Theory of Computation

Lesson 14a - CFGs simulating PDAs

Review

Last Time:

1. Chomsky Normal Forms
   - Derivations of words of length $n \geq 1$ have $2n$-1 steps (substitutions)
2. Pushdown Automata (PDAs)
   - NFA + stack
   - transitions (q, σ, a) (p, b) if in state q read σ on tape: replace a on top of stack with b and enter state p.
3. PDAs Examples
   - Designing PDAs
   - Proving that they work
4. CFLs are the languages recognized by PDAs
   - Every CFG can be simulated by a PDA
     - Simulate leftmost derivations on the stack
     - Extended PDAs as a tool

PDA to CFG

Theorem. A language is context-free if and only if some pushdown automaton recognizes it.

Theorem part 2. If L is recognized by some pushdown automaton P, then L is context free.

Proof. Idea: more complex version of simulation of DFA by regular grammar.

Omitted (see textbook p121-125).

Theory of Computation

Lesson 14b - Closure Properties

Closure Properties

Theorem. Context-free languages are closed under the regular operations: union, concatenation, star. They are also closed under reversal.

Proof. Let L_1, L_2 be context-free languages.

$$\left.\begin{array}{ll} L_1 = L(G_1), & G_1 = (V_1, \Sigma, R_1, S_1) \text{ cfg} \\ L_2 = L(G_2), & G_2 = (V_2, \Sigma, R_2, S_2) \text{ cfg} \end{array}\right\} \text{can assume } V_1 \cap V_2 = \varnothing$$

(a) Union. Construct cfg for $L_1 \cup L_2$

$G = (V, \Sigma, R, S)$ where

$V = V_1 \cup V_2 \cup \{S\}$ $\longleftarrow$ new variable $S \notin V_1 \cup V_2$ is start variable

$R = R_1 \cup R_2 \cup \{S \rightarrow S_1, S \rightarrow S_2\}$

Closure Properties

(b) Concatenation. Construct cfg for $L_1 \circ L_2$

$G = (V, \Sigma, R, S)$ where

$V = V_1 \cup V_2 \cup \{S\}$ $\longleftarrow$ new variable $S \notin V_1 \cup V_2$ is start variable

$R = R_1 \cup R_2 \cup \{S \rightarrow S_1 S_2\}$

(c) Star. Construct cfg for $L_1{}^*$

$G = (V, \Sigma, R, S)$ where

$V = V_1 \cup \{S\}$ $\longleftarrow$ new variable $S \notin V_1$ is start variable

$R = R_1 \cup \{S \rightarrow S_1 S, S \rightarrow \varepsilon\}$ word of $L_1{}^*$ is empty or it is a word of L_1 concatenated with a word of $L_1{}^*$

Closure Properties

(d) Reversal. Construct cfg for L_1^R

$G = (V_1, \Sigma, R, S_1)$ where

R is obtained by reversing right-hand sides of rules:

$A \to \alpha_1\, \alpha_2 \ldots \alpha_k \qquad (\alpha_i \in V \cup \Sigma)$

becomes

$A \to \alpha_k \ldots \alpha_2\, \alpha_1$

Remark. Context-free languages are not closed under intersection, complementation, and set difference.

Proof: soon. Intuition: can't simulate two PDAs at the same time as simulations would compete for the stack.

Theory of Computation

Lesson 14c - More Closure Properties

Closure Properties

Theorem. The intersection of a context-free language with a regular language is always context-free.

Proof.

$P = (Q, \Sigma, \Gamma, \delta, s, F)$ PDA accepts the context-free language

$D = (Q', \Sigma, \delta', s', F')$ DFA accepts the regular language

Want pushdown automaton SIM that simulates simultaneous runs of P and D on the same input.

SIM accepts input iff both P and D accept.

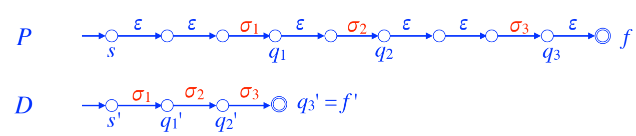
SIM $= (Q \times Q', \Sigma, \Gamma, \Delta, (s, s'), F \times F')$

Closure Properties

SIM $= (Q \times Q', \Sigma, \Gamma, \Delta, (s, s'), F \times F')$

In simulation, two cases:

- D and P read same input symbol $\sigma \in \Sigma$ at the same time.
- D waits without changing state while P reads ε.



Closure Properties

SIM $= (Q \times Q', \Sigma, \Gamma, \Delta, (s, s'), F \times F')$

In simulation, two cases:

- D and P read same input symbol $\sigma \in \Sigma$ at the same time.
- D waits without changing state while P reads ε.

