Theory of Machines and Languages

Fatemeh Deldar

1403-1404

Pushdown Automata for Context-Free Languages

- **Example**

$$S \rightarrow aSbb|a.$$

We first transform the grammar into Greibach normal form

$$S \rightarrow aSA|a,$$
$$A \rightarrow bB,$$
$$B \rightarrow b.$$

The start symbol S is put on the stack

$$\delta(q_0, \lambda, z) = \{(q_1, Sz)\}$$

$$\delta(q_1, a, S) = \{(q_1, SA), (q_1, \lambda)\}$$
$$\delta(q_1, b, A) = \{(q_1, B)\}$$
$$\delta(q_1, b, B) = \{(q_1, \lambda)\}$$

- For the production $S \rightarrow aSA$
  - Remove S from the stack and replace it with SA, while reading a from the input
- For the production $S \rightarrow a$
  - Read a from the input while removing S from the stack
- …

By appearing the start symbol on top of the stack the pda is put into its final state

$$\delta(q_1, \lambda, z) = \{(q_2, \lambda)\}$$

Pushdown Automata for Context-Free Languages

- **Example**

$$
\begin{aligned}
S &\to aA, \\
A &\to aABC \,|bB|\, a, \\
B &\to b, \\
C &\to c.
\end{aligned}
$$

$$
\begin{aligned}
&\delta(q_0, \lambda, z) = \{(q_1, Sz)\} \\
&\delta(q_1, \lambda, z) = \{(q_f, z)\}, \\
&\delta(q_1, a, S) = \{(q_1, A)\}, \\
&\delta(q_1, a, A) = \{(q_1, ABC), (q_1, \lambda)\}, \\
&\delta(q_1, b, A) = \{(q_1, B)\}, \\
&\delta(q_1, b, B) = \{(q_1, \lambda)\}, \\
&\delta(q_1, c, C) = \{(q_1, \lambda)\}.
\end{aligned}
$$

The sequence of moves made by *M* in processing *aaabc* is

$$
\begin{aligned}
(q_0, aaabc, z) &\vdash (q_1, aaabc, Sz) \\
&\vdash (q_1, aabc, Az) \\
&\vdash (q_1, abc, ABCz) \\
&\vdash (q_1, bc, BCz) \\
&\vdash (q_1, c, Cz) \\
&\vdash (q_1, \lambda, z) \\
&\vdash (q_f, \lambda, z).
\end{aligned}
$$

Deterministic Pushdown Automata and Deterministic Context-Free Languages

- **A deterministic pushdown accepter (dpda) is a pushdown automaton with the following restrictions:**

for every $q \in Q$, $a \in \Sigma \cup \{\lambda\}$ and $b \in \Gamma$

1. $\delta(q, a, b)$ contains at most one element,

2. if $\delta(q, \lambda, b)$ is not empty, then $\delta(q, c, b)$ must be empty for every $c \in \Sigma$.

For any given input symbol and any stack top, at most one move can be made

when a λ-move is possible for some configuration, no input-consuming alternative is available

Deterministic Pushdown Automata and Deterministic Context-Free Languages

- A language L is said to be a **deterministic context-free language** if and only if there exists a dpda M such that $L = L(M)$.

- **Example**

The language

$$L = \{a^n b^n : n \geq 0\}$$

is a deterministic context-free language. The pda $M = (\{q_0, q_1, q_2\}, \{a, b\}, \{0, 1\}, \delta, q_0, 0, \{q_0\})$ with

$$\begin{aligned}
\delta(q_0, a, 0) &= \{(q_1, 10)\}, \\
\delta(q_1, a, 1) &= \{(q_1, 11)\}, \\
\delta(q_1, b, 1) &= \{(q_2, \lambda)\}, \\
\delta(q_2, b, 1) &= \{(q_2, \lambda)\}, \\
\delta(q_2, \lambda, 0) &= \{(q_0, \lambda)\}
\end{aligned}$$

accepts the given language.