CIS 770: Formal Language Theory

Pavithra Prabhakar

Kansas State University

Spring 2015

From PDA to CFG

Proposition

For any PDA P, there is a CFG G such that $L(P) = L(G)$.

Proof Outline

1. For every PDA P there is a normalized PDA P_N such that $L(P) = L(P_N)$.
2. For every normalized PDA P_N there is a CFG G such that $L(P_N) = L(G)$.

Normalized PDAs

Definition

A PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, F)$ is normalized iff it satisfies the following conditions.

- Has exactly one accept state, i.e., $F = \{q_a\}$ for some $q_a \in Q$
- Empties its stack before accepting, i.e., if $\langle q_0, \epsilon \rangle \xrightarrow{w}_P \langle q_a, \sigma \rangle$ then $\sigma = \epsilon$.
- Each transition either pushes one symbol, or it pops one symbol. There are no transitions that both *push and pop*, nor transitions that leave the stack unaffected.

Normalizing a PDA

Proposition

For every PDA P, there is a normalized PDA P_N such that $L(P) = L(P_N)$

Proof Sketch

We will transform P is a series of steps, each time ensuring that the language does not change.

- We will ensure that there is only one accept state
- Next, we will ensure that all symbols are popped before accept state is reached.
- Finally, we will transform transitions to be either push or pop (not both or neither).

Normalizing a PDA

One accept state

To ensure one accept state, add ϵ-transitions (which do not change the stack) from old accept states to a new accept state.
Formally, given $P = (Q, \Sigma, \Gamma, \delta, q_0, F)$, let $P' = (Q', \Sigma, \Gamma, \delta', q_0, F')$ where

- $Q' = Q \cup \{q_a\}$, where $q_a \notin Q$
- $F' = \{q_a\}$
- $\delta'(q, x, a) = \delta(q, x, a)$ if $q \in Q \setminus F$ or $x \neq \epsilon$ or $a \neq \epsilon$, and $\delta'(q, \epsilon, \epsilon) = \delta(q, \epsilon, \epsilon) \cup \{(q_a, \epsilon)\}$ for $q \in F$, and $\delta'(q_a, x, a) = \emptyset$.

Normalizing a PDA

Emptying stack before acceptance

First push a new symbol \$ before starting computation, and from sole accept state, pop all symbols before popping \$ and moving to a new accept state.

i.e., given $P = (Q, \Sigma, \Gamma, \delta, q_0, \{q_a\})$, let $P' = (Q', \Sigma, \Gamma', \delta', q_0', F')$:

- $\Gamma' = \Gamma \cup \{\$\}$, where $\$ \notin \Gamma$
- $Q' = Q \cup \{q_i, q_p, q_f\}$ where $q_i, q_p, q_f \notin Q$
- $q_0' = q_i$
- $F' = \{q_f\}$
- $\delta'(q, x, a) = \delta(q, x, a)$ for $q \in Q \setminus \{q_a\}$ or $x \neq \epsilon$ or $a \neq \epsilon$. For q_a, we have $\delta'(q_a, \epsilon, \epsilon) = \delta(q_a, \epsilon, \epsilon) \cup \{(q_p, \epsilon)\}$. In addition, we have $\delta'(q_i, \epsilon, \epsilon) = \{(q_0, \$)\}$, and $\delta'(q_p, \epsilon, a) = \{(q_p, \epsilon)\}$ for $a \in \Gamma$, and $\delta'(q_p, \epsilon, \$) = (q_f, \epsilon)\}$. In all other cases δ' is $\emptyset$.

Normalizing a PDA

Only pushes or pops

There are two kinds of transitions that need fixing

- Transition of the form $q \xrightarrow{x,a \to b} q'$, where $a, b \in \Gamma$, i.e., those that push and pop in one step
  - Replace this by two steps, where you first pop a and then push b: $q \xrightarrow{x,a \to \epsilon} q'' \xrightarrow{\epsilon,\epsilon \to b} q'$. ($q''$ is a new state involved in only these transitions.)
- Transition of the form $q \xrightarrow{x,\epsilon \to \epsilon} q'$, i.e., those that neither push nor pop
  - Replace this by two steps, where first a dummy symbol is pushed, and then in the second step the dummy symbol is popped: $q \xrightarrow{x,\epsilon \to \partial} q'' \xrightarrow{\epsilon,\partial \to \epsilon} q'$. ($q''$ is a new state involved in only these transitions.)

(Formal definition skipped.)

