

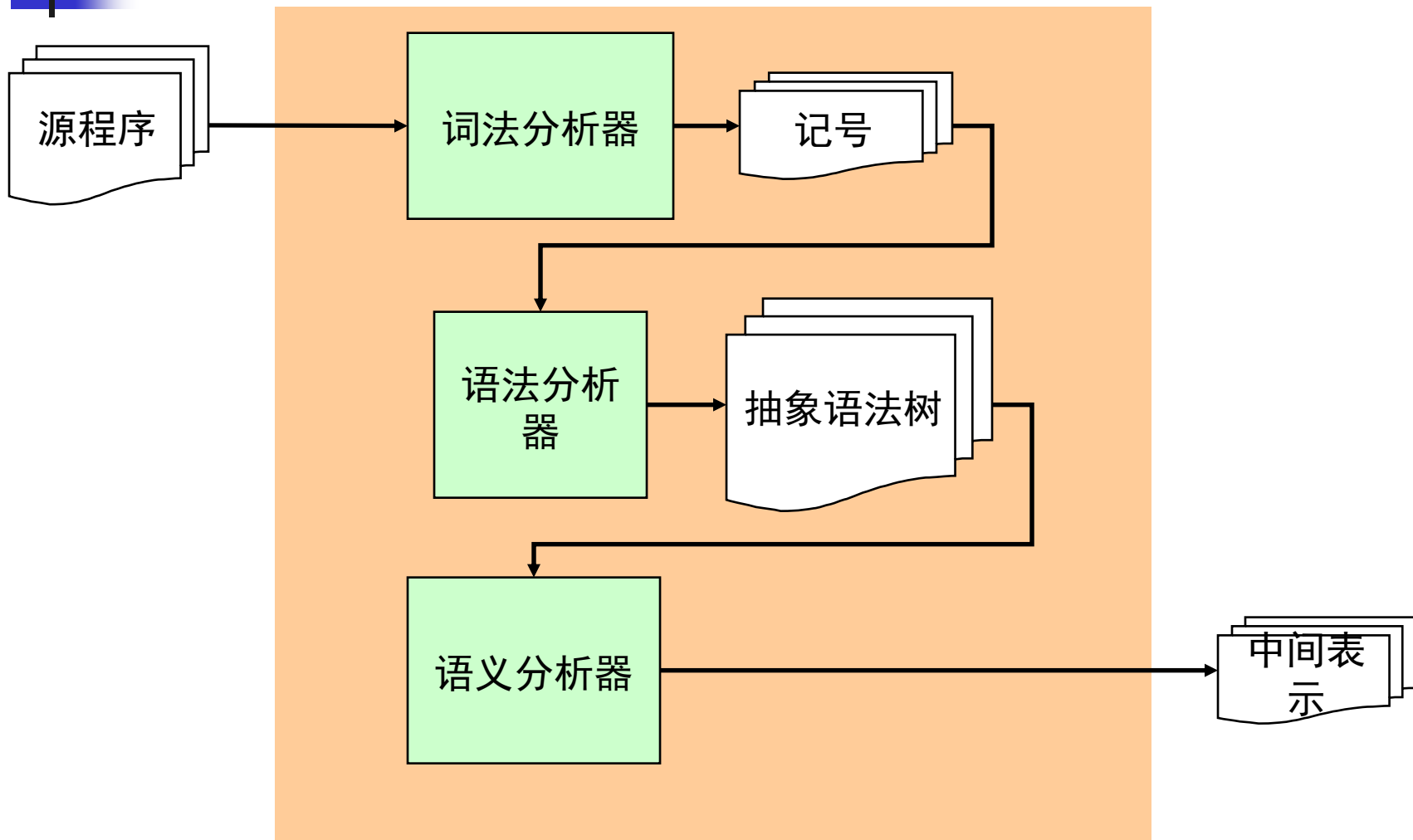
语法分析： LR(1)分析算法

编译原理

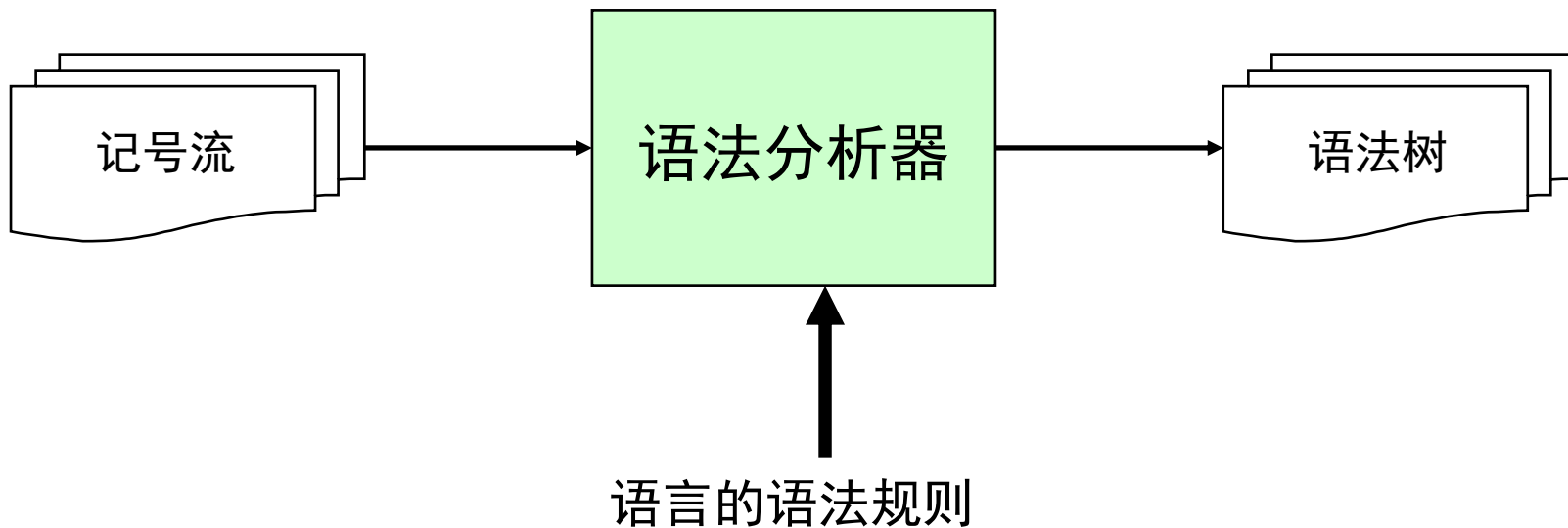
华保健

