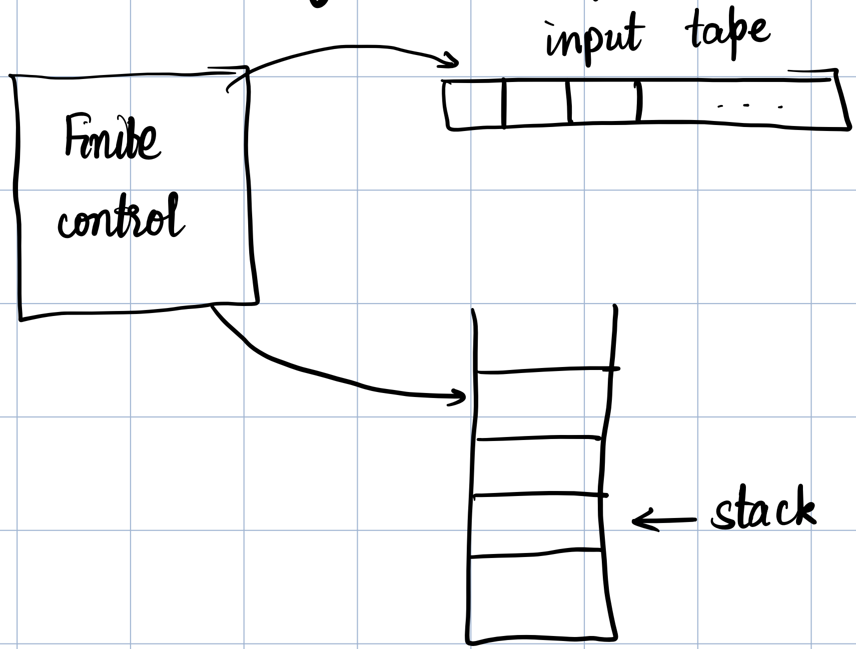
17 Feb 2025 – Theory of Computation – Week 07

(N) PDA



Finite control $\rightsquigarrow$ NFA

n PDAs are more powerful than d PDAs

$$(q, a, A) \longrightarrow (q', b)$$

e.g. $\{ xx^R \mid x \in \{0,1\}^* \}$ is a CFL ??

Theorem: L is a CFL iff L is recognized by a PDA

($\Rightarrow$) we will prove

($\Leftarrow$) know it as a fact

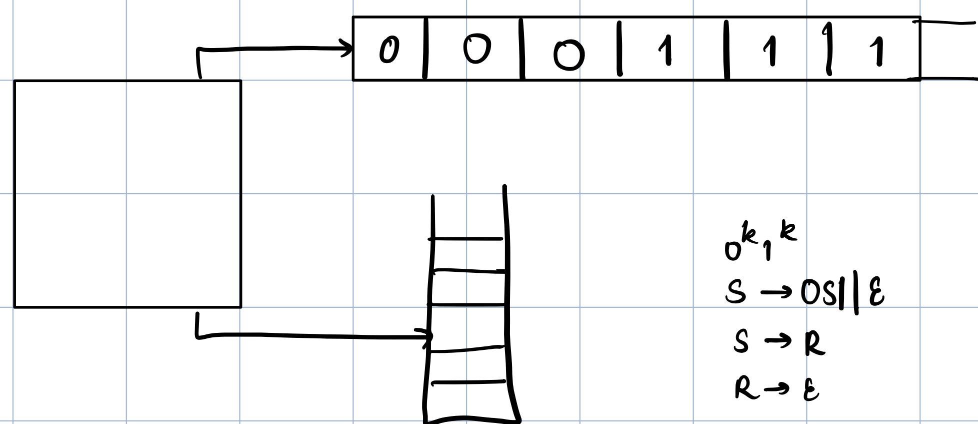
Why are nPDAs more powerful than dPDAs but NFA $\equiv$ DFA?

Given: L is a CFL.

$\exists$ a CFG, $G = (\underset{\text{variables}}{V}, \underset{\text{terminals}}{T}, \underset{\text{rules}}{R}, \underset{\text{start var}}{S})$

s.t. $L(G) = L$

Idea: Use stack to non-determi-nistically (uniformly at random)

