Introduction to the Theory of Computation

XU Ming

School of Software Engineering, East China Normal University

May 16, 2024

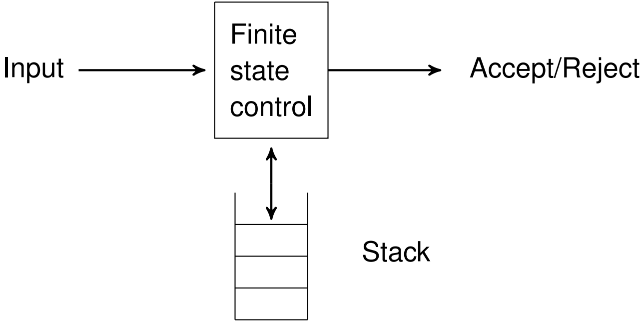
OUTLINE

- Pushdown Automata
- The Language of a Pushdown Automaton

Pushdown Automata

PDA Informally

A pushdown automaton (PDA) is essentially an ϵ-NFA with a stack.



On a transition in the PDA:

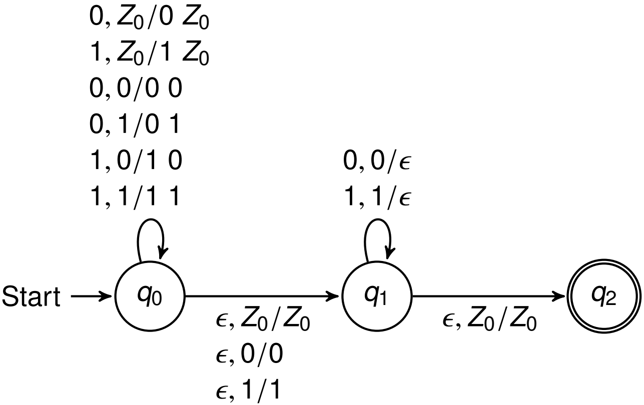
1. Consumes an input symbol.
2. Goes to a new state (or stays in the old).
3. Replaces the symbol at the top of the stack by any string. It could be
    - ϵ (pops the stack);
    - same symbol (does nothing);
    - one other symbol (changes the top of the stack, neigher pushes nor pops);
    - two or more symbols, (changes the top stack symbol, and then pushes one or more new symbols onto the stack).

Example

Let's consider $L_{wwr} = \{ww^R \mid w \in \{0, 1\}^*\}$ with grammar $S \to 0S0 \mid 1S1 \mid \epsilon$.
A PDA for L_{wwr} has three states, and operates as follows:

1. Guess that you're reading w. Stay in q_0, and push the input symbol onto the stack.
2. Guess that you're in the middle of ww^R. Go spontaneously to state q_1.
3. You're reading the head of w^R. Compare it to the top of the stack. If match, pop the stack, and remain in state q_1. If not match, go to sleep.
4. If the stack is empty, go to state q_2 and accept.

The PDA for L_{wwr} as a transition diagram:

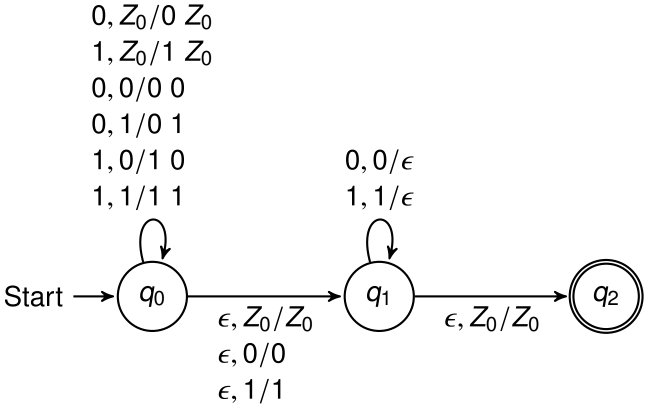


A pushdown automaton (PDA) is a 7-tuple $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$, in which

- Q is a finite set of *states*,
- Σ is a finite *input alphabet*,
- Γ is a finite *stack alphabet*,
- δ is a *transition function* from $Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma$ to $2^{Q \times \Gamma^*}$,
- $q_0 \in Q$ is the *start state*,
- $Z_0 \in \Gamma$ is the *start symbol* for the stack, and
- $F \subseteq Q$ is a set of *accepting* states.

