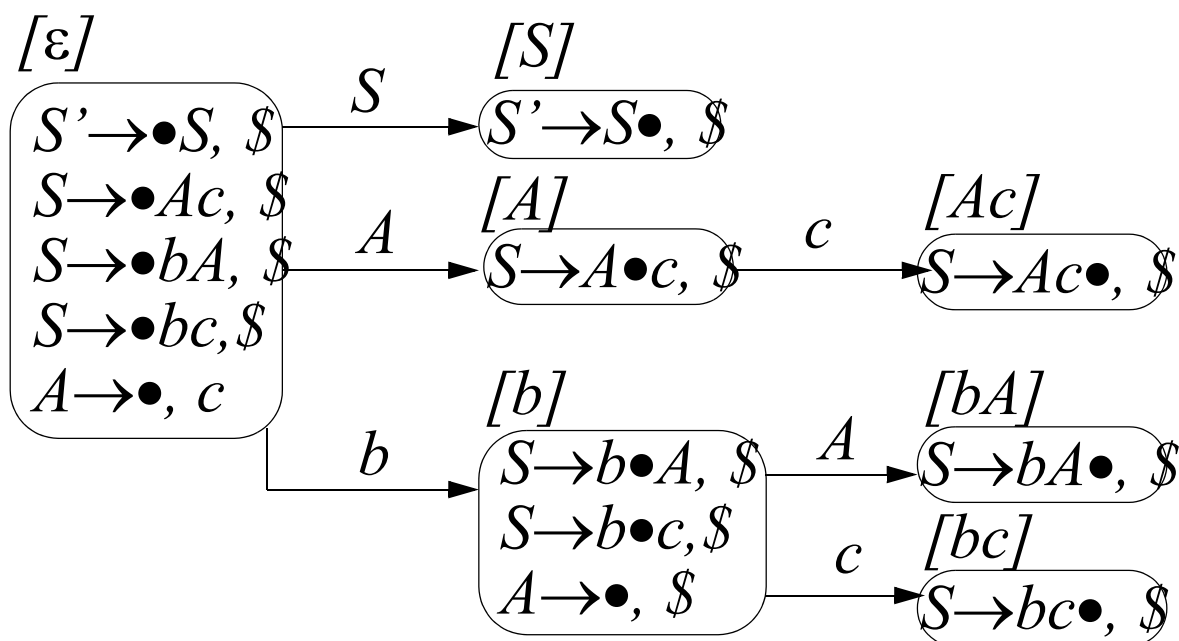
If  $G$  is ambiguous,

*Example*

$$G: \quad S \rightarrow Ac \mid bA \mid bc$$

$$A \rightarrow \varepsilon$$

$$\text{Follow}(A) = \{\$, c\}$$

$$G \text{ is not SLR}(1) \text{ in state } [b]$$


$$T_k(G): \quad ([\varepsilon], S) \rightarrow ([\varepsilon], A) c \mid b ([b], A) \mid b c$$

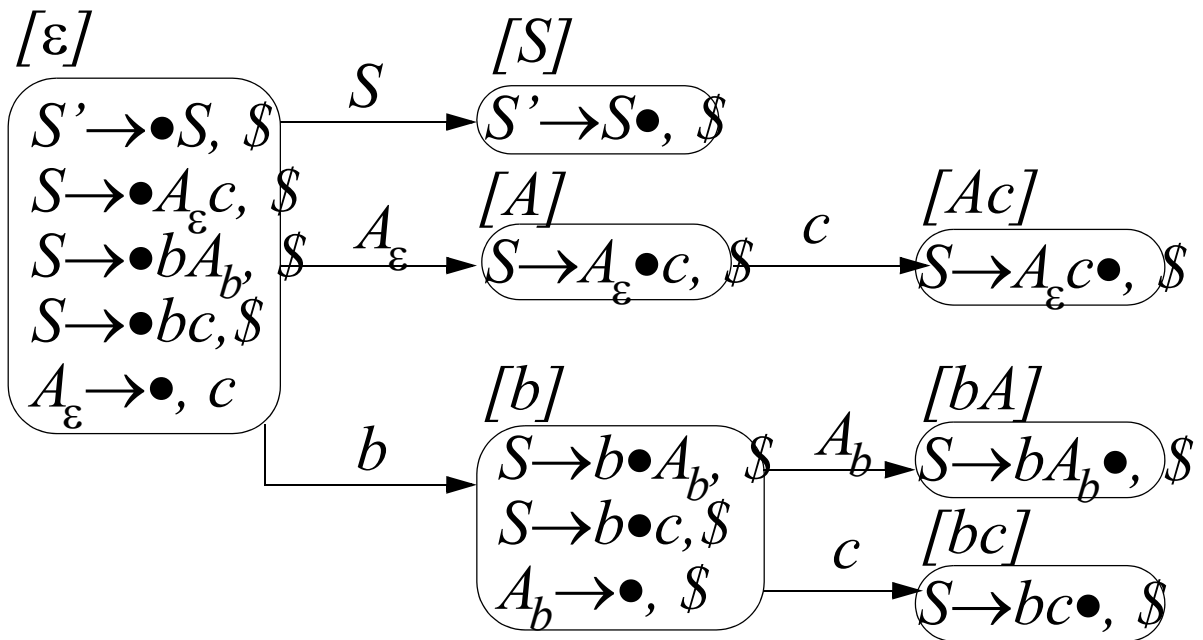
$$([\varepsilon], A) \rightarrow \varepsilon$$

$$([b], A) \rightarrow \varepsilon$$

$$S_\varepsilon \rightarrow A_\varepsilon c \mid b A_b \mid b c$$

$$A_\varepsilon \rightarrow \varepsilon$$

$$A_b \rightarrow \varepsilon$$



$$\text{Follow}(A_\varepsilon) = \{c\}$$

$$\text{Follow}(A_b) = \{\$\}$$

$T_k(G)$  is SLR(1)

$T_k(G)$  right-to-right covers  $G$  w.r.t.  $h_k$  (T6.64)

$\forall w \in L(\hat{G})$  where  $\hat{\pi}$  is a right parse of  $w$  in  $T_k(G)$ ,

$h_k(\hat{\pi})$  is a right parse of  $w$  in  $G$ . (L6.62)

$\forall w \in L(G)$ , where  $\pi$  is a right parse of  $w$  in  $G$

$\exists \hat{\pi} . \exists . h_k(\hat{\pi}) = \pi$ ,  $\hat{\pi}$  is a right parse of  $w$  in  $T_k(G)$ .

(L6.63)

**Lemma 6.62** If  $[\gamma A] \in [G]_k$   $([\gamma], A) \Rightarrow^{\hat{\pi}} \Phi$  in  $T_k(G)$ .

Then let  $\Phi = U_1 \dots U_m y$ ,

$\gamma X_1 \dots X_m \in [G]_k$  and

$A \Rightarrow^{hk(\hat{\pi})} X_1 \dots X_m y$  in  $G$ , where

$$\begin{aligned} U_i &= ([\gamma X_1 \dots X_{i-1}], X_i) \text{ if } X_i \in N, \\ &= X_i \text{ if } X_i \in \Sigma \cup \{\$ \}. \end{aligned}$$

**Proof** by induction on  $|\hat{\pi}|$ .

IB:  $\hat{\pi} = \varepsilon$

IH:  $\hat{\pi} = \hat{\pi}_1 \hat{r}$ ,  $\hat{r} = W \rightarrow W_1 \dots W_p \in \hat{P}$ .

