Pushdown Automata - Equivalence to CFGs

Lecture 19
Section 2.2

Robb T. Koether

Hampden-Sydney College

Wed, Oct 10, 2012

Outline

1. Equivalence of PDAs and CFGs
   - Proof $\Rightarrow$
   - Proof $\Leftarrow$
2. Long Example
3. Short Example
4. Assignment

Outline

1. Equivalence of PDAs and CFGs
   - Proof $\Rightarrow$
   - Proof $\Leftarrow$
2. Long Example
3. Short Example
4. Assignment

Equivalence of PDAs and CFGs

Theorem (Equivalence of PDAs and CFGs)

A language is context-free if and only if it is accepted by some PDA.

Outline

1. Equivalence of PDAs and CFGs
   - Proof $\Rightarrow$
   - Proof $\Leftarrow$
2. Long Example
3. Short Example
4. Assignment

Proof ($\Rightarrow$).

- We are given a CFG G and we will construct a PDA M.
- Let M have four states:
  - $Q = \{q_0, q_1, q_2, q_3\}$.
  - q_0 is the start state.
  - q_3 is the accept state.

$\square$

Equivalence of PDAs and CFGs

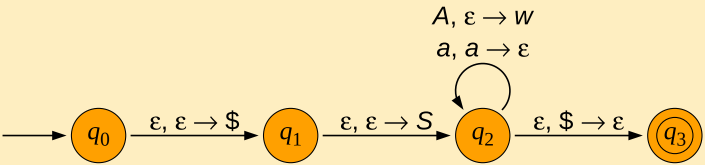
Proof ($\Rightarrow$).

- The transitions are
  - $\delta(q_0, \varepsilon, \varepsilon) = \{(q_1, \$)\}$
  - $\delta(q_1, \varepsilon, \varepsilon) = \{(q_2, S)\}$
  - $\delta(q_2, \varepsilon, A) = \{(q_2, w)\}$, where $A \to w$ is a rule.
  - $\delta(q_2, a, a) = \{(q_2, \varepsilon)\}$, for all $a \in \Sigma$.
  - $\delta(q_2, \varepsilon, \$) = \{(q_3, \varepsilon)\}$
- It is clear that $L(M) = L(G)$.

$\square$

Equivalence of PDAs and CFGs

Proof (⇒).



□

Outline

1. Equivalence of PDAs and CFGs
   - Proof $\Rightarrow$
   - Proof $\Leftarrow$
2. Long Example
3. Short Example
4. Assignment

Proof ($\Leftarrow$).

- Given a PDA M, we must construct a grammar G that generates $L(M)$.
- Modify M so that
  - M has a single accept state.
  - M empties its stack before accepting.
  - Each transition either pushes one symbol or pops one symbol, but not both.

$\square$

The Variables

Proof ($\Leftarrow$).

- For every pair of states p and q in M, we create a variable A_{pq}.
- The variable A_{pq} is interpreted to mean

  "Get from state p with an empty stack to state q with an empty stack."
- Since the stack starts and ends empty, we must push a symbol at the beginning and pop a symbol at the end.

$\square$

The Grammar Rules

Proof ($\Leftarrow$).

- There are two possibilities in going from p to q.
  - The first symbol pushed does not match the last symbol popped.
  - The first symbol pushed matches the last symbol popped.

$\square$

The Grammar Rules

Proof (⇐).

- For the rules in Group 1, use all pairs of transitions where the first transition pushes a symbol and the second transition pops a symbol, whether or not they are the same symbol.
- For each such pair of transitions $\delta(p, a, \varepsilon) = (s, c)$ and $\delta(t, b, d) = (q, \varepsilon)$, write the rules

$$A_{pq} \to A_{pr}A_{rq}$$

for all intermediate states $r \in Q$.

