Mathematical Foundation of Computer Sciences VI

Algorithms on Pushdown Automata

Guoqiang Li

School of Software, Shanghai Jiao Tong University

Summary

Language	regular	context-free
Machine	DFA/NFA	PDA
Syntax	regular expression	context-free grammar

General Computations

Closure Properties

The context-free languages are closed under union, concatenation, and kleene star.

Closure Properties - Union

Proof

$N_1 = (V_1, \Sigma_1, R_1, S_1)$ recognize A_1,
$N_2 = (V_2, \Sigma_2, R_2, S_2)$ recognize A_2. w.l.o.g. $V_1 \cap V_2 = \emptyset$.

- **Union.** S is a new symbol. Let $N = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R, S)$, where $R = R_1 \cup R_2 \cup \{S \to S_1, S \to S_2\}$.

Proof

$N_1 = (V_1, \Sigma_1, R_1, S_1)$ recognize A_1,
$N_2 = (V_2, \Sigma_2, R_2, S_2)$ recognize A_2. w.l.o.g. $V_1 \cap V_2 = \emptyset$.

- **Concatenation.** S is a new symbol. Let $N = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R, S)$, where $R = R_1 \cup R_2 \cup \{S \to S_1 S_2\}$.

Proof

$N_1 = (V_1, \Sigma_1, R_1, S_1)$ recognize A_1.

- **Kleene Star.** S is a new symbol. Let $N = (V_1 \cup \{S\}, \Sigma_1, R, S)$, where $R = R_1 \cup \{S \to \epsilon, S \to SS_1\}$.

Intersection of a CFL and a RL

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

Proof

PDA $M_1 = (Q_1, \Sigma, \Gamma_1, \delta_1, s_1, F_1)$ and DFA $M_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$.

Build $M = (Q, \Sigma, \Gamma_1, \Delta, s, F)$, where

- $Q = Q_1 \times Q_2$;
- $s = (s_1, s_2)$;
- $F = (F_1, F_2)$, and
- Δ is defined as follows
  1. for each PDA rule $(q_1, a, \beta) \to (p_1, r)$ and each $q_2 \in Q_2$ add the following rule to Δ
  $$((q_1, q_2), a, \beta) \to ((p_1, \delta_2(q_2, a)), r)$$
  2. for each PDA rule $(q_1, \epsilon, \beta) \to (p_1, r)$ and each $q_2 \in Q_2$ add the following rule to Δ
  $$((q_1, q_2), \epsilon, \beta) \to ((p_1, q_2), r)$$

The context free language are not closed under intersection or complementation.

Proof

Clearly $\{a^n b^n c^m \mid m, n \geq 0\}$ and $\{a^m b^n c^n \mid m, n \geq 0\}$ are both CFL. However their intersection, $\{a^n b^n c^n \mid n \geq 0\}$, is not.

To the second part of the statement,

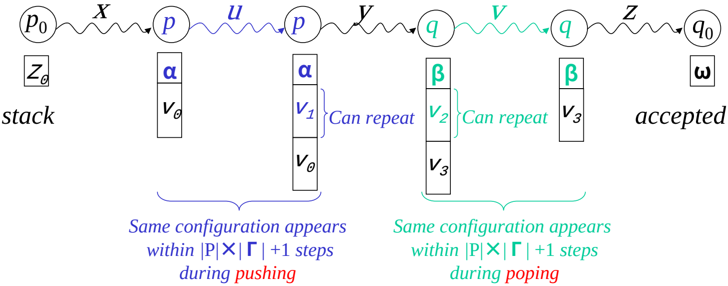
$$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$$

rules out the closure under complementation.

Pumping Lemma

If A is a context-free language, then there is a number p (the pumping length) where, if s is any string in A of length at least p, then s may be divided as $s = xuyvz$ satisfying the conditions

1. for each $i \geq 0$, $xu^i yv^i z \in A$,
2. $|uv| > 0$,
3. $|uyv| < p$.



Proof

Let G be a CFG for CFL A. Let b be the maximum number of symbols in the right-hand side of a rule. In any parse tree using this grammar, every node can have no more than b children.

If the height of the parse tree is at most h, the length of the string generated is at most b^h.