Construction of Δ (visual). For every $q' \in Q'$ do:

For each transition

$q \xrightarrow{\varepsilon,\, a \to b} p$ of P

add $(q, q') \xrightarrow{\varepsilon,\, a \to b} (p, q')$

For each transition

$q \xrightarrow{\sigma,\, a \to b} p$ of P with $\sigma \in \Sigma$:

add $(q, q') \xrightarrow{\sigma,\, a \to b} (p, p')$

where $p' = \delta'(q', \sigma)$

Closure Properties

SIM $= (Q \times Q', \Sigma, \Gamma, \Delta, (s, s'), F \times F')$

In simulation, two cases:

- D waits without changing state while P reads ε.
- D and P read same input symbol $\sigma \in \Sigma$ at the same time.

Construction of Δ. For every $q' \in Q'$ do:

For each transition $(q, \varepsilon, a)\,(p, b)$ of P:

add $(\,(q, q'), \varepsilon, a\,)\;(\,(p, q'), b\,)$

For each transition $(q, \sigma, a)\,(p, b)$ of P with $\sigma \in \Sigma$:

add $(\,(q, q'), \sigma, a\,)\,(\,(p, p'), b\,)$ where $p' = \delta'(q', \sigma)$

Closure Properties

Exercise. $\Sigma = \{a, b\}$. Let L be the set of words that have the same number of a's and b's and do not have a substring *abab* or *baba*. Show that L is context-free.

Solution. Consider languages

$$\left.\begin{array}{l} A = \{w \in \Sigma^* \mid w \text{ has same number of } a\text{'s and } b\text{'s}\} \\ B = \{w \in \Sigma^* \mid w \text{ has a substring } abab\} \\ C = \{w \in \Sigma^* \mid w \text{ has a substring } baba\} \end{array}\right\} L = A \cap \overline{B} \cap \overline{C}$$

A is context-free by lecture 12a

B is regular as $B = (a \cup b)^*\, abab\, (a \cup b)^*$

C is regular as $C = (a \cup b)^*\, baba\, (a \cup b)^*$

Closure properties of reg. languages $\Rightarrow$ $(\overline{B} \cap \overline{C})$ is regular

Lemma implies $A \cap (\overline{B} \cap \overline{C})$ is context-free.

Theory of Computation

Lesson 14d - Pumping Lemma Intro

Pumping Lemma

Poll 14.1 (Graded on attendance) Based on your knowledge or intuition, which of the following languages are context-free?

(a) $\{a^n b^n \mid n \geq 0\}$ $\longleftarrow$ yes

(b) $\{a^n b^n c^k d^k \mid n, k \geq 0\}$ $\longleftarrow$ yes

(c) $\{a^n b^n a^n \mid n \geq 0\}$ $\longleftarrow$ no

(d) $\{a^n b^k a^n b^k \mid n, k \geq 0\}$ $\longleftarrow$ no

Pumping Lemma

Theorem (Pumping Lemma). Let L be a context-free language. Then there exists a number $p \geq 0$ (*pumping length*) such that each word $w \in L$ with $|w| \geq p$ can be written as

$$w = u \circ v \circ x \circ y \circ z$$

with

(1) $|v \circ x \circ y| \leq p$

(2) $v \neq \varepsilon$ or $y \neq \varepsilon$

(3) $u \circ v^n \circ x \circ y^n \circ z \in L$ for all integers $n \geq 0$.

Proof. Soon.

Note. $p = 0 \Leftrightarrow L = \varnothing$

Pumping Lemma

Pumping Lemma. Let L be a context-free language.

Then there exists a number $p \geq 0$ (*pumping length*) such that each word $w \in L$ with $|w| \geq p$ can be written as

$$w = u \circ v \circ x \circ y \circ z$$

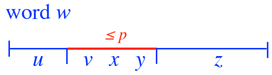
with

(1) $|v \circ x \circ y| \leq p$

(2) $v \neq \varepsilon$ or $y \neq \varepsilon$

(3) $u \circ v^n \circ x \circ y^n \circ z \in L$ for all $n \geq 0$.

(1) and (2) mean:



v and y are not both empty and lie within an interval of length p of w

(3) means: can pump down $(n = 0)$ $\quad u\,x\,z \in L$

can pump up $(n \geq 2)$ $\quad u\,v^2\,x\,y^2\,z\,,\ u\,v^3\,x\,y^3\,z\,, \ldots \in L$

Pumping Lemma

Pumping Lemma.
L context-free language.
$\Rightarrow$
exists number $p \geq 0$ (pumping length) such that each word $w \in L$ with $|w| \geq p$ can be written as

$$w = u \circ v \circ x \circ y \circ z$$

with

(1) $|v \circ x \circ y| \leq p$

(2) $v \neq \varepsilon$ or $y \neq \varepsilon$

(3) $u \circ v^n \circ x \circ y^n \circ z \in L$ for all $n \geq 0$.

