COL352 Lecture 19

Contents

1 Recap

Grammar $\implies$ PDA completed.

More precisely, we showed in the last to last lecture that L is recognized by a fast PDA $\iff$ L is recognized by a PDA, and in the last lecture that L is recognized by a fast PDA if L is generated by a grammar.

2 Definitions

Definition 1

A (non-deterministic) pushdown automaton ((N)PDA) is a 6-tuple $(Q, \Sigma, \Gamma, \Delta, q_0, A)$ where

1. Q – finite nonempty set of states
2. Σ – finite nonempty input alphabet
3. Γ – finite stack alphabet
4. $q_0 \in Q$ – initial state
5. A – set of accepting states
6. $\Delta \subseteq Q \times \Sigma_\epsilon \times \Gamma_\epsilon \times Q \times \Gamma_\epsilon$, where X_ϵ is defined as $X \cup \{\epsilon\}$

Note that in an NFA, $\Delta \subseteq Q \times (\Sigma \cup \{\epsilon\}) \times Q$.

Definition 2

Let $P = (Q, \Sigma, \Gamma, \Delta, q_0, A)$ be a PDA. An instantaneous description (i.d.) of P is a tuple (q, x, α) where $q \in Q$, $x \in \Sigma^*$, $\alpha \in \Gamma^*$. (The set of instantaneous descriptions is $Q \times \Sigma^* \times \Gamma^*$).

Definition 3

Let $P = (Q, \Sigma, \Gamma, \Delta, q_0, A)$ be a PDA. The relation $\vdash_P$ (read as "changes to") is defined on the set of i.d.s as follows:

If $(q, a, B, q', B') \in \Delta$, then $(q, ax, B\alpha) \vdash_P (q', x, B'\alpha)$, and no other pairs of i.d.s are related.

In other words:

$$(q, y, \beta) \vdash_P (q', y', \beta') \iff \exists a \in \Sigma_\epsilon, B \in \Gamma_\epsilon, \alpha \in \Gamma^*, B' \in \Gamma_\epsilon \text{ such that } y = ay', \beta = B\alpha, \beta' = B'\alpha, (q, a, B, q', B') \in \Delta$$

Definition 4

$\vdash_P^*$ is defined as the reflexive transitive closure of $\vdash$ (read as "changes to in finitely many steps").

Definition 5

$x \in \Sigma^*$ is said to be accepted by PDA $P = (Q, \Sigma^*, \Gamma, \Delta, q_0, A)$ iff

$$(q_0, x, \epsilon) \vdash_P^* (q, \epsilon, \alpha)$$

for some $q \in A$ and some $\alpha \in \Gamma^*$.

Definition 6

The language recognized by PDA P denoted by $\mathcal{L}(P)$ is $\{x \in \Sigma^* \mid P \text{ accepts } x\}$.

Definition 7

We define a simple PDA P to be a PDA such that

1. $\Delta = \Delta_{push} \uplus \Delta_{pop}$, where
   (a) Δ_{push} contains transitions (q, a, ϵ, q', B) where $q, q' \in Q, a \in \Sigma_\epsilon, B \in \Gamma$ (i.e., not allowed to pop, must push), and
   (b) Δ_{pop} contains transitions (q, a, B, q', ϵ) where $q, q' \in Q, a \in \Sigma_\epsilon, B \in \Gamma$ (i.e., must pop, not allowed to push).
2. $|A| = 1$, i.e., unique accepting state.
3. If x is accepted, then x is accepted with an empty stack, i.e., $(q_{init}, x, \epsilon) \vdash^* (q_{acc}, \epsilon, \alpha)$ for some $\alpha \in \Sigma^*$ iff $q_{init}, x, \epsilon) \vdash^* (q_{acc}, \epsilon, \epsilon)$.

3 Content

Now we need the following: If L is recognized by a PDA, L is generated by a grammar.

We shall define something called a simple PDA, and show that it is as powerful as a PDA.

