

13. SLR 파싱테이블

충북대학교

이재성





학습내용

- 아이템 집합군(Canonical LR(0)) 구성
- First와 Follow 함수
- SLR 파싱테이블 작성 방법



SLR파싱 테이블의 구성

■ LR(0) 아이템

- 문법 G의 생성규칙 오른쪽 임의의 위치에 점이 찍혀 있는 규칙

$A \rightarrow \bullet XYZ$

$A \rightarrow X \bullet YZ$

$A \rightarrow XY \bullet Z$

$A \rightarrow XYZ \bullet$

■ SLR 방법의 중심 개념

- 바이어블 프리픽스를 인식하는 결정 유한 오토마トン 구성
- 아이템들은 SLR파서의 상태에 해당하는 집합
 - NFA의 상태에 해당
 - “부분집합구성 방법”으로 아이템들을 함께 묶음



Closure 연산

■ Closure(I), 아이템 집합의 구성

- I에 있는 모든 아이템을 closure(I)에 넣는다.
- $A \rightarrow a \bullet B\beta$ 가 closure(I)에 속해 있고, $B \rightarrow \gamma$ 가 생성규칙이라면 아이템 $B \rightarrow \bullet \gamma$ 를 I에 추가 (단, $B \rightarrow \bullet \gamma$ 가 closure(I)에 없을 경우)

■ Goto 연산

- goto(I, X): I는 아이템 집합, X는 문법기호
- goto(I, X)는 $A \rightarrow \alpha \bullet X\beta$ 가 I에 있을 때 모든 아이템 $A \rightarrow \alpha X \bullet \beta$ 의 closure로 정의
- I가 바이어블 프리픽스 γ 에 유효하다면, goto(I, X)는 바이어블 프리픽스 γX 에 대한 유효한 아이템 집합

$$\begin{array}{ccc} I: & \Rightarrow & \text{goto}(I, X): \\ \gamma \bullet X\beta & & \gamma X \bullet \beta \\ \gamma \bullet Y & & \end{array}$$



아이템 집합군 구성

canonical LR(0)의 아이템 집합군 구성 알고리즘

```
procedure items(G');
begin
    C := {closure({ [S' -> S] })};
    repeat
        for C의 각 아이템 집합 I와 문법기호 X에 대해
            (단, goto(I, X)는 빈 것이 아니고 C에도 포함되지 않음)
        do
            goto(I, X)를 C에 추가
        until C에 추가할 아이템 집합이 더 이상 없다.
end
```



Canonical LR(0) 아이템 구성 예(1)

