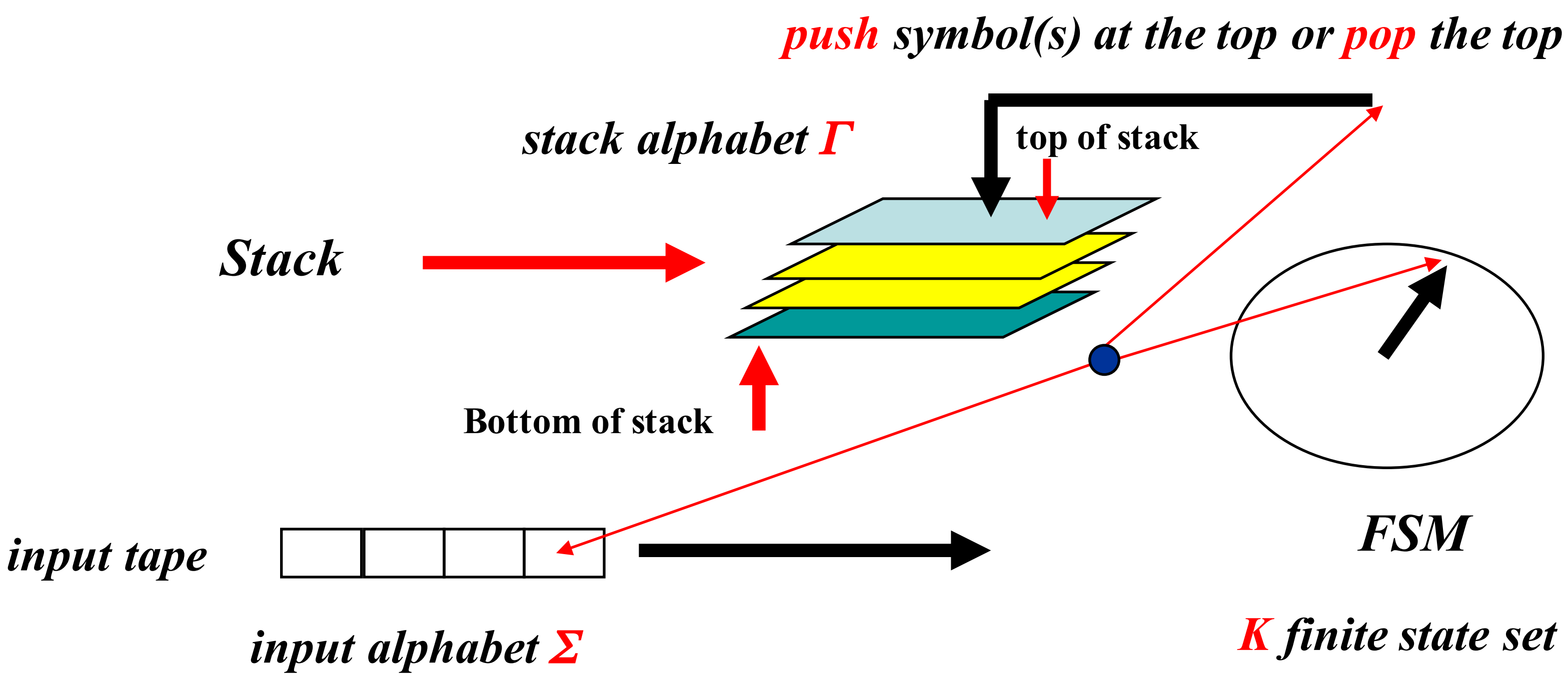
Pushdown Automata



$$\delta : Q \times (\Sigma \cup e) \times \Gamma \rightarrow 2^{(K \times \Gamma^*)}$$

$$(q, \sigma, X) \rightarrow ((p_1, \gamma_1), \ldots, (p_k, \gamma_k))$$

$$p_i \in Q, \gamma_i \in \Gamma^*$$

Formal Definition of Pushdown Automaton (PDA)