bjhua@ustc.edu.cn

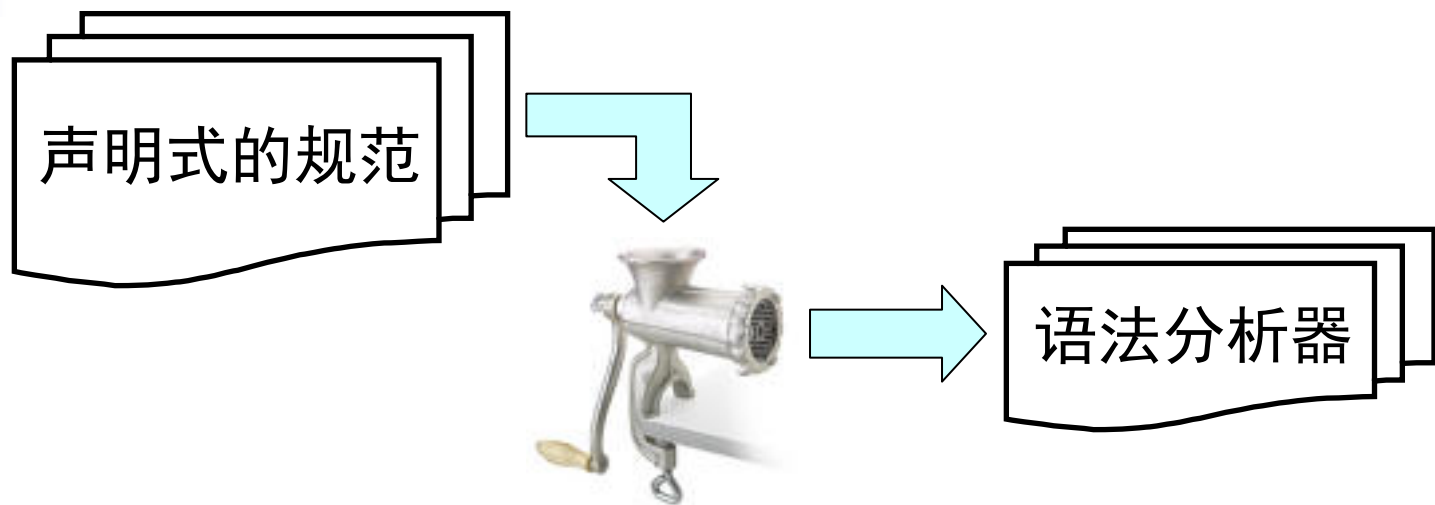
前端



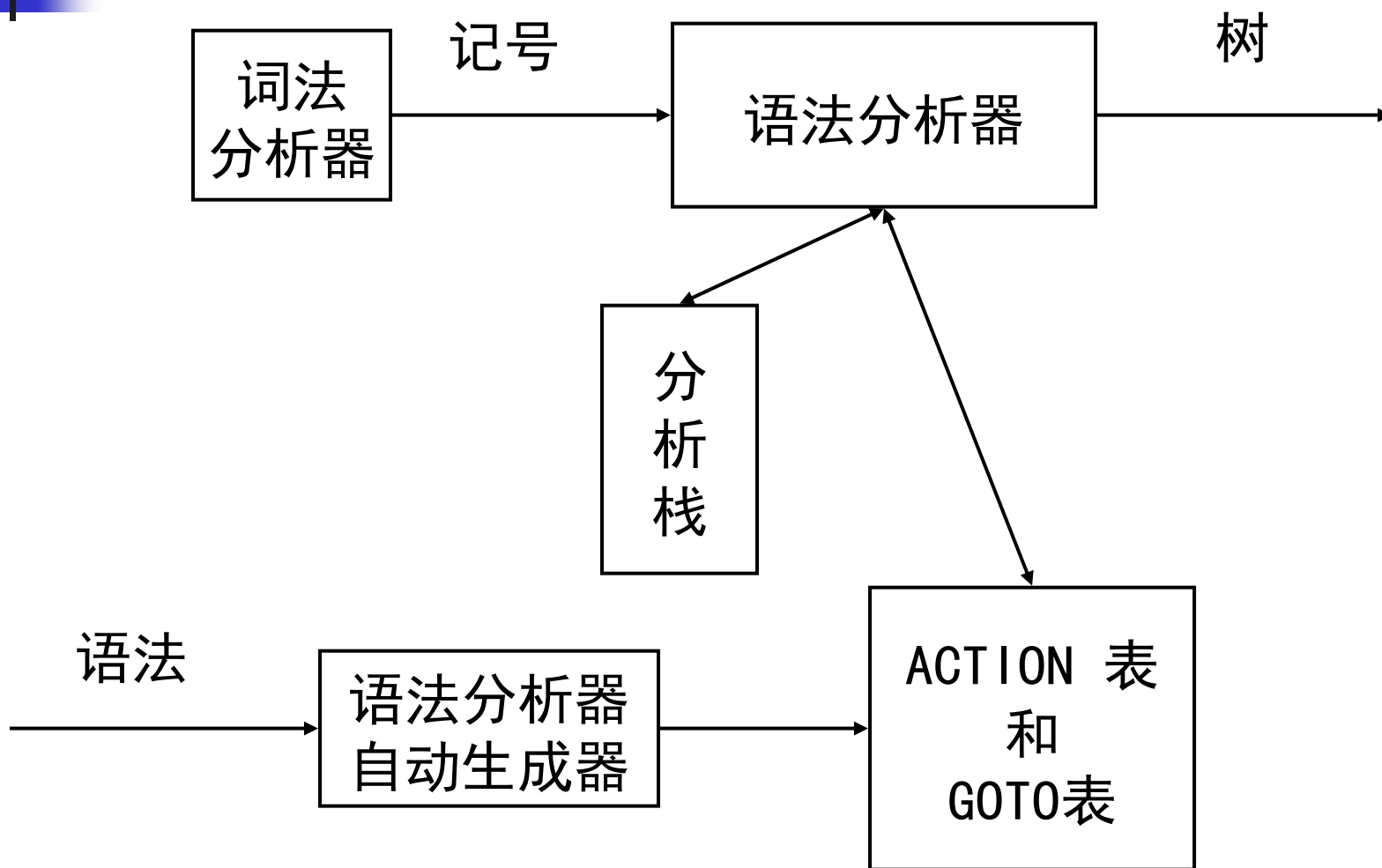
语法分析器的任务



