Context Free Pumping Lemma

Some languages are not context free!

Sipser pages 123 - 128

CFL Pumping Lemma

- A *CFL pump* consists of two non-overlapping substrings that can be pumped simultaneously while staying in the language.
- Precisely, two substrings u and v constitute a CFL pump for a string w of L when

1. $uv \neq \varepsilon$ (|uv| > 0, which means that at least one of u or v is not empty)
2. And we can write w = xuyvz

   so that for every $i \geq 0$
3. $xu^iyv^iz \in L$

Pumping Lemma

- Let L be a CFL. Then there exists a number n (depending on L) such that every string w in L of length greater than n contains a CFL pump.
- Moreover, there exists a CFL pump such that (with the notation as above), $|uyv| \leq n$.

- For example, take L= $\{0^i1^i \mid i \geq 0\}$: there are no (RE) pumps in any of its strings, but there are plenty of CFL pumps.

The pumping Lemma Game

- We want to prove L is not context-free. For a proof, it suffices to give a winning strategy for this game.

1. The demon first plays n.
2. We respond with $w \in L$ such that $|w| \geq n$.
3. The demon factors w into five substrings, w=xuyvz, with the proviso that $uv \neq \varepsilon$ and $|uyv| \leq n$
4. Finally, we play an integer $i \geq 0$, and we win if $xu^iyv^iz \notin L$.

Example 1

- We prove that L= $\{0^i1^i2^i \mid i \geq 0\}$ is not context-free.
-
- In response to the demon's n, we play $w=0^n1^n2^n$.
- The middle segment uyv of the demon's factorization of w = xuyvz, cannot have an occurrence of both 0 and 2 (because we can assume $|uyv| \leq n$).
- Suppose 2 does not occur in uyv (the other case is similar).
  1. We play i = 0.
  2. Then the total number of 0's and 1's in w_0=xyz will be smaller than 2n,
  3. while the number of 2's in w_0 will be n.
  4. Thus, $w_0 \notin L$.

Example 2

- Let L be the set of all strings over {0,1} whose length is a perfect square.

1. The demon plays n
2. We respond with $w = 0^{n^2}$
3. The demon plays a factorization $0^{n^2} = xuyvz$ with $1 \leq |uyv| \leq n$.
4. We play i=2.
5. The length of the resulting string $w_2 = xu^2yv^2z$ is between n^2+1 and n^2+n.
6. In that interval, there are no perfect squares, so $w_2 \notin L$.

Proof of the pumping lemma

- Strategy in several steps

1. Define fanout
2. Define height yield
3. Prove a lemma about height yield
4. Apply the lemma to prove pumping lemma

Fanout – max rhs length

- Let fanout(G) denote the maximal length of the rhs of any production in the grammar G.
- E.g. For the Grammar
  - $S \rightarrow S\ S$
  - $S \rightarrow (\ S\ )$
  - $S \rightarrow \varepsilon$
- The fanout is 3

Height Yield

- The proof of Pumping Lemma depends on this simple fact about parse trees.

- The *height* of a tree is the maximal length of any path from the root to any leaf.
- The yield of a parse tree is the string it represents (the terminals from a left-to-right in-order walk)

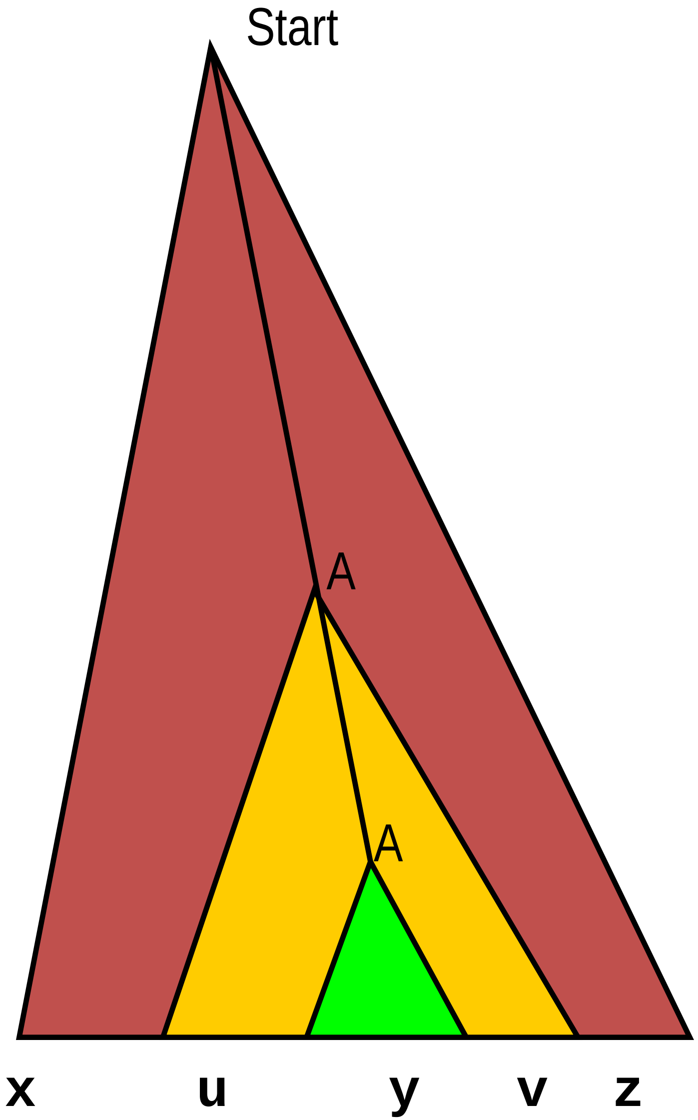
- **Lemma.** If a parse tree of G has height h, and the fanout(G)=b, then its yield has size at most b^h
  - Conversely if a string has length > b^h (i.e. it has length $b^h + 1$ or more) then its parse tree is at least h+1 deep