Example

The PDA



is actually the 7-tuple.

$$P = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\}, \delta, q_0, Z_0, \{q_2\})$$

where δ is given by the following table (set brackets omitted for conciseness):

	$0, Z_0$	$1, Z_0$	$0, 0$	$1, 0$	$0, 1$	$1, 1$	ϵ, Z_0	$\epsilon, 0$	$\epsilon, 1$
$\rightarrow q_0$	$q_0, 0Z_0$	$q_0, 1Z_0$	$q_0, 00$	$q_0, 10$	$q_0, 01$	$q_0, 11$	q_1, Z_0	$q_1, 0$	$q_1, 1$
q_1	–	–	q_1, ϵ	–	–	q_1, ϵ	q_2, Z_0	–	–
$\star q_2$	–	–	–	–	–	–	–	–	–

Instantaneous Description

A PDA goes from configurations to configurations when consuming input.

To reason about PDA computation, we use instantaneous description (ID) of the PDA. An ID is a triple

$$(q, w, \gamma)$$

where q is the state, w is the remaining input, and γ is the stack contents.

Let $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ be a PDA. Then $\forall w \in \Sigma^*, \beta \in \Gamma^*$:

$$(p, \alpha) \in \delta(q, a, X) \Rightarrow (q, aw, X\beta) \vdash (p, w, \alpha\beta).$$

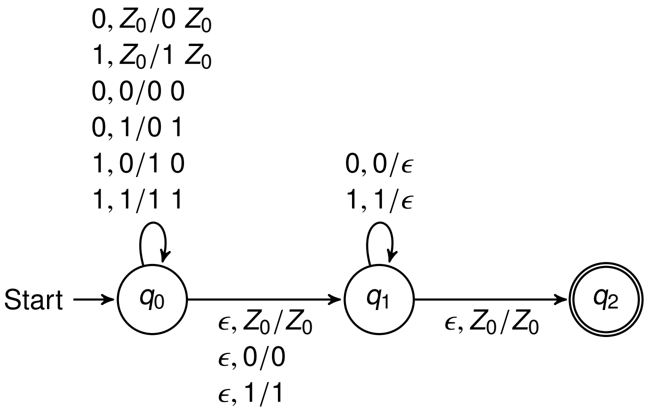
We define $\overset{*}{\vdash}$ to be the reflexive-transitive closure of $\vdash$:

Basis step: $I \overset{*}{\vdash} I$ for any ID I.

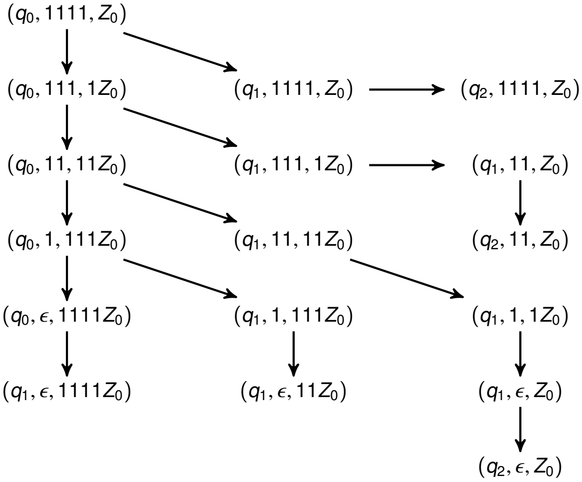
Inductive step: $I \overset{*}{\vdash} J$ if there exists some ID K such that $I \vdash K$ and $K \overset{*}{\vdash} J$.

