Introduction to the Theory of Computation

XU Ming

School of Software Engineering, East China Normal University

May 23, 2024

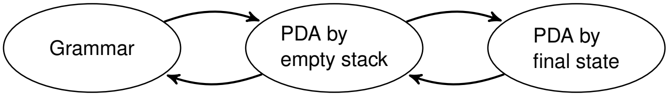
OUTLINE

- Equivalence between CFG's and PDA's
- Deterministic Pushdown Automata

We'll prove that a language is generated by a CFG

- if and only if it is accepted by a PDA upon empty stack,
- if and only if it is accepted by a PDA upon final state.

We already know how to go between empty stack and final state.



From CFG's to PDA's

From CFG's to PDA's

Given a CFG G, we construct a PDA that simulates the leftmost derivations $\overset{*}{\underset{lm}{\Longrightarrow}}$.

We write left-sentential forms as $xA\alpha$, in which:

- A is the leftmost variable,
- x is a string of terminals,
- and α is a string of terminals and variables.

The *tail* of $xA\alpha$ is $A\alpha$.

For instance,

$$\underbrace{(a+}_{x}\underbrace{\underbrace{E}_{A}\underbrace{)}_{\alpha}}_{\text{tail}}$$

Let $xA\alpha \underset{lm}{\Longrightarrow} x\beta\alpha$. This corresponds to the PDA first having consumed x and having $A\alpha$ on the stack, and then on ϵ it pops A and pushes β.

That is, letting $w = xy$, the PDA goes non-deterministically from configuration $(q, y, A\alpha)$ to $(q, y, \beta\alpha)$.

Formally, let $G = (V, T, R, S)$ be a CFG. Define $P_G = (\{q\}, T, V \cup T, \delta, q, S)$, where

- $\delta(q, \epsilon, A) = \{(q, \beta) \mid A \to \beta \in R\}$ for $A \in V$,
- $\delta(q, a, a) = \{(q, \epsilon)\}$ for $a \in T$.

Theorem 6.5

If PDA P_G is constructed from CFG G by the construction above, then $N(P_G) = L(G)$.

Proof ($\supseteq$-direction). Let $w \in L(G)$. Then

$$S = \gamma_1 \underset{lm}{\Longrightarrow} \gamma_2 \underset{lm}{\Longrightarrow} \cdots \underset{lm}{\Longrightarrow} \gamma_n = w.$$

Let $\gamma_i = x_i\alpha_i$. We show by induction on i that if $S \overset{*}{\underset{lm}{\Longrightarrow}} \gamma_i$, then $(q, w, S) \overset{*}{\vdash} (q, y_i, \alpha_i)$, where $w = x_i y_i$.

Since $\gamma_n = w$, we have $\alpha_n = y_n = \epsilon$ and $x_n = w$, thus $(q, w, S) \overset{*}{\underset{lm}{\vdash}} (q, \epsilon, \epsilon)$, i.e. $w \in N(P_G)$.

For instance, consider CFG G:

$$E \to I \mid E + E \mid E \times E \mid (E), \quad I \to a \mid b \mid Ia \mid Ib \mid I0 \mid I1.$$

Since $a \times b \in L(G)$

$$\underset{\gamma_1}{E} \Rightarrow \underset{\gamma_2}{E \times E} \Rightarrow \underset{\gamma_3}{I \times E} \Rightarrow \underset{\gamma_4}{a \times E} \Rightarrow \underset{\gamma_5}{a \times I} \Rightarrow \underset{\gamma_6}{a \times b}$$

then

$$\begin{array}{ll}
(q, a \times b, E) & \gamma_1 \\
\vdash (q, a \times b, E \times E) & \gamma_2 \\
\vdash (q, a \times b, I \times E) & \gamma_3 \\
\vdash (q, a \times b, a \times E) \vdash (q, \times b, \times E) \vdash (q, b, E) & \gamma_4 \\
\vdash (q, b, I) & \gamma_5 \\
\vdash (q, b, b) \vdash (q, \epsilon, \epsilon) & \gamma_6
\end{array}$$

We continue the proof.

