Pushdown Automata

CENG 280

CC BY NC ND

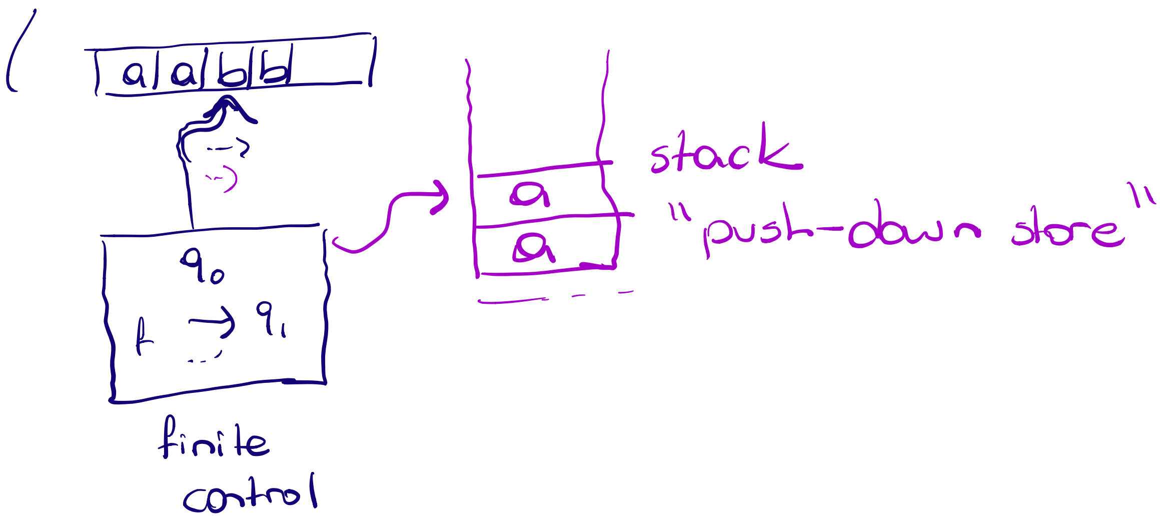
Course outline

- Preliminaries: Alphabets and languages
- Regular languages generator α, $L(\alpha)$
  acceptors FSA, NFA M $L(M)$
- Context-free languages
  - Context-free grammars generators
  - Parse trees
  - Push-down automaton acceptors
  - Push-down automaton - context-free languages
  - Languages that are and that are not context-free, Pumping lemma
- Turing-machines

Pushdown Automata

- What feature do we need to add to FSA so that it can recognize a CFL?
- Consider $a^n b^n$, ww^R.

read/write stack



Pushdown Automata

- What feature do we need to add to FSA so that it can recognize a CFL?
- Consider $a^n b^n$, ww^R.
- MEMORY !

Input tape, reading head, internal state, STACK (or pushdown store)

- Read at most one symbol at a time
- Read/write only the top of the stack
- Remove from top check against the current input
- The word is accepted, if when it is read, the automaton is in accepting state and the stack is empty.

Definition

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

- 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,
- $\Delta \subset (\underline{K} \times (\underline{\Sigma} \cup \{\underline{e}\}) \times \underline{\Gamma^\star}) \times (\underline{K} \times \Gamma^\star)$ is a finite transition relation.

$$((\underline{q}, \underline{a}, \underbrace{\alpha}_{\text{pop}}), (\underline{p}, \underbrace{\beta}_{\text{push}}))$$

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 (pop) with γ. (push)
- $((p, a, \delta), (q, \gamma))$ 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

Push/Pop transitions

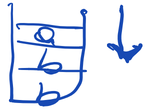
- $((p, a, e), (q, b))$: **push** transition, read a push b to the top of the stack. do not read/ pop

$\not\equiv$ $((p, a, e), (q, e))$: do not pop /push

$((p, a, a), (q, a))$: the content of the stack remains the same

pop a   push a

- $((p, a, b), (q, e))$: **pop** transition, read a pop b from the top of the stack.
  - if b is at the top of the stack
  - pops b from the stack

(p, x, α)  $\alpha = abb$

- The **configuration** of a pushdown automaton is a member of $\underline{K \times \Sigma^\star} \times \Gamma^\star$: the current state, unread part of the input type, the contents of the stack (read top-down).
- A configuration (p, x, α) of M **yields** (q, y, ζ) (shown as $(p, x, \alpha) \vdash_M (q, y, \zeta)$) if there is a transition $((p, a, \beta), (q, \gamma))$ such that
  - $x = ay$,
  - $\alpha = \beta\nu$,
  - $\zeta = \gamma\nu$ for some $\nu \in \Gamma^\star$.