□

The Grammar Rules

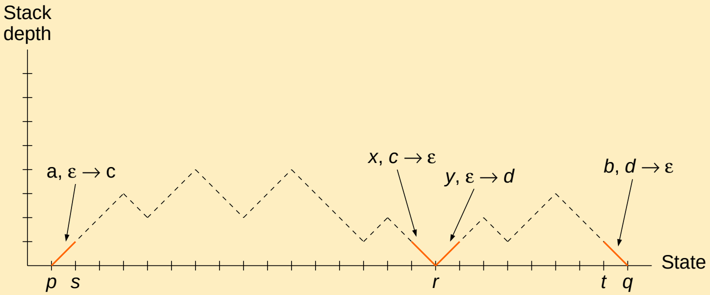
Proof ($\Leftarrow$).

- The Group 1 rules allow us to break the processing up into segments that begin and end with an empty stack, and have an empty stack at some intermediate point.

$\square$

The Grammar Rules

Proof (⇐).



□

The Grammar Rules

Proof ($\Leftarrow$), continued.

- For the rules in Group 2, use all pairs of transitions where the second transition pops the same symbol that the first transition pushes.
- For each such pair of transitions $\delta(p, a, \varepsilon) = (r, c)$ and $\delta(s, b, c) = (q, \varepsilon)$, write the rule

$$A_{pq} \to aA_{rs}b.$$

$\square$

The Grammar Rules

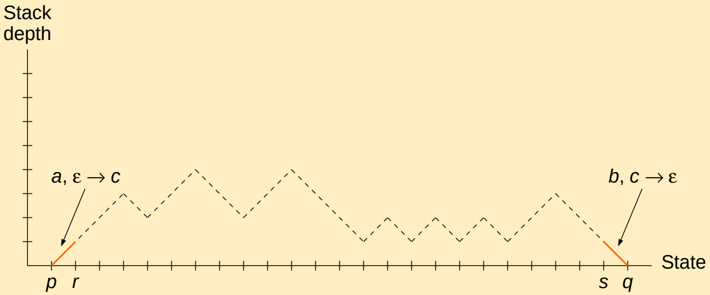
Proof ($\Leftarrow$).

- The Group 2 rules allow us to break the processing up into segments that begin and end with an empty stack, but during which the stack is never empty.

$\square$

The Grammar Rules

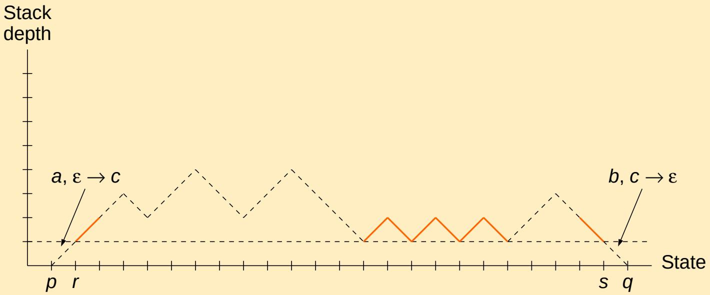
Proof ($\Leftarrow$).



$\square$

The Grammar Rules

Proof (⇐).



□

The Grammar Rules

Proof ($\Leftarrow$), continued.

- Finally, the Group 3 rules are of the form

$$A_{pp} \rightarrow \varepsilon$$

for every state p.

$\square$

The Start Symbol

Proof ($\Leftarrow$), conclusion.

- Let q_{start} be the start state and q_{accept} be the accept state.
- Then the start symbol is $A_{q_{\text{start}}q_{\text{accept}}}$.
- If we can eventually replace the start symbol with all terminals, then the string of terminals must be in the language.
- Therefore, $L(G) = L(M)$.

$\square$

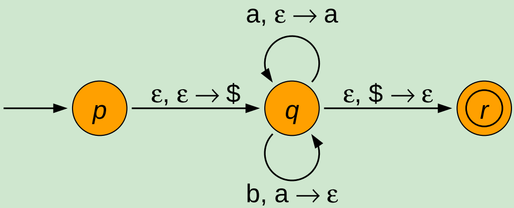
Outline

1. Equivalence of PDAs and CFGs
   - Proof $\Rightarrow$
   - Proof $\Leftarrow$
2. Long Example
3. Short Example
4. Assignment

Example

Example (Convert a PDA to a CFG)

- Find a grammar for the following PDA.



Example

Example (Convert a PDA to a CFG)

- The variables are A_{pp}, A_{pq}, A_{pr}, A_{qp}, A_{qq}, A_{qr}, A_{rp}, A_{rq}, and A_{rr}.