自动生成



表驱动的LR分析器架构



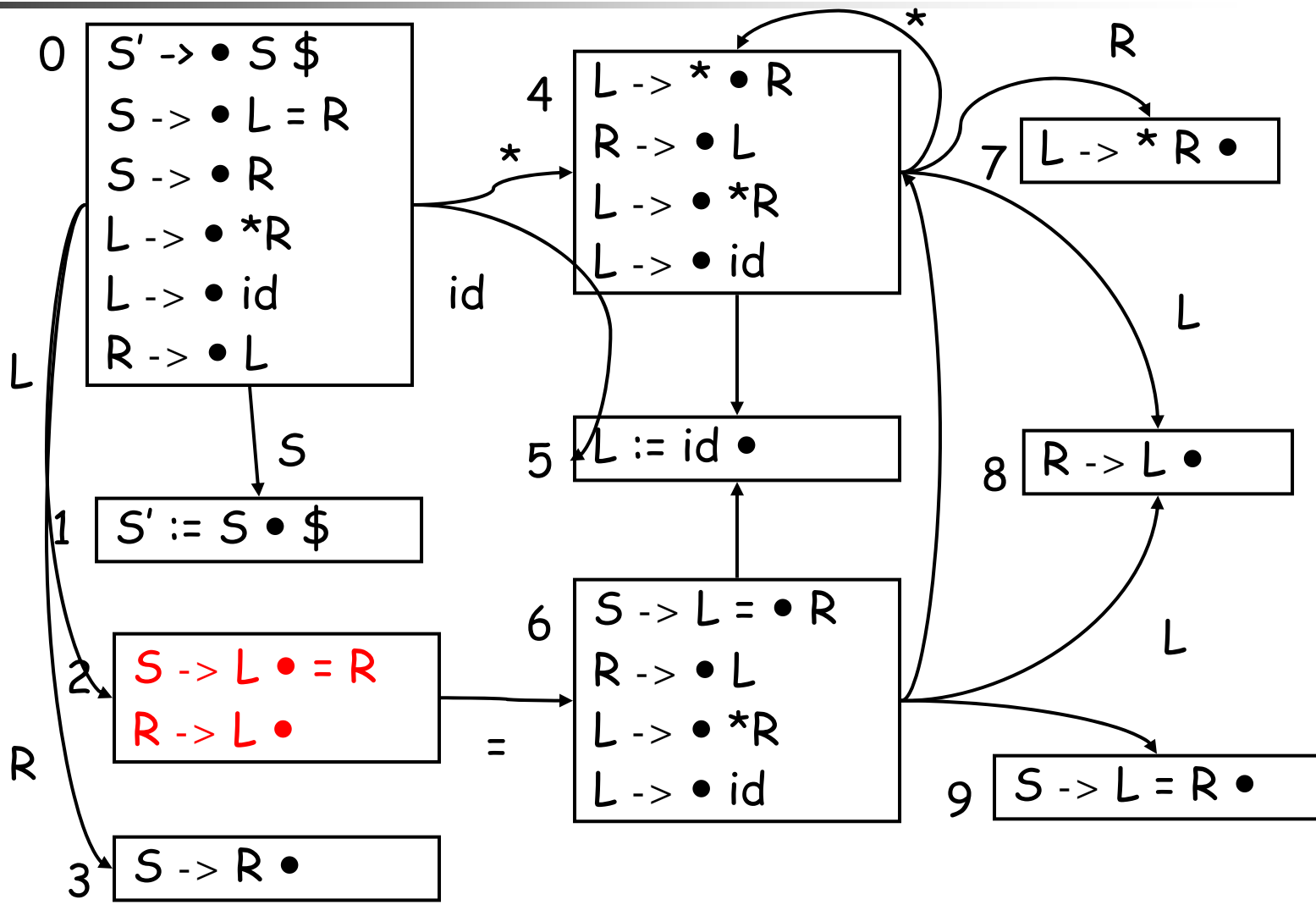


SLR分析算法的思想

- 基于LR(0)，通过进一步判断一个前看符号，来决定是否执行归约动作
 - $X \rightarrow \alpha$ • 归约，当且仅当 $y \in FOLLOW(X)$
- 优点：
 - 有可能减少需要归约的情况
 - 有可能去除需要移进-归约冲突
- 缺点：
 - 仍然有冲突出现的可能

