COL 352 Introduction to Automata and Theory of Computation

Nikhil Balaji

Bharti 420
Indian Institute of Technology, Delhi
nbalaji@cse.iitd.ac.in

February 23, 2023

Lecture 15: Pushdown Automata

Recap

Definition

A 2DFA $A = (Q, \Sigma \cup \{\#, \$\}, \delta, q_0, q_{\text{acc}}, q_{\text{rej}})$, where

Q :	set of states,	Σ:	input alphabet
$\#$:	left endmarker	$\$$:	right endmarker
q_0:	start state		
q_{acc}:	accept state	q_{rej}:	reject state

$\delta : Q \times (\Sigma \cup \{\#, \$\} \to Q \times \{L, R\}$

The following conditions are forced:

$\forall q \in Q,\ \exists q', q'' \in Q$ s.t. $\delta(q, \#) = (q', R)$ and $\delta(q, \$) = (q'', L)$.

Recap

Definition

A 2DFA $A = (Q, \Sigma \cup \{\#, \$\}, \delta, q_0, q_{\text{acc}}, q_{\text{rej}})$, where

Q :	set of states,	Σ:	input alphabet
$\#$:	left endmarker	$\$$:	right endmarker
q_0:	start state		
q_{acc}:	accept state	q_{rej}:	reject state

$\delta : Q \times (\Sigma \cup \{\#, \$\} \to Q \times \{L, R\}$

The following conditions are forced:

$\forall q \in Q$, $\exists q', q'' \in Q$ s.t. $\delta(q, \#) = (q', R)$ and $\delta(q, \$) = (q'', L)$.

Exercise: Come up with a suitable definition of 2-NFA. Redo closure properties of regular languages, but now using 2-DFA/2-NFA.

Power of 2DFAs

Lemma

The class of language recognized by 2DFAs is regular.

Proof.

Let $T_x : Q \cup \{\bowtie\} \to Q \cup \{\perp\}$, which is defined as follows:

$T_x(p) := q$ if whenever A enters x on p it leaves x on q.

$T_x(\bowtie) := q$ q is the state in which A emerges on x the first time.

$T_x(q) := \perp$ if A loops on x forever.

□

Power of 2DFAs

Lemma

The class of language recognized by 2DFAs is regular.

Proof.

Let $T_x : Q \cup \{\bowtie\} \to Q \cup \{\perp\}$, which is defined as follows:

$T_x(p) := q$ if whenever A enters x on p it leaves x on q.

$T_x(\bowtie) := q$ q is the state in which A emerges on x the first time.

$T_x(q) := \perp$ if A loops on x forever.

Total number of functions of the type

$T_x \leq (|Q| + 1)^{(|Q|+1)}$

Power of 2DFAs

Lemma

The class of language recognized by 2DFAs is regular.

Proof.

Let $T_x : Q \cup \{\bowtie\} \to Q \cup \{\perp\}$, which is defined as follows:

$$
\begin{aligned}
T_x(p) &:= q && \text{if whenever } A \text{ enters } x \text{ on } p \\
& && \text{it leaves } x \text{ on } q. \\
T_x(\bowtie) &:= q && q \text{ is the state in which } A \text{ emerges} \\
& && \text{on } x \text{ the first time.} \\
T_x(q) &:= \perp && \text{if } A \text{ loops on } x \text{ forever.}
\end{aligned}
$$

Total number of functions of the type

$T_x \leq (|Q| + 1)^{(|Q|+1)}$

$T_x = T_y \Rightarrow \forall z (xz \in F \Leftrightarrow yz \in F)$. Prove this.

Power of 2DFAs

Lemma

The class of language recognized by 2DFAs is regular.

Proof.

Let $T_x : Q \cup \{\bowtie\} \to Q \cup \{\perp\}$, which is defined as follows:

$T_x(p) := q$ if whenever A enters x on p it leaves x on q.

$T_x(\bowtie) := q$ q is the state in which A emerges on x the first time.

$T_x(q) := \perp$ if A loops on x forever.

Total number of functions of the type

$T_x \leq (|Q| + 1)^{(|Q|+1)}$

$T_x = T_y \Rightarrow \forall z(xz \in F \Leftrightarrow yz \in F)$. Prove this.