The Grammar Rules (Group 1)

Example (Convert a PDA to a CFG)

- Then we get the Group 1 grammar rules, based on the transitions.

Pushes	Pops
$\delta(p, \varepsilon, \varepsilon) = (q, \$)$	$\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$
$\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$	$\delta(q, \varepsilon, \$) = (r, \varepsilon)$

- Find all pairs of transitions where
  - The first transition pushes a symbol x.
  - The second transition pops a symbol y.

The Grammar Rules (Group 1)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(p, \varepsilon, \varepsilon) = (q, \$)$
  - $\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$

  gives the rules $A_{pq} \to A_{px}A_{xq}$ for all $x \in Q$.
- That is,

$$
\begin{aligned}
A_{pq} &\to A_{pp}A_{pq} \\
A_{pq} &\to A_{pq}A_{qq} \\
A_{pq} &\to A_{pr}A_{rq}
\end{aligned}
$$

The Grammar Rules (Group 1)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(p, \varepsilon, \varepsilon) = (q, \$)$
  - $\delta(q, \varepsilon, \$) = (r, \varepsilon)$

  gives the rules $A_{pr} \to A_{px}A_{xr}$ for all $x \in Q$.
- That is,

$$A_{pr} \to A_{pp}A_{pr}$$
$$A_{pr} \to A_{pq}A_{qr}$$
$$A_{pr} \to A_{pr}A_{rr}$$

The Grammar Rules (Group 1)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$
  - $\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$

  gives the rules $A_{qq} \to A_{qx}A_{xq}$ for all $x \in Q$.
- That is,

$$\begin{aligned} A_{qq} &\to A_{qp}A_{pq} \\ A_{qq} &\to A_{qq}A_{qq} \\ A_{qq} &\to A_{qr}A_{rq} \end{aligned}$$

The Grammar Rules (Group 1)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$
  - $\delta(q, \varepsilon, \$) = (r, \varepsilon)$

  gives the rules $A_{qr} \to A_{qx}A_{xr}$ for all $x \in Q$.
- That is,

$$\begin{aligned} A_{qr} &\to A_{qp}A_{pr} \\ A_{qr} &\to A_{qq}A_{qr} \\ A_{qr} &\to A_{qr}A_{rr} \end{aligned}$$

The Grammar Rules (Group 1)

Example (Convert a PDA to a CFG)

- We get the first group of rules.

$$\begin{array}{ll}
A_{pq} \to A_{pp}A_{pq} & A_{qq} \to A_{qp}A_{pq} \\
A_{pq} \to A_{pq}A_{qq} & A_{qq} \to A_{qq}A_{qq} \\
A_{pq} \to A_{pr}A_{rq} & A_{qq} \to A_{qr}A_{rq} \\
A_{pr} \to A_{pp}A_{pr} & A_{qr} \to A_{qp}A_{pr} \\
A_{pr} \to A_{pq}A_{qr} & A_{qr} \to A_{qq}A_{qr} \\
A_{pr} \to A_{pr}A_{rr} & A_{qr} \to A_{qr}A_{rr}
\end{array}$$

The Grammar Rules (Group 1)

Example (Convert a PDA to a CFG)

- We may eliminate all rules with variables that represent impossible situations.

$$
\begin{array}{llll}
A_{pq} \rightarrow A_{pp}A_{pq} & & A_{qq} \rightarrow A_{qp}A_{pq} \\
A_{pq} \rightarrow A_{pq}A_{qq} & & A_{qq} \rightarrow A_{qq}A_{qq} \\
A_{pq} \rightarrow A_{pr}A_{rq} & & A_{qq} \rightarrow A_{qr}A_{rq} \\
A_{pr} \rightarrow A_{pp}A_{pr} & & A_{qr} \rightarrow A_{qp}A_{pr} \\
A_{pr} \rightarrow A_{pq}A_{qr} & & A_{qr} \rightarrow A_{qq}A_{qr} \\
A_{pr} \rightarrow A_{pr}A_{rr} & & A_{qr} \rightarrow A_{qr}A_{rr}
\end{array}
$$