$([\gamma], A) \Rightarrow^{\hat{\pi}_1} U_1 \dots U_n W y$

$$\Rightarrow^{\hat{r}} U_1 \dots U_n W_1 \dots W_p y = \Phi.$$

By definition of  $T_k(G)$ ,

$W = ([\delta], B)$ ,  $\delta = \gamma h_k(U_1 \dots U_n)$ , and

$$\begin{aligned} W_i &= ([\delta Z_1 \dots Z_{i-1}], Z_i), \text{ if } Z_i \in N, \\ &= Z_i, \text{ if } Z_i \in \Sigma \cup \{\$ \}. \end{aligned}$$

By IH,  $\exists \gamma X_1 \dots X_n B \in [G]_k$  and,

$A \Rightarrow^{hk(\hat{\pi}_1)} X_1 \dots X_n B y$  in  $G$ ,

$[\gamma X_1 \dots X_n] = [\delta]$ , and by right invariance,

$$[\delta Z_1 \dots Z_{i-1}] = [\gamma X_1 \dots X_n Z_1 \dots Z_{i-1}].$$

If choose  $m = n+p$ ,  $1 \leq i \leq p$ ,

$U_{n+i} = W_i$ ,  $X_{n+i} = Z_i$ , then the lemma is proved.



**Lemma 6.63**  $\gamma A \in [G]_k$   $A \Rightarrow^\pi X_1 \dots X_m \gamma$  in  $G$ , and either  $X_1 \dots X_m = \varepsilon$  or  $X_m \in N$ .

Then  $\exists \hat{\pi} \in \hat{P}^*$  . $\exists$ .  $h_k(\hat{\pi}) = \pi$  and

$$([\gamma], A) \Rightarrow^{\hat{\pi}} U_1 \dots U_m \gamma \text{ in } T_k(G).$$

**Proof** by induction on  $|\pi|$ .

*Base:*  $\pi = \varepsilon$ . Then  $\hat{\pi} = \varepsilon$  and  $U_1 = ([\gamma], A)$ .

*Induction Step:*  $\pi = \pi_1 r$ ,  $r = B \rightarrow Z_1 \dots Z_p$ .

$$\begin{aligned} A &\xRightarrow{rm}^{\pi_1} Y_1 \dots Y_n B y_1 \xRightarrow{rm}^r Y_1 \dots Y_n Z_1 \dots Z_p y_1 \\ &= X_1 \dots X_m \gamma \text{ in } G. (m=n+p) \end{aligned}$$

$\exists \bar{\pi}_1$  of  $T_k(G)$  . $\exists$ .

$$h_k(\bar{\pi}_1) = \pi_1 \text{ and}$$

$$([\gamma], A) \xRightarrow{rm}^{\bar{\pi}_1} U_1 \dots U_n ([\gamma Y_1 \dots Y_n], B) y_1$$

$$\begin{aligned} \text{where } U_i &= ([\gamma Y_1 \dots Y_{i-1}], Y_i) \text{ if } Y_i \in N, \\ &= Y_i \text{ if } Y_i \in \Sigma. \end{aligned}$$

Then  $\exists \bar{r} = ([\gamma Y_1 \dots Y_n], B) \rightarrow U_{n+1} \dots U_{n+p}$  in  $T_k(G)$

$$\begin{aligned} \text{where } U_i &= ([\gamma Y_1 \dots Y_n Z_1 \dots Z_{i-1}], Z_i) \text{ if } Z_i \in N, \\ &= Z_i \text{ if } Z_i \in \Sigma. \end{aligned}$$

$\therefore h_k(\bar{\pi}_1 \bar{r}) = h_k(\bar{\pi}_1) h_k(\bar{r}) = \pi_1 r = \pi$ , and

$$\begin{aligned} ([\gamma], A) &\xRightarrow{rm}^{\pi_1 r} U_1 \dots U_{n+p} y_1 = U_1 \dots U_m z y_1 \\ &= U_1 \dots U_m \gamma. \end{aligned}$$

**Theorem 6.64**  $T_k(G)$  right-to-right covers  $G$  w.r.t.  $h_k$ .

**Corollary 6.65** If  $(M, \tau)$  is a right parser of  $T_k(G)$ , then  $(M, \tau h_k)$  is a right parser of  $G$ .

**Lemma 6.66**  $y \in \text{Follow}_k([\gamma]_k A)$  in  $T_k(G)$  iff  $S \Rightarrow^* \delta Az$  in  $G$ ,  $[\delta]_k = [\gamma]_k$   $l:z = y$ .

In other words,

$y$  is a follower of  $([\gamma]_k A)$  in  $T_k(G)$

iff  $y$  is a follower of  $A$  in  $G$

in some context LR(k)-equivalent to  $\gamma$ .

**Lemma 6.67**

$[U \rightarrow U_m \dots U_i \bullet U_{i+1} \dots U_p, y] \in \langle U_1 \dots U_i \rangle_l$

in  $T_k(G)$  iff

$[A \rightarrow X_m \dots X_i \bullet X_{i+1} \dots X_p, y] \in \langle X_1 \dots X_i \rangle_l$ ,

$U = ([X_1 \dots X_{m-1}], A)$ , and  $1 \leq j \leq p$ ,

$U_j = ([X_1 \dots X_{j-1}], X_j)$  if  $X_j \in N$ ,

$= X_j$  if  $X_j \in \Sigma \cup \{\$\}$ .

**Lemma 6.68** *If  $G$  is non-LR(k), then so is  $T_k(G)$ .*

**Proof**

(i)  $S \xRightarrow{rm}^+ S$ , then  $([\varepsilon]_k S) \xRightarrow{rm}^+ ([\varepsilon]_k S)$ ,

$\therefore T_k(G)$  is non-LR(k).

(ii)  $\langle X_1 \dots X_i \rangle_k$  contains a conflict. Then

$[A \rightarrow X_m \dots X_i \bullet, y] \in \langle X_1 \dots X_i \rangle_k$

$[B \rightarrow X_n \dots X_i \bullet X_{i+1} \dots X_p, u] \in \langle X_1 \dots X_i \rangle_k$

$y \in \text{First}_k(X_{i+1} \dots X_p u)$ .

Then by lemma 6.67,

$[U \rightarrow U_m \dots U_i \bullet, y] \in \langle U_1 \dots U_i \rangle_k$  and

$[W \rightarrow U_n \dots U_i \bullet U_{i+1} \dots U_p, u] \in \langle U_1 \dots U_i \rangle_k$

where  $U = ([X_1 \dots X_{m-1}]_k A)$ ,

$W = ([X_1 \dots X_{n-1}]_k B)$ ,

$U_j = ([X_1 \dots X_{j-1}]_k X_j)$  if  $X_j \in N$ ,

$= X_j$  if  $X_j \in \Sigma \cup \{\$\}$ .

$U \neq W$  or  $m \neq n$  or  $i+1 \leq p$ ,

$y \in \text{First}_k(U_{i+1} \dots U_p u)$ .

$\therefore T_k(G)$  is non-LR(k).

