Pushdown Automata

CENG 280

Course outline

- Preliminaries: Alphabets and languages
- Regular languages
- Context-free languages
  - Context-free grammars
  - Parse trees
  - Push-down automaton
  - Push-down automaton - context-free languages
  - Languages that are and that are not context-free, Pumping lemma
- Turing-machines

Definition

Pushdown automaton is a sextuple $M = (K, \Sigma, \Gamma, \Delta, s, F)$ where

- K is a finite set of states,
- Σ is an alphabet (input symbols)
- Γ is an alphabet (stack symbols)
- $s \in K$ is the initial state
- $F \in K$ is the set of final states, and,
- $\Delta \subset (K \times (\Sigma \cup \{e\}) \times \Gamma^\star) \times (K \times \Gamma^\star)$ is a finite transition relation.

- If $((p, a, \delta), (q, \gamma)) \in \Delta$, then when M is in state p, if it reads $a \in \Sigma$ (or if a is e without reading a symbol) and if the top of the stack is δ, it enters state q and replaces δ with γ.
- $((p, a, \delta), (q, \gamma))$ is called a transition of M.
- Since Δ is a relation, several transitions can be applicable at a point. The machine chooses non-deterministically from the applicable transitions

Example

Write a grammar G such that $L(G) = \{w \in \{0,1\}^\star \mid$ the number of $0's$ in w is different than the number of $1's\}$. Write a PDA M such that $L(M) = L(G)$. Write a derivation generating 001 and a computation accepting 001. $(s,001,e) \vdash_M (q,001,\bot) \vdash (q,01,0\bot) \vdash_M (q,1,00\bot) \vdash_M (q,e,0\bot) \vdash_M (f,e,\bot) \vdash_M (f,e,e)$

$001 \in L(M)$

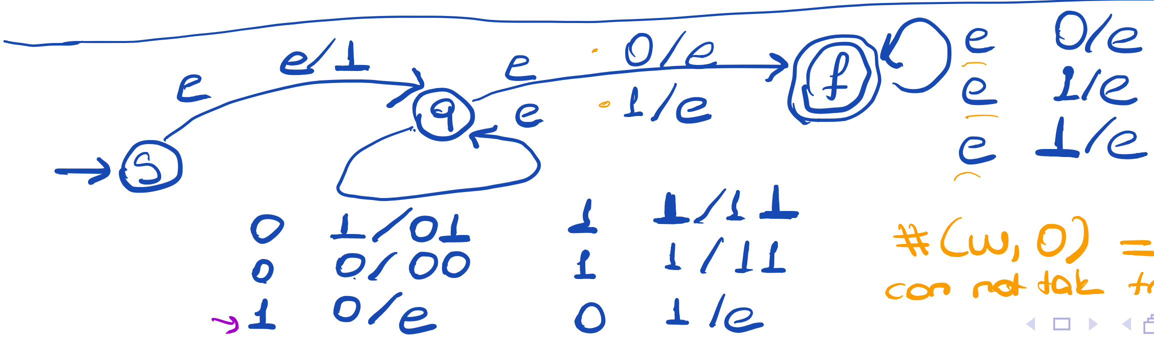
$S \to T0T \mid P1P$

$T \to 0T1T \mid 1T0T \mid TT \mid e, \mid 0T$

generates strings with equal # of 0's and 1's

$P \to 0P1P \mid 1P0P \mid PP \mid e \mid 1P$

$S \Rightarrow T0T \Rightarrow 0T \Rightarrow 00T1T \Rightarrow 001T \Rightarrow 001$



$\#(w,0) = \#(w,1)$ can not take transition to f

Pushdown automata and context-free grammars

Theorem

The class of languages accepted by pushdown automata is exactly the class of context free languages.

Pushdown automata and context-free grammars

Theorem

The class of languages accepted by pushdown automata is exactly the class of context free languages.

Lemma

Each context free language is accepted by some pushdown automaton.

L, $L = L(G)$, M, $L(M) = L(G)$

Lemma

If a language is accepted by a pushdown automaton, then it is a context-free language.

$L = L(M)$, G $L(G) = L(M)$

Pushdown automata and context-free grammars

given G write M

Lemma

Each context free language is accepted by some pushdown automaton.

Constructive proof: Given a CFG $G = (V, \Sigma, R, S)$, construct a PDA $M = (K, \Sigma, \Gamma, \Delta, s, F)$ such that $L(G) = L(M)$.

Pushdown automata and context-free grammars

Lemma

Each context free language is accepted by some pushdown automaton.

Constructive proof: Given a CFG $G = (V, \Sigma, R, S)$, construct a PDA $M = (K, \Sigma, \Gamma, \Delta, s, F)$ such that $L(G) = L(M)$.