- **Proof**. Induction on h
- qed

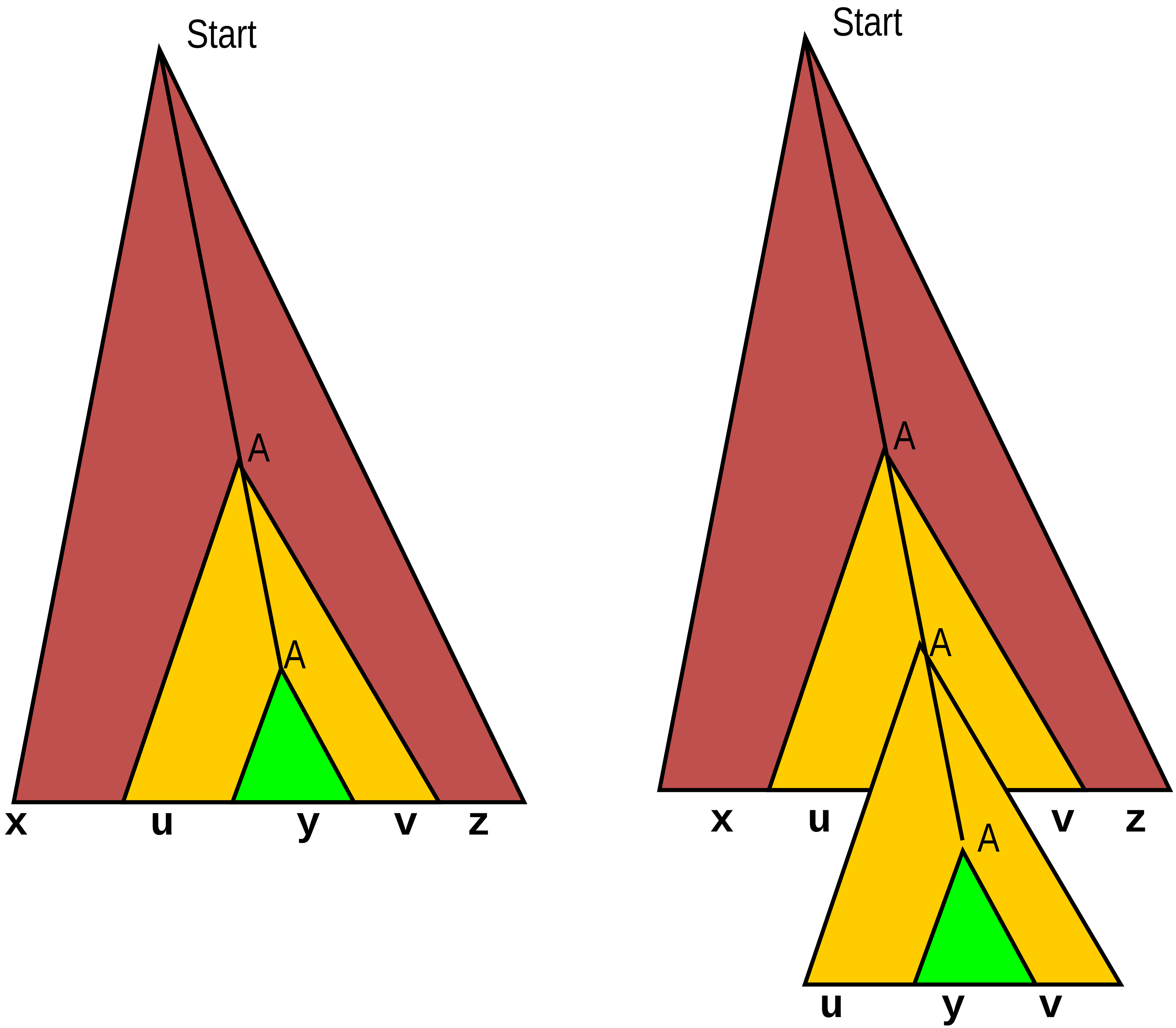
The actual Proof

- The constant n for the grammar G is $\text{fanout}(G)^{|V|}$ where V is the set of variables of G.
- Suppose $w \in L(G)$ and $|w| \geq n$.
- Take a parse tree of w with the smallest possible number of nodes.
- By the Height-Yield Lemma, any parse tree of w must have height $\geq |V|$.
- Therefore, there must be two occurrences of the same variable on a path from root to a leaf.
- Consider the last two occurrences of the same variable (say A) on that path.
- They determine a factorization of the yield w=xuyvz as in the picture on the next slide

Diagram



- We have
-
- $S \Rightarrow^* xAz$
- $A \Rightarrow^* uAv$
- $A \Rightarrow^* y$
- so clearly $S \Rightarrow^* xu^iyv^iz$ for any $i \geq 0$.



Start
A
A
x
u
y
v
z
Start
A
A
A
x
u
v
z
u
y
v

Proof continued

- We also need to check that $uv \neq \varepsilon$. Indeed, if $uv= \varepsilon$, we can get a smaller parse tree for the same w by ignoring the productions "between the two As". But we have chosen the smallest possible parse tree for w! Which leads to a Contradiction.

- Finally, we need to check that $|uyv| \leq n$. This follows from the Height-Yield Lemma because the nodes on our chosen path from the first depicted occurrence of A, onward, are labeled with necessarily distinct variables.

- qed