$(p, x, \alpha) \vdash_M (q, y, \zeta)$

$(p, ay, \beta\nu) \vdash_M (q, y, \gamma\nu)$

$a \in \Sigma \cup \{e\}, \ \beta \in \Gamma^*$

$\rightarrow ((p, a, \beta), (q, \gamma))$

- The reflexive transitive closure of $\vdash_M$ is denoted by $\vdash_M^\star$.
- M accepts a word $w \in \Sigma^\star$ if and only if for some $f \in F$

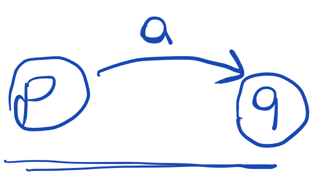
(s, w, e) $\rightarrow$ $$(s, w, e) \vdash_M^\star (f, e, e)$$ $w \in L(M)$

$$C_0 \vdash_M C_1 \vdash_M \cdots \vdash_M C_n = (f, e, e)$$

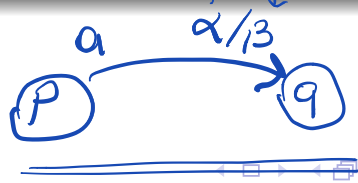
- Any sequence $C_0, \ldots, C_n$ with $C_i \vdash_M C_{i+1}$ is called a **computation** of M. It has length n (or n steps).
- The language accepted by M, $L(M)$ is the set of strings accepted by M.

Transition diagram

Show $((p, a, \alpha), (q, \beta))$ using an arrow from p to q with label "$a \quad \alpha/\beta$"



$(p, a, q) \in \Delta$



Example

PDA for $L = \{wcw^R \mid w \in \{a,b\}^*\}$. Accepting computation over $abcba$.

$M = (K, \Sigma, \Gamma, \Delta, s, F)$

$K = \{s, f\}$

$\Sigma = \{a, b, c\}$

$\Gamma = \{a, b\}$

$F = \{f\}$

1. $((s, a, e), (s, a))$ // push
2. $((s, b, e), (s, b))$ // push b
3. $((s, c, e), (f, e))$
4. $((f, a, a), (f, e))$ // compare and pop
5. $((f, b, b), (f, e))$ // compare & pop

State	unread part	stack	transition used
s	abcba	e	
s	bcba	a	1
s	cba	ba	2
f	ba	ba	3
f	a	a	5
f	e	e	4

Pushdown automata examples

Example

PDA for $L = \{wcw^R \mid w \in \{a, b\}^\star\}$. Accepting computation over $abcba$

$$(s, \underbrace{abcba}_{w}, e) \vdash_M (s, bcba, a) \vdash_M (s, cba, ba)$$

$$\vdash_M (f, ba, ba) \vdash_M (f, a, a) \vdash_M (f, e, e)$$

Pushdown automata examples

Example

Write a PDA for $L = \{ww^R \mid w \in \{a, b\}^*\}$, and computations over $abba$

$K = \{s, f\}$

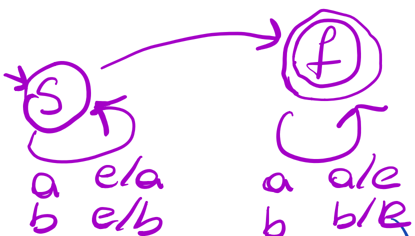
$\Sigma = \{a, b\}$

$\Gamma = \{a, b\}$

$F = \{f\}$

Δ:

1. $((s, a, e), (s, a))$ read & push
2. $((s, b, e), (s, b))$
3. $((s, e, e), (f, e))$
4. $((f, a, a), (f, e))$ read, compare, pop
5. $((f, b, b), (f, e))$



$(s, abba, e) \vdash_M (f, abba, e)$ stuck

1, 3

no information reporting

$abba \in L(M)$

$(s, abba, e) \vdash_M (s, bba, a) \vdash_M (s, ba, ba) \vdash_M (f, ba, ba) \vdash_M \ldots (f, e, e)$?

Example

Write a PDA M such that $L(M) = \{w \in \{a, b\}^* \mid w$ has the same number of $a's$ and $b's\}$ ($Grammar$?). $S \to e \mid SS \mid aSb \mid bSa$

$M = (K, \Sigma, \Gamma, \Delta, s, F)$

stack: hold a's / b's

seen more a's / seen more b's

$K = \{s, q, f\}$

$\Sigma = \{a, b\}$

$\Gamma = \{a, b, c\}$

$F = \{f\}$

1. $((s, e, e), (q, c))$ mark the bottom
2. $((q, a, c), (q, ac))$
3. $((q, a, a), (q, aa))$
4. $((q, b, a), (q, e))$
5. $((q, b, c), (q, bc))$
6. $((q, b, b), (q, bb))$
7. $((q, a, b), (q, e))$
8. $((q, e, c), (f, e))$

more a's then b's

$\#a - \#b =$ #a's in our stack

Example

Write a PDA M such that $L(M) = \{w \in \{0,1\}^\star \mid \sum_{i=0}^{|w|} w_i \leq \frac{|w|}{2}\}$.

number of 0's is more than or equal to the # of 1's

1. ((s,e,e), (q,c)) mark the bottom
2. ((q,0,c), (q,0c)) push
3. ((q,0,0), (q,00)) push
4. ((q,1,0), (q,e)) pop
5. ((q,1,c), (q,1c)) push
6. ((q,1,1), (q,11)) push
7. ((q,0,1), (q,e)) pop
8. ((q,e,e), (f,e))
9. ((f,e,0), (f,e))
10. ((f,e,c), (f,e))

(f,e,e)

Pushdown automata examples

Example

Given $G = (V, \Sigma, R, S)$ with $V = \{S, (,), [,]\}$, $\Sigma = \{(,), [,]\}$, and R:

$$S \to e \mid (S) \mid [S] \mid SS$$

Construct a PDA M such that $L(M) = L(G)$.

(( ) [ ])