$P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

Q = states of the FSM

Σ = input alphabet set

Γ = stack alphabet set

$\delta : (Q \times (\Sigma \cup e) \times \Gamma) \rightarrow 2^{(Q \times \Gamma^*)}$ = transition function

q_0 = initial state

Z_0 = initial bottom of stack in Γ

F = final state set , $F \subseteq Q$

Interpretation of the PDA transition notation

Two notations for transitions

$(q', \gamma') \in \delta(q, a, X) = \{(q_1, \gamma_1), \ldots, (q_p, \gamma_p)\}$

or

$(q, a, X) \rightarrow (q', \gamma')$

(q, e, X) *means that the result is* ***independent*** *of the current input symbol in the list and that input is* ***not consumed*** *for the current transition*

Note that (q, a, e) *is* ***NOT*** *defined since domain of* δ *is* $(Q \times (\Sigma \cup e) \times \Gamma)$ *and* $e \notin \Gamma$

$(q, a, X) \rightarrow (q', \gamma')$ *means that the symbol* ***X*** *at the top of the stack is removed and replaced by the sequence* γ' *of stack symbols ; if* $\gamma' = e$ *then* ***X*** *is said to be* ***'pop'ped***

Instantaneous Description (ID) of a PDA

left to right

$(q, v, \beta) \in Q \times \Sigma^* \times \Gamma^*$

$q \in Q$ *(current state)*, $v \in \Sigma^*$ *(rest of the (unconsumed) list of the inputs)*,

$\beta \in \Gamma^*$ *(current stack contents)* ← ***top at left***

P accepts *input* $w \in \Sigma^*$ ***in the L(P) sense*** *(or by **final state**) iff*

$(q_0, w, Z_0) \vdash^* (f, e, \gamma)$, *where* $f \in F$, e = *empty string*, $\gamma \in \Gamma^*$

P accepts *input* $w \in \Sigma^*$ ***in the N(P) sense*** *(or by **empty stack**) iff*

$(q_0, w, Z_0) \vdash^* (q, e, e)$, *where* e = *empty string*

Acceptance by final state :

$$L(P) := \{ w \in T^* \mid (q_0, w, Z_0) \vdash^* (f, e, \gamma), f \in F\}$$

Conversion is simple !!

L to N **: *Whenever any final state f is entered empty the stack by continuously popping the stack at f !***

N to L **: *Whenever the stack is empty move to a final state !***

(in this case initially put an extra stack symbol say Z_{00} in N and replace all Z_0's in N by Z_{00} ; then when the stack is emptied in N, in L the top of the stack is Z_0 ; then move into a newly defined final state f)

Acceptance by empty stack :

$$N(P) := \{ w \in T^* \mid (q_0, w, Z_0) \vdash^* (q, e, e), q \in Q\}$$

*Examples : **PDAs** that accept the languages (i) **wcw^R** and (ii) **ww^R** ; $w \in \{a,b\}^*$*

$Q=\{q_0, q, f\}$, $\Sigma = \{a,b,c\}$, $\Gamma=\{Z_0, a,b,c\}$

*(i) Transitions (**X** = generic variable)*

$(q_0, a, X) \rightarrow (q_0, aX)$

$(q_0, b, X) \rightarrow (q_0, bX)$

$(q_0, c, X) \rightarrow (q, X)$

$(q, a, a) \rightarrow (q, e)$

$(q, b, b) \rightarrow (q, e)$

$(q, e, Z_0) \rightarrow (f, Z_0)$

*(ii) Transitions (**X** = generic variable)*

$(q_0, a, X) \rightarrow (q_0, aX)$

$(q_0, b, X) \rightarrow (q_0, bX)$

$(q_0, e, X) \rightarrow (q, X)$

$(q, a, a) \rightarrow (q, e)$

$(q, b, b) \rightarrow (q, e)$

$(q, e, Z_0) \rightarrow (f, Z_0)$

Example (iii): Also an example for L(P) and N(P) acceptance

PDA *accepts the language* $(w \in \{a,b\}^* \mid \#'a's = \#'b's \text{ in } w)$