Example

On input 1111 the PDA



has the following computation sequences.


$(q_0, 1111, Z_0)$
$(q_0, 111, 1Z_0)$
$(q_1, 1111, Z_0)$
$(q_2, 1111, Z_0)$
$(q_0, 11, 11Z_0)$
$(q_1, 111, 1Z_0)$
$(q_1, 11, Z_0)$
$(q_0, 1, 111Z_0)$
$(q_1, 11, 11Z_0)$
$(q_2, 11, Z_0)$
$(q_0, \epsilon, 1111Z_0)$
$(q_1, 1, 111Z_0)$
$(q_1, 1, 1Z_0)$
$(q_1, \epsilon, 1111Z_0)$
$(q_1, \epsilon, 11Z_0)$
(q_1, ϵ, Z_0)
(q_2, ϵ, Z_0)

There are three important principles about ID's and their transitions:

1. If an ID sequence is a legal computation for a PDA, then so is the sequence obtained by adding an additional string at the end of the second component.
2. If an ID sequence is a legal computation for a PDA, then so is the sequence obtained by adding an additional string at the bottom of the third component.
3. If an ID sequence is a legal computation for a PDA, and some tail of the input is never consumed, then removing this tail from all ID's results in a legal computation sequence.

Theorem 6.1

Let $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ *be a PDA. Then* $\forall w \in \Sigma^*, \gamma \in \Gamma^*$:

$$(q, x, \alpha) \overset{*}{\vdash} (p, y, \beta) \Rightarrow (q, xw, \alpha\gamma) \overset{*}{\vdash} (p, yw, \beta\gamma).$$

Proof Induction on the number of steps in the sequence of ID's that take (q, x, α) to (p, y, β). □

Note that if $\gamma = \epsilon$ we have the first principle above, and if $w = \epsilon$, then we have the second principle.

The reverse of the theorem is false.

We state the third principle formally as:

Theorem 6.2

Let $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ be a PDA.

$$(q, xw, \alpha) \overset{*}{\vdash} (p, yw, \beta) \Rightarrow (q, x, \alpha) \overset{*}{\vdash} (p, y, \beta).$$

Although a FA has no stack, we could use a pair (q, w) as the ID of a finite automaton. For any FA, we could show that $\hat{\delta}(q, w) = p$ if and only if $(q, wx) \overset{*}{\vdash} (p, x)$ for all strings x. The fact that x can be anything we wish without influencing the behavior of the FA is a theorem analogous to Theorems 6.1 and 6.2.

The Language of a Pushdown Automaton

We have two approaches to defining the language of a PDA: “acceptance by final state” and “acceptance by empty stack”.

- These two methods are equivalent, in the sense that a language L has a PDA that accepts it by final state iff L has a PDA that accepts it by empty stack.
- For a given PDA P, the languages that P accepts by final state and by empty stack are usually different.

Acceptance by the Final State

Let $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ be a PDA. The language accepted by P by final state is

$$L(P) = \{w \mid (q_0, w, Z_0) \overset{*}{\vdash} (q, \epsilon, \alpha), q \in F\}.$$

Example

Below PDA accepts exactly L_{wwr} by final state.

	$0, Z_0$	$1, Z_0$	$0, 0$	$1, 0$	$0, 1$	$1, 1$	ϵ, Z_0	$\epsilon, 0$	$\epsilon, 1$
$\rightarrow q_0$	$q_0, 0Z_0$	$q_0, 1Z_0$	$q_0, 00$	$q_0, 10$	$q_0, 01$	$q_0, 11$	q_1, Z_0	$q_1, 0$	$q_1, 1$
q_1	–	–	q_1, ϵ	–	–	q_1, ϵ	q_2, Z_0	–	–
$\star q_2$	–	–	–	–	–	–	–	–	–

Let $P = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\}, \delta, q_0, Z_0, \{q_2\})$ be the machine. We prove that $L(P) = L_{wwr}$.