Basis step: For $i = 1, \gamma_1 = S$. Thus $x_1 = \epsilon$, and $y = w$. Clearly, $(q, w, S) \overset{*}{\vdash} (q, w, S)$.

Inductive step: The induction hypothesis is $(q, w, S) \overset{*}{\vdash} (q, y_i, \alpha_i)$. We have to show that

$$(q, y_i, \alpha_i) \overset{*}{\vdash} (q, y_{i+1}, \alpha_{i+1}).$$

Now α_i begins with a variable A, and we have the form (e.g. suppose $\beta = vB\mu$)

$$\underbrace{x_i A \chi}_{\gamma_i} \underset{lm}{\Longrightarrow} x_i \beta \chi = \underbrace{x_{i+1} B \mu \chi}_{\gamma_{i+1}}$$

By the induction hypothesis $A\chi$ is on the stack, y_i is unconsumed. From the construction of P_G, it follows that we can make the move

$$(q, y_i, A\chi) \vdash (q, y_i, \beta\chi) = (q, y_i, vB\mu\chi).$$

If β has a prefix of terminals (in our example, only has v), we can pop them with matching terminals in a prefix of y_i, ending up in configuration $(q, y_{i+1}, \alpha_{i+1})$, where $\alpha_{i+1} = B\mu\chi$ is the tail of the sentential γ_{i+1}.

($\subseteq$-direction). We shall show by an induction on the length of $\overset{*}{\vdash}$ that

$$\text{If } (q, x, A) \overset{*}{\vdash} (q, \epsilon, \epsilon), \text{then } A \overset{*}{\Rightarrow} x.$$

Suppose $w \in N(P_G)$, choose $A = S$, and $x = w$, then $(q, w, S) \overset{*}{\vdash} (q, \epsilon, \epsilon)$. We have to show $S \overset{*}{\Rightarrow} w$, meaning $w \in L(G)$.

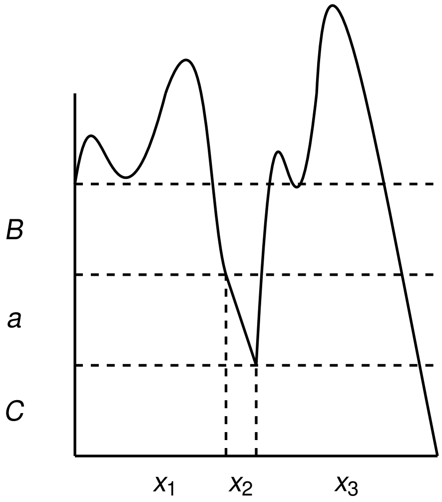
Basis step: Length is 1. Then it must be that $A \to \epsilon$ is a production in G, and we have $(q, \epsilon) \in \delta(q, \epsilon, A)$. In this case, $x = \epsilon$, and we know that $A \overset{*}{\Rightarrow} \epsilon$.

Inductive step: Length is $n > 1$, and the induction hypothesis holds for length $< n$. Since A is a variable, we must have

$$(q, x, A) \vdash (q, x, Y_1 Y_2 \cdots Y_k) \vdash \cdots \vdash (q, \epsilon, \epsilon)$$

where $A \rightarrow Y_1 Y_2 \cdots Y_k$ is a production in G.

We can now write x as $x_1 x_2 \cdots x_k$, analogous to the figure in next slide, where $Y_1 = B$, $Y_2 = a$ and $Y_3 = C$.


B
a
C
x1
x2
x3

Now we can conclude that

$$(q, x_i x_{i+1} \cdots x_k, Y_i) \overset{*}{\vdash} (q, x_{i+1} \cdots x_k, \epsilon)$$

has less than n steps, for each $i = 1, 2, \cdots, k$.

