EECS 2001A : Introduction to the Theory of Computation

Suprakash Datta

Course page: `http://www.eecs.yorku.ca/course/2001`
Also on Moodle

Equivalence of PDA, CFL

- Theorem 2.20 (2.12 in 2nd Ed): A language L is context-free if and only if there is a pushdown automata M that recognizes L.
- Two step proof:
  1) Given a CFG G, construct a PDA M_G
  2) Given a PDA M, make a CFG G_M
- Converting a CFL to a PDA: Lemma 2.21
  - The PDA should simulate the derivation of a word in the CFG and accept if there is a derivation.
  - Need to store intermediate strings of terminals and variables. How?

Converting a CFG to an Equivalent PDA

- First idea: Store all intermediate strings in the derivation in the stack
  - Does not work
- Store only a suffix of the string of terminals and variables derived at the moment starting with the first variable
- The prefix of terminals up to but not including the first variable is checked against the input

Converting a CFG to an Equivalent PDA - 2

Informal description

- Push the usual \$ marker into the empty stack
- Repeat forever:
  - If the top of stack symbol is a variable A, pop A, choose a rule $A \rightarrow \ldots$ nondeterministically and put the RHS of the rule into the stack
  - If the top of the stack is a terminal a, match it against the input. If it does not match reject, else continue
  - If the top of the stack is a \$, accept
- A 3 state PDA is enough: p 120 3rd Ed.

Converting a PDA to an Equivalent CFG

- Lemma 2.27 in 3rd Ed
- Design a grammar equivalent to a PDA
- Idea: For each pair of states p, q we have a variable A_{pq} that generates all strings that take the automaton from p to q (empty stack to empty stack).

Converting a PDA to an Equivalent CFG - Details

- Assume
  - Single accept state
  - Stack emptied before accepting
  - Each transition either pops or pushes a symbol
- Can create rules for all the possible cases (p 122 in 3rd Ed)

Beyond CFL's

Are there problems not solvable by PDA?

Is there a Non-CFL?

The language $L = \{a^n b^n c^n | n \geq 0, n \in \mathbb{Z}\}$ does not **appear** to be context-free.

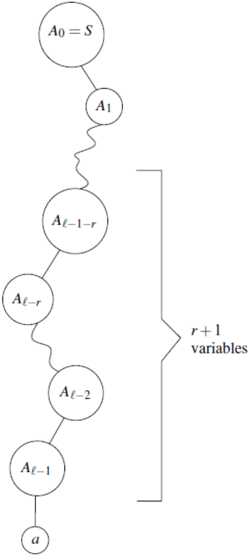
- Informal: The problem is that every variable can (only) act 'by itself' (context-free)
- We can only keep the numbers of 2 of a, b, c equal
- If we think of a PDA, again we can only keep the numbers of 2 of a, b, c equal

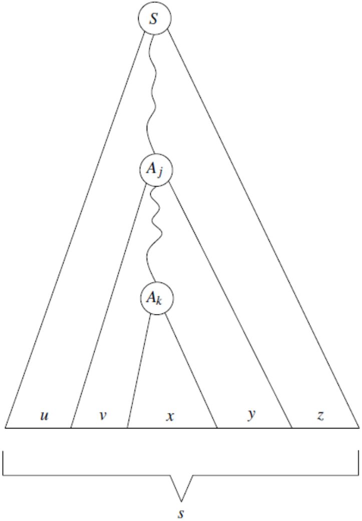
Can we "pump" CFL's?

If so, we may be able to use it to prove that a language is not a CFL. What repeats if we have to derive long strings?

- One possibility is variables
- Some variable(s) must be repeated to derive long strings
- Idea: If we can prove that some derivations use the step $A \Rightarrow^* vAy$, then a new form of 'pumping' holds: $A \Rightarrow^* vAy \Rightarrow^* v^2Ay^2 \Rightarrow^* v^3Ay^3 \Rightarrow^* \ldots$
- For this to happen the word derived must be long enough

Pumping CFLs





Pumping Lemma for CFL's

Let L be a CFL. Then there exists a pumping length $p \geq 1$, and every string $s \in L$, with $|s| \geq p$, can be written as $s = uvxyz$, such that

1. $|vy| \geq 1$ (i.e., v and y are not both empty),
2. $|vxy| \leq p$, and
3. $uv^i xy^i z \in L$, for all $i \geq 0$

Note

- 3) implies that $uxz \in L$ ("pumping down")
- 2) is not always used

Using the Pumping Lemma for CFL's

Prove: $\{a^n b^n c^n | n \geq 0\}$ is not a CFL

- Assume that $B = \{a^n b^n c^n | n \geq 0\}$ is CFL
- Let p be the pumping length, and $s = a^p b^p c^p \in B$
- P.L.: $s = uvxyz = a^p b^p c^p$, with $uv^i xy^i z \in B$ for all $i \geq 0$
- Options for $|vxy|$:
  - The strings v and y are uniform: ($v = a \ldots a$ and $y = c \ldots c$, for example)
    Then uv^2xy^2z will not contain the same number of a's, b's and c's, hence $uv^2xy^2z \notin B$
  - v and y are not uniform.
    Then uv^2xy^2z will not be $a \ldots ab \ldots bc \ldots c$. Hence $uv^2xy^2z \notin B$
- So B is not a CFL

Using the Pumping Lemma for CFL's - 2

Prove: $C = \{a^i b^j c^j | k \geq j \geq i \geq 0\}$ is not a CFL

- Assume that C is CFL
- Let p be the pumping length, and $s = a^p b^p c^p \in C$
- P.L.: $s = uvxyz = a^p b^p c^p$, with $uv^i xy^i z \in C$ for all $i \geq 0$
- Options for $1 \leq |vxy| \leq p$:
  - $v = a^* b^*$: Then uv^2xy^2z will not contain enough c's, so $uv^2xy^2z \notin C$
  - $v = b^* c^*$: Then $uv^0xy^0z = uxz$ will have twwo many a's. Hence $uv^0xy^0z \notin C$
- So C is not a CFL

Using the Pumping Lemma for CFL's - 3

Prove: $D = \{ww | w \in \{0,1\}^*\}$ is not a CFL

- Assume that D is CFL
- Let p be the pumping length, and $s = 0^p1^p0^p1^p \in D$
- P.L.: $s = uvxyz = 0^p1^p0^p1^p$, with $uv^ixy^iz \in D$ for all $i \geq 0$
- Options for $1 \leq |vxy| \leq p$:
  - If a part of y is to the left of $|$ in $0^p1^p|0^p1^p$, then second half of uv^2xy^2z starts with '1', so $uv^2xy^2z \notin D$
  - Same reasoning if a part of v is to the right of the middle of $0^p1^p|0^p1^p$, hence $uv^2xy^2z \notin D$
  - If x is in the middle of $0^p1^p|0^p1^p$, then uxz equals $0^p1^i0^j1^p \notin D$ (because i or j is less than p)
- So D is not a CFL