**Lemma 6.69** *If  $T_k(G)$  is non-SLR(k), then  $G$  is non-LR(k).*

**Proof** *If  $T_k(G)$  is non-LR(k), we have*

$$[U \rightarrow U_m \dots U_i \bullet] \in \langle U_1 \dots U_i \rangle_0,$$

$$[W \rightarrow U_n \dots U_i \bullet U_{i+1} \dots U_p] \in \langle U_1 \dots U_i \rangle_0,$$

$$y \in \text{Follow}_k(U) \cap \text{First}_k(U_{i+1} \dots U_p \text{Follow}_k(W)),$$

*$U_{i+1} \in \Sigma$  whenever  $i+1 \leq p$ . Then there exist*

$$U = ([X_1 \dots X_{m-1}]_k A),$$

$$W = ([X_1 \dots X_{n-1}]_k B), \text{ and}$$

$$U_j = ([X_1 \dots X_{j-1}]_k X_j) \text{ if } X_j \in N,$$

$$= X_j \text{ if } X_j \in \Sigma \cup \{\$, \}, 1 \leq \forall j \leq p.$$

*By lemma 6.66,*

$$S \Rightarrow^* \gamma Az, [\gamma]_k = [X_1 \dots X_{m-1}]_k \text{ k:z} = y,$$

$$S \Rightarrow^* \delta Bu, [\delta]_k = [X_1 \dots X_{n-1}]_k \text{ k:xu} = y, \text{ and}$$

$$x \in \text{First}_k(U_{i+1} \dots U_p).$$

(i) *If  $U_{i+1} \in \Sigma$ ,  $X_{i+1} \in \Sigma$ , too. Then*

$$[A \rightarrow X_1 \dots X_i \bullet, \text{k:z\$}], [B \rightarrow X_1 \dots X_i \bullet X_{i+1} \dots X_p, \text{k:u\$}]$$

*exhibit a conflict.*

(ii)  *$i \leq p$ , by the right invariance,  $i = p$ , then reduce-reduce conflict.*

**Theorem 6.70** *Any grammar  $G$  can be transformed into a structurally equivalent grammar which is SLR( $k$ ) iff  $G$  is LR( $k$ ).*

**Theorem 6.71** *For any  $k \geq 0$ ,  
the families of LR( $k$ ) languages,  
LALR( $k$ ) languages, and  
SLR( $k$ ) languages  
are all equal.*

## 6.7. Covering LR(k) Grammars by LR(1) Grammars

- LR(k) language  $\equiv$  LR(1) language  
     LR(k) grammar  $\Rightarrow$  LR(1) grammar  
     right-to-right cover
- deterministic language  $\Rightarrow$  SLR(1) parsing

$T_{k, 1}(G)$  right-to-right covers  $G$ , and  
 $T_{k, 1}(G)$  is LR(1), iff  $G$  is LR(k+1)

*Idea*

- shift the derivation trees in  $G$   
    $k$  symbols to the right
- reduce actions are postponed until 1 symbol  
   lookahead is sufficient to resolve uniquely.

$A$  is replaced by set of  $(x, A, y)$ 's, where  
 $y \in \text{Follow}_k(A)$ ,  $x \in \text{First}_k(Ay)$ .  
 $|y| \leq k$  and  $|x| = k$ .

$L((x, A, y)) =$   
 $\{z \mid S \Rightarrow^* uAw \Rightarrow^* uvw, y = k:w, x = k:vw, xz =$   
 $vy, |x| = k\}$

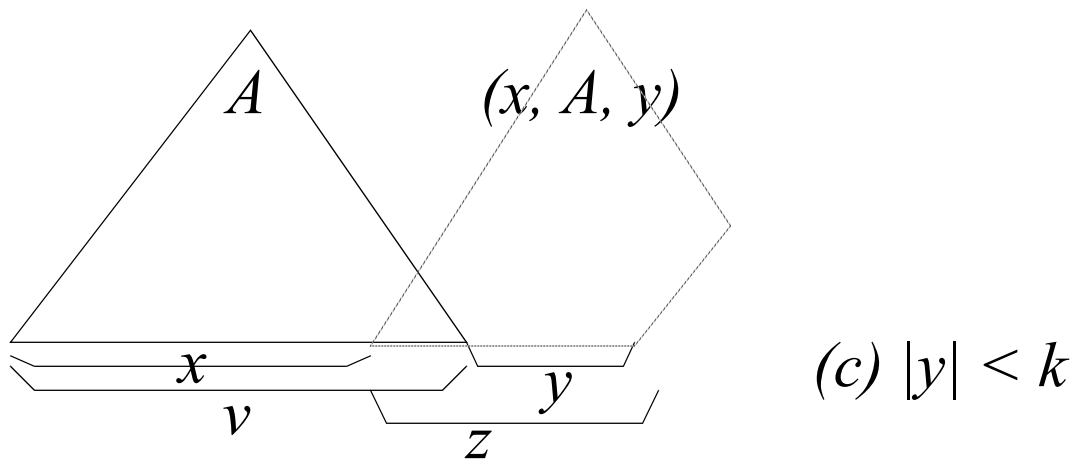
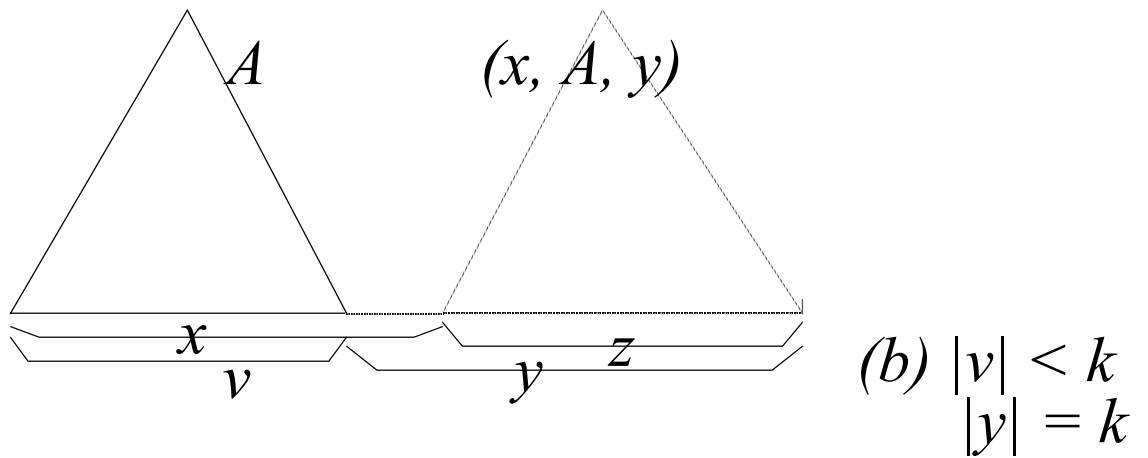
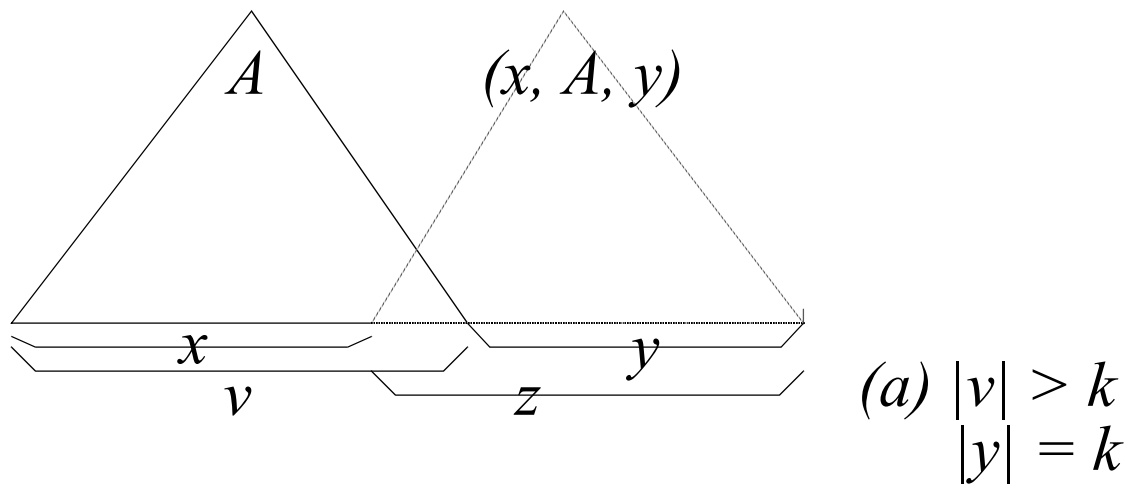