Common transitions for L(P) and N(P)

$$(q_0, a, Z_0) \rightarrow (q_0, aZ_0)$$

$$(q_0, b, Z_0) \rightarrow (q_0, bZ_0)$$

$$(q_0, a, a) \rightarrow (q_0, aa)$$

$$(q_0, b, b) \rightarrow (q_0, bb)$$

$$(q_0, a, b) \rightarrow (q_0, e)$$

$$(q_0, b, a) \rightarrow (q_0, e)$$

L(P)

$$(q_0, e, Z_0) \rightarrow (f, Z_0)$$

N(P)

$$(q_0, e, Z_0) \rightarrow (q_0, e)$$

Equivalence of CFGs and PDAs

Theorem

A language is generated by a CFG

if and only if

it is accepted by a PDA

Theorem 1 (only if)

For every language L_G *where* **G** *is a* **CFG**

there exists a **PDA** *that accepts it*

Theorem 1 *(restated) Given a CFG,* $G = (V,T,P,S)$ *there*

exists a PDA, $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ *such that*

$w \in L_G$ *if and only if* $w \in L_P$

The PDA $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ *used in the proof of* **Theorem 1**

$G = (V, T, P, S)$ *is the given CFG*

$Q = \{q_0, q, f\}$ $\quad \Sigma = T$ $\quad \Gamma = V \cup T \cup \{Z_0\}$ $\quad F = \{f\}$

(1) *Starting transition*

$\delta(q_0, e, Z_0) = (q, SZ_0)$

(2) *For each* $A \in V$ *of* G

$\delta(q, e, A) := ((q, \beta) \mid A \rightarrow \beta$ *a production in* P *of* G $)$

production transition

(3) *For each* $a \in T$ *of* G

$\delta(q, a, a) := (q, e)$

input shaving transition

(4) *Finally for* $L(P)$; *OR for* $N(P)$

$\delta(q, e, Z_0) = (f, Z_0)$ $\quad \delta(q, e, Z_0) = (q, e)$

Proof relies on relating a leftmost derivation of G

to an accepting computation of P using induction

$$S \Rightarrow_{lm} \gamma_1 \Rightarrow_{lm} \gamma_2 \cdots \Rightarrow_{lm} \gamma_n = w \in L_G$$

$$\Updownarrow$$

Clue $k = n+|w|$

$$(q_0, w, Z_0) \mid\!-\!-_P (q, w, SZ_0) \mid\!-\!-_P \alpha_1 \mid\!-\!-_P \alpha_2 \ldots \mid\!-\!-_P \alpha_k = (q, e, Z_0) \mid\!-\!-_P (f, e, Z_0)$$

initialization

final state step

Example

$G=(V,T,P,S)$ $\quad V=\{S,A,B\}$ $\quad T=\{0,1\}$

$S \rightarrow AB \quad A \rightarrow 0A1 \mid e \quad B \rightarrow 1B0 \mid e$

$L_G = \{ 0^n \; 1^{(n+m)} \; 0^m \; ; n,m \geq 0\}$

$S \Rightarrow AB \Rightarrow 0A1B \Rightarrow 0e1B \Rightarrow 011B0 \Rightarrow 011e0 = 0110$

$(q_0, 0110, Z_0) \mid\!\!-\!-_P (q, 0110, SZ_0) \mid\!\!-\!-_P (q, 0110, ABZ_0) \mid\!\!-\!-_P$

$(q, 0110, 0A1BZ_0) \mid\!\!-\!-_P (q, 110, A1BZ_0) \mid\!\!-\!-_P$

$(q_0, 110, 1BZ_0) \mid\!\!-\!-_P (q, 10, BZ_0) \mid\!\!-\!-_P (q, 10, eZ_0) \mid\!\!-\!-_P (f, 10, Z_0)$ *wrong !*

$(q, 10, 1B0Z_0) \mid\!\!-\!-_P (q, 0, B0Z_0) \mid\!\!-\!-_P (q, 0, 0Z_0)\,) \mid\!\!-\!-_P (q, e, Z_0) \mid\!\!-\!-_P (f, e, Z_0)$