($\supseteq$-direction) Let $x \in L_{wwr}$. Then $x = ww^R$, and the following is a legal computation sequence

$$(q_0, ww^R, Z_0) \overset{*}{\vdash} (q_0, w^R, w^R Z_0) \vdash (q_1, w^R, w^R Z_0) \overset{*}{\vdash} (q_1, \epsilon, Z_0) \vdash (q_2, \epsilon, Z_0).$$

($\subseteq$-direction) Observe that the only way the PDA can enter q_2 is if it is in state q_1 with only element Z_0. Thus it suffices to show that if $(q_0, x, Z_0) \overset{*}{\vdash} (q_1, \epsilon, Z_0)$ then $x = ww^R$, for some word w.

We'll show by an induction on $|x|$ that

$$(q_0, x, \alpha) \overset{*}{\vdash} (q_1, \epsilon, \alpha) \Rightarrow x = ww^R.$$

Basis step: If $x = \epsilon$ then x is a palindrome.

Inductive step: Suppose $x = a_1 a_2 \cdots a_n$, where $n > 0$, and the induction hypothesis holds for shorter strings. There are two moves for the PDA from ID(q_0, x, α):

Move 1: The spontaneous $(q_0, x, \alpha) \vdash (q_1, x, \alpha)$. Now $(q_1, x, \alpha) \overset{*}{\vdash} (q_1, \epsilon, \beta)$ implies $|\beta| < |\alpha|$ and thus $\beta \neq \alpha$.

Move 2: Loop and push $(q_0, a_1a_2\cdots a_n, \alpha) \vdash (q_0, a_2\cdots a_n, a_1\alpha)$. In this case there is a sequence

$$(q_0, a_1a_2\cdots a_n, \alpha) \vdash (q_0, a_2\cdots a_n, a_1\alpha) \vdash \cdots \vdash (q_1, a_n, a_1\alpha) \vdash (q_1, \epsilon, \alpha).$$

Thus $a_1 = a_n$ and $(q_0, a_2\cdots a_n, a_1\alpha) \overset{*}{\vdash} (q_1, a_n, a_1\alpha)$. By Theorem 6.2 we can remove a_n. Therefore

$$(q_0, a_2\cdots a_{n-1}, a_1\alpha) \overset{*}{\vdash} (q_1, \epsilon, a_1\alpha).$$

Then, by the induction hypothesis $a_2\cdots a_{n-1} = yy^R$. Then $x = a_1yy^Ra_n$ is a palindrome. $\square$

Acceptance by the Empty Stack

Let $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ be a PDA. The language accepted by P by empty stack is

$$N(P) = \{w \mid (q_0, w, Z_0) \overset{*}{\vdash} (q, \epsilon, \epsilon)\}$$

for any state q.

Question How to modify the palindrome PDA to accept by empty stack?

Since the set of accepting states is irrelevant, we would write a PDA P as a 6-tuple $(Q, \Sigma, \Gamma, \delta, q_0, Z_0)$ when all we care about is the language that P accepts by empty stack.

From Empty Stack to Final State

Theorem 6.3

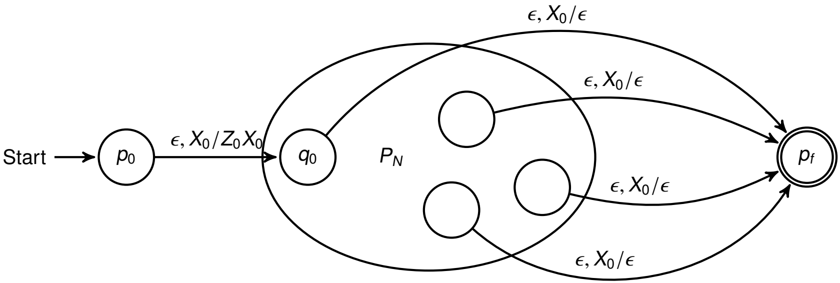
If $L = N(P_N)$ for some PDA $P_N = (Q, \Sigma, \Gamma, \delta_N, q_0, Z_0)$, then there exists a PDA P_F such that $L = L(P_F)$.