Example 1. Find the minimum pumping length of $L = \{a^n b^n \mid n \geq 0\}$.

p = 1 ? Doesn't work with $w = ab$

$w \in L$ and $|w| \geq 1$ but w can't be pumped within an interval of length 1:

(1) & (2) would imply that one of v, y is ε, the other is a or b. Pumping would change only one letter count.

p = 2 ? Works: given any $w \in L$ with $|w| \geq 2$

$$w = \underbrace{a \ldots a}_{u}\ \underset{v}{a}\ \underset{y}{b}\ \underbrace{b \ldots b}_{z} \quad (\text{with } x = \varepsilon)$$

Pumping Lemma

Pumping Lemma.
L context-free language.
$\Rightarrow$
exists number $p \ge 0$
(pumping length)
such that each word
$w \in L$ with $|w| \ge p$
can be written as

$w = u \circ v \circ x \circ y \circ z$

with
(1) $| v \circ x \circ y | \le p$
(2) $v \ne \varepsilon$ or $y \ne \varepsilon$
(3) $u \circ v^n \circ x \circ y^n \circ z \in L$
for all $n \ge 0$.

Example 2. Find the minimum pumping length of $L = \{a^n \# b^n \mid n \ge 0\}$.

$p \le 2$? Doesn't work with $w = a \# b$.

$w \in L$ and $|w| \ge 2$ but w can't be pumped within an interval of length 2:

- vxy is contained in $a\#$: pumping changes the count of a or $\#$ but not of b.
- vxy is contained in $\# b$: pumping changes the count of b or $\#$ but not of a.

$p = 3$? Works: given any $w \in L$ with $|w| \ge 3$

$$w = \underbrace{a \ldots a}_{u}\ \underbrace{a}_{v}\ \underbrace{\#}_{x}\ \underbrace{b}_{y}\ \underbrace{b \ldots b}_{z}$$

Pumping Lemma

Pumping Lemma.
L context-free language.
$\Rightarrow$
exists number $p \ge 0$
(pumping length)
such that each word
$w \in L$ with $|w| \ge p$
can be written as
$w = u \circ v \circ x \circ y \circ z$
with
(1) $| v \circ x \circ y | \le p$
(2) $v \ne \varepsilon$ or $y \ne \varepsilon$
(3) $u \circ v^n \circ x \circ y^n \circ z \in L$
for all $n \ge 0$.

Example 3. Show that $L = \{a^n b^n c^n \mid n \ge 0\}$ is not context-free.

Proof (by contradiction). Suppose that L is context-free, pumping length p.

Consider $w = a^p b^p c^p$.

We have $w \in L$ and $|w| \ge p$.

Cannot pump within interval of length p :

- interval contains at most two of letters a, b, c.
- pumping would change count of one or two letters, but not of the third.

CONTRADICTION

Pumping Lemma

Pumping Lemma.
L context-free language.
$\Rightarrow$
exists number $p \ge 0$
(pumping length)
such that each word
$w \in L$ with $|w| \ge p$
can be written as
$w = u \circ v \circ x \circ y \circ z$
with
(1) $| v \circ x \circ y | \le p$
(2) $v \ne \varepsilon$ or $y \ne \varepsilon$
(3) $u \circ v^n \circ x \circ y^n \circ z \in L$
for all $n \ge 0$.

Example 3. Show that $L = \{a^n b^n c^n \mid n \ge 0\}$ is not context-free.

Proof (by contradiction). Suppose that L is context-free, pumping length p.

Consider $w = a^p b^p c^p$.

We have $w \in L$ and $|w| \ge p$.

Thus $w = u\, v\, x\, y\, z$ with properties (1), (2), (3).

By (1), vxy cannot contain both a's and c's
By (2), $v \ne \varepsilon$ or $y \ne \varepsilon$.

$\Rightarrow$ pumping changes the counts of at least 1 and at most 2 letters.

Thus $u\, v^2\, x\, y^2\, z \notin L$. CONTRADICTION to (3)

Pumping Lemma

Pumping Lemma.
L context-free language.
$\Rightarrow$
exists number $p \ge 0$
(pumping length)
such that each word
$w \in L$ with $|w| \ge p$
can be written as
$w = u \circ v \circ x \circ y \circ z$
with
(1) $| v \circ x \circ y | \le p$
(2) $v \ne \varepsilon$ or $y \ne \varepsilon$
(3) $u \circ v^n \circ x \circ y^n \circ z \in L$
for all $n \ge 0$.