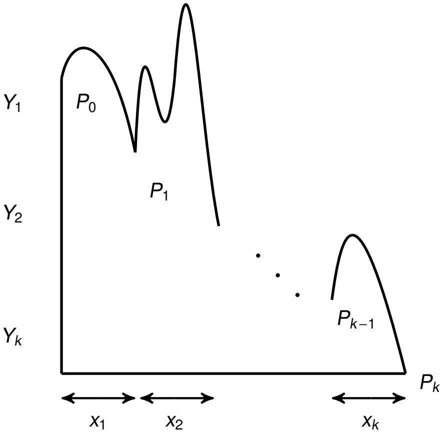
- If Y_i is a variable we have by the induction hypothesis and Theorem 6.2 that $Y_i \overset{*}{\Rightarrow} x_i$.
- If Y_i is a terminal, we have $|x_i| = 1$, and $Y_i = x_i$. Thus $Y_i \overset{*}{\Rightarrow} x_i$ by the reflexivity of $\overset{*}{\Rightarrow}$.

So, $A \Rightarrow Y_1 Y_2 \cdots Y_k \overset{*}{\Rightarrow} x_1 Y_2 \cdots Y_k \overset{*}{\Rightarrow} x_1 x_2 \cdots Y_k \overset{*}{\Rightarrow} x_1 x_2 \cdots x_k = x$. □

From PDA's to CFG's

From PDA's to CFG's

Let's first look at how a PDA can consume $x = x_1 x_2 \cdots x_k$ and empty the stack.



We shall define a grammar with variables of the form $[p_{i-1}Y_ip_i]$ representing going from p_{i-1} to p_i with net effect of popping Y_i.

Formally, let $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$ be a PDA. Define $G_P = (V, \Sigma, R, S)$, where

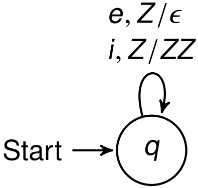
- $V = \{[pXq] \mid p, q \in Q, X \in \Gamma\} \cup \{S\}$.
- $R = \{S \to [q_0Z_0p] \mid p \in Q\} \cup$

$$\begin{aligned}\{[qXr_k] \to a[rY_1r_1]\cdots[r_{k-1}Y_kr_k] \mid\ & a \in \Sigma \cup \{\epsilon\} \wedge \\ & r_1, \ldots, r_k \in Q \wedge \\ & (r, Y_1 \cdots Y_k) \in \delta(q, a, X)\}.\end{aligned}$$

Note that k can be 0, in which case $(r, \epsilon) \in \delta(q, a, X)$. Then production is $[qXr] \to a$.

Example

Let's convert $P = (\{q\}, \{i, e\}, \{Z\}, \delta, q, Z)$, where $\delta(q, i, Z) = \{(q, ZZ)\}$ and $\delta(q, e, Z) = \{(q, \epsilon)\}$, to a grammar.



By the construction above, we get $G_P = (V, \{i, e\}, R, S)$ where $V = \{[qZq], S\}$, and $R = \{S \rightarrow [qZq], [qZq] \rightarrow i[qZq][qZq], [qZq] \rightarrow e\}$.

Replacing $[qZq]$ by A, we get productions $S \rightarrow A$ and $A \rightarrow iAA \,|\, e$, or more directly $S \rightarrow iSS \,|\, e$.

Example

Let $P = (\{p, q\}, \{0, 1\}, \{X, Z_0\}, \delta, q, Z_0)$, where δ is given by

$$
\begin{aligned}
&1.\ \delta(q, 1, Z_0) = \{(q, XZ_0)\} \\
&2.\ \delta(q, 1, X) = \{(q, XX)\} \\
&3.\ \delta(q, 0, X) = \{(p, X)\} \\
&4.\ \delta(q, \epsilon, Z_0) = \{(q, \epsilon)\} \\
&5.\ \delta(p, 1, X) = \{(p, \epsilon)\} \\
&6.\ \delta(p, 0, Z_0) = \{(q, Z_0)\}
\end{aligned}
$$

Construct the CFG G_P.