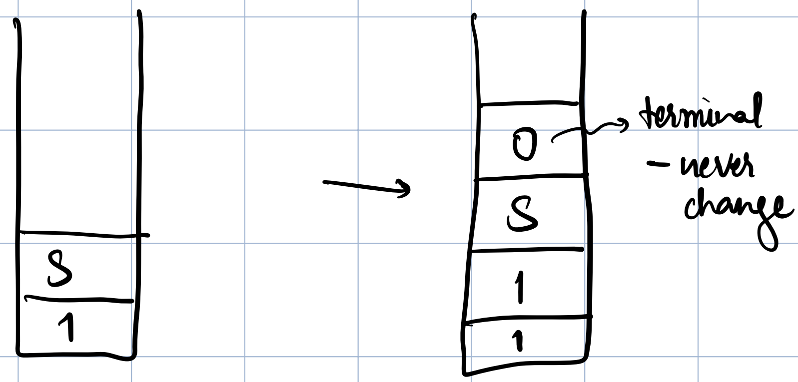


0^k1^k

$S \to 0S1 \mid \varepsilon$

$S \to R$

$R \to \varepsilon$



$$(q, \varepsilon, S) \longrightarrow (q, 0S1)$$

push order

$$\Sigma = \{0, 1\}$$

$$\Gamma = \{0, 1, S, R, 1\}^R$$

$$= V \cup T$$

Problem: → we have to remove 0 and store it somewhere → no other place

Solution: → terminals in stack won't change.
match partially.

$$(q, 0, 0) \longrightarrow (q, \varepsilon)$$

$(q_0, \varepsilon, 1) \to (q_0, S1)$

$(q_0, \varepsilon, S) \to (q_0, 0S1)$

$(q_0, \varepsilon, S) \to (q_0, \varepsilon)$

$(q_0, \varepsilon, S) \to (q_0, R)$

$(q_0, \varepsilon, R) \to (q_0, \varepsilon)$

$(q_0, 0, 0) \to (q_1, \varepsilon)$

$(q_1, 1, 1) \to (q_1, \varepsilon)$

$(q_0, 0, 1) \to (q_{rej}, \varepsilon)$

$(q_0, \varepsilon, 1) \to (q_{accept}, \varepsilon)$

$(q_1, \varepsilon, S) \to (q_0, S)$

$(q_1, \varepsilon, R) \to (q_0, R)$

$(q_1, 0, 0) \to (q_1, \varepsilon)$

$(q_1, 1, 1) \to (q_1, \varepsilon)$

$$|P| + |T| + \binom{|T|}{2} + 1 + |V|$$

→ Closure properties

→ Pumping lemma for CFL

→ Membership problem

Chomsky Normal Form and other normal forms

$A \to BC$

$A \to a$

19 Feb 2025

* Write down proof by yourself once.

think in modular terms
- start
- generation
- end/matching

Things we know:

→ CFG ⟷ PDA

* From an arbitrary PDA you can construct an equivalent PDA with only one state

⇐

If A is regular, then A is a CFL.

→ PDA which does not use a stack.

→ Det. PDA ⊂ PDA

There are languages that can be recognized by a PDA but no det PDA
e.g. $\{xx^R \mid x \in \Sigma^*\}$

What is context-free about CFLs?

$0A \mid BC$

substitution does not depend on context.

Chomsky Normal Forms

Rules are of the form:

$$A \longrightarrow BC \qquad A, B, C \in V$$

$$A \longrightarrow a \qquad \text{where } a \in V$$

Griebach Normal Form

$$A \longrightarrow a B_1 \ldots B_k \qquad \text{where } k \geq 0$$

$$\text{and } a \in \Sigma$$

Thm: Let G be a CFG. Then, $\exists$ a G_1 in CNF and G_2 in GNF such that

$$L(G_1) = L(G_2) = L(G) \setminus \{\varepsilon\}$$

Proof: $G = (V, \Sigma, P, S)$

$S \to a A b B c$

$A \to \varepsilon$ ε-rules

$A \to b$

$A \to B$ unit-rules

If you don't have these rules, at every step, then you always make a measurable progress

$\hat{G}$ in CNF $\}$ $\hat{G} = (V, \Sigma, \hat{P}, S)$

(1) If $\exists$ a rule of kind

$$A \rightarrow \alpha B \gamma \quad , \quad \alpha\gamma \neq \varepsilon$$

not needed
see next lecture

and $B \rightarrow \varepsilon$

then add $A \rightarrow \alpha\gamma$ in $\hat{P}$

where α and $\gamma \in (\Sigma \cup V)^*$

$\rightarrow$ like canonical cover in DBMS

(2) if $A \rightarrow \alpha B \gamma$ | then add $A \rightarrow \alpha C \gamma$

$B \rightarrow C$

Claim 1 : $L(\hat{G}) = L(G)$

Claim 2 : Consider any $x \in \Sigma^*$ and its shortest derivation in $\hat{G}$.