$T_x = T_y \Leftrightarrow x \equiv_A y$

□

Moving on

How to we add expressive power to DFA/NFA so that we can compute more functions?

Pushdown automata

Pushdown automata

- $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.

Pushdown automata

- $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
- $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

Pushdown automata

- $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
- $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

Pushdown automata

- $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
- $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

NFA needs more memory to solve them.

Pushdown automata

- $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.

- $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

NFA needs more memory to solve them.What if the NFA had a stack?

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q: set of states

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q: set of states Σ: input alphabet

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet		

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA) $A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol		

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol	F:	set of final states

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA) $A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol	F:	set of final states

$$\delta \subseteq Q \times \Sigma \times \Gamma \times Q \times \Gamma^*.$$

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA) $A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol	F:	set of final states

$$\delta \subseteq Q \times \Sigma \times \Gamma \times Q \times \Gamma^*.$$

Understanding δ

For $q \in Q, a \in \Sigma$ and $X \in \Gamma$, if $\delta(q, a, X) = (p, \gamma)$,

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol	F:	set of final states

$$\delta \subseteq Q \times \Sigma \times \Gamma \times Q \times \Gamma^*.$$

Understanding δ

For $q \in Q, a \in \Sigma$ and $X \in \Gamma$, if $\delta(q, a, X) = (p, \gamma)$,
then p is the new state and γ replaces X in the stack.

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA) $A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol	F:	set of final states

$$\delta \subseteq Q \times \Sigma \times \Gamma \times Q \times \Gamma^*.$$

Understanding δ

For $q \in Q, a \in \Sigma$ and $X \in \Gamma$, if $\delta(q, a, X) = (p, \gamma)$,
then p is the new state and γ replaces X in the stack.

if $\gamma = \epsilon$ then X is popped.

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol	F:	set of final states

$$\delta \subseteq Q \times \Sigma \times \Gamma \times Q \times \Gamma^*.$$

Understanding δ

For $q \in Q, a \in \Sigma$ and $X \in \Gamma$, if $\delta(q, a, X) = (p, \gamma)$,
then p is the new state and γ replaces X in the stack.

- if $\gamma = \epsilon$ then X is popped.
- if $\gamma = X$ then X stays unchanges on the top of the stack.

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA) $A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol	F:	set of final states

$$\delta \subseteq Q \times \Sigma \times \Gamma \times Q \times \Gamma^*.$$

Understanding δ

For $q \in Q, a \in \Sigma$ and $X \in \Gamma$, if $\delta(q, a, X) = (p, \gamma)$, then p is the new state and γ replaces X in the stack.

if $\gamma = \epsilon$ then X is popped.

if $\gamma = X$ then X stays unchanges on the top of the stack.

if $\gamma = \gamma_1\gamma_2 \dots \gamma_k$ then X is replaced by γ_k

Pushdown automata: formal definition

Definition

A non-deterministic pushdown automaton (NPDA)
$A = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$, where

Q:	set of states	Σ:	input alphabet
Γ:	stack alphabet	q_0:	start state
$\perp$:	start symbol	F:	set of final states

$$\delta \subseteq Q \times \Sigma \times \Gamma \times Q \times \Gamma^*.$$

Understanding δ

For $q \in Q, a \in \Sigma$ and $X \in \Gamma$, if $\delta(q, a, X) = (p, \gamma)$,
then p is the new state and γ replaces X in the stack.

if $\gamma = \epsilon$ then X is popped.
if $\gamma = X$ then X stays unchanges on the top of the stack.
if $\gamma = \gamma_1\gamma_2 \ldots \gamma_k$ then X is replaced by γ_k
and $\gamma_1\gamma_2 \ldots \gamma_{k-1}$ are pushed on top of that.