문법확장후, 아이템

집합군 구성

$$E' \rightarrow E$$

$$E \rightarrow E + T$$

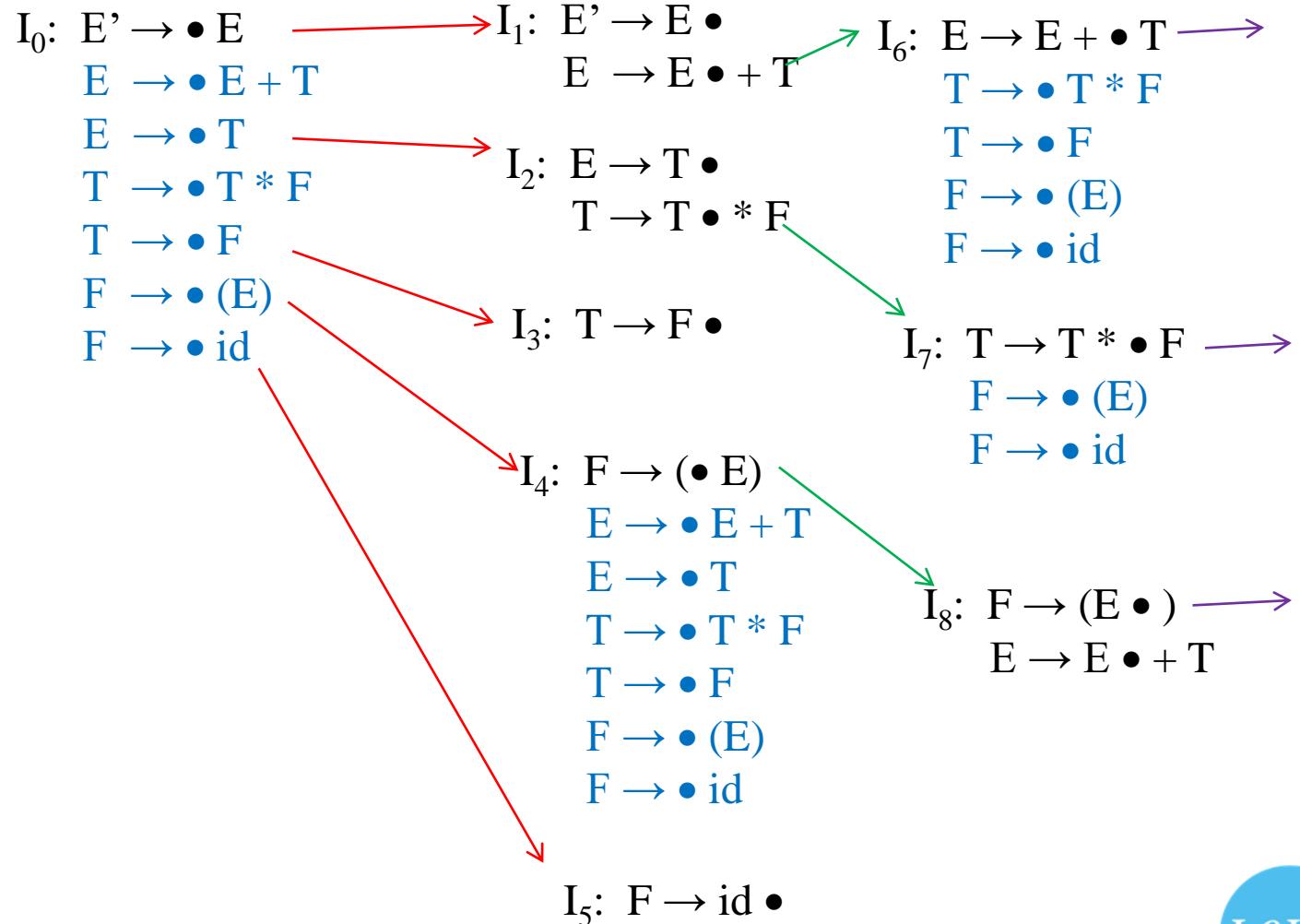
$$E \rightarrow T$$

$$T \rightarrow T * F$$

$$T \rightarrow F$$

$$F \rightarrow (E)$$

$$F \rightarrow id$$



13. SLR 파싱 테이블



Canonical LR(0) 아이템 구성 예(2)

I₀: E' → • E

E → • E + T

E → • T

T → • T * F

T → • F

F → • (E)

F → • id

I₄: F → (• E)

E → • E + T

E → • T

T → • T * F

T → • F

F → • (E)

F → • id

I₇: T → T * • F

F → • (E)

F → • id

I₈: F → (E •)