Figure 6.20 (p85)

$vy = xz$ , where  $A \Rightarrow^* v$ ,  $(x, A, y) \Rightarrow^* z$ ,  
and  $|x| = k$ .

Let  $G = (N, \Sigma, P, S)$  be a grammar.

$T_{k,1}(G) = (N', \Sigma, P', S')$ , where

$$N' = \{S'\}$$

$$\cup \{(x, X, y) \mid y \in \text{Follow}_k(X), x \in \text{First}_k(Xy)\}$$

$$P' = \{S' \rightarrow x(x, S, \varepsilon) \mid x \in \text{First}_k(S)\}$$

$$\cup \{(y_0, A, y_m) \rightarrow$$

$$(y_0, X_1, y_1)(y_1, X_2, y_2) \dots (y_{m-1}, X_m, y_m)$$

$$\mid m \geq 0, A \rightarrow X_1 \dots X_m \in P, y_m \in \text{Follow}_k(A),$$

$$0 \leq \forall i < m, y_i \in \text{First}_k(X_{i+1}y_{i+1}),$$

$$\text{Follow}_k, \text{First}_k \text{ in the context of } A \Rightarrow^* y_0 \dots \}$$

$$\cup \{(ax, a, xb) \rightarrow b \mid xb \in \text{Follow}_k(a), |xb| = k\}$$

$$\cup \{(ax, a, x) \rightarrow \varepsilon \mid x \in \text{Follow}_k(a), |x| < k\}.$$

$$y_i \in \text{Follow}_k(X_i)$$

$$h_{k,1}: P' \rightarrow P \cup \{\varepsilon\}$$

$$h_{k,1}(S' \rightarrow x(x, S, \varepsilon)) = \varepsilon,$$

$$h_{k,1}((y_0, A, y_m) \rightarrow (y_0, X_1, y_1) \dots (y_{m-1}, X_m, y_m))$$

$$= A \rightarrow X_1 \dots X_m,$$

$$h_{k,1}((ax, a, xb) \rightarrow b) = \varepsilon, \text{ and}$$

$$h_{k,1}((ax, a, x) \rightarrow \varepsilon) = \varepsilon.$$

## Two problems

i)  $(T_{k,1}(G), h_{k,1})$  is a right-to-right cover of  $G$ .

**T6.78** ( $\Rightarrow$ : L6.72-74,  $\Leftarrow$ : L6.75-77)

ii)  $T_{k,1}(G)$  is LR(1) if and only if  $G$  is LR( $k+1$ ).

**T6.85** ( $\Leftarrow$ : L6.79-81,  $\Rightarrow$ : L6.82-84)



*Example*

	$First_1$	$Follow_1$
$S \rightarrow Abb \mid Bb$	$\{a\}$	$\{\epsilon\}$
$A \rightarrow aA \mid a$	$\{a\}$	$\{b\}$
$B \rightarrow aB \mid a$	$\{a\}$	$\{b\}$

$S_0 \rightarrow a (a, S, \epsilon)$		$\epsilon$
$(a, S, \epsilon) \rightarrow (a, A, b)(b, b, b)(b, b, \epsilon)$		$S \rightarrow Abb$
$(a, B, b) (b, b, \epsilon)$		$S \rightarrow Bb$
$(a, A, b) \rightarrow (a, a, a) (a, A, b)$		$A \rightarrow aA$
$(a, a, b)$		$A \rightarrow a$
$(aa, B, b) \rightarrow (a, a, a) (a, B, b)$		$B \rightarrow aB$
$(a, a, b)$		$B \rightarrow a$
$(a, a, a) \rightarrow a$		$\epsilon$
$(a, a, b) \rightarrow b$		$\epsilon$
$(b, a, b) \rightarrow b$		$\epsilon$
$(b, b, \epsilon) \rightarrow \epsilon$		$\epsilon$

*Example*

	$First_2$	$Follow_2$
$S \rightarrow Abb \mid Bb$	$\{aa, ab\}$	$\{\varepsilon\}$
$A \rightarrow aA \mid a$	$\{aa, a\}$	$\{bb\}$
$B \rightarrow aB \mid a$	$\{aa, a\}$	$\{b\}$

$S_0 \rightarrow aa (aa, S, \varepsilon)$	$\varepsilon$
$ab (ab, S, \varepsilon)$	$\varepsilon$
$(aa, S, \varepsilon) \rightarrow (aa, A, bb)(bb, b, b)(b, b, \varepsilon)$	$S \rightarrow Abb$
$(aa, B, b) (b, b, \varepsilon)$	$S \rightarrow Bb$
$(ab, S, \varepsilon) \rightarrow (ab, A, bb)(bb, b, b)(b, b, \varepsilon)$	$S \rightarrow Abb$
$(ab, B, b) (b, b, \varepsilon)$	$S \rightarrow Bb$
$(aa, A, bb) \rightarrow (aa, a, aa) (aa, A, bb)$	$A \rightarrow aA$
$(aa, a, ab) (ab, A, bb)$	$A \rightarrow aA$
$(ab, A, bb) \rightarrow (ab, a, bb)$	$A \rightarrow a$
$(aa, B, b) \rightarrow (aa, a, aa) (aa, B, b)$	$B \rightarrow aB$
$(aa, a, ab) (ab, B, b)$	$B \rightarrow aB$
$(ab, B, b) \rightarrow (ab, a, b)$	$B \rightarrow a$
$(aa, a, aa) \rightarrow a$	$\varepsilon$
$(aa, a, ab) \rightarrow b$	$\varepsilon$
$(ab, a, bb) \rightarrow b$	$\varepsilon$
$(ab, a, b) \rightarrow \varepsilon$	$\varepsilon$
$(bb, b, b) \rightarrow \varepsilon$	$\varepsilon$
$(b, b, \varepsilon) \rightarrow \varepsilon$	$\varepsilon$

**Lemma 6.72** Consider  $G$  and  $T_{k, 1}(G)$  be grammar.

If  $(x, A, y) \Rightarrow^{\pi'} (y_0, X_1, y_1) \dots (y_{m-1}, X_m, y_m)z$   
 $= \Phi$  in  $T_{k, 1}(G)$ .

