Lecture 24

PDAs and CFGs

In this lecture and the next we will show that nondeterministic pushdown automata and context-free grammars are equivalent in expressive power: the languages accepted by NPDAs are exactly the context-free languages. In this lecture we will show how to convert a given CFG to an equivalent NPDA. We will do the other direction in Lecture 25.

Suppose we are given a CFG $G = (N, \Sigma, P, S)$. We wish to construct an NPDA M such that $L(M) = L(G)$. By a simple construction from Lecture 21, we can assume without loss of generality that all productions of G are of the form

$$A \to cB_1B_2 \cdots B_k,$$

where $c \in \Sigma \cup \{\epsilon\}$ and $k \geq 0$.

We construct from G an equivalent NPDA M with only one state that accepts by empty stack. Let

$$M = (\{q\}, \Sigma, N, \delta, q, S, \varnothing),$$

where

- q is the only state,
- Σ, the set of terminals of G, is the input alphabet of M,

- N, the set of nonterminals of G, is the stack alphabet of M,
- δ is the transition relation defined below,
- q is the start state,
- S, the start symbol of G, is the initial stack symbol of M,
- $\varnothing$, the null set, is the set of final states (actually, this is irrelevant, since M accepts by empty stack).

The transition relation δ is defined as follows. For each production

$$A \to cB_1B_2 \cdots B_k$$

in P, let δ contain the transition

$$((q, c, A), (q, B_1B_2 \cdots B_k)).$$

Thus δ has one transition for each production of G. Recall that for $c \in \Sigma$, this says, "When in state q scanning input symbol c with A on top of the stack, scan past the c, pop A off the stack, push $B_1B_2 \cdots B_k$ onto the stack (B_k first), and enter state q," and for $c = \epsilon$, "When in state q with A on top of the stack, without scanning an input symbol, pop A off the stack, push $B_1B_2 \cdots B_k$ onto the stack (B_k first), and enter state q."

That completes the description of M. Before we prove $L(M) = L(G)$, let's look at an example.

Consider the set of nonnull balanced strings of parentheses [ ]. Below we give a list of productions of a grammar in Greibach normal form for this set. Beside each production, we give the corresponding transition of the NPDA as specified by the construction above.

(i)	$S \to [BS$	$((q, [, S), (q, BS))$
(ii)	$S \to [B$	$((q, [, S), (q, B))$
(iii)	$S \to [SB$	$((q, [, S), (q, SB))$
(iv)	$S \to [SBS$	$((q, [, S), (q, SBS))$
(v)	$B \to]$	$((q,], B), (q, \epsilon))$

Recall that a *leftmost derivation* is one in which productions are always applied to the leftmost nonterminal in the sentential form. We will show that a leftmost derivation in G of a terminal string x corresponds exactly to an accepting computation of M on input x. The sequence of sentential forms in the leftmost derivation corresponds to the sequence of configurations of M in the computation.

For example, consider the input $x = [\,[\,[\,]\,]\,[\,]\,]$. In the middle column below is a sequence of sentential forms in a leftmost derivation of x in G. In the right column is the corresponding sequence of configurations of M. In the left column is the number of the production or transition applied.

Rule applied	*Sentential forms in a leftmost derivation of x in G*	*Configurations of M in an accepting computation of M on input x*
	S	$(q, [\,[\,[\,]\,]\,[\,]\,], S)$
(iii)	$[SB$	$(q, \;[\,[\,]\,]\,[\,]\,], SB)$
(iv)	$[\,[SBSB$	$(q, \;\;[\,]\,]\,[\,]\,], SBSB)$
(ii)	$[\,[\,[BBSB$	$(q, \;\;\;]\,]\,[\,]\,], BBSB)$
(v)	$[\,[\,[\,]BSB$	$(q, \;\;\;\;]\,[\,]\,], BSB)$
(v)	$[\,[\,[\,]\,]SB$	$(q, \;\;\;\;\;[\,]\,], SB)$
(ii)	$[\,[\,[\,]\,]\,[BB$	$(q, \;\;\;\;\;\;]\,], BB)$
(v)	$[\,[\,[\,]\,]\,[\,]B$	$(q, \;\;\;\;\;\;\;], B)$
(v)	$[\,[\,[\,]\,]\,[\,]\,]$	$(q, \;\;\;\;\;\;\;\epsilon, \epsilon)$

In the middle column, the first sentential form is the start symbol of G and the last sentential form is the terminal string x. In the right column the first configuration is the start configuration of M on input x and the last configuration is an accept configuration (the two ϵ's denote that the entire input has been read and the stack is empty).