Question What's the language $N(P)$?

We get $G_P = (V, \{0,1\}, R, S)$ where

$$V = \{S, [pXp], [pXq], [pZ_0p], [pZ_0q], [qXp], [qXq], [qZ_0p], [qZ_0q]\}$$

and the productions in R are

$$S \to [qZ_0q] \mid [qZ_0p]$$

and the following: From Rule (1) that $\delta(q, 1, Z_0) = \{(q, XZ_0)\}$, we have

$$[qZ_0q] \to 1[qXq][qZ_0q] \mid 1[qXp][pZ_0q]$$
$$[qZ_0p] \to 1[qXq][qZ_0p] \mid 1[qXp][pZ_0p]$$

From Rule (2) that $\delta(q, 1, X) = \{(q, XX)\}$, we have

$$[qXq] \to 1[qXq][qXq] \mid 1[qXp][pXq]$$
$$[qXp] \to 1[qXq][qXp] \mid 1[qXp][pXp]$$

From Rule (3) that $\delta(q, 0, X) = \{(p, X)\}$, we have

$$[qXq] \rightarrow 0[pXq]$$
$$[qXp] \rightarrow 0[pXp]$$

From Rule (4) that $\delta(q, \epsilon, Z_0) = \{(q, \epsilon)\}$, we have

$$[qZ_0q] \rightarrow \epsilon$$

From Rule (5) that $\delta(p, 1, X) = \{(p, \epsilon)\}$, we have

$$[pXp] \rightarrow 1$$

From Rule (6) that $\delta(p, 0, Z_0) = \{(q, Z_0)\}$, we have

$$[pZ_0q] \rightarrow 0[qZ_0q]$$
$$[pZ_0p] \rightarrow 0[qZ_0p]$$

Theorem 6.6

If CFG G_P is constructed from PDA P by the construction above, then $L(G_P) = N(P)$.

Proof ($\supseteq$-direction). We shall show by an induction on the length of the sequence $\overset{*}{\vdash}$ that

$$\text{If } (q, w, X) \overset{*}{\vdash} (p, \epsilon, \epsilon), \text{ then } [qXp] \overset{*}{\Rightarrow} w.$$

Suppose $w \in N(P)$, then $(q_0, w, Z_0) \overset{*}{\vdash} (p, \epsilon, \epsilon)$ for some p. We have $[q_0Z_0p] \overset{*}{\Rightarrow} w$, i.e. $S \overset{*}{\Rightarrow} w$, because the rules for start symbol S are constructed. That means $w \in L(G_P)$.

Basis step: Length is 1. Then w is an a or ϵ, and $(p, \epsilon) \in \delta(q, w, X)$. By the construction of G_P we have $[qXp] \to w$ and thus $[qXp] \overset{*}{\Rightarrow} w$.

Inductive step: Length is $n > 1$, and the induction hypothesis holds for lengths $< n$. We must have

$$(q, w, X) \vdash (r_0, x, Y_1 Y_2 \cdots Y_k) \vdash \cdots \vdash (p, \epsilon, \epsilon)$$

where $w = ax$ (a is either a symbol in Σ or $a = \epsilon$). It follows that $(r_0, Y_1 Y_2 \cdots Y_k) \in \delta(q, a, X)$. Then we have a production

$$[qXr_k] \to a[r_0 Y_1 r_1] \cdots [r_{k-1} Y_k r_k],$$

for any $r_1, \ldots, r_k \in Q$.