Proof Let $P_F = (Q \cup \{p_0, p_f\}, \Sigma, \Gamma \cup \{X_0\}, \delta_F, p_0, X_0, \{p_f\})$ where $\delta_F(p_0, \epsilon, X_0) = \{(q_0, Z_0X_0)\}$, and for all $q \in Q$, $a \in \Sigma \cup \{\epsilon\}$, $Y \in \Gamma$

$$\delta_F(q, a, Y) = \delta_N(q, a, Y),$$

and in addition $(p_f, \epsilon) \in \delta_F(q, \epsilon, X_0)$.

The diagram is



We have shown that $L(P_F) = N(P_N)$.

($\subseteq$-direction) Inspect the diagram.

($\supseteq$-direction) Let $w \in N(P_N)$. Then $(q_0, w, Z_0) \overset{*}{\underset{N}{\vdash}} (q, \epsilon, \epsilon)$, for some q. From Theorem 6.1 we get

$$(q_0, w, Z_0X_0) \overset{*}{\underset{N}{\vdash}} (q, \epsilon, X_0).$$

Since $\delta_N \subset \delta_F$ we have

$$(q_0, w, Z_0X_0) \overset{*}{\underset{F}{\vdash}} (q, \epsilon, X_0).$$

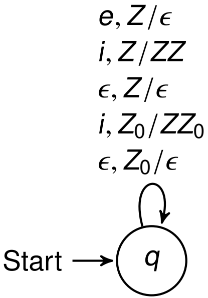
We conclude that

$$(p_0, w, X_0) \underset{F}{\vdash} (q_0, w, Z_0X_0) \overset{*}{\underset{F}{\vdash}} (q, \epsilon, X_0) \underset{F}{\vdash} (p_f, \epsilon, \epsilon).$$

□

Example

Let's design P_N for the *if-else*-grammar G: $S \to \epsilon \,|\, SS \,|\, iS \,|\, iSe$.
The diagram for P_N is



Formally, $P_N = (\{q\}, \{i, e\}, \{Z, Z_0\}, \delta_N, q, Z_0)$, where

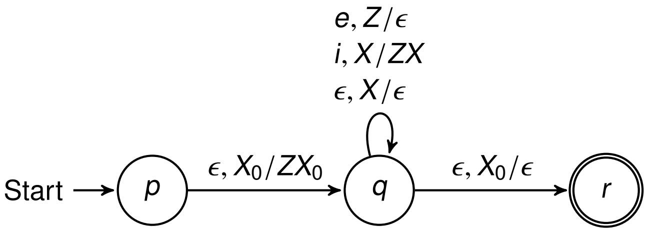
$$\delta_N(q, i, X) = \{(q, ZX)\} \quad \delta_N(q, e, Z) = \{(q, \epsilon)\} \quad \delta_N(q, \epsilon, X) = \{(q, \epsilon)\}$$

for $X = Z$ or $X = Z_0$.

From P_N we can construct $P_F = (\{p, q, r\}, \{i, e\}, \{Z, X_0\}, \delta_F, p, X_0, \{r\})$, where

$$\begin{aligned}
&\delta_F(p, \epsilon, X_0) = \{(q, ZX_0)\}, \\
&\delta_F(q, i, X) = \delta_N(q, i, X) = \{(q, ZX)\}, \\
&\delta_F(q, e, Z) = \delta_N(q, e, Z) = \{(q, \epsilon)\}, \\
&\delta_F(q, \epsilon, X) = \delta_N(q, \epsilon, X) = \{(q, \epsilon)\}, \\
&\delta_F(q, \epsilon, X_0) = \{(r, \epsilon)\}.
\end{aligned}$$