Correspondence : $\gamma_m = w_m A_m \beta_m$ *with* $|w_m| = f(m)$, $A_m \in V$

corresponds to $\alpha_{m+f(m)} = (q\ ,\ u_m\ ,\ A_m\ \beta_m Z_0)$ *where* $w = w_m u_m$

using the production $A_m \to \Phi$ *express* $\Phi\ \beta_m$ *as* $\Phi\ \beta_m = v_m\ A_{m+1}\ \beta_{m+1}$

$\gamma_m = w_m\ A_m\ \beta_m \Rightarrow_G w_m\ \Phi\ \ \beta_m = w_m\ v_m\ A_{m+1}\ \beta_{m+1}$; *hence*

$w_{m+1} = w_m\ v_m$ *and* $\gamma_{m+1} = w_{m+1}\ A_{m+1}\ \beta_{m+1}$, *which leads to the production transition :*

$$\alpha_{m+f(m)} := (q\ ,\ u_m\ ,\ A_m\ \beta_m Z_0\ ) \mid\!\!--_P (q,\ u_m\ ,\ v_m\ A_{m+1}\ \beta_{m+1} Z_0\ ) = \alpha_{m+f(m)+1}$$

and following this there are $|v_m|$ *input shaving transitions ,hence setting* $j = |v_m|$, *that leads to :*

$$(q,\ u_m\ ,\ v_m\ A_{m+1}\ \beta_{m+1} Z_0\ ) \mid\!\!--_P{}^j\ (q,\ u_{m+1}\ ,\ A_{m+1}\ \beta_{m+1} Z_0\ ) = \alpha_{m+f(m)+1+j} = \alpha_{m+1+f(m+1)}$$

where $f(m+1) := f(m)+|v_m| = |w_{m+1}|$; $v_m\ u_{m+1} = u_m$, ; $w_m\ v_m\ u_{m+1} = w_m\ u_m = w_{m+1} u_{m+1} = w$

Hence the correspondence is valid for **m+1** *and with* $A_n = u_n = \beta_n = e$ *and* $w_n = w$, *the proof is complete*

Theorem 2 (if)

For every language $\boldsymbol{L}$ *accepted by a* ***PDA***

there is a CFG , $\boldsymbol{G}$ *with* $\boldsymbol{L_G = L}$

Theorem 2 *(restated) Given a PDA,*

$\boldsymbol{P = (Q\ ,\ \Sigma,\ \Gamma,\ \delta\ ,\ q_0\ ,\ Z_0\ ,\ F\ )}$ *there exists a CFG ,*

$\boldsymbol{G = (\ V,\ T,\ P\ ,\ S)}$ *such that* $\boldsymbol{w} \in \boldsymbol{L_P}$ *if and only if* $\boldsymbol{w} \in \boldsymbol{L_G}$

Given a PDA, $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ find a CFG

$G = (T, V, P, S)$ such that $L_P = L_G$

$T = \Sigma$,

$V = \{ [p\,X\,q] \mid p, q \in Q, X \in \Gamma \} \cup \{ S \}$

Productions in P :

(1) $S \rightarrow [q_0 Z_0 p]$, for all $p \in Q$

(2) For each transition component with :

$(r, Y_1 Y_2 \ldots Y_k) \in \delta(q, a, X)$; $r, q \in Q$; $Y_j \in \Gamma$, $j = 1, \ldots, k$;

$X \in \Gamma$; $a \in \Sigma \cup e$

the productions :

$[q\,X\,r_k] \rightarrow a\,[r\,Y_1\,r_1]\,[r_1\,Y_2\,r_2] \ldots [r_{k-1}\,Y_k\,r_k]$

all $r_1, r_2, \ldots, r_k \in Q$

Interpretation of *[q X p]* *:* ***P*** *moves from state* ***q*** *to some* ***p*** *eventually popping* ***X*** *from its stack and in the process consuming the input string* ***w***

Precise statement to be proved by induction on the steps of derivation (only if) and computation (if) respectively :

$[q\ X\ p] \Rightarrow_G^* w$ *if and only if* $(q\ ,\ w\ ,\ X)\ |\!\!-\!\!-_P^*\ (p\ ,e,\ e)$

(we use the convention : acceptance by empty stack, for ***P*** *)*

Example for constructing $G = (V, T, P, S)$ *from* $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