E → E • + T

I₁: E' → E •

E → E • + T

I₅: F → id •

I₉: E → E + T •

T → T • * F

I₂: E → T •

T → T • * F

I₆: E → E + • T

T → • T * F

I₁₀: T → T * F •

T → • F

I₃: T → F •

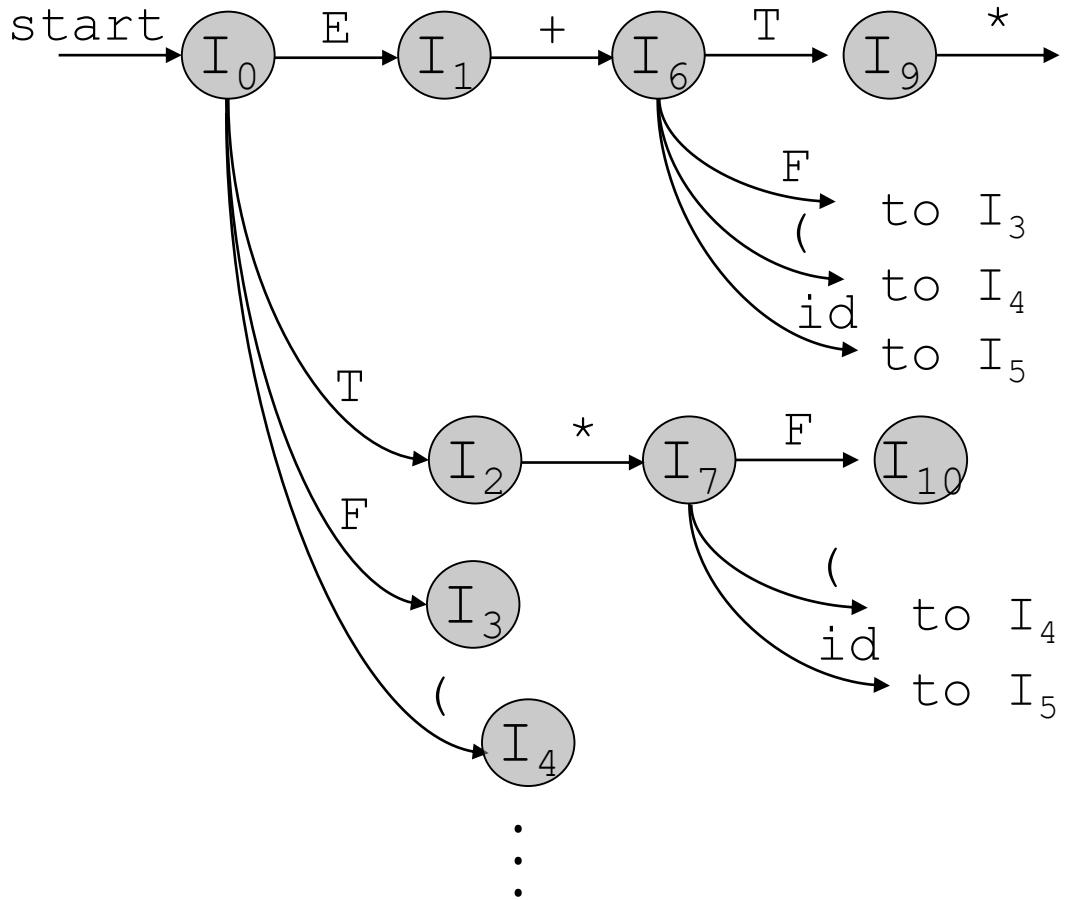
F → • (E)

F → • id

I₁₁: F → (E) •



바이어블 프리픽스에 대한 DFA D의 전이도 예



to I_7

$I_4: F \rightarrow (\bullet E)$
⋮

$I_0: E' \rightarrow \bullet E$
 $E \rightarrow \bullet E + T$
 $E \rightarrow \bullet T$
 $T \rightarrow \bullet T * F$
 $T \rightarrow \bullet F$
 $F \rightarrow \bullet (E)$
 $F \rightarrow \bullet id$

$I_6: E \rightarrow E + \bullet T$
 $T \rightarrow \bullet T * F$
 $T \rightarrow \bullet F$
 $F \rightarrow \bullet (E)$
 $F \rightarrow \bullet id$

$I_1: E' \rightarrow E \bullet$
 $E \rightarrow E \bullet + T$

$I_2: E \rightarrow T \bullet$
 $T \rightarrow T \bullet * F$

$I_9: E \rightarrow E + T \bullet$
 $T \rightarrow T \bullet * F$

$I_3: T \rightarrow F \bullet$



First

■ FIRST(X)