Recall that PDA $P = (Q, \Sigma, \Gamma, \Delta, q_{init}, A)$ where $q_{init} \in Q, A \subseteq Q$, $\Delta \subseteq Q \times \Sigma_\epsilon \times \Gamma_\epsilon \times Q \times \Gamma_\epsilon$.
A simple PDA is defined as in the previous section.

Claim 0.1

L is recognized by a PDA iff L is recognized by a simple PDA.

Proof. $\Longleftarrow$ is trivial. We'll look at the other direction.

Suppose L is recognized by PDA $P = (Q, \Sigma, \Gamma, \Delta, q_{init}, A)$.

To ensure condition 2, add a new state q_{acc} to Q, make it the unique accepting state, and add transitions $(q, \epsilon, \epsilon, q_{acc}, \epsilon)$ for each $q \in A$.

To ensure condition 3, add transitions $(q_{acc}, \epsilon, B, q_{acc}, \epsilon)$ $\forall B \in \Gamma$.

To ensure condition 1, we need to break each transition that pushes as well as pops into two, i.e., replace the transition (q, a, B, q', C) where $a \in \Sigma_\epsilon, B, C \in \Gamma$ by (q, a, B, q'', ϵ) and $(q'', \epsilon, C, q', \epsilon)$, and replace $(q, a, \epsilon, q', \epsilon)$ with $(q, a, \epsilon, q'', \$)$ and $(q'', \epsilon, \$, q', \epsilon)$.

$P' = (Q \uplus \{q_{acc}\} \uplus \text{intermediate states}, \Sigma, \Gamma \uplus \{\$\}, q_{init}, \{q_{acc}\})$. □

Now our goal shall be the following.

Question 1

Given simple PDA $P = (Q, \Sigma, \Gamma, \Delta, q_{init}, \{q_{acc}\})$, $\Delta = \Delta_{push} \uplus \Delta_{pop}$, construct a grammar $G = (N, \Sigma, R, S)$ such that

$$\forall x \in \Sigma^* : ((q_{init}, x, \epsilon) \vdash^* (q_{acc}, \epsilon, \epsilon) \iff S \xrightarrow{*} x)$$

Note 1

Idea: $N = \{V_{qq'} \mid (q, q') \in Q \times Q\}$. R should ensure that $V_{qq'} \overset{*}{\Longrightarrow} x \iff (q, x, \epsilon) \vdash^* (q', \epsilon, \epsilon)$ (i.e., $\exists$ a run of P on x starting from q with empty stack and ending in q' with empty stack).

We also set S to $v_{q_{init} q_{acc}}$.

The derivation should be something like $(q, x, \epsilon) \vdash \_ \vdash \_ \cdots \vdash \_ \vdash (q', \epsilon, \epsilon)$.

Is there an intermediate instantaneous description in which the stack is empty? If there is, then in the first part of the run, we have read off some prefix of x, and in the second part of the run, we have read off the remaining suffix of x, say $x = x_1 x_2$. We hope that $V_{qr} \overset{*}{\Longrightarrow} x_1$ and $V_{rq'} \overset{*}{\Longrightarrow} x_2$.

We want $V_{qq'} \overset{*}{\Longrightarrow} x_1 x_2$, so we add the rule $V_{qq'} \to V_{qr} V_{rq'}$ to R, whihc will give us $V_{qq'} \Longrightarrow V_{qr} V_{rq'} \overset{*}{\Longrightarrow} x_1 x_2 = x$.

Now suppose the answer is no. Then the first transition must be a push transition and the last must be a pop transition, i.e., we go from (q, x, ϵ) to $(r, \_, B)$ where $(q, a, \epsilon, r, B) \in \Delta$ where $a = \epsilon$ or $x[1]$, and from $(r', \_, B')$ to (q', ϵ, ϵ) where (r', b, B', q, ϵ) where $b = \epsilon$ or $x[n]$. Note that the stack has never been empty, so $B' = B$ (look at it from the perspective of the evolution of the stack over time).