PDA *accepts the language* ($w \in \{a,b\}^*$ | #'**a**'s = #'**b**'s in w)

Transitions for $N(P)$

$(q, a, Z_0) \rightarrow (q, aZ_0)$

$(q, b, Z_0) \rightarrow (q, bZ_0)$

$(q, a, a) \rightarrow (q, aa)$

$(q, b, b) \rightarrow (q, bb)$

$(q, a, b) \rightarrow (q, e)$

$(q, b, a) \rightarrow (q, e)$

$(q, e, Z_0) \rightarrow (q, e)$

Let $Z := [qZ_0q]$, $A := [qaq]$, $B := [qbq]$,

$(q, a, Z_0) \rightarrow (q, aZ_0)$; $Z \rightarrow a\,AZ$

$(q, b, Z_0) \rightarrow (q, bZ_0)$; $Z \rightarrow b\,BZ$

$(q, a, a) \rightarrow (q, aa)$; $A \rightarrow a\,AA$

$(q, b, b) \rightarrow (q, bb)$; $B \rightarrow b\,BB$

$(q, a, b) \rightarrow (q, e)$; $B \rightarrow a$

$(q, b, a) \rightarrow (q, e)$; $A \rightarrow b$

$(q, e, Z_0) \rightarrow (q, e)$; $Z \rightarrow e$

Example for constructing $G=(V,T,P,S)$ from $P=(Q,\Sigma,\Gamma,\delta,q_0,Z_0,F)$

PDA *accepts the language* $(w \in \{a,b\}^* \mid \#'a's = \#'b's \text{ in } w)$

Transitions for N(P)

$(q, a, Z_0) \rightarrow (q, aZ_0)$

$(q, b, Z_0) \rightarrow (q, bZ_0)$

$(q, a, a) \rightarrow (q, aa)$

$(q, b, b) \rightarrow (q, bb)$

$(q, a, b) \rightarrow (q, e)$

$(q, b, a) \rightarrow (q, e)$

$(q, e, Z_0) \rightarrow (q, e)$

$G = (\{Z,A,B\}, \{a,b\}, P, Z)$

$Z \rightarrow a\,AZ \mid b\,BZ \mid e$

$A \rightarrow a\,AA \mid b$

$B \rightarrow b\,BB \mid a$

$(q, e, Z) \rightarrow (q, aAZ)$

$(q, e, Z) \rightarrow (q, bBZ)$

$(q, e, Z) \rightarrow (q, e)$

$(q, e, A) \rightarrow (q, aAA)$

$(q, e, A) \rightarrow (q, b)$

$(q, e, B) = (q, bBB)$

$(q, e, B) = (q, b)$

$(q, a, a) \rightarrow (q, e)$

$(q, b, b) \rightarrow (q, e)$

$(q, e, Z_0) \rightarrow (q, ZZ_0)$

$(q, e, Z_0) \rightarrow (q, e)$

Transitions for different PDA   Accept : abba

$(q, a, Z_0) \rightarrow (q, aZ_0)$ $(q, b, Z_0) \rightarrow (q, bZ_0)$ $(q, a, a) \rightarrow (q, aa)$ $(q, b, b) \rightarrow (q, bb)$

$(q, a, b) \rightarrow (q, e)$ $(q, b, a) \rightarrow (q, e)$ $(q, e, Z_0) \rightarrow (q, e)$

$(q, abba, Z_0) \vdash (q, bba, aZ_0) \vdash (q, ba, Z_0) \vdash (q, a, bZ_0) \vdash (q, e, Z_0) \vdash (q, e, e)$

$(q, e, Z_0) \rightarrow (q, ZZ_0)$ $(q, e, Z) \rightarrow (q, aAZ)$ $(q, e, Z) \rightarrow (q, bBZ)$ $(q, e, Z) \rightarrow (q, e)$

$(q, e, A) \rightarrow (q, aAA)$ $(q, e, A) \rightarrow (q, b)$ $(q, e, B) = (q, bBB)$ $(q, e, B) = (q, a)$

$(q, a, a) \rightarrow (q, e)$ $(q, b, b) \rightarrow (q, e)$ $(q, e, Z_0) \rightarrow (q, e)$

