Chapter II: Context-Free Languages

Section 1: Context-free grammars
Section 2: Chomski normal form
Section 3: Pusdown automata
Section 4: Equivalence of CFG and PDA
Section 5: Non-context-free languages
Section 6: Pumping lemma for context-free languages

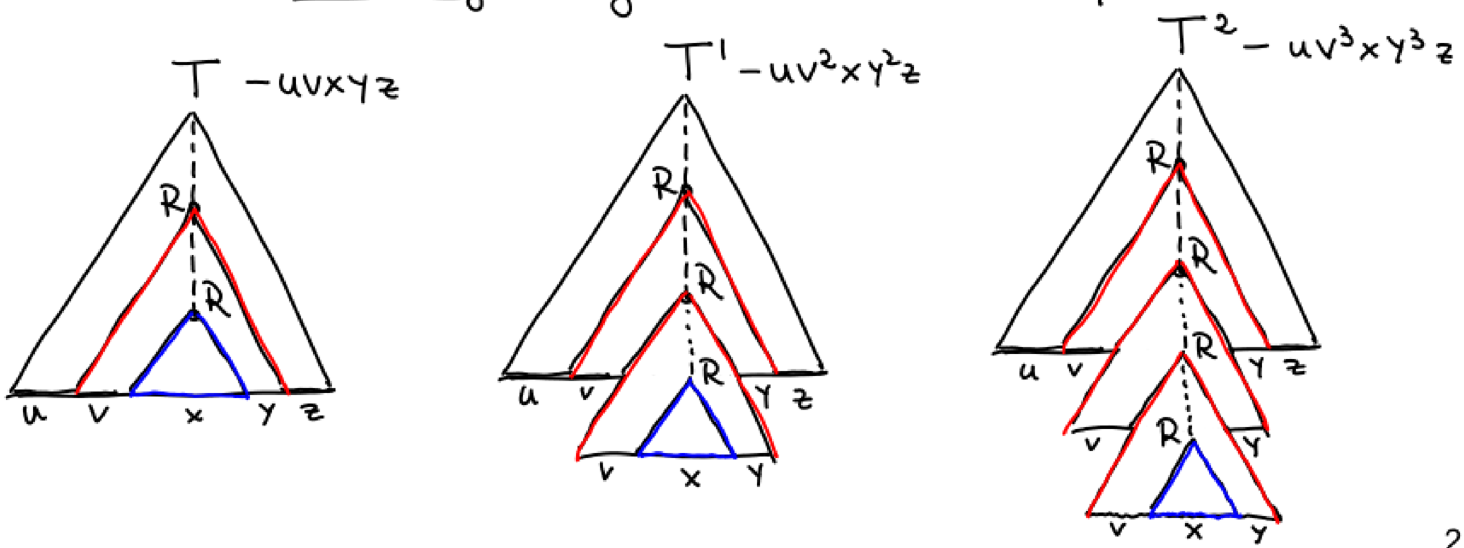
Pumping lemma for context-free languages:

If L is a context free language, then there exists a number $p \geq 1$ called the pumping length such that for any string $w \in L$ such that $|w| \geq p$, there exist u, v, x, y, z substrings of w satisfying the following conditions:

1. $w = uvxyz$
2. $\forall i \geq 0, \quad uv^i xy^i z \in L$
3. $|vy| > 0$
4. $|vxy| \leq p$

Proof: Let G be a CFL such that $L = \mathcal{L}(G)$.

Let w be a very long string in L and T below its pars-tree.



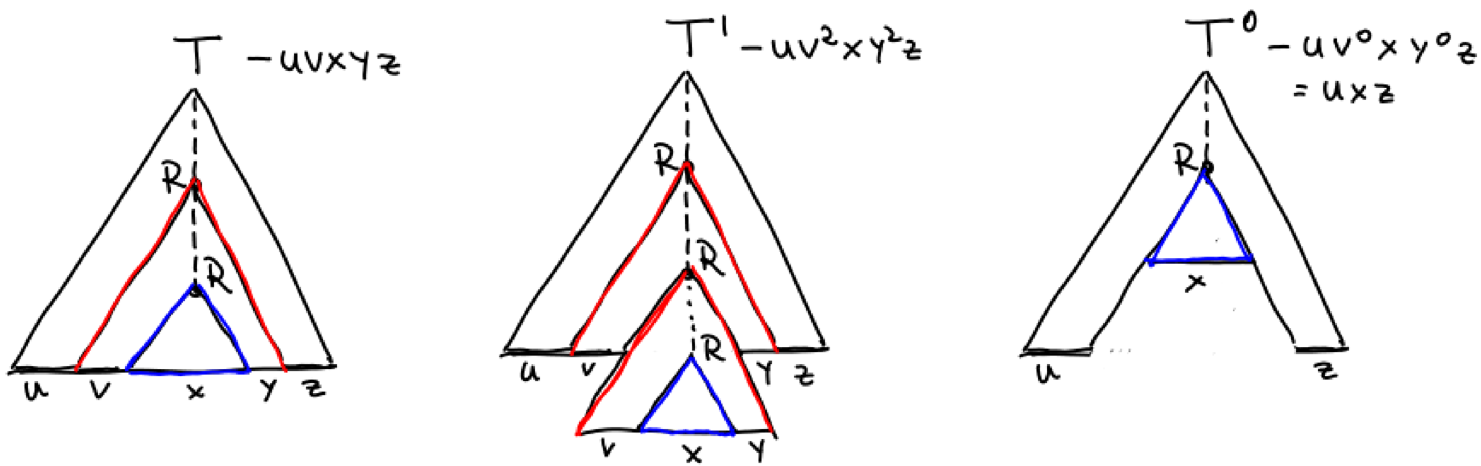
Pumping lemma for context-free languages:

