NPDAs accept only CFLs.

Step 1. Every PDA can be simulated by a PDA with one state.

Step 2. Every PDA with one state has an equivalent CFG.

→ "Invert" the construction in the previous lecture.

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

Define $G = (\Gamma, \Sigma, P, \perp)$ as follows.

For every transition $((q, c, A), (q, B_1 \cdots B_k)) \in \delta$,

add the production $A \to cB_1 B_2 \cdots B_k$ in P.

Lemma 1. For any $z, y \in \Sigma^*$, $\gamma \in N^*$ and $A \in N$,

$A \xrightarrow[G]{n} z\gamma$ by a leftmost derivation iff $(q, zy, A) \xrightarrow[M]{n} (q, y, \gamma)$

Theorem $L(G) = L(M)$.

Claim An arbitrary PDA can be simulated by a PDA with one state.

Main idea - Maintain all state information on the stack.

Given a PDA M_1 can construct a PDA M_2 s.t M_2 has a single final state t and M_2 can empty its stack after it enters state t.
That is, for M_2 acceptance by final state and empty stack coincide.

Wlog assume $\underbrace{M = (Q, \Sigma, \Gamma, S, \delta, \perp, \{t\})}$

Single final state and can empty the stack after M enters state t.

$\Gamma' = Q \times \Gamma \times Q$ $\quad\quad$ $\langle p A q \rangle$

$M' = (\{z\}, \Sigma, \Gamma', \delta', z, \langle s \perp t \rangle, \phi)$

M' can scan string x with $\langle p A q \rangle$ on its stack and end up with an empty stack iff M can scan x in state p with A on its stack and end in q with empty stack.

Transition relation δ': For each transition

$$((p, c, A), (q_0, B_1 B_2 \cdots B_k)) \in \delta, \quad c \in \Sigma \cup \{\epsilon\}$$

include in δ'

$$((z, c, \langle p A q_k \rangle), (z, \langle q_0 B_1 q_1 \rangle \langle q_1 B_2 q_2 \rangle \cdots \langle q_{k-1} B_k q_k \rangle))$$

for all possible choices of $q_1, q_2, \ldots q_k$.

For $k=0$ this implies:

if $((p, c, A), (q_0, \epsilon)) \in \delta$ then add

$$((z, c, \langle p A q_0 \rangle), (z, \epsilon)) \text{ in } \delta'.$$

Intuition: M' simulates M guessing the state M will be at future points of computation saving the guesses on the stack and verifying later.

Lemma 2. Let M' be the PDA constructed above from M. Then $(p, x, B_1 \cdots B_k) \xrightarrow[M]{n} (q, \epsilon, \epsilon)$ iff $\exists q_0 q_1 \cdots q_k$ such that $p = q_0$, $q = q_k$ and

$$(z, x, \langle q_0 B_1 q_1 \rangle \langle q_1 B_2 q_2 \rangle \cdots \langle q_{k-1} B_k q_k \rangle) \xrightarrow[M']{n} (z, \epsilon, \epsilon)$$

In particular:

$$(p, x, B) \xrightarrow[M]{n} (q, \epsilon, \epsilon) \text{ iff } (z, x, \langle p B q \rangle) \xrightarrow[M']{n} (z, \epsilon, \epsilon)$$

Proof. Induction on n.

Theorem. $L(M') = L(M)$

Proof. For all $x \in \Sigma^*$,

$$x \in L(M') \text{ iff } (z, x, \langle s \perp t \rangle) \xrightarrow[M']{*} (\ , \epsilon, \epsilon)$$

$$\text{iff } (s, x, \perp) \xrightarrow[M]{*} (t, \epsilon, \epsilon) \quad \text{[Lemma 2]}$$

$$\text{iff } x \in L(M).$$

CFL and closure under intersection

Theorem. CFLs are closed under intersection with regular sets.

if $A \subseteq \Sigma^*$ is a CFL and $B \subseteq \Sigma^*$ is a regular set then $A \cap B$ is a CFL.

Proof idea. Consider PDA M_1 and DFA M_2 s.t $L(M_1) = A$ and $L(M_2) = B$

Construct PDA N by applying the product construction on M_1 & M_2.

Stack of N simulates stack of M_1.

States of N are product of states of M_1 & M_2.

Another example using pumping lemma.

$A = \{ww \mid w \in \{a,b\}^*\}$. is not a CFL.

Suffices to show that
$A' = A \cap L(a^* b^* a^* b^*) = \{a^n b^m a^n b^m \mid n, m \geq 0\}$ is not CFL

Consider any k. Choose $z = a^k b^k a^k b^k$

We have $|z| \geq k$. No matter which way z is split

$z = uvwxy$ where $vx \neq \epsilon$ and $|vwx| \leq k$

with $i = 2$ it can be shown that $uv^i w x^i y \notin A'$.

By pumping lemma, A' is not regular.
so A is not regular.