- $K = \{s, q\}$, $F = \{q\}$
- $\Gamma = V$
- Δ :
  1. $((s, e, e), (q, S))$ (push the start symbol)
  2. $((q, e, A), (q, x))$ for each rule $A \to x \in R$ (replace the top nonterminal with a corresponding rule)
  3. $((q, a, a), (q, e))$ for each symbol $a \in \Sigma$ (pop the topmost symbol if it matches the next input symbol)

Type

pop push to stack

Type

Σ^*

Mimics the leftmost derivation of the input string.

Pushdown automata and context-free grammars

Example

Construct a PDA that accepts $L(G)$, where $G = (V, \Sigma, R, S)$,
$V = \{a, b, c, S\}$, R : $\Sigma = \{a, b, c\}$ $\rightarrow ((s, e, e), (q, S))$
$K = \{s, q\}$ $F = \{q\}$
$\Gamma = V$

$((q, a, a), (q, e))$
$((q, b, b), (q, e))$
$((q, c, c), (q, e))$

$S \rightarrow aSa$ $((q, e, S), (q, aSa))$
$S \rightarrow bSb$ $((q, e, S), (q, bSb))$
$S \rightarrow c$ $\rightarrow ((q, e, S), (q, c))$

type 2

Show the computation along "aca"

$(s, aca, e) \vdash_M (q, aca, S) \vdash_M (q, aca, aSa) \vdash_M$
$(q, ca, Sa) \vdash_M (q, ca, ca) \vdash_M (q, a, a) \vdash_M (q, e, e)$

Lemma

Each context free language is accepted by some pushdown automaton.

To complete the proof, we need to show that $L(M) = L(G)$.
Claim: let $w \in \Sigma^\star$ and $\alpha \in (V \setminus \Sigma)V^\star \cup \{e\}$, then:

$$S \overset{L}{\Rightarrow}^\star w\alpha \quad \text{iff} \quad (q, w, S) \vdash_M^\star (q, e, \alpha)$$

$w \in L(G)$ iff $w \in L(M)$

Why this claim is sufficient for language equivalence?

$\alpha = e$ $\quad S \overset{L}{\Rightarrow}^* w \quad$ iff $(q, w, S) \vdash_M^* (q, e, e)$

$w \in L(G) \quad (s, w, e) \vdash_M (q, w, S) \quad w \in L(M)$

Proof in two parts:

(1) $S \overset{L}{\Rightarrow}^\star w\alpha$ implies $(q, w, S) \vdash_M^\star (q, e, \alpha)$

(2) $(q, w, S) \vdash_M^\star (q, e, \alpha)$ implies $S \overset{L}{\Rightarrow}^\star w\alpha$

$*$ (1) $S \overset{L}{\Rightarrow}^{\star} w\alpha$ implies $(q, w, S) \vdash_M^{\star} (q, e, \alpha)$

$\Rightarrow$

By induction on the length of the derivation.

- Basis step, derivation length is 0.

$S \overset{*}{\Rightarrow} S$ $\quad w\alpha = S$ $\quad w = e$ $\quad , \; (q, e, S) \vdash_M^{*} (q, e, S)$

$\in \Sigma^*$ $\quad , \quad \alpha = S$

- **IH**: If $S \overset{L}{\Rightarrow}^{\star} w\alpha$ by a derivation of length n or less, then $(q, w, S) \vdash_M^{\star} (q, e, \alpha)$

$S \overset{L}{\Rightarrow}^{*} w\alpha$ by a derivation of length $n+1$

- **IS**: Show the implication holds for a derivation of length $n+1$.

$S = u_0 \overset{L}{\Rightarrow} u_1 \overset{L}{\Rightarrow} u_2 \overset{L}{\Rightarrow} \cdots \Rightarrow u_n \overset{L}{\Rightarrow} u_{n+1} \overset{L}{=} w\alpha$

show $(q, w, S) \vdash_M^{*} (q, e, \alpha)$

$x \in \Sigma^*$

$A \in V \setminus \Sigma$, $A \to \gamma \in R$ $\; \underbrace{x A \beta}_{u_n} \overset{L}{\Rightarrow} x \underbrace{\gamma}_{u_{n+1}} \beta$

$\beta \in V^*$

$w \in L(G) \Rightarrow$
$w \in L(M)$

(1) $S \overset{L}{\Rightarrow}^{\star} w\alpha$ implies $(q, w, S) \vdash_M^{\star} (q, e, \alpha)$

By induction on the length of the derivation. $(q, xy, S) \vdash_M^{*} (q, y, y\alpha) \vdash_M^{*} (q, e, \cancel{\alpha})$

IH: If $S \overset{L}{\Rightarrow}^{\star} w\alpha$ by a derivation of length n or less, then $(q, w, S) \vdash_M^{\star} (q, e, \alpha)$

IS: Show the implication holds for a derivation of length $n + 1$.