$(q, abba, Z_0) \vdash (q, abba, ZZ_0) \vdash (q, abba, aAZZ_0) \vdash (q, bba, AZZ_0) \vdash (q, bba, bZZ_0) \vdash$
$(q, ba, ZZ_0) \vdash (q, ba, bBZZ_0) \vdash (q, a, BZZ_0) \vdash (q, a, aZZ_0) \vdash (q, e, ZZ_0) \vdash (q, e, Z_0)$
(q, e, e)

Lemma *Given a PDA* $\boldsymbol{R}$ *with an input string* $\boldsymbol{w}$*, states* $\boldsymbol{p_1}$ *and* $\boldsymbol{p_{n+1}}$ *and stack elements* $\boldsymbol{X_1 , X_2 \ldots , X_n}$ *; then*

$$(p_1 , w , X_1 X_2 \ldots X_n) \mid\!\!-\!-_R^k \ (p_{n+1} , e , e)$$

if and only if

for some $\boldsymbol{p_2 , \ldots , p_n}$ *and* $\boldsymbol{w_1 , w_2 , \ldots , w_n}$ *with* $\boldsymbol{w := w_1 w_2 \ldots w_n}$ **:**

$$(p_i , w_i , X_i) \mid\!\!-\!-^* (p_{i+1} , e , e) , i = 1, 2, \ldots, n$$

Proof *Use induction on* $\boldsymbol{n}$ *(for* $\boldsymbol{n = 1}$ *obvious !!) (for details refer to Fact 1- Fact 3 in the NOTES distributed in class)*

Now let

$(p_1, w, X_1 X_2 \ldots X_n) \mid\!\!--_R^* (p_{n+1}, e, e)$ *then by definition of a* ***PDA*** *there will come a* ***first*** *computational instance when* X_1 *pops, that is :*

$(p_1, w, X_1 X_2 \ldots X_n) \mid\!\!--_R^* (p_2, u_1, X_2 \ldots X_n)$ *for some* p_2 *and* u_1 *since multiple* ***pops*** *are not allowed within a single step of computation. Hence*

$(p_1, w_1 u_1, X_1 (X_2 \ldots X_n)) \mid\!\!--_R^* (p_2, u_1, X_2 \ldots X_n)$ *which implies that :*

$(p_1, w_1, X_1) \mid\!\!--_R^* (p_2, e, e)$ *since* X_2 *never became the top of the stack and had no influence on prior steps of computation. The result follows by induction applied to :*

$(p_2, u_1, X_2 \ldots X_n) \mid\!\!--_R^* (p_{n+1}, e, e)$

Exercise : *Prove the* ***('if ',*** *that is the* ***'$\Leftarrow$ '*** *part of the* ***Lemma)***

Proving the main result

Part 1 *If* $(q, u, X) \vdash_P^{k} (q_{n+1}, e, e)$ *(a* k *step computation)*

show that $[q\,X\,q_{n+1}] \Rightarrow_G^* u$ *using induction on* k

$(q, av, X) \vdash_P (q_1, v, Y_1 Y_2 \ldots Y_n) \vdash_P^* (q_{n+1}, e, e)$

where $(q_1, Y_1 Y_2 \ldots Y_n) \in \delta(q, a, X)$

Now apply the **Lemma (only if)** ; *then for some* $q_2, q_3, \ldots q_n$ *and*

$u_1, u_2, \ldots, u_n$ *we have* $v = u_1 u_2 \ldots u_n$ *and*

$(q_i, u_i, Y_i) \vdash_P^* (q_{i+1}, e, e), i = 1, \ldots, n$

By definition of the grammar G *we have the production*

$[q\,X\,q_{n+1}] \rightarrow a\,[q_1\,Y_1\,q_2] \ldots [q_n\,Y_n\,q_{n+1}]$

Since by induction hypothesis the computation steps $r(i)$ *below is :* $r(i) < k$

$(q_i, u_i, Y_i) \vdash_P^{r(i)} (q_{i+1}, e, e)$ *implies that* $[q_i\,Y_i\,q_{i+1}] \Rightarrow_G^* u_i$