The Grammar Rules (Group 1)

Example (Convert a PDA to a CFG)

- The Group 1 rules are

$$
\begin{aligned}
A_{pq} &\rightarrow A_{pp}A_{pq} \\
A_{pq} &\rightarrow A_{pq}A_{qq} \\
A_{pr} &\rightarrow A_{pp}A_{pr} \\
A_{pr} &\rightarrow A_{pq}A_{qr} \\
A_{pr} &\rightarrow A_{pr}A_{rr} \\
A_{qq} &\rightarrow A_{qq}A_{qq} \\
A_{qr} &\rightarrow A_{qq}A_{qr} \\
A_{qr} &\rightarrow A_{qr}A_{rr}
\end{aligned}
$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- Then we get the Group 2 grammar rules, based on the transitions.

Pushes	Pops
$\delta(p, \varepsilon, \varepsilon) = (q, \$)$	$\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$
$\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$	$\delta(q, \varepsilon, \$) = (r, \varepsilon)$

- Find all pairs of transitions where
  - The first transition pushes a symbol x.
  - The second transition pops the *same* symbol x.

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- For every stack symbol s and for every pair of transitions, one of which pushes s and the other of which pops s, we write a grammar rule

$$A_{xy} \to aA_{zw}b$$

where a is the symbol read when s is pushed and b is the symbol read when s is popped.

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(p, \varepsilon, \varepsilon) = (q, \$)$
  - $\delta(q, \varepsilon, \$) = (r, \varepsilon)$

  gives the rule

$$A_{pr} \to \varepsilon A_{qq} \varepsilon = A_{qq}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(p, \varepsilon, \varepsilon) = (q, \$)$
  - $\delta(q, \varepsilon, \$) = (r, \varepsilon)$

  gives the rule

$$A_{pr} \to \varepsilon A_{qq} \varepsilon = A_{qq}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(p, \varepsilon, \varepsilon) = (q, \$)$
  - $\delta(q, \varepsilon, \$) = (r, \varepsilon)$

  gives the rule

$$A_{pr} \rightarrow \varepsilon A_{qq} \varepsilon = A_{qq}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(p, \varepsilon, \varepsilon) = (q, \$)$
  - $\delta(q, \varepsilon, \$) = (r, \varepsilon)$

  gives the rule

$$A_{pr} \rightarrow \varepsilon A_{qq} \varepsilon = A_{qq}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(p, \varepsilon, \varepsilon) = (q, \$)$
  - $\delta(q, \varepsilon, \$) = (r, \varepsilon)$

  gives the rule

$$A_{pr} \rightarrow \varepsilon A_{qq} \varepsilon = A_{qq}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$
  - $\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$

  gives the rule

$$A_{qq} \to \mathbf{a}A_{qq}\mathbf{b}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$
  - $\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$

  gives the rule

$$A_{qq} \to \mathbf{a}A_{qq}\mathbf{b}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$
  - $\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$

  gives the rule

$$A_{qq} \to \mathbf{a} A_{qq} \mathbf{b}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$
  - $\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$

  gives the rule

$$A_{qq} \rightarrow \mathbf{a}A_{qq}\mathbf{b}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The pair
  - $\delta(q, \mathbf{a}, \varepsilon) = (q, \mathbf{a})$
  - $\delta(q, \mathbf{b}, \mathbf{a}) = (q, \varepsilon)$

  gives the rule

$$A_{qq} \rightarrow \mathbf{a}A_{qq}\mathbf{b}$$

The Grammar Rules (Group 2)

Example (Convert a PDA to a CFG)

- The Group 2 rules are

$$A_{pr} \to A_{qq}$$
$$A_{qq} \to \mathbf{a}A_{qq}\mathbf{b}$$

The Grammar Rules (Group 3)

Example (Convert a PDA to a CFG)

- Finally, the Group 3 rules are

$$A_{pp} \longrightarrow \varepsilon$$
$$A_{qq} \longrightarrow \varepsilon$$
$$A_{rr} \longrightarrow \varepsilon$$

The Grammar Rules

Example (Convert a PDA to a CFG)