We may now choose r_i to be the state in the sequence $\overset{*}{\vdash}$ when Y_i is popped, finally, $r_k = p$ when Y_k is popped. Let $x = w_1 w_2 \cdots w_k$, where w_i is consumed while Y_i is popped. Then

$$(r_{i-1}, w_i, Y_i) \overset{*}{\vdash} (r_i, \epsilon, \epsilon).$$

By the induction hypothesis we get $[r_{i-1} Y_i r_i] \overset{*}{\Rightarrow} w_i$.
We then get the following derivation sequence:

$$\begin{aligned}
[qXr_k] &\Rightarrow a[r_0 Y_1 r_1] \cdots [r_{k-1} Y_k r_k] \\
&\overset{*}{\Rightarrow} aw_1 [r_1 Y_2 r_2][r_2 Y_3 r_3] \cdots [r_{k-1} Y_k r_k] \\
&\overset{*}{\Rightarrow} aw_1 w_2 [r_2 Y_3 r_3] \cdots [r_{k-1} Y_k r_k] \\
&\overset{*}{\Rightarrow} \cdots \\
&\overset{*}{\Rightarrow} aw_1 w_2 \cdots w_k = w
\end{aligned}$$

where $r_k = p$.

($\subseteq$-direction). We shall show by an induction on the length of the derivation $\overset{*}{\Rightarrow}$ that

$$\text{If } [qXp] \overset{*}{\Rightarrow} w, \text{ then } (q, w, X) \overset{*}{\vdash} (p, \epsilon, \epsilon).$$

Suppose $w \in L(G_P)$, then $S \overset{*}{\Rightarrow} w$. There is a state p such that $[q_0 Z_0 p] \overset{*}{\Rightarrow} w$, because we have only productions $S \rightarrow [q_0 Z_0 p]$ for the start symbol S.

Now we have $(q_0, w, Z_0) \overset{*}{\vdash} (p, \epsilon, \epsilon)$. That means $w \in N(P)$.

Basis step: One step. Then we have a production $[qXp] \rightarrow w$. From the construction of G_P it follows that $(p, \epsilon) \in \delta(q, a, X)$, where $w = a$. But then $(q, w, X) \vdash (p, \epsilon, \epsilon)$.

Inductive step: Length is $n > 1$, and the induction hypothesis holds for lengths $< n$. We must have

$$[qXr_k] \Rightarrow a[r_0 Y_1 r_1][r_1 Y_2 r_2] \cdots [r_{k-1} Y_k r_k] \overset{*}{\Rightarrow} w$$

where $r_k = p$.

We can break w into $aw_1 \cdots w_k$ such that $[r_{i-1} Y_i r_i] \overset{*}{\Rightarrow} w_i$. From the induction hypothesis we get $(r_{i-1}, w_i, Y_i) \overset{*}{\vdash} (r_i, \epsilon, \epsilon)$.

From Theorem 6.1 we get

$$(r_{i-1}, w_i w_{i+1} \cdots w_k, Y_i Y_{i+1} \cdots Y_k) \overset{*}{\vdash} (r_i, w_{i+1} \cdots w_k, Y_{i+1} \cdots Y_k).$$

Since this holds for all $i = 1, 2, \cdots, k$, we put all these sequences together and get

$$\begin{aligned}(q, aw_1 w_2 \cdots w_k, X) &\vdash (r_0, w_1 w_2 \cdots w_k, Y_1 Y_2 \cdots Y_k)\\ &\overset{*}{\vdash} (r_1, w_2 \cdots w_k, Y_2 \cdots Y_k)\\ &\overset{*}{\vdash} (r_2, w_3 \cdots w_k, Y_3 \cdots Y_k)\\ &\overset{*}{\vdash} \cdots\\ &\overset{*}{\vdash} (r_k, \epsilon, \epsilon).\end{aligned}$$

Since $r_k = p$, we have shown that $(q, w, X) \overset{*}{\vdash} (p, \epsilon, \epsilon)$. □