Then  $y_0 = x$ ,  $y_m z = vy$ ; and

$A \Rightarrow^{\pi} X_1 X_2 \dots X_m v$  in  $G$ .

Moreover, if  $|y_m| < k$ , then  $z = \varepsilon$ .

**Proof** by induction on  $|\pi'|$

IB:  $m = 1$ ,  $y_0 = x$ ,  $y_m = y$ ,  $z = v = \varepsilon$ , and  $X_1 = A$ .

IH:  $\pi' = \pi_1' r'$

$(x, A, y) \Rightarrow^{\pi_1'} (y_0, X_1, y_1) \dots (y_{n-2}, X_{n-1}, y_{n-1})(y_{n-1}, X,$   
 $y'_n)z_1 \Rightarrow^{r'} (y_0, X_1, y_1) \dots (y_{n-2}, X_{n-1}, y_{n-1})\omega z_1 = \Phi$

Moreover,  $\exists v_1$  s.t.

$\psi_0 = x$ ,  $y'_n z_1 = v_1 y$ , and

$A \Rightarrow^{\pi} X_1 X_2 \dots X_{n-1} X v_1$  in  $G$ .

Case 1:

$r' = (y_{n-1}, X, y'_n) \rightarrow (y_{n-1}, X_n, y_n) \dots (y_{m-1}, X_m, y_m)$ ,  
 $m \geq n - 1$ .

By definition,  $y_m = y'_n$  and  $h(r') = X \rightarrow X_n \dots X_m \in P$ .

Then we have:

$y_0 = x$ ,  $y_m z_1 = y'_n z_1 = v_1 y$ , and

$A \Rightarrow^{\pi_1'} X_1 X_2 \dots X_{n-1} X v_1 \Rightarrow^{r'} X_1 \dots X_m v_1$  in  $T_{k, 1}(G)$ .

$z = z_1$ ,  $v = v_1$ .

Case 2:  $r' = (au, a, ub) \rightarrow b$ .  $h(r') = \varepsilon$ .

If  $m = n - 1$ ,  $z = bz_1$ , and  $v = av_1$ , we then have:

$$\begin{aligned} \Phi &= (y_0, X_1, y_1) \dots (y_{n-2}, X_{n-1}, y_{n-1})bz_1 \\ &= (y_0, X_1, y_1) \dots (y_{m-1}, X_m, y_m)z, \quad (y_n' = ub) \end{aligned}$$

$$y_0 = x, y_mz = y_{n-1}bz_1 = ay_n'z_1 = av_1y = vy, \text{ and}$$

$$A \Rightarrow^{h(\pi'r')} X_1 \dots X_{n-1}Xv = X_1 \dots X_mav_1 = X_1 \dots X_mv$$

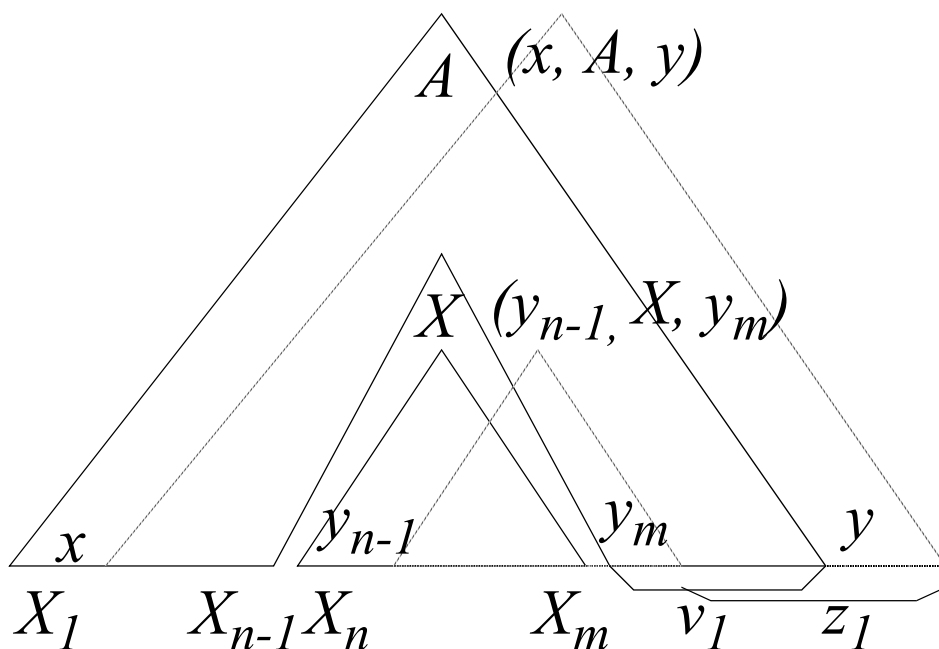
Case 3:  $r' = (ay_n', a, y_n') \rightarrow \varepsilon$ .  $|y_n'| < k$ ,  $h(r') = \varepsilon$ .

If  $m = n - 1$ ,  $z = z_1$ , and  $v = av_1$ , we then have:

$$\begin{aligned} \Phi &= (y_0, X_1, y_1) \dots (y_{n-2}, X_{n-1}, y_{n-1})z_1 \\ &= (y_0, X_1, y_1) \dots (y_{m-1}, X_m, y_m)z, \end{aligned}$$

$$y_0 = x, y_mz = y_{n-1}z_1 = ay_n'z_1 = av_1y = vy, \text{ and}$$

$$A \Rightarrow^{h(\pi'r')} X_1 \dots X_{n-1}Xv = X_1 \dots X_mv_1 = X_1 \dots X_mv$$



**Lemma 6.73** *If*

$(x, A, y) \Rightarrow^{\pi'} z$  in  $T_{k, 1}(G)$ , then

$A \Rightarrow^{h(\pi')} v$  in  $G$ , where  $vy = xz$ .

**Proof**

$\Phi = z$  in **L6.72**.

**Lemma 6.74** *If  $\pi'$  is a right parse of  $w$  in  $T_{k, 1}(G)$ , then  $h(\pi')$  is a right parse of  $w$  in  $G$ .*

**Lemma 6.75** Let  $G = (N, \Sigma, P, S)$

$$A \Rightarrow^\pi X_1 X_2 \dots X_m v \text{ in } G,$$

$m = 0$  or  $X_m$  is a nonterminal,

$$y \in \text{Follow}_k(A), y_m = k:vy, y_m z = vy, \text{ and}$$

$$y_i \in \text{First}_k(X_{i+1}y_{i+1}) \quad 0 \leq i < m.$$

Then there is a rule string  $\pi'$  of  $T_{k,1}(G)$  . $\exists$ .

$$h(\pi') = \pi, \text{ and}$$

$$(y_0, A, y) \Rightarrow^{\pi'} (y_0, X_1, y_1) \dots (y_{m-1}, X_m, y_m) z$$

in  $T_{k,1}(G)$ .