$\underbrace{xA\beta}_{u_n} \overset{L}{\Rightarrow} \underbrace{x\gamma\beta}_{u_{n+1}} = w\alpha \quad w \in \Sigma^{*}$

$w = xy$

$x\gamma\beta = xy\alpha$

$\gamma\beta = y\alpha$

$S \overset{L}{\Rightarrow}^{*} xA\beta$ derivation of length n

read x

$^{*}(q, x, S) \vdash_M^{*} (q, e, A\beta)$ By IH

$^{*}(q, e, A\beta) \vdash_M (q, e, \underbrace{\gamma\beta}_{y\alpha})$ By constr. of M, Type 2

$^{*}(q, y, y\alpha) \vdash_M^{*} (q, e, \alpha)$ via Type 3 transitions

$w \in L(M)$ then $w \in L(G)$

(2) $(q, w, S) \vdash_M^\star (q, e, \alpha)$ implies $S \overset{L}{\Rightarrow}^\star w\alpha$

By induction on the number of type-2 transitions. $A \to \gamma$ $((q, e, A), (q, \gamma))$

Basis step, 0 type-2s transition.

$(q, \underset{e}{w}, S) \vdash_M^\star (q, e, \underset{S}{\alpha})$ $w = e$, $\alpha = S$ $S \overset{L\star}{\Rightarrow} eS = S$

IH: If $(q, w, S) \vdash_M^\star (q, e, \alpha)$ by a computation of with n push (type 2) transitions, then $S \overset{L}{\Rightarrow}^\star w\alpha$

IS: Show the implication holds for a computation $n+1$ type-2 transitions.

$(q, w, S) \vdash_M^\star (q, e, \alpha)$ $n+1$ type 2 transitions $\gamma\beta = y\alpha$

$(q, w, S) \vdash_M^\star (q, y, A\beta) \vdash_M (q, y, \gamma\beta) \vdash_M^\star (q, e, \alpha)$

$w = xy$ read x n $(A \to \gamma \in R)$ $n+1$th T2 $y\alpha$ only T3 transition

Pushdown automata and context-free grammars

(2) $(q, w, S) \vdash_M^\star (q, e, \alpha)$ implies $S \overset{L}{\Rightarrow}^\star w\alpha$

By induction on the number of type-2 transitions.
IH: If $(q, w, S) \vdash_M^\star (q, e, \alpha)$ by a computation of with n push (type 2) transitions, then $S \overset{L}{\Rightarrow}^\star w\alpha$ **IS**: Show the implication holds for a computation $n+1$ type-2 transitions.

$\boxed{S \overset{L}{\Rightarrow}^\star x A\beta \quad \text{By IH}}$

$\rightarrow (q, xy, S) \vdash_M^\star (q, y, A\beta)$

$\rightarrow (q, x, S) \vdash_M^\star (q, e, A\beta)$ $n+2$ transitions

$S \overset{L}{\Rightarrow}^\star x A\beta \overset{L}{\Rightarrow} x\gamma\beta = w\alpha$

$A \to \gamma$

$= xy\alpha$

w

Pushdown automata and context-free grammars

$G \qquad \underline{M} \qquad L(G) = L(M)$

$w \in L(G) \Rightarrow w \in L(M),$
$w \in L(M) \Rightarrow w \in L(G)$

Theorem

The class of languages accepted by pushdown automata is exactly the class of context free languages.

Lemma

Each context free language is accepted by some pushdown automaton.

Lemma

If a language is accepted by a pushdown automaton, then it is a context-free language.

$M, \quad G \quad$ s.t. $\quad L(M) = L(G)$

Pushdown automata and context-free grammars

Lemma

If a language is accepted by a pushdown automaton, then it is a context-free language.

Proof idea: For any $M = (K, \Sigma, \Gamma, \Delta, s, F)$, there exists $G = (V, \Sigma, R, S)$ with $L(M) = L(G)$.

Pushdown automata and context-free grammars

Lemma

If a language is accepted by a pushdown automaton, then it is a context-free language.

Proof idea: For any $M = (K, \Sigma, \Gamma, \Delta, s, F)$, there exists $G = (V, \Sigma, R, S)$ with $L(M) = L(G)$.