Deterministic Pushdown Automata

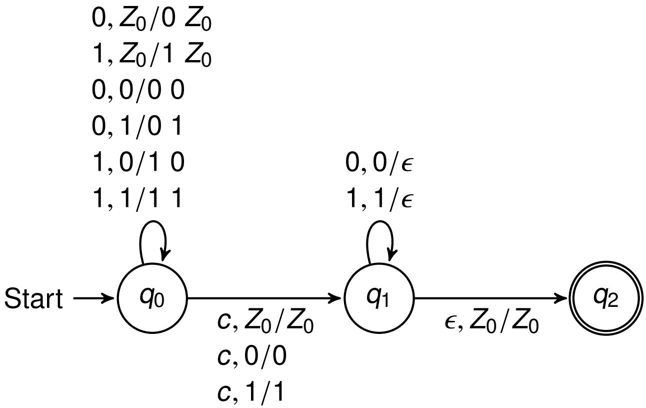
A PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ is deterministic (DPDA) if

1. $\delta(q, a, X)$ is always empty or a singleton for any $q \in Q$, $a \in \Sigma$ or $a = \epsilon$, and $X \in \Gamma$.
2. Given a pair of $q \in Q$ and $X \in \Gamma$, if $\delta(q, a, X)$ is nonempty for some $a \in \Sigma$, then $\delta(q, \epsilon, X)$ must be empty.

Example

Let's define $L_{wcwr} = \{wcw^R \mid w \in \{0, 1\}^*\}$. This language can be recognized by a DPDA.

The DPDA for L_{wcwr} is shown as follows.



DPDA's and Regular Languages as Well as CFL's

We'll show that

$$\text{Regular Languages} \subset L(\text{DPDA}) \subset \text{Context-free Languages}$$

Theorem 6.7

If L is regular language, then $L = L(P)$ for some DPDA P.

Proof Since L is regular there is a DFA A s.t. $L = L(A)$. Let $A = (Q, \Sigma, \delta_A, q_0, F)$. We define the DPDA $P = (Q, \Sigma, \{Z_0\}, \delta_P, q_0, Z_0, F)$, where the transition function $\delta_P(q, a, Z_0) = \{(\delta_A(q, a), Z_0)\}$, for all $q \in Q$, and $a \in \Sigma$.

We shall show by an induction on the length of w that

$$(q, w, Z_0) \overset{*}{\vdash} (p, \epsilon, Z_0) \quad \text{iff} \quad \hat{\delta}_A(q, w) = p$$

Choose $q = q_0$, since both A and P accept by entering one of the states of F, we conclude their languages are the same.

We only give a proof for "only if" part.

Basis step: Let $w = \epsilon$. From $(q, \epsilon, Z_0) \vdash (p, \epsilon, Z_0)$, we know $(p, Z_0) \in \delta_P(q, \epsilon, Z_0)$. Since $\delta_P(q, \epsilon, Z_0)$ is a singleton, we must have $p = \delta_A(q, \epsilon) = q$.

Inductive step: Let $w = ax$. We have

$$(q, w, Z_0) = (q, ax, Z_0) \vdash (\delta_A(q, a), x, Z_0) \overset{*}{\vdash} (p, \epsilon, Z_0).$$

By the induction hypothesis, we get $\hat{\delta}_A(\delta_A(q, a), x) = p$.

Notice that we mentioned the formula for the $\hat{\delta}$:

$$\hat{\delta}(q, ax) = \hat{\delta}(\delta(q, a), x)$$

for any state q, string x, and input symbol a.

We conclude that $\hat{\delta}_A(q, w) = p$. □

- We have shown that the DPDA languages include all the regular languages.
- We have already seen that a DPDA can accept language like L_{wcwr} that is not regular.
- Are there CFL's that can not be accepted by any DPDA?

Yes, for example L_{wwr}! But a formal proof is complex.

The two modes of acceptance – final state and empty stack – are not same for DPDA's. What about DPDA's that accept by empty stack?

A language L has the prefix property if there are no two distinct strings in L, such that one is a prefix of the other.