**Proof** by induction on  $|\pi|$

*IB:*  $m = 1, y_0 = x, y_m = y, z = v = \varepsilon, \text{ and } X_1 = A.$

*IH:*  $\pi = \pi_1 r.$  Then

$$A \Rightarrow^{\pi_1} X_1 X_2 \dots X_n B v_1$$

$$\Rightarrow^r X_1 X_2 \dots X_n X_{n+1} \dots X_p v_1$$

$$= X_1 \dots X_m v \text{ in } G$$

Here,  $v = X_{m+1} \dots X_p v_1$ , because  $r = B \rightarrow X_{n+1} \dots X_p$ .

If  $p > m$ , let  $y_p = k:v_1 y, \exists z_1 . \exists. y_p z_1 = v_1 y.$

And let  $y_i = k:X_{i+1}y_{i+1}, m < i < p.$

$$\text{Then } y_m = k:vy = k:X_{m+1} \dots X_p v_1 y = k:X_{m+1} y_{m+1},$$

$$\text{and } y_m \in \text{First}_k(B y_p).$$

By *IH:*

$$h(\pi_1') = \pi_1, \text{ and}$$

$$(y_0, A, y) \Rightarrow^{\pi_1'} (y_0, A, y) \\ \Rightarrow^{\pi_1'} (y_0, X_1, y_1) \dots (y_{n-1}, X, y'_n) z_1 \text{ in } T_{k, 1}(G).$$

And  $\exists r' . \exists$ .

$$r' = (y_n, X_{n+1}, y_{n+1}) \dots (y_{p-1}, X_p, y_p) \\ h(r') = r.$$

Then

$$(y_0, A, y) \Rightarrow^{\pi_1' r'} (y_0, X_1, y_1) \dots (y_{p-1}, X_p, y_p) z_1 \\ (y_m, X_{m+1}, y_{m+1}) \dots (y_{p-1}, X_p, y_p) \Rightarrow^{\pi_2'} u \in \Sigma^* \\ \text{in } T_{k, 1}(G), \text{ where } \pi_2' = p - m \text{ rules of the form} \\ (ax, a, xb) \rightarrow b \text{ or } (ax, a, x) \rightarrow \varepsilon.$$

Then

$$(y_0, A, y) \Rightarrow^{\pi_1' r' \pi_2'} (y_0, X_1, y_1) \dots (y_{m-1}, X_m, y_m) u z_1 \\ \text{in } T_{k, 1}(G).$$

Let  $\pi' = \pi_1' r' \pi_2'$ , then  $h(\pi') = \pi$ .

And  $y_m u z_1 = v y = y_m z$  implying  $u z_1 = z$ , as claimed.

**Lemma 6.76** *If*

$A \Rightarrow^\pi v$  in  $G$ , and

$y \in \text{Follow}_k(A)$ ,  $x = k:vy$ , and  $xz = vy$ ,

then for some  $\pi'$  of  $T_{k, 1}(G)$ ,

$h(\pi') = \pi$  and

$(x, A, y) \Rightarrow^{\pi'} z$  in  $T_{k, 1}(G)$ .

**Proof**

$m=0$  and  $y_0 = x$  in **L6.75**.

**Lemma 6.77** *If  $\pi$  is a right parse of  $w$  in  $G$ , then  $w$  has in  $T_{k, 1}(G)$  a right parse  $\pi'$ .  $\exists$ .  $h(\pi') = \pi$ .*

**Theorem 6.78** *For all grammars  $G$  and  $k > 0$ ,  $T_{k, 1}(G)$  right-to-right covers  $G$  w.r.t. the homomorphism  $h$ .*



**Lemma 6.79**

$(y_0, X_1, y_1) \dots (y_{n-1}, X_n, y_n) \xRightarrow{rm}^* z$  in  $T_{k, 1}(G)$ .

Then  $\exists v . \exists . X_1 \dots X_n \xRightarrow{rm}^* v$  in  $G$  and

$$vy_n = y_0z.$$

**Lemma 6.80**

$[U \rightarrow \phi \cdot \psi, d] \in \langle \Phi \rangle_1$ .

Then the form of  $\Phi$  and  $[U \rightarrow \phi \cdot \psi, d]$  are

(i)  $\Phi = x, [S_0 \rightarrow x \cdot y(xy, S, \varepsilon), \$]$ .

(ii)  $\Phi = x(x, S, \varepsilon), [S_0 \rightarrow x(x, S, \varepsilon); \$]$ .

(iii)  $\Phi = y_0(y_0, X_1, y_1) \dots (y_{r-1}, X_r, y_r),$

$$[(y_m, A, y_n) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{r-1}, X_r, y_r) \cdot \\ (y_r, X_{r+1}, y_{r+1}) \dots (y_{n-1}, X_n, y_n), d],$$

where  $[A \rightarrow X_{m+1} \dots X_r \cdot X_{r+1} \dots X_n, y_n d]$

$$\in \langle X_1 \dots X_r \rangle_{k+1}.$$

(iv)  $\Phi = y_0(y_0, X_1, y_1) \dots (y_{r-1}, X_r, ax)$  and

$$[(ax, a, xb) \rightarrow \cdot b, d]$$

where  $[A \rightarrow \alpha \cdot a \beta, y'] \in \langle X_1 \dots X_r \rangle_{k+1}$

and  $xbd \in \text{First}_{k+1}(\beta y')$ .

(v)  $\Phi = y_0(y_0, X_1, y_1) \dots (y_{r-1}, X_r, ax)b$  and

$[(ax, a, xb) \rightarrow \cdot b, d]$  where... same to (iv).

(vi)  $\Phi = y_0(y_0, X_1, y_1) \dots (y_{r-1}, X_r, ax)b$  and

$$[(ax, a, xb) \rightarrow \cdot, d]$$

where  $[A \rightarrow \alpha \cdot a \beta, y'] \in \langle X_1 \dots X_r \rangle_{k+1}$

and  $x\$ \in \text{First}_{k+1}(\beta y')$ .

**Proof** cases on the form of  $U$ .

Case1:  $U = S_0$ . Then (i) or (ii) is true.

Case2:  $U = (x, A, y)$ . Then

$$\begin{aligned} S_0' &\xRightarrow{rm} S_0\$ \xRightarrow{rm} y_0(y_0, S, \varepsilon)\$ \xRightarrow{rm}^* y_0\gamma(x, A, y)z\$ \\ &\xRightarrow{rm} y_0\gamma\phi\psi z\$ = \Phi\psi z\$ \text{ in } T_{k, 1}(G)' \end{aligned}$$

and  $1:z\$ = d$ .

$$\therefore (y_0, S, \varepsilon) \xRightarrow{rm}^* \gamma(x, A, y)z \text{ in } T_{k, 1}(G).$$

$$\gamma(x, A, y)z = (y_0, X_1, Y_1) \dots (y_{m-1}, X_m, x)(x, A, y)z$$

and  $S \xRightarrow{rm}^* X_1 \dots X_m A y z$  in  $G$ .