- This gives us the grammar rules

$$
\begin{array}{ll}
A_{pq} \rightarrow A_{pp}A_{pq} & A_{pr} \rightarrow A_{qq} \\
A_{pq} \rightarrow A_{pq}A_{qq} & A_{qq} \rightarrow \mathbf{a}A_{qq}\mathbf{b} \\
A_{pr} \rightarrow A_{pp}A_{pr} & A_{pp} \rightarrow \varepsilon \\
A_{pr} \rightarrow A_{pq}A_{qr} & A_{qq} \rightarrow \varepsilon \\
A_{pr} \rightarrow A_{pr}A_{rr} & A_{rr} \rightarrow \varepsilon \\
A_{qq} \rightarrow A_{qq}A_{qq} & \\
A_{qr} \rightarrow A_{qq}A_{qr} & \\
A_{qr} \rightarrow A_{qr}A_{rr} &
\end{array}
$$

The Grammar

Example (Convert a PDA to a CFG)

- The start symbol is A_{pr}.

The Grammar

Example (Convert a PDA to a CFG)

- The complete grammar is

$$
\begin{aligned}
A_{pr} &\to A_{pp}A_{pr} \mid A_{pq}A_{qr} \mid A_{pr}A_{rr} \mid A_{qq} \\
A_{pp} &\to \varepsilon \\
A_{pq} &\to A_{pp}A_{pq} \mid A_{pq}A_{qq} \\
A_{qq} &\to \mathbf{a}A_{qq}\mathbf{b} \mid A_{qq}A_{qq} \mid \varepsilon \\
A_{qr} &\to A_{qq}A_{qr} \mid A_{qr}A_{rr} \\
A_{rr} &\to \varepsilon.
\end{aligned}
$$

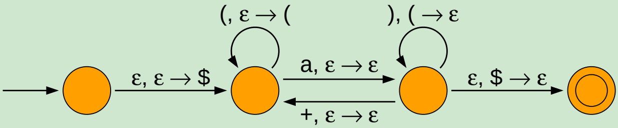
Outline

1. Equivalence of PDAs and CFGs
   - Proof $\Rightarrow$
   - Proof $\Leftarrow$
2. Long Example
3. Short Example
4. Assignment

Example

Example (Convert a PDA to a CFG)

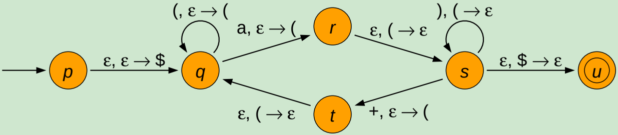
- Find a grammar for the language of the following PDA.



Example

Example (Convert a PDA to a CFG)

- First, we need to modify the PDA:



Example

Example (Convert a PDA to a CFG)

- For Group 1, the transitions are

Pushes	Pops
$\delta(p, \varepsilon, \varepsilon) = (q, \$)$	$\delta(r, \varepsilon, () = (s, \varepsilon)$
$\delta(q, (, \varepsilon) = (q, ()$	$\delta(s,), () = (s, \varepsilon)$
$\delta(q, \mathbf{a}, \varepsilon) = (r, ()$	$\delta(t, \varepsilon, () = (q, \varepsilon)$
$\delta(s, +, \varepsilon) = (t, ()$	$\delta(s, \varepsilon, \$) = (u, \varepsilon)$

Group 1 Rules

Example (Convert a PDA to a CFG)

- For the Group 1 rules, there are 16 combinations of a transition that pushes with a transition that pops.
- However, there are only three beginning states $\{p, q, s\}$ and three ending states $\{s, q, u\}$.
- So there are only 9 sets of rules:

$$\begin{array}{c}
A_{ps} \rightarrow A_{px}A_{xs} \text{ for all } x \in Q \\
A_{pq} \rightarrow A_{px}A_{xq} \text{ for all } x \in Q \\
A_{pu} \rightarrow A_{px}A_{xu} \text{ for all } x \in Q \\
\hline
A_{qs} \rightarrow A_{qx}A_{xs} \text{ for all } x \in Q \\
A_{qq} \rightarrow A_{qx}A_{xq} \text{ for all } x \in Q \\
A_{qu} \rightarrow A_{qx}A_{xu} \text{ for all } x \in Q \\
\hline
A_{ss} \rightarrow A_{sx}A_{xs} \text{ for all } x \in Q \\
A_{sq} \rightarrow A_{sx}A_{xq} \text{ for all } x \in Q \\
A_{su} \rightarrow A_{sx}A_{xu} \text{ for all } x \in Q
\end{array}$$