One can see from this example the correspondence between the sentential forms and the configurations. In the sentential forms, the terminal string x is generated from left to right, one terminal in each step, just like the input string x in the automaton is scanned off from left to right, one symbol in each step. Thus the two strings of terminals appearing in each row always concatenate to give x. Moreover, the string of nonterminals in each sentential form is exactly the contents of the stack in the corresponding configuration of the PDA.

We can formalize this observation in a general lemma that relates the sentential forms in leftmost derivations of $x \in G$ and the configurations of M in accepting computations of M on input x. This lemma holds not just for the example above but in general.

Lemma 24.1 *For any $z, y \in \Sigma^*$, $\gamma \in N^*$, and $A \in N$, $A \xrightarrow[G]{n} z\gamma$ via a leftmost derivation if and only if $(q, zy, A) \xrightarrow[M]{n} (q, y, \gamma)$.*

For example, in the fourth row of the table above, we would have $z = [\,[\,[$, $y =]\,]\,[\,]\,]$, $\gamma = BBSB$, $A = S$, and $n = 3$.

Proof. The proof is by induction on n.

Basis

For $n = 0$, we have

$$\begin{aligned} A \xrightarrow[G]{0} z\gamma &\iff A = z\gamma \\ &\iff z = \epsilon \text{ and } \gamma = A \\ &\iff (q, zy, A) = (q, y, \gamma) \\ &\iff (q, zy, A) \xrightarrow[M]{0} (q, y, \gamma). \end{aligned}$$

Induction step

We do the two implications $\Rightarrow$ and $\Leftarrow$ separately for clarity.

First suppose $A \xrightarrow[G]{n+1} z\gamma$ via a leftmost derivation. Suppose that $B \to c\beta$ was the last production applied, where $c \in \Sigma \cup \{\epsilon\}$ and $\beta \in N^*$. Then

$$A \xrightarrow[G]{n} uB\alpha \xrightarrow[G]{1} uc\beta\alpha = z\gamma,$$

where $z = uc$ and $\gamma = \beta\alpha$. By the induction hypothesis,

$$(q, ucy, A) \xrightarrow[M]{n} (q, cy, B\alpha). \tag{24.1}$$

By the definition of M,

$$((q, c, B), (q, \beta)) \in \delta,$$

thus

$$(q, cy, B\alpha) \xrightarrow[M]{1} (q, y, \beta\alpha). \tag{24.2}$$

Combining (24.1) and (24.2), we have

$$(q, zy, A) = (q, ucy, A) \xrightarrow[M]{n+1} (q, y, \beta\alpha) = (q, y, \gamma).$$

Conversely, suppose

$$(q, zy, A) \xrightarrow[M]{n+1} (q, y, \gamma),$$

and let

$$((q, c, B), (q, \beta)) \in \delta$$

be the last transition taken. Then $z = uc$ for some $u \in \Sigma^*$, $\gamma = \beta\alpha$ for some $\alpha \in \Gamma^*$, and

$$(q, ucy, A) \xrightarrow[M]{n} (q, cy, B\alpha) \xrightarrow[M]{1} (q, y, \beta\alpha).$$

By the induction hypothesis,

$$A \xrightarrow[G]{n} uB\alpha$$

via a leftmost derivation in G, and by construction of M,

$$B \to c\beta$$

is a production of G. Applying this production to the sentential form $uB\alpha$, we get

$$A \xrightarrow[G]{n} uB\alpha \xrightarrow[G]{1} uc\beta\alpha = z\gamma$$

via a leftmost derivation. □

Theorem 24.2 $L(M) = L(G)$.

Proof.

$$\begin{aligned}
& x \in L(G) \\
\iff & S \xrightarrow[G]{*} x \text{ by a leftmost derivation} && \text{definition of } L(G) \\
\iff & (q, x, S) \xrightarrow[M]{*} (q, \epsilon, \epsilon) && \text{Lemma 24.1} \\
\iff & x \in L(M) && \text{definition of } L(M).
\end{aligned}$$

□