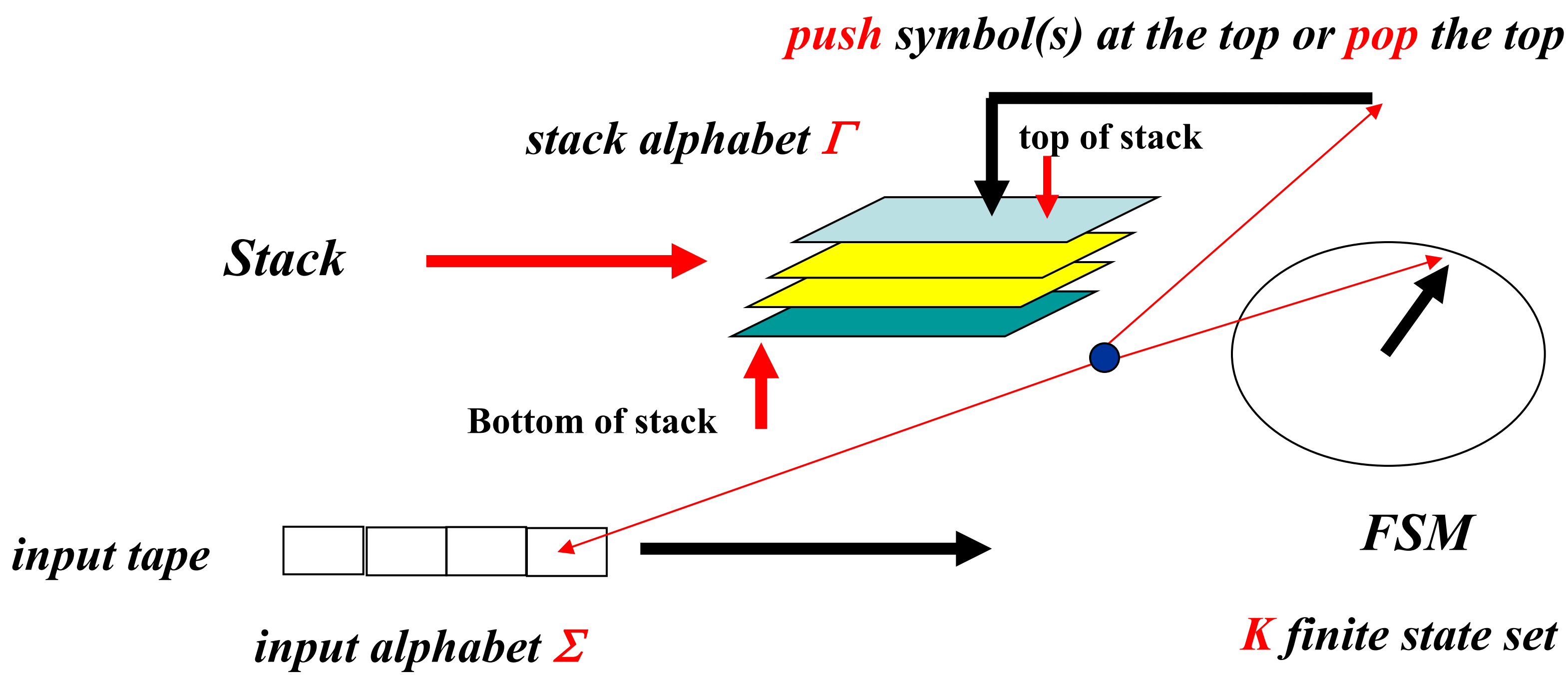
Pushdown Automata



$$\delta : Q \times (\Sigma \cup e) \times \Gamma \rightarrow 2^{(K \times \Gamma^*)} \qquad (q, \sigma, X) \rightarrow ((p_1, \gamma_1), \ldots, (p_k, \gamma_k))$$

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

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) |-- * (f, e, γ) , 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) |-- * (q, e, e) , where e = empty string

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

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

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

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

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

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

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

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

$(q, e, Z_0) \to (f, Z_0)$ accept by L(P)

{$(q, e, Z_0) \to (q, e)$ accept by N(P)}

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

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

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

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

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

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

$(q, e, Z_0) \to (f, Z_0)$ accept by L(P)

{$(q, e, Z_0) \to (q, e)$ accept by N(P)}

Example (iii):($w \in \{a,b\}^* \mid \#'a's = \#'b's \text{ in } w$)

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

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

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

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

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

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

L(P)

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

N(P)

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

Acceptance by final state :

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

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

Conversion is simple !!

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 :

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

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 $\boldsymbol{L_G}$ *where* **G** *is a* ***CFG***

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

Theorem 1 *(restated) Given a CFG,* $\boldsymbol{G = (V,T,R,S)}$ *there*

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

$\boldsymbol{w \in L_G}$ *if and only if* $\boldsymbol{w \in L_P}$

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

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

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

(1) Starting transition

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

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

$\delta(q, e, A) := ((q, \beta) \mid A \to \beta$ a production in R of G) production transitions

(3) For each $a \in T$ of G input shaving transitions

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

(4) For **L(P)** acceptance For **N(P)** acceptance

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

Note that if $e \in L_G$ a **single state**, namely $q_0 = q$, is sufficient for **N(P)** acceptance by definition.