Example

L_{wcwr} has the prefix property. But $\{0\}^*$ does not have the prefix property.

Theorem 6.8

L is $N(P)$ for some DPDA P if and only if L has the prefix property and L is $L(P')$ for some DPDA P'.

That is, the languages accepted by empty stack are exactly those of the languages accepted by final state that have the prefix property. We'll show this in three parts.

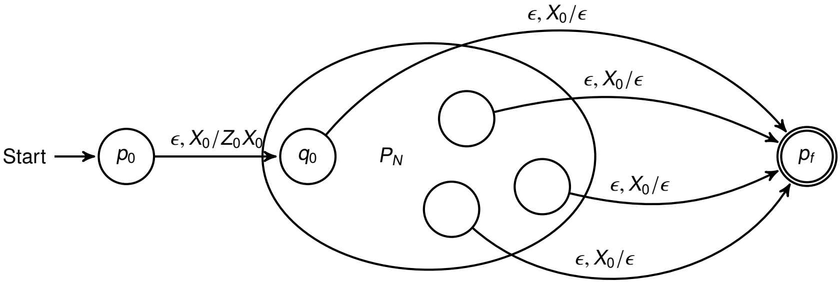
- If $L = N(P)$ for some DPDA P, then L has the prefix property.

Proof   Suppose L does not have the prefix property, i.e. P accepts both w and wx by empty stack, where $x \neq \epsilon$. Then $(q_0, w, Z_0) \overset{*}{\vdash} (q, \epsilon, \epsilon)$ for some state q, where q_0 and Z_0 are the start state and symbol of P. It does so by a unique sequence of moves because P is deterministic. Thus $(q_0, wx, Z_0) \overset{*}{\vdash} (q, x, \epsilon)$.

However, it is not possible that $(q, x, \epsilon) \overset{*}{\vdash} (p, \epsilon, \epsilon)$ for some state p, because we know x is not ϵ, and a PDA cannot have a move with an empty stack. This observation contradicts the assumption that wx is in $N(P)$.

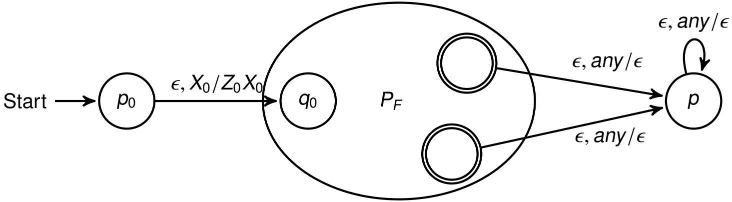
- If $L = N(P)$ for some DPDA P, then there exists a DPDA P' such that $L = L(P')$.

Proof We can convert P to P' just as P_N to P_F.



- If L has the prefix property and is $L(P')$ for some DPDA P', then there exists a DPDA P such that $L = N(P)$.

Proof Converting P' to P just as P_F to P_N, we find that P is not deterministic unless $L(P')$ has the prefix property.



DPDA's and Ambiguous Grammars

We can refine the power of the DPDA's by noting the languages they accept, all of which have unambiguous grammars. Unfortunately, the DPDA languages are not exactly equal to the subset of the CFL's that are not inherently ambiguous.

For instance, L_{wwr} has an unambiguous grammar

$$S \to 0S0 \mid 1S1 \mid \epsilon$$

even though it is not a DPDA language.

For the converse, we have

Theorem 6.9

If $L = N(P)$ for some DPDA P then L has an unambiguous CFG.

Proof By inspecting the proof of Theorem 6.6 we see that if the construction is applied to a DPDA the result is a CFG with unique leftmost derivations. □

This theorem can be strengthened as follows, but we omit the proof.

Theorem 6.10

If $L = L(P)$ for some DPDA P then L has an unambiguous CFG.

Homework

Exercises 6.3.2, 6.3.7