- 문법기호 X의 맨 처음에 나올 수 있는 단말기호
- 규칙
 1. X가 단말이면 $\text{FIRST}(X) = \{X\}$
 2. 만약 $X \rightarrow \epsilon$ 가 있으면, $\text{FIRST}(X)$ 에 ϵ 를 추가
 3. X가 비단말이고, $X \rightarrow Y_1Y_2\dots Y_k$ 라는 생성식이 있을 경우
 - $\text{FIRST}(Y_1)$ 의 원소 중 ϵ 가 아닌 모든 기호를 추가
 - $\text{FIRST}(Y_1)$ 에 ϵ 원소가 있었을 경우,
 - $\text{FIRST}(Y_2)$ 의 ϵ 가 아닌 모든 원소 추가
 - $\text{FIRST}(Y_1)$ 과 $\text{FIRST}(Y_2)$ 에 모두 ϵ 가 있을 경우,
 - $\text{FIRST}(Y_3)$ 의 ϵ 가 아닌 모든 원소추가
 - 위의 과정 반복
 - 모든 i에 대해 $\text{FIRST}(Y_i)$ 가 ϵ 를 가지고 있다면 $\text{FIRST}(X)$ 에 ϵ 추가



Follow

■ FOLLOW(A)

1. FOLLOW(S)에 \$를 넣음 (S: 시작기호, \$: 입력의 끝)
2. $A \rightarrow \alpha B \beta$ 라는 생성식이 있으면,
 - ε 를 제외한 FIRST(β)의 모든 원소들은 FOLLOW(B)에 속함
3. $A \rightarrow \alpha B \beta$ 라는 생성식이 있고, $A \rightarrow \alpha B$ 라는 생성식이 있거나 FIRST(β)가 ε 를 가질 경우,
 - FOLLOW(A)의 모든 원소는 FOLLOW(B)에 속함



FIRST 와 FOLLOW 예 1

$\text{FIRST}(E) = \text{FIRST}(T) = \text{FIRST}(F) = \{ (, \text{id}) \}$

$E \rightarrow E + T$

$E \rightarrow T$

$\text{FOLLOW}(E) = \{ +,), \$ \}$

$T \rightarrow T * F$

$T \rightarrow F$

$\text{FOLLOW}(T) = \text{FOLLOW}(E) \cup \{ * \} = \{ +,), \$, * \}$

$F \rightarrow (E)$

$F \rightarrow \text{id}$

$\text{FOLLOW}(F) = \text{FOLLOW}(T)$



FIRST 와 FOLLOW 예2

$\text{FIRST}(E) = \text{FIRST}(T) = \text{FIRST}(F) = \{ (, \text{id}) \}$

$\text{FIRST}(E') = \{ +, \epsilon \}$

$\text{FIRST}(T') = \{ *, \epsilon \}$

$\text{FOLLOW}(E) = \{ \}, \$ \}$

$\text{FOLLOW}(E) = \text{FOLLOW}(E')$

$E \rightarrow TE'$

$E' \rightarrow +TE' | \epsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT' | \epsilon$

$F \rightarrow (E) | \text{id}$

$\text{FOLLOW}(T) = \text{FIRST}'(E') \cup \text{FOLLOW}(E) = \{ +, \}, \$ \} (\epsilon \text{ 제외})$

$\text{FOLLOW}(F) = \text{FIRST}'(T') \cup \text{FOLLOW}(T) = \{ +, *, \}, \$ \} (\epsilon \text{ 제외})$



SLR파싱 테이블의 구성 알고리즘

- 입력: 확장문법 G'
- 출력: G'에 대한 SLR 파싱 테이블 함수 action과 goto
- 방법
 1. G'에 대한 LR(0) 아이템 집합군 C={I₀, I₁, ..., I_n}구성
 2. I_i에서 상태i를 구성
 - action[i, a]=shift j
 - [A → α • aβ]가 I_i안에 있고 goto(I_i, a)=I_j일 경우
 - action[i, a]= reduce A → α (FOLLOW(A)에 있는 모든 a)
 - [A → α •]이 I_i안에 있을 경우
 - action[i, \$] = accept
 - [S' → S •]이 I_i에 있을 경우
 3. 상태 i에서 goto전이
 - goto(I_i, A) = I_j라면 goto[i, A]=j
 4. 기타 항목은 여러 (2,3에서 정의되지 않은 것)
 5. 초기 상태는 [S' → S]를 담고 있는 아이템 집합



SLR파싱 테이블 작성 예

$I_0: E' \rightarrow \bullet E$
 $E \rightarrow \bullet E + T$
 $E \rightarrow \bullet T$
 $T \rightarrow \bullet T * F$
 $T \rightarrow \bullet F$
 $F \rightarrow \bullet (E)$
 $F \rightarrow \bullet id$

