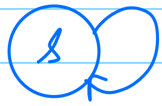
Equivalence between PDAs and CFGs

* Going from CFGs to PDAs

$G = (N, \Sigma, P, S)$ $\qquad\qquad\qquad$ $\Gamma = N \cup \Sigma \cup \{\perp\}$



$\varepsilon, \perp \to S\perp$

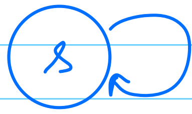
$\varepsilon, A \to \alpha$ $\quad$ if $A \to \alpha \in P$

$\sigma, \sigma \to \varepsilon$ $\quad$ $\sigma \in \Sigma$

$\varepsilon, \perp \to \varepsilon$

Example

Dyck Language: $S \to [S] \mid SS \mid \varepsilon$



$\varepsilon, \perp \to S\perp$

$\varepsilon, S \to [S]$

$\varepsilon, S \to SS$

$\varepsilon, S \to \varepsilon$

$[, [\to \varepsilon$

$],] \to \varepsilon$

$\varepsilon, \perp \to \varepsilon$

$\omega = [\,]\,[\,[\,]\,]$

stack contents during execution

$\perp,\; S\perp,\; SS\perp,\; [S]S\perp,\; S]S\perp,\;]S\perp,\; S\perp,\; [S]\perp,\; S]\perp,\; [S]]\perp,\; S]]\perp,\;]]\perp,\;]\perp,\; \perp,\; \varepsilon$

* Going from PDAs to CFGs

① PDAs $\rightarrow$ PDAs with a single accept state & that empties the stack.

Add ε-transitions to the single final state & empty the stack

② PDAs with one final state & empty stack

$\rightarrow$ PDA with a single state that accepts via the empty stack

$\downarrow$

CFGs

$M = (Q, \Sigma, \Gamma, \delta, s, \perp, \{f\})$

$M' = (\{q\}, \Sigma, \Gamma', \delta', \{q\}, (s, \perp, f), \phi\}$

$\Gamma' = Q \times \Gamma \times Q$

Let $\gamma \in \Gamma$. M on x starts with only γ on the stack from state p and reaches p' after reading x and emptying the stack

$\iff$ M' on x starting with (p, γ, p') on the stack, empties the stack after reading x.

if $(p, \sigma, A), (p', B_1 B_2 \cdots B_k) \in \delta$

do the following:

$\forall\ q_0, q_1, \ldots, q_k$ where $q_0 = p'$

add $(q, \sigma, (pAq_k)), (q, (q_0 B_1 q_1)(q_1 B_2 q_2) \cdots$

$\cdots (q_{k-1} B_k q_k))$

if $((p, \sigma, A), (p', \varepsilon)) \in \delta$

add $((q, \sigma, (pAp')), (q, \varepsilon))$ to δ'

Converting PDA with one state to CFG

if $((q, \sigma, A), (q, B_1 \ldots B_k)) \in \delta$

Add the production $A \to \sigma B_1 B_2 \cdots B_k$