If  $y_m = x, y_n = y, A \rightarrow X_{m+1} \dots X_n \in P$ ,

$$U \rightarrow \phi\psi$$

$$= (y_m, A, y_n) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{n-1}, X_n, y_n).$$

Then

$$S' \xRightarrow{rm} X_1 \dots X_m A y z \$ \xRightarrow{rm} X_1 \dots X_m X_{m+1} \dots X_n y z \$ \text{ in } G'$$

and  $[A \rightarrow X_{m+1} \dots X_r X_{r+1} \dots X_n, k+1:yz\$]$

$$\langle X_1 \dots X_r \rangle_{k+1}.$$

$\therefore$  (iii) is true.

Case3:  $U = (ax, a, xb)$ .

By lemma 6.72,

$$(y_0, S, \varepsilon)\$ \xRightarrow{rm}^+ y_0(y_0, X_1, Y_1) \dots (y_{r-1}, X_r, ax)(ax, a, xb)z,$$

and  $\exists A' \rightarrow X_{i+1} \dots X_n \in P$ ,

$$(y_0, S, \varepsilon)\$ \xRightarrow{rm}^+ y_0(y_0, X_1, y_1) \dots (y_{i-1}, X_i, y_i)(y_i, A', y_n)u$$

$$\xRightarrow{rm} y_0(y_0, X_1, y_1) \dots (y_{n-1}, X_n, y_n)u, \text{ and}$$

$$(y_{r+1}, X_{r+2}, y_{r+2}) \dots (y_{n-1}, X_n, y_n) u \xRightarrow{rm}^* z.$$

$$\therefore S' \xRightarrow{rm}^* X_1 \dots X_i A' y_n u \$ \xRightarrow{rm} X_1 \dots X_n y_n u \$$$

in  $G$  by lemma 6.72.

$$[A' \rightarrow X_{i+1} \dots X_r \cdot a X_{r+2} \dots X_n, k+1 : y_n u \$] \\ \in \langle X_1 \dots X_r \rangle_{k+1},$$

$$\text{and } X_{r+2} \dots X_n \xRightarrow{rm} v \text{ in } G, v y_n u = y_{r+1} z.$$

$$x b d = k+1 : y_{r+1} z \$ \in \text{First}_{k+1}(X_{r+2} \dots X_n y_n u \$) \\ = \text{First}_{k+1}(X_{r+2} \dots X_n (k+1 : y_n u \$)).$$

$\therefore$  one of (iv) and (v) is true.

Case4:  $U = (ax, a, x)$ , similar to Case3, (vi) is true.

**Lemma 6.81** If  $T_{k, 1}(G)$  is non-LR(1), then  $G$  is non-LR( $k+1$ ).

**Proof**

Let  $\Phi$  be a viable prefix  $\exists I, J \in \langle \Phi \rangle_1$

which cause a conflict.

$$\text{Case1: } \Phi = x(x, S, \varepsilon), I = [S_0 \rightarrow x(x, S, \varepsilon) \cdot, \$],$$

$$J = [(x, A, \varepsilon) \rightarrow (x, S, \varepsilon) \cdot, \$].$$

Then  $[A \rightarrow S \cdot, \$] \in \langle S \rangle_{k+1}$ ,

$$S' \xRightarrow{rm} S \$ \xRightarrow{rm}^* A \$ \xRightarrow{rm} S \$ \text{ in } G', \text{ and } S \xRightarrow{rm}^+ S \text{ in } G.$$

$$\text{Case2: } \Phi = x(x, S, \varepsilon), I = [S_0 \rightarrow x(x, S, \varepsilon) \cdot, \$],$$

$$J = [(x, A, \varepsilon) \rightarrow \cdot, \$].$$

Then  $[A \rightarrow \cdot, \$] \in \langle S \rangle_{k+1}$ , and  $S' \xRightarrow{rm} S \$ \xRightarrow{rm}^* A \$ \xRightarrow{rm} S \$$  in  $G$ .

Case3:  $\Phi = y_0(y_0, X_1, Y_1) \dots (y_{r-1}, X_r, y_r),$

$I = [(y_m, A, y_r) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{r-1}, X_r, y_r) \bullet,$   
 $d]$

$J = [(y_p, B, y_r) \rightarrow (y_p, X_{p+1}, y_{p+1}) \dots (y_{r-1}, X_r, y_r) \bullet, d].$

Then  $[A \rightarrow X_{m+1} \dots X_r \bullet, d]$  and  $[B \rightarrow X_{p+1} \dots X_r \bullet, d]$

cause a conflict in  $\langle X_1 \dots X_r \rangle_{k+1}$ .

Case4:  $\Phi = y_0(y_0, X_1, Y_1) \dots (y_{r-1}, X_r, ax),$

$I = [(y_m, A, y_r) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{r-1}, X_r, ax) \bullet,$   
 $b]$

$J = [(ax, a, xb) \rightarrow \bullet b, d].$

Then  $[A \rightarrow X_{m+1} \dots X_r \bullet, axb]$  and  $[B \rightarrow \alpha \bullet a \beta, y']$

cause a s-r conflict in  $\langle X_1 \dots X_r \rangle_{k+1}$

because  $xbd \in \text{First}_{k+1}(\beta y')$ .

Case5:  $\Phi = y_0(y_0, X_1, Y_1) \dots (y_{r-1}, X_r, ax),$

$I = [(y_m, A, y_r) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{r-1}, X_r, ax) \bullet,$   
 $\$]$

$J = [(ax, a, xb) \rightarrow \bullet, \$].$

Then  $[A \rightarrow X_{m+1} \dots X_r \bullet, ax\$]$  and  $[B \rightarrow \alpha \bullet a \beta, y']$

cause a s-r conflict in  $\langle X_1 \dots X_r \rangle_{k+1}$

because  $x\$ \in \text{First}_{k+1}(\beta y')$ .

**Lemma 6.82** Let

$$X_i \xRightarrow{rm}^* v_i, y_i \in \text{Follow}_k(X_i), \text{ and } y_{i-1} = k:v_i y_i.$$

Then

$$(y_0, X_1, Y_1) \dots (y_{n-1}, X_n, y_n) \xRightarrow{rm}^* z, \\ \text{where } v_1 \dots v_n y_n = y_0 z.$$

**Lemma 6.83** Let

$$[A \rightarrow X_{m+1} \dots X_r \cdot X_{r+1} \dots X_n, y_n d] \\ \in \langle X_1 \dots X_r \rangle_{k+1},$$

$$X_i \xRightarrow{rm}^* v_i \text{ and } y_{i-1} = k:v_i y_i, 1 \leq \forall i \leq n, 0 \leq m \leq r \leq n.$$

Then

$$(b) [(y_m, A, y_n) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{r-1}, X_r, y_r) \cdot \\ (y_r, X_{r+1}, y_{r+1}) \dots (y_{n-1}, X_n, y_n), d] \\ \in \langle y_0(y_0, X_1, Y_1) \dots (y_{r-1}, X_r, y_r) \rangle_1.$$

Moreover, if  $r < n$  and

$$(y_r, X_{r+1}, y_{r+1}) = (ax, a, xb), \text{ then}$$

$$(c) [(ax, a, xb) \rightarrow \cdot b, 1:ud] \\ \in \langle y_0(y_0, X_1, Y_1) \dots (y_{r-1}, X_r, y_r) \rangle_1, \\ \text{where } xbu = v_{i+1} \dots v_n y_n.$$

Similarly, if  $r < n$  and

$$(y_r, X_{r+1}, y_{r+1}) = (ax, a, x), \text{ then}$$

$$(d) [(ax, a, xb) \rightarrow \cdot, \$] \\ \in \langle y_0(y_0, X_1, Y_1) \dots (y_{r-1}, X_r, y_r) \rangle_1.$$

**Proof**

(i) By definition,  $\exists v \in \Sigma^* . \exists$ .

$$S' \Rightarrow S\$ \Rightarrow^* X_1 \dots X_m A v \$ \Rightarrow X_1 \dots X_m X_{m+1} \dots X_n v \$$$

and  $k+1:v\$ = y_n d$ .