Pushdown automata

Example:

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \perp) = (q_0, B \perp)$$

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \perp) = (q_0, B \perp)$$

$$\delta(q_0, a, B) = (q_0, B)$$

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push* B *while reading* a *on* q_0

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push* B *while reading* a *on* q_0

$\delta(q_0, b, B) = (q_1, \epsilon)$

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push B while reading a on q_0*

$\delta(q_0, b, B) = (q_1, \epsilon)$ *move to q_1 on seeing b*

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push B while reading a on q_0*

$\delta(q_0, b, B) = (q_1, \epsilon)$ *move to q_1 on seeing b*

$\delta(q_1, b, B) = (q_1, \epsilon)$

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push* B *while reading* a *on* q_0

$\delta(q_0, b, B) = (q_1, \epsilon)$ *move to* q_1 *on seeing* b

$\delta(q_1, b, B) = (q_1, \epsilon)$ *pop* B *while reading* b *on* q_1

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push* B *while reading* a *on* q_0

$\delta(q_0, b, B) = (q_1, \epsilon)$ *move to* q_1 *on seeing* b

$\delta(q_1, b, B) = (q_1, \epsilon)$ *pop* B *while reading* b *on* q_1

$\delta(q_1, b, \perp) = (q', \perp)$

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B\perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push* B *while reading* a *on* q_0

$\delta(q_0, b, B) = (q_1, \epsilon)$ *move to* q_1 *on seeing* b

$\delta(q_1, b, B) = (q_1, \epsilon)$ *pop* B *while reading* b *on* q_1

$\delta(q_1, b, \perp) = (q', \perp)$ *extra* b *in the input then go to some state* q'

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push B while reading a on q_0*

$\delta(q_0, b, B) = (q_1, \epsilon)$ *move to q_1 on seeing b*

$\delta(q_1, b, B) = (q_1, \epsilon)$ *pop B while reading b on q_1*

$\delta(q_1, b, \perp) = (q', \perp)$ *extra b in the input then go to some state q'*

$\delta(q_1, \epsilon, \perp) = (q_2, \perp)$

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push* B *while reading* a *on* q_0

$\delta(q_0, b, B) = (q_1, \epsilon)$ *move to* q_1 *on seeing* b

$\delta(q_1, b, B) = (q_1, \epsilon)$ *pop* B *while reading* b *on* q_1

$\delta(q_1, b, \perp) = (q', \perp)$ *extra* b *in the input then go to some state* q'

$\delta(q_1, \epsilon, \perp) = (q_2, \perp)$ *else go to accepting state* q_2

Pushdown automata

Example: $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{B\}, q_0, F = \{q_2\})$.

$\delta(q_0, a, \perp) = (q_0, B \perp)$

$\delta(q_0, a, B) = (q_0, B)$ *push* B *while reading* a *on* q_0

$\delta(q_0, b, B) = (q_1, \epsilon)$ *move to* q_1 *on seeing* b

$\delta(q_1, b, B) = (q_1, \epsilon)$ *pop* B *while reading* b *on* q_1

$\delta(q_1, b, \perp) = (q', \perp)$ *extra* b *in the input then go to some state* q'

$\delta(q_1, \epsilon, \perp) = (q_2, \perp)$ *else go to accepting state* q_2

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{A, B\}, q_0, F = \{q_2\})$.

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{A, B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \perp) = \{(q_0, A \perp), (q_1, A \perp)\}$$

$$\delta(q_0, b, \perp) = \{(q_0, B \perp), (q_1, B \perp)\}$$

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{A, B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \perp) = \{(q_0, A \perp), (q_1, A \perp)\}$$

$$\delta(q_0, b, \perp) = \{(q_0, B \perp), (q_1, B \perp)\}$$

$$\delta(q_0, a, A) = \{(q_0, AA), (q_1, AA)\}, \ \delta(q_0, a, B) = \{(q_0, AB), (q_1, AB)\}$$

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.
$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{A, B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \bot) = \{(q_0, A\bot), (q_1, A\bot)\}$$
$$\delta(q_0, b, \bot) = \{(q_0, B\bot), (q_1, B\bot)\}$$
$$\delta(q_0, a, A) = \{(q_0, AA), (q_1, AA)\},\ \delta(q_0, a, B) = \{(q_0, AB), (q_1, AB)\}$$
$$\delta(q_0, b, A) = \{(q_0, BA), (q_1, BA)\},$$
$$\delta(q_0, b, B) = \{(q_0, BB), (q_1, BB)\}$$