$goto(0, E)=1$
 $goto(0, T)=2$
 $goto(0, F)=3$

\longrightarrow action[0, ()]=shift 4
 \longrightarrow action[0, id]=shift 5

$I_1: E' \rightarrow E \bullet$
 $E \rightarrow E \bullet + T$

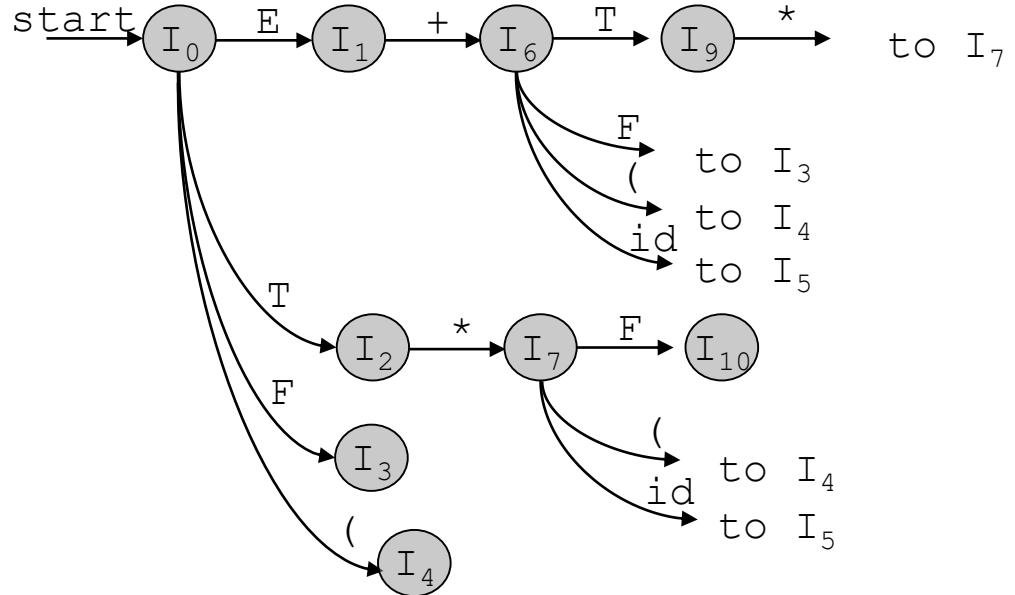
\longrightarrow action[1, \$]=accept
 \longrightarrow action[1, +]=shift 6

$I_2: E \rightarrow T \bullet$
 $T \rightarrow T \bullet * F$

\longrightarrow FOLLOW(E) = {\$, +, ()\$} 0|므로 :
 $action[2, $]=action[2, +]=action[2, ()]=reduce E \rightarrow T$
 \longrightarrow action[2, *]=shift 7

$I_3: T \rightarrow F \bullet$

\longrightarrow FOLLOW(T) = {\$, +, (), *\$} 0|므로
 $action[3, $]=action[3, +]=action[3, ()]=action[3, *]=reduce T \rightarrow F$





SLR 파싱테이블 예

상태	action					goto			
	id	+	*	()	\$	E	T	F
0	s5			s4			1	2	3
1		s6			acc				
2		r2	s7		r2	r2			
3		r4	r4		r4	r4			
4	s5			s4			8	2	3
5		r6	r6		r6	r6			
6	s5			s4			9	3	
7	s5			s4				10	
8		s6			s11				
9		r1	s7		r1	r1			
10		r3	r3		r3	r3			
11		r5	r5		r5	r5			

문법

1. $E \rightarrow E + T$
2. $E \rightarrow T$
3. $T \rightarrow T * F$
4. $T \rightarrow F$
5. $F \rightarrow (E)$
6. $F \rightarrow id$



참고 문헌

- [1] Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, “Compilers – Principles, Techniques, and Tools,” Bell Telephone Laboratories, Incorporated, 1986.
- [2] 오세만, “컴파일러 입문”, 정의사, 2004.