Then

$$(y_0, S, \varepsilon) \Rightarrow^* (y_0, X_1, y_1) \dots (y_{m-1}, X_m, y_m) (y_m, A, y_n) z,$$

$$(y_m, A, y_n) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{n-1}, X_n, y_n)$$

in  $T_{k,1}(G)$ .

Then

$$S_0' \Rightarrow S_0\$ \Rightarrow y_0(y_0, S, \varepsilon)\$$$

$$\Rightarrow^* y_0(y_0, X_1, y_1) \dots (y_{m-1}, X_m, y_m) (y_m, A, y_n) z \$$$

$$\Rightarrow y_0(y_0, X_1, y_1) (y_{n-1}, X_n, y_n) z \$.$$

$\therefore$  (b) is true.

(ii) Assume that  $r < n$ ,  $(y_r, X_{r+1}, y_{r+1}) = (ax, a, xb)$ .

Then  $(ax, a, xb) \rightarrow b$  in  $T_{k,1}(G)$  and

$$X_i \Rightarrow^* v_i, y_i = k:v_{i+1} \dots v_n y_n \in \text{Follow}_k(X_i),$$

$$(y_{r+1}, X_{r+2}, y_{r+2}) \dots (y_{n-1}, X_n, y_n) \Rightarrow^* u \text{ by lem 6.82}$$

where  $v_{r+2} \dots v_n y_n = y_{r+1} u = xbu$ .

Then by (b) and lemma 6.17, (c) is true.

(iii) in a same manner (d) is true.

**Lemma 6.84** Let  $G = (N, \Sigma, P, S)$  and  $k \geq 0$ .  
If  $G$  is non-LR( $k+1$ ), then  $T_{k, 1}(G)$  is non-LR(1).

*Proof*

(i) if  $S \Rightarrow^+ S$  in  $G$ , then  $T_{k, 1}(G)$  is ambiguous.

$$S \Rightarrow^+ A_1 \dots A_m S, A_i \Rightarrow^* \varepsilon \text{ for all } i.$$

then  $\exists x$  in  $T_{k, 1}(G)$ ,

$$(x, S, \varepsilon) \Rightarrow^+ (x, A_1, x) \dots (x, A_m, x) (x, S, \varepsilon).$$

By lemma 6.75,  $A_i \Rightarrow^* \varepsilon$  implies  $(x, A_i, x) \Rightarrow^* \varepsilon$  for all  $i$ .

$\therefore (x, S, \varepsilon)$  derives itself and  $T_{k, 1}(G)$  is ambiguous.

(ii)

$$[A \rightarrow X_{m+1} \dots X_r; w'], [B \rightarrow X_{p+1} \dots X_r; w'] \\ \in \langle X_1 \dots X_r \rangle_{k+1}.$$

then for  $y_r d = w'$ ,  $y_i = k: v_{i+1} y_{i+1}$ ,

$$[(y_m, A, y_r) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{r-1}, X_r, y_r); d], \\ [(y_p, A, y_r) \rightarrow (y_p, X_{p+1}, y_{p+1}) \dots (y_{r-1}, X_r, y_r); d], \\ \in \langle y_0(y_0, X_1, y_1) \dots (y_{r-1}, X_r, y_r) \rangle_{k+1}.$$

(iii)

$$[A \rightarrow X_{m+1} \dots X_r; w_1], [B \rightarrow X_{p+1} \dots X_r \cdot X_{r+1} \dots X_n, w_2] \\ \in \langle X_1 \dots X_r \rangle_{k+1}, w_1 \in \text{First}_{k+1}(X_{r+1} \dots X_n w_2).$$

then  $\exists v_i \dots X_i \Rightarrow^* v_i$  and  $k+1: v_{r+1} \dots v_n w_2 = w_1$ .

let  $y_i = k: v_{i+1} y_{i+1}$ , then

$$y_r = k: v_{r+1} y_{i+1} = k: v_{r+1} v_{r+2} y_{r+2} \\ = \dots = k: v_{r+1} \dots v_n y_n,$$

$$\begin{aligned} \text{and } w_1 &= k+1:v_{r+1}\dots v_n w_2 \\ &= k+1:v_{r+1}\dots v_n y_n d_2. \end{aligned}$$

Then by lem6.83,

$$\begin{aligned} [(y_m, A, y_r) \rightarrow (y_m, X_{m+1}, y_{m+1}) \dots (y_{r-1}, X_r, y_r) \cdot, d_1] \\ \in \langle y_0(y_0, X_1, y_1) \dots (y_{r-1}, X_r, y_r) \rangle_1. \end{aligned}$$

If  $(y_r, X_{r+1}, y_{r+1}) = (ax, a, xb)$ ,

$$\begin{aligned} [(ax, a, xb) \rightarrow \bullet b, 1:ud_2] \\ \in \langle y_0(y_0, X_1, y_1) \dots (y_{r-1}, X_r, y_r) \rangle_1, \\ xbu \in v_{r+1}\dots v_n y_n. \end{aligned}$$

$$\begin{aligned} \text{Then } w_1 &= k+1:v_{r+1}\dots v_n y_n d_2 = k+1:av_{r+2}\dots v_n y_n d_2 \\ &= k+1:axbud_2 = k+1:y_r bud_2. \end{aligned}$$

$\therefore \exists$  a shift-reduce conflict.

If  $(y_r, X_{r+1}, y_{r+1}) = (ax, a, x)$ , by lem6.83,  $d_2 = \$, m$

$$\begin{aligned} [(ax, a, x) \rightarrow \bullet, \$] \\ \in \langle y_0(y_0, X_1, y_1) \dots (y_{r-1}, X_r, y_r) \rangle_1. \end{aligned}$$

$$y_{r+1} = k:v_{r+2}\dots v_n y_n = v_{r+2}\dots v_n y_n$$

$$y_r = ay_{r+1} = v_{r+1}\dots v_n y_n$$

$\therefore d_1 = \$$  and  $\exists$  a reduce-reduce conflict.



**Theorem 6.85**

For any reduced grammar  $G = (N, \Sigma, P, S)$  and  $k \geq 0$ ,  
 $T_{k, 1}(G)$  is LR(1) iff  $G$  is LR( $k+1$ ).

**Theorem 6.86** For  $k \geq 1$ , any reduced grammar  $G$ ,  
 $G$  can be transformed into  $G' .\exists$ .

$G'$  is an equivalent grammar,  
 $G'$  right-to-right covers  $G$ , and  
 $G'$  is LR(1) iff  $G$  is LR( $k$ ).

**Theorem 6.87** For any alphabet  $\Sigma$ ,  
the family of deterministic languages over  $\Sigma$   
coincides with  
the family of SLR(1) languages over  $\Sigma$ .