Group 1 Rules

Example (Convert a PDA to a CFG)

- That gives 54 rules.
- We can eliminate the ones containing variables of the form A_{xp} for $x \neq p$ and A_{ux} for $x \neq u$.
- That eliminates 12 rules, leaving 42.

$$
\begin{array}{l}
A_{ps} \to A_{px}A_{xs} \text{ for all } x \in Q,\ x \neq u \\
A_{pq} \to A_{px}A_{xq} \text{ for all } x \in Q,\ x \neq u \\
A_{pu} \to A_{px}A_{xu} \text{ for all } x \in Q \\
\hline
A_{qs} \to A_{qx}A_{xs} \text{ for all } x \in Q,\ x \neq p, u \\
A_{qq} \to A_{qx}A_{xq} \text{ for all } x \in Q,\ x \neq p, u \\
A_{qu} \to A_{qx}A_{xu} \text{ for all } x \in Q,\ x \neq p \\
\hline
A_{ss} \to A_{sx}A_{xs} \text{ for all } x \in Q,\ x \neq p, u \\
A_{sq} \to A_{sx}A_{xq} \text{ for all } x \in Q,\ x \neq p, u \\
A_{su} \to A_{sx}A_{xu} \text{ for all } x \in Q,\ x \neq p
\end{array}
$$

Example

Example (Convert a PDA to a CFG)

- For Group 2, the transitions are

Pushes	Pops
$\delta(p, \varepsilon, \varepsilon) = (q, \$)$	$\delta(r, \varepsilon, () = (s, \varepsilon)$
$\delta(q, (, \varepsilon) = (q, ()$	$\delta(s,), () = (s, \varepsilon)$
$\delta(q, \mathbf{a}, \varepsilon) = (r, ()$	$\delta(t, \varepsilon, () = (q, \varepsilon)$
$\delta(s, +, \varepsilon) = (t, ()$	$\delta(s, \varepsilon, \$) = (u, \varepsilon)$

Group 2 Rules

Example (Convert a PDA to a CFG)

- There is 1 combination of transitions that pushes and pops \$.
- There are 9 combinations of transitions that push and pop (.
- So there are 10 rules in Group 2.

$$
\begin{array}{l}
A_{pu} \to A_{qs} \\
\hline
A_{qs} \to \texttt{(} A_{qr} \\
A_{qs} \to \texttt{(} A_{qs} \texttt{)} \\
A_{qq} \to \texttt{(} A_{qt} \\
A_{qs} \to \mathbf{a} A_{rr} \\
A_{qs} \to \mathbf{a} A_{rs} \texttt{)} \\
A_{qq} \to \mathbf{a} A_{rt} \\
A_{ss} \to + A_{tr} \\
A_{ss} \to + A_{ts} \texttt{)} \\
A_{sq} \to + A_{tt}
\end{array}
$$

Group 3 Rules

Example (Convert a PDA to a CFG)

- There are 6 rules in Group 3.

$$\begin{aligned} A_{pp} &\rightarrow \varepsilon \\ A_{qq} &\rightarrow \varepsilon \\ A_{rr} &\rightarrow \varepsilon \\ A_{ss} &\rightarrow \varepsilon \\ A_{tt} &\rightarrow \varepsilon \\ A_{uu} &\rightarrow \varepsilon \end{aligned}$$

Outline

1. Equivalence of PDAs and CFGs
   - Proof $\Rightarrow$
   - Proof $\Leftarrow$
2. Long Example
3. Short Example
4. Assignment

Assignment

Homework

- Read Section 2.2, pages 117 - 125.
- Exercises 11, 12, page 154.
- Find a grammar for the language

$$L = \{w \mid w \text{ contains an equal number of } \mathbf{a}\text{'s and } \mathbf{b}\text{'s}\}$$

with PDA