Hence result follows by a leftmost derivation

Part 2

If $[q\ X\ q_{n+1}] \Rightarrow_G^k u$ *(a* **k** *step derivation)*

show that using induction on **k** , $(q, u, X) \mid\!\!--_P{}^* (q_{n+1}, e, e)$

$[q\ X\ q_{n+1}] \to a\ [q_1\ Y_1\ q_2] \ldots [q_n\ Y_n\ q_{n+1}]$

where we assume that $(q_1, Y_1\ Y_2 \ldots Y_n) \in \delta(q, a, X)$

A leftmost derivation reveals that :

$[q_i\ Y_i\ q_{i+1}] \Rightarrow^{r(i)} v_i$ *and* $u = a\ v_1 \ldots v_n$ *where, necessarily* $r(i) < k$

Hence by induction hypothesis :

$(q_i, v_i, Y_i) \mid\!\!--_P{}^* (q_{i+1}, e, e)$, $i=1,\ldots,n$ *and by the* ***Lemma (if)***

$(q_1, v_1\ v_2 \ldots v_n, Y_1\ Y_2 \ldots Y_n) \mid\!\!--_P{}^* (q_{n+1}, e, e)$ *and adding the first transition*

$(q,u,X) = (q, a\ v_1\ v_2 \ldots v_n, X) \mid\!\!--_P (q_1, v_1\ v_2 \ldots v_n, Y_1\ Y_2 \ldots Y_n) \mid\!\!--_P{}^* (q_{n+1}, e, e)$

Hence

$w \in L(P)$

iff

$(q_0, w, Z_0) \mid\!\!-\!-_P^* (f, e, Z_0)$ (or (f, e, e) for $N(P)$)

iff

$S \Rightarrow_G [q_0 Z_0 f] \Rightarrow_G^* w$

iff

$w \in L_G$

Deterministic Pushdown Automata (DPDA)

Definition *A PDA* $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ *is said to be deterministic if*

(1) $|\delta(q, a, X)| \leq 1, \forall (q \in Q, a \in \Sigma \cup e, X \in \Gamma)$

(2) *If* $|\delta(q, a, X)| > 0$ *for some* $a \in \Sigma$ *then* $|\delta(q, e, X)| = 0$

Theorem *Every regular language is accepted by a DPDA*

Proof : Use a DPDA that does not use its stack !!

Fact *: there is a* ***DPDA*** *that accepts* $\{wcw^R\}$ *but none that accepts* $\{ww^R\}$ *!!!*

A language **L** *has the* ***prefix property*** *if there are NO distinct* **x, y** *in* **L** *such that* $\boldsymbol{y = x.\, u}$ *for some* **u** *(i.e* **x** *is a prefix of* **y***)*

$L = \{w.c.w^R \mid w \in (0+1)^*\}$ *has the prefix property whereas* $L' = 0^*$ *or* $L' = \{w.w^R \mid w \in (0+1)^*\}$ *does NOT have the prefix property !*

Theorem *A language* **L** *is* **N(P)** *for some DPDA* **P** *if and only if :*

(1) L *has the prefix property*

(2) L *is* **L(P')** *for some DPDA* **P'**

Proof *:*

(⇐) Assume (1) and (2) and by (2) assume a **P'** *for* **L(P')** *and convert this* **P'** *to some* **P** *for* **N(P)** *which is possible since (1) holds !*

(⇒) If **L** *is* **N(P)** *for some DPDA* **P** *then:(1)* **L** *must have the prefix property and : (2)* **L** *is* **L(P')** *for some DPDA* **P'**

$(\Leftarrow)$

Let $\boldsymbol{P'}$ *accept language* $\boldsymbol{L}$ *as* $\boldsymbol{L(P')}$ *(by final state* $\boldsymbol{f}$*)*

Let $(q_0, u, Z_0) \vdash_{P'} (q_1, u_1, \alpha_1) \vdash_{P'} \dots (q_n, u_n, \alpha_n) \vdash_{P'} (f, e, \alpha_{n+1})$

be any accepting computation of $\boldsymbol{P'}$

Then $q_j \neq f = \textbf{final state}$*, for any* $j=,0, \dots, n$ *, by* **(1)** *since :*