If a generated string is at least $b^h + 1$ long, each of its parse trees must be at least $h + 1$ high.

We choose the pumping length

$$p = b^{|V|+1}$$

For any string $s \in A$ with $|s| \geq p$, any of its parse trees must be at least $|V| + 1$ high.

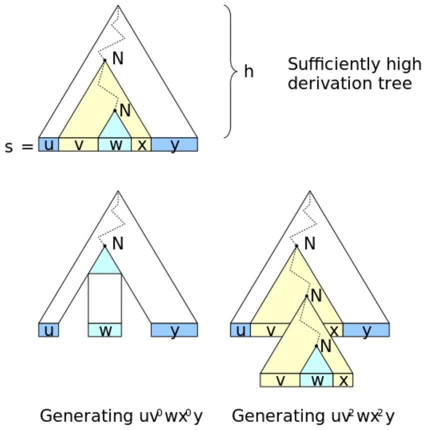
Proof

Let τ be one parse tree of s with smallest number of nodes, whose height is at least $|V|+1$. So τ has a path from the root to a leaf of length $|V|+1$ with $|V|+2$ nodes. One variable R must appear at least twice in the last $|V|+1$ variable nodes on this path.

We divide s into $xuyvz$:

- x from the leftmost leaf of τ to the leaf left next to the leftmost leaf of the subtree hanging on the first R,
- u from the leftmost leaf of the subtree hanging on the first R to the leaf left next to the leftmost leaf of the subtree hanging on the second R,
- y for all the leaves of the subtree hanging on the second R,
- v from the leaf right next to the rightmost leaf of the subtree hanging on the second R to the rightmost leaf of the subtree hanging on the first R,
- z from the leaf right next to the rightmost leaf of the subtree hanging on the first R to the rightmost leaf of τ.

Pumping Lemma



from Wikipedia

Proof

Condition 1. Replace the subtree of the second R by the subtree of the first R would validate that for each $i \geq 0$, $xu^i yv^i z \in A$.

Condition 2. If $|uv| = 0$, i.e., $u = v = \epsilon$, then τ cannot have the smallest number of nodes.

Condition 3. To see $|uyv| \leq p = b^{|V|+1}$, note that uyv is generated by the first R. We can always choose R so that its last two occurrences fall within the bottom $|V| + 1$ high. A tree of this height can generate a string of length at most $b^{|V|+1} = p$.

Example

$\{a^n b^n c^n \mid n \geq 0\}$ is not context free.

Assume otherwise, and let p be the pumping length. Consider $s = a^p b^p c^p$ and divide it to $xuyvz$ according to the Pumping Lemma.

- When both u and v contain only one type of symbols, i.e., one of a, b, c, then xu^2yv^2z cannot contain equal number of a's, b's, and c's.
- If either u or v contains more than one type of symbols, then xu^2yv^2z would have symbols interleaved.

Example

$\{ww \mid w \in \{0,1\}^*\}$ is not context free.

Assume otherwise, and let p be the pumping length. Consider $s = 0^p 1^p 0^p 1^p$ and divide it to $xuyvz$ with $|uyv| \leq p$.

- If uyv occurs only in the first half of s, then the second half of xu^2yv^2z must start with an 1. This is impossible.
- Similarly uyv cannot occur only in the second half of s.
- If uyv straddles the midpoint of s, then pumping s to the form $0^p 1^i 0^j 1^p$ cannot ensure $i = j = p$.

Emptiness

The Emptiness of PDA is decidable.

Some decision problems related to FA

Problems from Formal Language Theory

Decision Problems

- Acceptance: does a given string belong to a given language?
- Emptiness: is a given language empty?
- Equality: are given two languages equal?

Language Problems Concerning CFL

The following three problems:

- Acceptance: Given a CFG G and a string w, does G accept w?
- Emptiness: Given a CFG G is the language $L(G)$ empty?
- Equality: Given two DFA CF(NFA) A and B is $L(A)$ equal to $L(B)$?

The Acceptance and Emptiness problem for CFG are decidable, the Equality problem is not decidable.

Assignment 2

Assignment 2

Exercises 2.4 (b, c, d); 2.6 (a, d); 2.14; 2.18 (b); 2.20; 2.26; 2.30; 2.40; 2.42

deadline Apr. 8