If L is a context free language, then there exists a number $p \geq 1$ called the pumping length such that for any string $w \in L$ such that $|w| \geq p$, there exist u, v, x, y, z substrings of w satisfying the following conditions:

1. $w = uvxyz$
2. $\forall i \geq 0, \quad uv^i x y^i z \in L$
3. $|vy| > 0$
4. $|vxy| \leq p$

Proof: Let G be a CFL such that $L = \mathcal{L}(G)$.

Let w be a very long string in L and T below its pars-tree.



Pumping lemma for context-free languages:

If L is a context free language, then there exists a number $p \geq 1$ called the pumping length such that for any string $w \in L$ such that $|w| \geq p$, there exist u, v, x, y, z substrings of w satisfying the following conditions:

1. $w = uvxyz$
2. $\forall i \geq 0, \quad uv^ixy^iz \in L$
3. $|vy| > 0$
4. $|vxy| \leq p$

Proof: Let b be the maximum no of symbols in the right hand side of the rules. Hence, in a parse tree a node cannot have more than b children.

Consequently, we get at most b leaves at one step from the root $\Rightarrow$

$\Rightarrow$ we get at most b^2 leaves at two steps from the root $\Rightarrow$

$\Rightarrow$ we get at most b^k leaves at k-steps from the root

- Contraposition: If the length of the generated string is at least $b^k + 1$, then the height of the parse tree is at least $k + 1$

Pumping lemma for context-free languages:

If L is a context free language, then there exists a number $p \geq 1$ called the pumping length such that for any string $w \in L$ such that $|w| \geq p$, there exist u, v, x, y, z substrings of w satisfying the following conditions:

1. $w = uvxyz$
2. $\forall i \geq 0, \quad uv^ixy^iz \in L$
3. $|vy| > 0$
4. $|vxy| \leq p$

Proof: Let b be the maximum no of symbols in the right hand side of the rules. and $|V|$ the number of variables of G.

If $p \geq b^{|v|+1} (\geq b^{|v|} + 1)$, then any string $|w| \geq p$ has a parse tree at least $|V|+1$ high.

Hence, the tree has at least one path of length at least $|V|+1$.

This path contains at least $|V|+2$ nodes, one terminal and $|V|+1$ variables

Consequently, at least one variable repeates.

This proves the conditions 1 and 2.

Pumping lemma for context-free languages:

If L is a context free language, then there exists a number $p \geq 1$ called the pumping length such that for any string $w \in L$ such that $|w| \geq p$, there exist u, v, x, y, z substrings of w satisfying the following conditions:

1. $w = uvxyz$
2. $\forall i \geq 0, \quad uv^ixy^iz \in L$
3. $|vy| > 0$
4. $|vxy| \leq p$

Proof: Suppose that $v = y = \varepsilon$.

Then, there is only one occurrence of the repeating variable.

Impossible! Hence, $v \neq \varepsilon$ or $y \neq \varepsilon$ or $v \neq \varepsilon \neq y$.

In any case, $|vy| > 0$ – condition 3.

In T, the upper occurrence of R generates vxy.

Since R is within the bottom $|V|+1$ variables, the string generated by R is at most $b^{|V|+1} \leq p$.

Hence $|vxy| \leq p$ – condition 4.

Theorem: Any regular language satisfies the pumping lemma for context-free languages.

Exercise: Prove this theorem without using the fact that a regular language is a context-free language.

Theorem: Let C be a context-free language and R a regular language. Then $R \cap C$ is a context-free language.

Proof: Let $P = (Q, \Sigma, \Gamma, \delta_P, q_P, F_P)$ be a PDA such that $\mathcal{L}(P) = C$ and $D = (Q', \Sigma, \delta_D, q_D, F_D)$ be a DFA such that $\mathcal{L}(D) = R$.

Construct the product automaton $P' = (Q \times Q', \Sigma, \Gamma, \delta', (q_P, q_D), F_P \times F_D)$ such that

$$\left.\begin{array}{l} - P' \text{ will do what } P \text{ does} \\ - P' \text{ keeps track of the transitions of } D \\ - \text{initial state of } P' \text{ is } (q_P, q_D) \\ - P' \text{ stops at a state } (q, q') \in F_P \times F_D \end{array}\right\} \text{Exercise}$$

By construction, P' is a PDA and $\mathcal{L}(P') = C \cap R$.

Theorem: Let C be a context-free language and R a regular language. Then $R \cap C$ is a context-free language.

Corollary: If C is a context-free language, and R is a regular expression, then $R \cap C$ is a context-free language.

Theorem: The class of context-free languages is not closed under intersection, i.e., if C_1 and C_2 are context-free languages, it is not always the case that $C_1 \cap C_2$ is context-free.

The proof is part of the Exercise Session 8.