$(q_0, u, Z_0) \vdash_{P'} (q_1, u_1, \alpha_1) \vdash_{P'} \dots (q_i = f, u_i, \alpha_i)$

implies that $u = w \,.\, u_i$ *and therefore*

$(q_0, w, Z_0) \vdash_{P'} \dots \vdash_{P'} (f, e, \alpha_i)$

and $\boldsymbol{P'}$ *accepts a prefix* $\boldsymbol{w}$ *of* $\boldsymbol{u}$ *a contradiction to* **(1)**

Hence adding the transitions $(f,e,X) \to (f,e)$ *for all* $X \epsilon \Gamma$ *solves the problem*

since this version of $\boldsymbol{P'}$ *,say* $\boldsymbol{P''}$*, is a* **DPDA** *and accepts* $\boldsymbol{u}$ *by* $\boldsymbol{N(P'')}$

(⇒)

*If **L** is **N(P)** for some DPDA **P** then we shall show that*

L** is **L(P')** for some DPDA **P'

*Let **P** be a DPDA that accepts **L** by empty stack*

*Insert a new state **q'**, a new bottom stack **Z***

and a new initialization transition $(q', e, Z) \rightarrow (q_0, Z_0 Z)$

*The **last** computation of **P** in accepting any word **w** will be for some*

***q,p,a** (where before **Z** was added resulted in empty stack !) :*

$(q, a, Z_0) \vdash_P (p, e)$ *which corresponds to the computation*

$(q, a, Z_0 Z) \vdash_{P'} (p, e, Z)$, *in **P'***

*Now add for all such distinct **p**s the transition(s) :* $(p, e, Z) \rightarrow (f, Z)$

*where **f** is the only final state of the new **P'***

(i) Example $\{wcw^R\}$ (X = generic variable)

$X = a$, b or Z_0

$(q_0, a, X) \rightarrow (q_0, aX)$

$(q_0, b, X) \rightarrow (q_0, bX)$

$(q_0, c, X) \rightarrow (q, X)$

$(q, a, a) \rightarrow (q, e)$

$(q, b, b) \rightarrow (q, e)$

$(q, e, Z_0) \rightarrow (f, Z_0)$

Is this a DPDA ?

Example ($w \in \{a,b\}^*$ | #'a's = #'b's in w)

$(q_0, a, Z_0) \rightarrow (q_0, aZ_0)$

$(q_0, b, Z_0) \rightarrow (q_0, bZ_0)$

$(q_0, a, a) \rightarrow (q_0, aa)$

$(q_0, b, b) \rightarrow (q_0, bb)$

$(q_0, a, b) \rightarrow (q_0, e)$

$(q_0, b, a) \rightarrow (q_0, e)$

$(q_0, e, Z_0) \rightarrow (f, Z_0)$

Is this a DPDA ?

$(q_0, e, Z_0) \rightarrow (f, Z_0)$

$(f, a, Z_0) \rightarrow (q_0, aZ_0)$

$(f, b, Z_0) \rightarrow (q_0, bZ_0)$

$(q_0, a, a) \rightarrow (q_0, aa)$

$(q_0, b, b) \rightarrow (q_0, bb)$

$(q_0, a, b) \rightarrow (q_0, e)$

$(q_0, b, a) \rightarrow (q_0, e)$

How about this ?

Ambiguous Grammars and DPDA

Theorem *If a language **L** is accepted by a DPDA **P** then it has an non-ambiguous CFG*

*Proof : For a DPDA **P** and **w** the unique (only) computation sequence is :*

$(q_0, w, Z_0) \vdash (q_1, u_1, \alpha_1) \vdash \ldots \vdash (q_k, u_k, \alpha_k)$

*and is **accepting** iff* $q_k = f$ *and* $u_k = e$, *for some final state* f *(or* $\alpha_k = e$*)*

*The corresponding CFG **G** has a leftmost derivation which is also unique*

$S \Rightarrow_G [q_0 Z_0 f] \Rightarrow_G \ldots \Rightarrow_G w$

Prove the above statement by using induction on the steps of computation !