Then,

$$\hat{\hat{P}} = \hat{P} \setminus \begin{matrix} \varepsilon\text{-rules} \\ \text{and unit-rules} \end{matrix}$$

Introduce new variables A_a for each $a \in \Sigma$

$$A_a \rightarrow a \qquad \forall a \in \Sigma$$

$$S \rightarrow aA\,bB\,c \longrightarrow \text{remove}$$

$$S \rightarrow A_a A\, A_b B\, A_c \longrightarrow \text{add}$$

$A \rightarrow ABCD$

$A \rightarrow a$

$A \rightarrow ABCD \rightarrow$ remove

$$\boxed{\begin{array}{l} A \rightarrow AE_1 \\ E_1 \rightarrow BE_2 \\ E_2 \rightarrow CD \end{array}} \rightarrow \text{add}$$

→ No. of rules are still finite.

→ This process will stop in finite time:

right-hand-side ⇝ length keeps reducing.

20 Feb 2025

1) $A \to \alpha B \gamma$ and $B \to \varepsilon$
then add $A \to \alpha\gamma$

α and γ can be anything

2) $A \to B$ and $B \to \gamma$
then add $A \to \gamma$

$\alpha, \gamma \in (\Sigma \cup V)^*$

Keep applying these rules as long as the grammar stops changing.

e.g $G: \quad S \to 0S1 \mid \varepsilon$

$L(G) = \{0^n 1^n \mid n \geq 0\}$

* Get rid of ε-rules (no unit rules)

$\hat{P}$

①

$S \to 0S1$

$S \to \varepsilon$

$S \to 01$ } add all rules

②

$S \to 0S1$

$S \to 01$

throw variables

③

$S \to ASB$

$S \to AB$

$A \to 0$

$B \to 1$

replace variables

④

$S \to AC \mid AB$

$C \to SB$

$A \to 0$

$B \to 1$

e.g: $S \rightarrow [S] \mid SS \mid \varepsilon$ (balanced parenthesis)

① $S \rightarrow [S] \mid SS \mid \varepsilon \mid [\,]$

② $S \rightarrow [S] \mid SS \mid [\,]$

③ $S \rightarrow ASB \mid SS \mid AB$

$A \rightarrow [$

$B \rightarrow]$

④ $S \rightarrow AC \mid SS \mid AB$

$C \rightarrow SB$

$A \rightarrow [$ $\quad B \rightarrow]$

* $S \rightarrow ASA \mid 0B$

$A \rightarrow B \mid S$

$B \rightarrow 1 \mid \varepsilon$

→ Remove ε - rule (add all possible rules first)

$A \rightarrow B$ and $B \rightarrow \varepsilon$

$\Rightarrow A \rightarrow \varepsilon$

$S \rightarrow ASA \mid 0B \mid 0 \mid AS \mid SA$

$A \rightarrow B \mid S \mid \cancel{\varepsilon} \mid ASA \mid 0B \mid 0 \mid AS \mid SA$

$B \rightarrow 1 \mid \cancel{\varepsilon}$

Throw away ε - rules and unit rules

$S \rightarrow ASA \mid 0B \mid 0 \mid AS \mid SA$

$A \rightarrow 1 \mid ASA \mid 0B \mid 0 \mid AS \mid SA$

$B \rightarrow 1$

handle ASA, replace terminals

Pumping Lemma for CFL

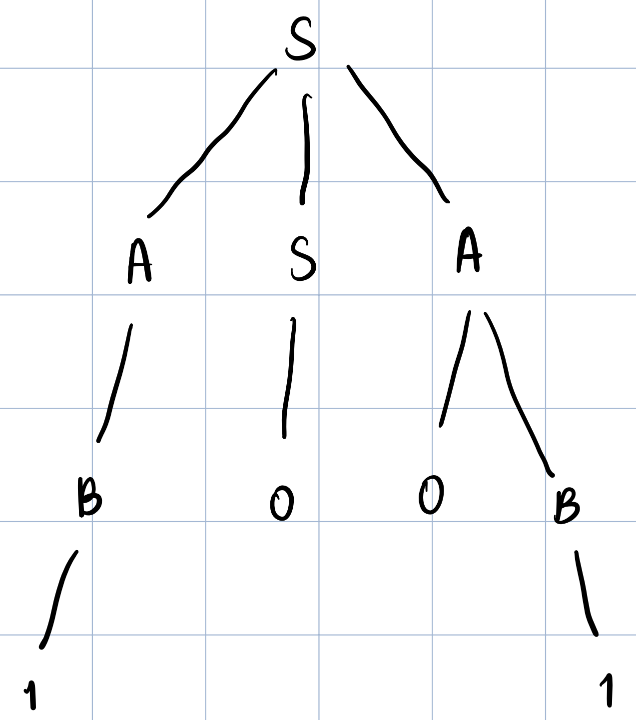
Let A be a CFL. Then $\exists\ k \geq 0$ s.t. $\forall s \in A$ of $|s| \geq k$ $\exists$ a subdivision $s = uvwxy$ and the following holds:

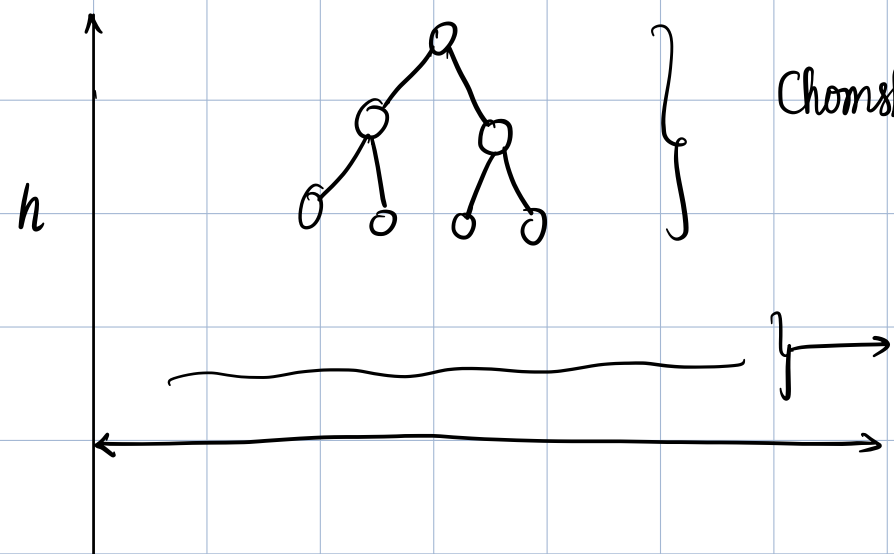
(1) for all $i \geq 0$; $uv^i w x^i y \in A$

(2) $vx \neq \varepsilon$ $\mapsto$ meaningful pumping

(3) $|vwx| \leq k$

If you have a long string s, observe its parse tree parse tree for 1001

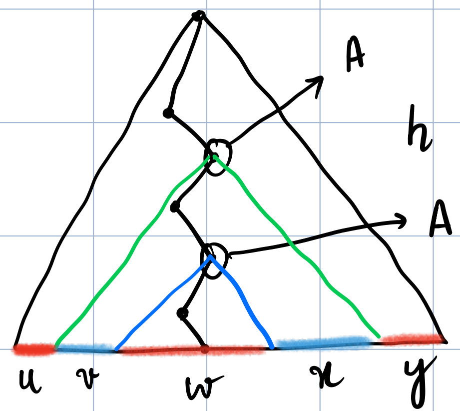




Chomsky Normal form

No. of terminals $\leq 2^h$

$|s| > 2^{n} \implies$ height of the parse tree that derives s $> n$



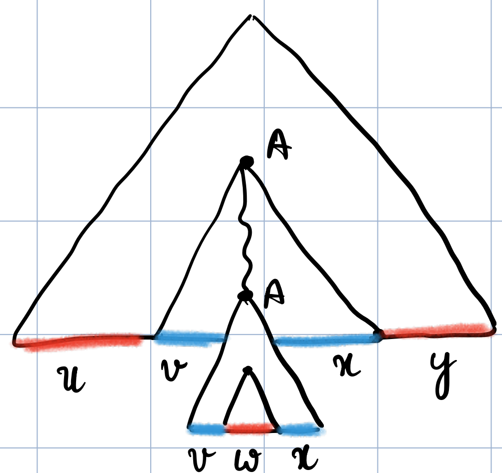
$$|s| \geq 2^{|V|+1}$$

$$\implies \underline{h \geq |V| + 1}$$

no. of variables in the path is at least $|V| + 1$.

But no. of variables are $|V|$.

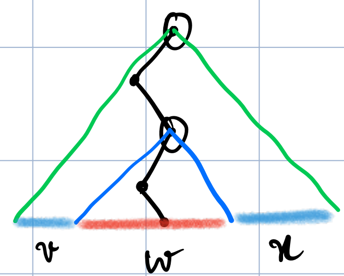
So, by pigeonhole principle, at least one variable must occur twice



$\boxed{vx \neq \varepsilon}$ ↓ By Chomsky Normal form

$|vwx| \leq k$ $\rightsquigarrow$ start from the bottom, find the first repetition

$\rightarrow$ must occur before $|V| + 1$ variables



$\Big\} \leq |V| + 1$

$\therefore \quad |vwx| \leq 2^{|V| + 1}$

$= k$