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{A, B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \perp) = \{(q_0, A \perp), (q_1, A \perp)\}$$

$$\delta(q_0, b, \perp) = \{(q_0, B \perp), (q_1, B \perp)\}$$

$$\delta(q_0, a, A) = \{(q_0, AA), (q_1, AA)\},\ \delta(q_0, a, B) = \{(q_0, AB), (q_1, AB)\}$$

$$\delta(q_0, b, A) = \{(q_0, BA), (q_1, BA)\},$$
$$\delta(q_0, b, B) = \{(q_0, BB), (q_1, BB)\}$$

$$\delta(q_1, a, A) = (q_1, \epsilon)$$

$$\delta(q_1, b, B) = (q_1, \epsilon)$$

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{A, B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \perp) = \{(q_0, A\perp), (q_1, A\perp)\}$$

$$\delta(q_0, b, \perp) = \{(q_0, B\perp), (q_1, B\perp)\}$$

$$\delta(q_0, a, A) = \{(q_0, AA), (q_1, AA)\},\ \delta(q_0, a, B) = \{(q_0, AB), (q_1, AB)\}$$

$$\delta(q_0, b, A) = \{(q_0, BA), (q_1, BA)\},$$
$$\delta(q_0, b, B) = \{(q_0, BB), (q_1, BB)\}$$

$$\delta(q_1, a, A) = (q_1, \epsilon)$$

$$\delta(q_1, b, B) = (q_1, \epsilon)$$

$$\delta(q_1, a, B) = (q', \epsilon)$$

$$\delta(q_1, b, A) = (q', \epsilon)$$

$$\delta(q_1, a, \perp) = (q', \epsilon)$$

$$\delta(q_1, b, \perp) = (q', \epsilon)$$

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{A, B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \perp) = \{(q_0, A\perp), (q_1, A\perp)\}$$

$$\delta(q_0, b, \perp) = \{(q_0, B\perp), (q_1, B\perp)\}$$

$$\delta(q_0, a, A) = \{(q_0, AA), (q_1, AA)\},\ \delta(q_0, a, B) = \{(q_0, AB), (q_1, AB)\}$$

$$\delta(q_0, b, A) = \{(q_0, BA), (q_1, BA)\},$$
$$\delta(q_0, b, B) = \{(q_0, BB), (q_1, BB)\}$$

$$\delta(q_1, a, A) = (q_1, \epsilon)$$

$$\delta(q_1, b, B) = (q_1, \epsilon)$$

$$\delta(q_1, a, B) = (q', \epsilon)$$

$$\delta(q_1, b, A) = (q', \epsilon)$$

$$\delta(q_1, a, \perp) = (q', \epsilon)$$

$$\delta(q_1, b, \perp) = (q', \epsilon)$$

$$\delta(q_1, \epsilon, \perp) = (q_2, \perp)$$

Nondeterministic pushdown automata

Example: $PAL = \{w \cdot w^R \mid w \in \Sigma^*\}$.

$N = (Q = \{q_0, q_1, q_2, q'\}, \Sigma = \{a, b\}, \Gamma = \{A, B\}, q_0, F = \{q_2\})$.

$$\delta(q_0, a, \perp) = \{(q_0, A \perp), (q_1, A \perp)\}$$

$$\delta(q_0, b, \perp) = \{(q_0, B \perp), (q_1, B \perp)\}$$

$$\delta(q_0, a, A) = \{(q_0, AA), (q_1, AA)\},\ \delta(q_0, a, B) = \{(q_0, AB), (q_1, AB)\}$$

$$\delta(q_0, b, A) = \{(q_0, BA), (q_1, BA)\},$$
$$\delta(q_0, b, B) = \{(q_0, BB), (q_1, BB)\}$$

$$\delta(q_1, a, A) = (q_1, \epsilon)$$

$$\delta(q_1, b, B) = (q_1, \epsilon)$$

$$\delta(q_1, a, B) = (q', \epsilon)$$

$$\delta(q_1, b, A) = (q', \epsilon)$$

$$\delta(q_1, a, \perp) = (q', \epsilon)$$

$$\delta(q_1, b, \perp) = (q', \epsilon)$$

$$\delta(q_1, \epsilon, \perp) = (q_2, \perp)$$