Example 3. Show that $L = \{a^n b^n c^n \mid n \ge 0\}$ is not context-free.

Note. Didn't really need property (1) in the proof. Was only used to show: v and y together contain at most two letters

This is implied by (3) and structure of L :

each of v, y contains at most one letter

otherwise, say $v = \ldots a \ldots b \ldots$

then $v^2 = \ldots a \ldots b \ldots a \ldots b \ldots$

out of order in L

Theory of Computation

Lesson 14e - Pumping Lemma

Pumping Lemma and Closure Properties

Corollary. Context-free languages are not closed under intersection, complement, and set difference.

Proof.

(a) Intersection.

$$L_1 = \{a^i b^j c^k \mid i = j\}$$
$$L_2 = \{a^i b^j c^k \mid j = k\}$$

easy to show L_1, L_2 context-free.

But

$$L_1 \cap L_2 = \{a^n b^n c^n \mid n \ge 0\}$$ is not context-free.

Pumping Lemma and Closure Properties

Corollary. Context-free languages are not closed under intersection, complement, and set difference.

Proof.

(b) Complement.

$$L_1 \cap L_2 = \overline{\overline{L_1} \cap \overline{L_2}} = \overline{\overline{L_1} \cup \overline{L_2}}$$

CFLs are closed under union. If they were also closed under complement, then they would be closed under intersection.

(c) Set difference.

$$\overline{L} = \Sigma^* - L$$

Σ^* is context-free. If CFLs were closed under set difference, then they would also be closed under complement.

Pumping Lemma

Pumping Lemma.
L context-free language.
$\Rightarrow$
exists number $p \geq 0$
(pumping length)
such that each word
$w \in L$ with $|w| \geq p$
can be written as
$w = u \circ v \circ x \circ y \circ z$
with
(1) $|v \circ x \circ y| \leq p$
(2) $v \neq \varepsilon$ or $y \neq \varepsilon$
(3) $u \circ v^n \circ x \circ y^n \circ z \in L$ for all $n \geq 0$.

Example 4. Show that

$$L = \{a^i b^j a^i b^j \mid i, j \geq 1\}$$

is not context-free.

Proof (by contradiction). Suppose that L is context-free, pumping length p.

Attempt 1. Consider $w = a^p b a^p b$.

- Would work to show L not regular!
- Doesn't work to show L not context-free:

$$w = \underset{u}{a^{p-1}}\ \underset{v}{a}\ \underset{x}{b}\ \underset{y}{a}\ \underbrace{a^{p-1}\ b}_{z}$$

Pumping Lemma

Pumping Lemma.
L context-free language.
$\Rightarrow$
exists number $p \geq 0$
(pumping length)
such that each word
$w \in L$ with $|w| \geq p$
can be written as
$w = u \circ v \circ x \circ y \circ z$
with
(1) $|v \circ x \circ y| \leq p$
(2) $v \neq \varepsilon$ or $y \neq \varepsilon$
(3) $u \circ v^n \circ x \circ y^n \circ z \in L$ for all $n \geq 0$.

Example 4. Show that

$$L = \{a^i b^j a^i b^j \mid i, j \geq 1\}$$

is not context-free.

Attempt 2. Consider $w = a^p b^p a^p b^p$.

We have $w \in L$ and $|w| \geq p$.

Cannot pump within interval of length p:

- Interval can't overlap two runs of a's or two runs of b's.
- Pumping changes counts in one run of a's but not in the other, or in one run of b's but not in the other.

CONTRADICTION

Pumping Lemma

Pumping Lemma.
L context-free language.
$\Rightarrow$
exists number $p \geq 0$
(pumping length)
such that each word
$w \in L$ with $|w| \geq p$
can be written as
$w = u \circ v \circ x \circ y \circ z$
with
(1) $|v \circ x \circ y| \leq p$
(2) $v \neq \varepsilon$ or $y \neq \varepsilon$
(3) $u \circ v^n \circ x \circ y^n \circ z \in L$ for all $n \geq 0$.

Example 5. Show that $L = \{ww \mid w \in \{a,b\}^*\}$ is not context-free.

Proof (by contradiction). Suppose that L is context-free.

Then $L_1 = L \cap \underbrace{a^+ b^+ a^+ b^+}_{\text{regular}}$ is context-free.

Claim: $L_1 = \{a^i b^j a^i b^j \mid i, j \geq 1\}$ ← contradicts example 4

Suppose that $w\,w \in a^+ b^+ a^+ b^+$

must begin with a

must end with b

then $w = a^i b^j$ where $i, j \geq 1$