CFGs for Normalized PDAs

Intuitions

- Let $P = (Q, \Sigma, \Gamma, \delta, q_0, \{q_a\})$ be a normalized PDA. $w \in L(P)$ iff $\langle q_0, \epsilon \rangle \xrightarrow{w}_P \langle q_a, \epsilon \rangle$.
- If, for every $p, q \in Q$, we can describe $L_{p,q} = \{w \mid \langle p, \epsilon \rangle \xrightarrow{w}_P \langle q, \epsilon \rangle\}$ using a CFG, then we are done because $L(P)$ is nothing but L_{q_0,q_a}.
- So CFG will have variables $A_{p,q}$ such that $A_{p,q} \stackrel{*}{\Rightarrow} w$ iff $w \in L_{p,q}$.
- What are the rules for $A_{p,q}$?

Rules for the grammar

Consider $w \in L_{p,q}$, and a computation corresponding to $\langle p, \epsilon \rangle \xrightarrow{w}_P \langle q, \epsilon \rangle$. Since the computation starts with empty stack, the first step must be a push, and last step must be a pop, since we end with empty stack.

- Case I: The first symbol pushed is popped only at the end. So we have $\langle p, \epsilon \rangle \xrightarrow{a} \langle r, A \rangle \xrightarrow{u} \langle s, A \rangle \xrightarrow{b} \langle q, \epsilon \rangle$, with $w = aub$. And $u \in L_{r,s}$. Can be captured by rule $A_{p,q} \rightarrow aA_{r,s}b$.
- Case II: First symbol pushed is popped in the middle of computation (and then stack is empty). So we have $\langle p, \epsilon \rangle \xrightarrow{u_1} \langle r, \epsilon \rangle \xrightarrow{u_2} \langle q, \epsilon \rangle$. Can be captured by rule $A_{p,q} \rightarrow A_{p,r}A_{r,q}$

Formal Construction

Let $P = (Q, \Sigma, \Gamma, \delta, q_0, \{q_a\})$ be a normalized PDA. Define $G_P = (V, \Sigma, R, S)$ where

- $V = \{A_{p,q} \mid p, q \in Q\}$,
- $S = A_{q_0,q_a}$
- And the rules in R are
  - For every $p \in Q$, $A_{p,p} \to \epsilon$
  - For every $p, q, r \in Q$, $A_{p,q} \to A_{p,r}A_{r,q}$
  - For every $p, q, r, s \in Q$, $\gamma \in \Gamma$, $a, b \in \Sigma \cup \{\epsilon\}$, if $(r, \gamma) \in \delta(p, a, \epsilon)$ and $(q, \epsilon) \in \delta(r, b, \gamma)$ then $A_{p,q} \to aA_{r,s}b$

Correctness of Construction

Proposition

Let P be a normalized PDA and let G_P be the corresponding CFG. Then $A_{p,q} \stackrel{*}{\Rightarrow} w$ iff $\langle p, \epsilon \rangle \xrightarrow{w}_P \langle q, \epsilon \rangle$.

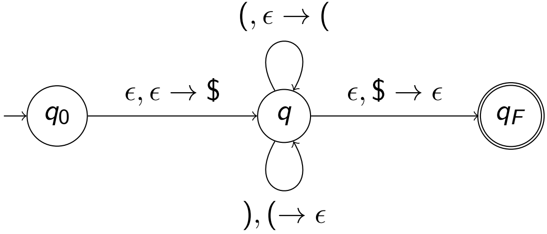
Proof.

The two directions are proved as follows

- $\Rightarrow$ By induction on the number of steps in the derivation.
- $\Leftarrow$ By induction on the number of steps in the computation.

$\square$

Example



- $A_{q_0,q_0} \to \epsilon$, $A_{q,q} \to \epsilon$, $A_{q_F,q_F} \to \epsilon$
- $A_{q_0,q_F} \to A_{q_0,q}A_{q,q_F}$, $A_{q_0,q} \to A_{q_0,q}A_{q,q}$, $A_{q,q_F} \to A_{q,q}A_{q,q_F}$, $A_{q,q} \to A_{q,q}A_{q,q}$ (We can write all other rules, however, they are not important for this particular example)
- $A_{q_0,q_F} \to A_{q,q}$, $A_{q,q} \to (A_{q,q})$

Tying all the Ends

Proposition

Let P be a PDA then $L(P)$ is context-free.

Proof.

- A normalized PDA P_N can be constructed such that $L(P) = L(P_N)$
- A grammar G_P can be constructed such that $L(G_P) = L(P_N) = L(P)$. This is because
  - $S = A_{q_0,q_a} \stackrel{*}{\Rightarrow} w$ iff $\langle q_0, \epsilon \rangle \xrightarrow{w}_{P_N} \langle q_a, \epsilon \rangle$ (by previous proposition) iff $w \in L(P_N) = L(P)$

$\square$