- Convert M to M' such that M' is a simple automaton: transitions of M' satisfies the following property (if $q \neq s$)

$$((q, a, \beta), (p, \gamma)) : \qquad \beta \in \Gamma \cup \{e\}, |\gamma| \leq 2$$

- Prove that for each M, there exists a simple M' with $L(M) = L(M')$
- Construct a grammar $G = (V, \Sigma, R, S)$ from M'.
- Prove that $L(G) = L(M')$ (thus $L(G) = L(M)$)

Pushdown automata and context-free grammars

Convert M to M' such that M' is a simple automaton: transitions of M' satisfies the following property (if $q \neq s$)

$$((q, a, \beta), (p, \gamma)) : \qquad \beta \in \Gamma \cup \{e\}, |\gamma| \leq 2$$

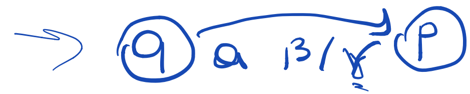
and $L(M) = L(M')$.

- $M = (K, \Sigma, \Gamma, \Delta, s, F)$, define $M' = (K', \Sigma, \Gamma \cup \{Z\}, \Delta', s', \{f'\})$
- $K' = K \cup \{s, f'\}$ $\bigcup$ ......
- $\Delta' = \Delta \cup \{((s', e, e), (s, Z))\} \cup \{((f, e, Z), (f', e) \mid f \in F\}$
- Replace each transition violating the requirement with a series of transitions, for each intermediate transition also add the intermediate states to K'.

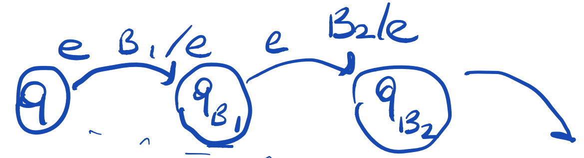
Replace the transitions violating the requirement that $|\beta| \leq 1$.

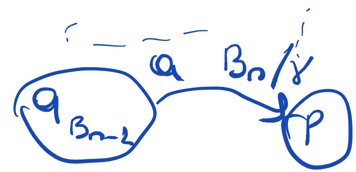
$$((q, a, \beta), (p, \gamma)) \in \Delta', \qquad \beta = B_1 \dots B_n, n > 1$$

Replace $((q, a, \beta), (p, \gamma))$ with



$$((q, e, B_1), (q_{B_1}, e))$$
$$((q_{B_1}, e, B_2), (q_{B_1 B_2}, e))$$
$$\dots$$
$$((q_{B_1 \dots B_{n-2}}, e, B_{n-1}), (q_{B_1 \dots B_{n-1}}, e))$$
$$((q_{B_1 \dots B_{n-1}}, a, B_n), (p, \gamma)),$$

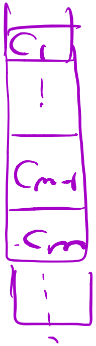




Add $q_{B_1}, \dots, q_{B_1 \dots B_{n-1}}$ to K' $((q, a, bb), (p, a))$

$((q, e, b), (q', e)), ((q', a, b), (p, a))$

Replace the transitions violating the requirement that $|\gamma| \leq 2$.



$$((q, a, \beta), (p, \gamma)) \in \Delta', \qquad \gamma = C_1 \ldots C_m, m > 2$$

Replace $((q, a, \beta), (p, \gamma))$ with

$$\begin{array}{l}
((q, a, \beta), (r_1, C_m)) \\
((r_1, e, e), (r_2, C_{m-1})) \\
\ldots \\
((r_{m-2}, e, e), (r_{m-1}, C_2)) \\
((r_{m-1}, e, e), (p, C_1))
\end{array}$$

Add $r_1, \ldots, r_{m-1}$ to K'

M is simple and $L(M) = L(M')$?

Construct $G = (V, \Sigma, R, S)$ from $M' = (K', \Sigma, \Gamma \cup \{Z\}, \Delta', s', \{f'\})$

non-terminal

- $V = \Sigma \cup \{S\} \cup \{\boxed{< q, A, p >} \mid q, p \in K', A \in \Sigma \cup \{e, Z\}\}$
- Rules R
  1. The rule $S \rightarrow < s, Z, f' >$
  2. For each $((q, a, B), (r, C)) \in \Delta'$ with $B, C \in \Gamma \cup \{e\}$ and for each $p \in K'$, add rule $< q, B, p > \rightarrow a < r, C, p >$
  3. For each $((q, a, B), (r, C_1 C_2))\Delta'$ with $B, C \in \Gamma \cup \{e\}$ and for each $p, p' \in K'$, add rule $< q, B, p > \rightarrow a < r, C_1, p' >< p', C_2, p >$
  4. For each $q' \in K'$, add $< q, e, q > \rightarrow e$

$L(G) = L(M')$

Proof: Home study.