SLR分析表中的冲突

$S' \rightarrow S\$$
 $S \rightarrow L = R$
 $S \rightarrow R$
 $L \rightarrow * R$
 $L \rightarrow id$
 $R \rightarrow L$





LR(1)项目

- $[X \rightarrow \alpha \bullet \beta, a]$ 的含义是:
 - α 在栈顶上
 - 剩余的输入能够匹配 βa
- 当归约 $X \rightarrow \alpha\beta$ 时, a 是前看符号
 - 把 'reduce by $X \rightarrow \alpha\beta$ ' 填入 $\text{ACTION}[s, a]$

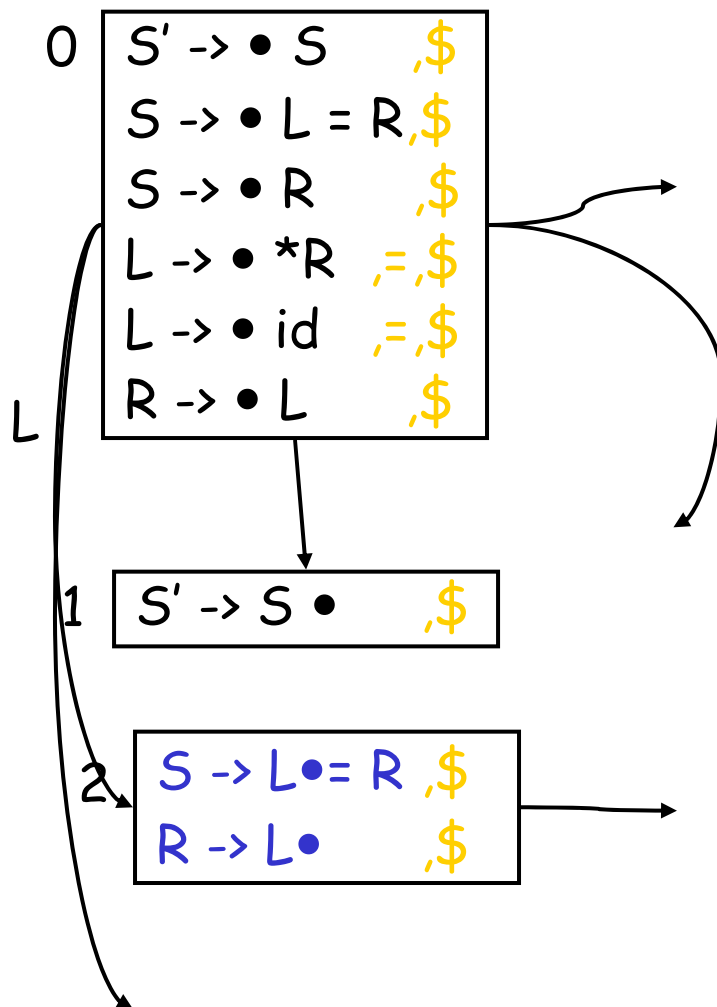


LR(1)项目的构造

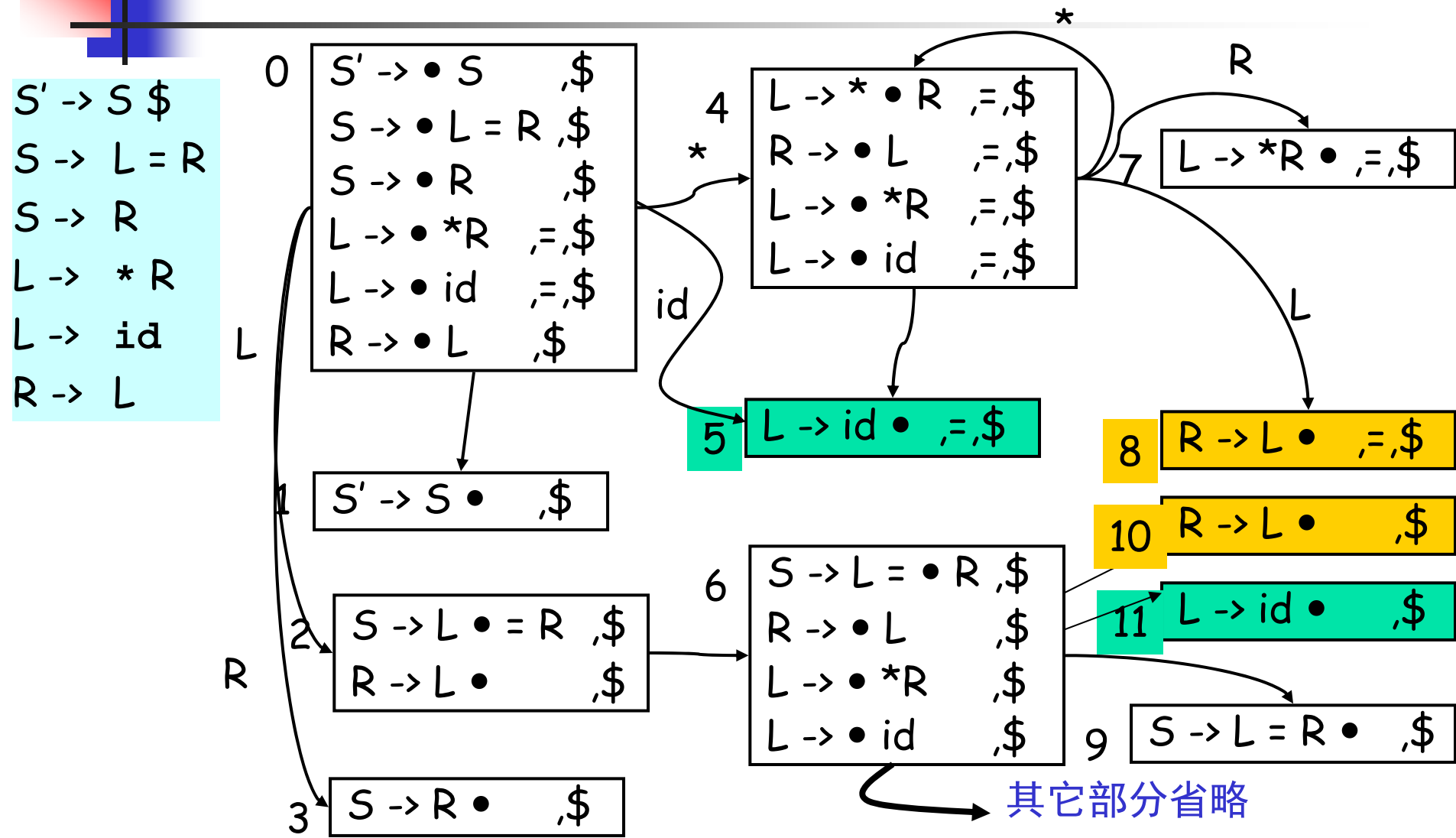
- 其他和LR(0)相同，仅闭包的计算不同：
 - 对项目 $[X \rightarrow \alpha \bullet Y \beta, a]$
 - 添加 $[Y \rightarrow \bullet \gamma, b]$ 到项目集，
 - 其中 $b \in \text{FIRST}_S(\beta a)$

LR(1)项目集 (部分)

$S' \rightarrow S\$$
 $S \rightarrow L = R$
 $S \rightarrow R$
 $L \rightarrow * R$
 $L \rightarrow id$
 $R \rightarrow L$



更多项目集





LALR分析算法

- 把类似的项目集进行合并
- 需要修改ACTION表和GOTO表，以反映合并的效果



对二义性文法的处理

- 二义性文法无法使用LR分析算法分析
- 不过，有几类二义性文法很容易理解，因此，在LR分析器的生成工具中，可以对它们特殊处理
 - 优先级
 - 结合性
 - 悬空else
 - 。 。 。

优先级和结合性

$$\begin{array}{l} E \rightarrow E + E \\ | E * E \\ | n \end{array}$$