The diagram for P_F is



From Final State to Empty Stack

Theorem 6.4

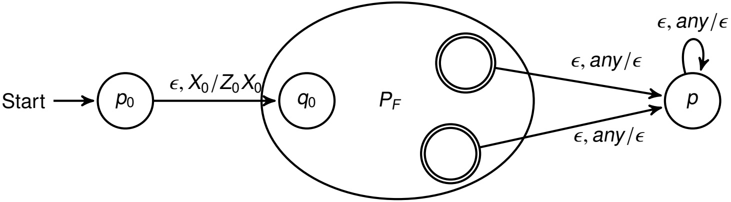
Let $L = L(P_F)$ for some PDA $P_F = (Q, \Sigma, \Gamma, \delta_F, q_0, Z_0, F)$. Then there is a PDA P_N such that $L = N(P_N)$.

Proof Let $P_N = (Q \cup \{p_0, p\}, \Sigma, \Gamma \cup \{X_0\}, \delta_N, p_0, X_0)$, where $\delta_N(p_0, \epsilon, X_0) = \{(q_0, Z_0X_0)\}, \delta_N(p, \epsilon, Y) = \{(p, \epsilon)\}$ for $Y \in \Gamma \cup \{X_0\}$, and for all $q \in Q$, $a \in \Sigma \cup \{\epsilon\}$, $Y \in \Gamma$

$$\delta_N(q, a, Y) = \delta_F(q, a, Y),$$

and in addition $\forall\, q \in F$ and $Y \in \Gamma \cup \{X_0\} : (p, \epsilon) \in \delta_N(q, \epsilon, Y)$.

The diagram is



We have shown that $N(P_N) = L(P_F)$.

($\subseteq$-direction) Inspect the diagram.

($\supseteq$-direction) Let $w \in L(P_F)$. Then $(q_0, w, Z_0) \overset{*}{\underset{F}{\vdash}} (q, \epsilon, \alpha)$, for some $q \in F, \alpha \in \Gamma^*$. Since $\delta_F \subset \delta_N$, and Theorem 6.1 says that X_0 can be slid under the stack, we get

$$(q_0, w, Z_0X_0) \overset{*}{\underset{N}{\vdash}} (q, \epsilon, \alpha X_0).$$

The P_N can compute

$$(p_0, w, X_0) \underset{N}{\vdash} (q_0, w, Z_0X_0) \overset{*}{\underset{N}{\vdash}} (q, \epsilon, \alpha X_0) \underset{N}{\vdash} (p, \epsilon, \alpha X_0) \overset{*}{\underset{N}{\vdash}} (p, \epsilon, \epsilon).$$

□

Example

What language is accepted by the following PDA with final state?

$$P = (\{q_0, q_1, q_2\}, \{a, b\}, \{a, b, Z_0\}, \delta, q_0, Z_0, \{q_2\})$$

with transitions $\delta(q_0, a, Z_0) = \{(q_1, a), (q_2, \epsilon)\}$, $\delta(q_1, b, a) = \{(q_1, b)\}$, $\delta(q_1, b, b) = \{(q_1, b)\}$, $\delta(q_1, a, b) = \{(q_2, \epsilon)\}$.

Solution $L = \{a\} \cup L(\mathbf{abb^*a})$.

What language is accepted upon empty stack?

Example

What language is accepted by the following PDA with final state?

$$P = (\{q_0, q_1, q_2\}, \{a, b\}, \{a, b, Z_0\}, \delta, q_0, Z_0, \{q_2\})$$

with transitions $\delta(q_0, a, Z_0) = \{(q_1, a), (q_2, \epsilon)\}$, $\delta(q_1, b, a) = \{(q_1, b)\}$, $\delta(q_1, b, b) = \{(q_1, b)\}$, $\delta(q_1, a, b) = \{(q_2, \epsilon)\}$.

Solution $L = \{a\} \cup L(\mathbf{abb^*a})$.

What language is accepted upon empty stack?

Homework

Exercises 6.1.1(a), 6.2.8