

## Conflits LR(0), grammaire LR(0)

L'automate LR(0) construit peut ne pas être déterministe (2 cas).

État autorisant 2 réductions (ou plus) :

conflit LR(0) **reduce/reduce**

Ex :

$[A \rightarrow b\bullet]$
$[B \rightarrow b\bullet]$

État autorisant 1 réduction et 1 lecture (ou plus) :

conflit LR(0) **shift/reduce**

Ex :

$[A \rightarrow \bullet b]$
$[B \rightarrow c\bullet]$

## Conflits LR(0), grammaire LR(0)

Une grammaire est dite **LR(0)** si aucun de ses états ne contient de conflit LR(0) :

- ▶ ni shift-reduce
- ▶ ni reduce-reduce

Les conflits shift/shift n'existent pas (aucun sens).

## Conflicts shift/reduce au sens SLR(1)

Un état de LR-AFD provoque un **conflit S/R** au sens **SLR(1)** s'il contient à la fois :

- ▶ un item de la forme  $[Y \rightarrow \dots \bullet a \dots]$
- ▶ un item de la forme  $[X \rightarrow \alpha \bullet]$  avec  $a \in \text{Suivant}(X)$

Comparer avec LR(0) : conflit S/R au sens LR(0) si l'état contient les items  $[Y \rightarrow \dots \bullet a \dots]$  et  $[X \rightarrow \alpha \bullet]$

## Conflits reduce/reduce au sens SLR(1)

Un état de LR-AFD provoque un **conflit R/R** au sens **SLR(1)** s'il contient à la fois :

- ▶ un item de la forme  $[Y \rightarrow \beta \bullet]$
- ▶ un item de la forme  $[X \rightarrow \alpha \bullet]$
- ▶ avec  $Suivant(X) \cap Suivant(Y) \neq \emptyset$

Comparer avec LR(0) : conflit R/R au sens LR(0) si l'état contient les items  $[Y \rightarrow \beta \bullet]$  et  $[X \rightarrow \alpha \bullet]$