Proof relies on relating a leftmost derivation of G to an accepting computation of P using induction

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

$$\Updownarrow$$

Clue: total no. of transitions = $n+|w|+2$

$$(q_0, w, Z_0) |\!-\!-_P (q, w, SZ_0) |\!-\!-_P \alpha_1 \dots |\!-\!-_P \alpha_k \dots |\!-\!-_P \alpha_n \dots |\!-\!-_P (q, e, Z_0) \ |\!-\!-_P (f, e, Z_0)$$

initialization →

final state step →

$S \Rightarrow_{lm} \gamma_1 \ldots \Rightarrow_{lm} \gamma_m \ldots \Rightarrow_{lm} \gamma_n = w \in L_G$; $\gamma_m = w_m A_m \beta_m$, $m=1,\ldots,n$; $A_n = \beta_n = null$

$A_m \rightarrow \Psi$; $A_m \beta_m \Rightarrow \Psi \beta_m = u_{m+1} A_{m+1} \beta_{m+1}$; $\gamma_{m+1} = w_m u_{m+1} A_{m+1} \beta_{m+1} = w_{m+1} A_{m+1} \beta_{m+1}$

$(q_0, w, Z_0) \vdash_P (q, w, SZ_0) \vdash_P \alpha_1 \ldots \vdash_P \alpha_k \ldots \vdash_P \alpha_n \ldots \vdash_P (q, e, Z_0) \vdash_P (f, e, Z_0)$

$\alpha_1 = (q, u_1 v_1, u_1 A_1 \beta_1 Z_0)$; $\alpha_k = (q, u_k v_k, u_k A_k \beta_k Z_0)$; $\alpha_n = (q, u_n, u_n Z_0)$

$w = u_1 u_2 u_3 \ldots u_n$; $v_k = u_{k+1} \ldots u_n$; $w_{m+1} = u_1 u_2 u_3 \ldots u_{m+1}$

after each $\boldsymbol{\alpha_k}$ *(ID triple) there are* $|\boldsymbol{u_k}|$ *shaving transitions for* $\boldsymbol{k=1,\ldots,n}$

total number of shaving transitions = $|\boldsymbol{w}| = |\boldsymbol{u_1}| + \ldots + |\boldsymbol{u_n}|$

total number of production transitions = $\boldsymbol{n}$

total number of transitions = $\boldsymbol{n} + |\boldsymbol{w}| + \boldsymbol{2}$

Example

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

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

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

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

$(q_0, 0110, Z_0) \vdash_P (q, 0110, SZ_0) \vdash_P (q, 0110, ABZ_0) \vdash_P$

$(q, 0110, 0A1BZ_0) \vdash_P (q, 110, A1BZ_0) \vdash_P$

$(q, 110, 1BZ_0) \vdash_P (q, 10, BZ_0)$ $\vdash_P (q, 10, eZ_0) \vdash_P (f, 10, Z_0)$ wrong !

$(q, 10, 1B0Z_0) \vdash_P (q, 0, B0Z_0) \vdash_P (q, 0, 0Z_0)\ ) \vdash_P (q, e, Z_0) \vdash_P (f, e, Z_0)$

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* ,

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

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

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

$T = \Sigma$,

$V = \{ [p\, X\, q] \mid p, q \in Q,\ X \in \Gamma \cup \Sigma \} \cup \{ S \}$; $|V| = |Q|^2 \, |\Gamma \cup \Sigma| + 1$

Productions in R :

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

(2) For each transition component with :

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

$(q, a, X) \rightarrow (r, Y_1 Y_2 \ldots Y_k)$

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 $G = (V, T, R, S)$ 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)

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

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

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

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

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

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

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

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

if $(q, a, X) \rightarrow (r, Y_1 Y_2 \ldots Y_k)$

then

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

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

if $(q, a, X) \rightarrow (r, e)$

then

$[q\,X\,r] \rightarrow a$

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

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

$Z \to a\,AZ$
$Z \to b\,BZ$
$Z \to e$

Transitions for N(P)

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

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

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

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

$A \to a\,AA$
$A \to b$

$B \to bBB$
$B \to a$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Transitions for different PDA   Accept : abba

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

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

$(q, abba, Z_0)$ |-- (q, bba, aZ_0) |-- (q, ba, Z_0) |-- (q, a, bZ_0) |-- (q, e, Z_0) |-- (q, e, e)

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

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

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

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

Lemma *Given a PDA* $\boldsymbol{P}$ *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*

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

if and only if

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

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

For a proof of ***if*** ***(⇒)*** *part we use induction on* ***n*** *below , hence let :*

$(p_1 , w , X_1 X_2 \ldots X_n) \mid\!-\!-_P^* (p_{n+1} , e , e)$ *then by definition of a* ***PDA*** *there will come a* ***first*** *computational instance when* X_1 *pops ; that is, for some* $k > 0$ *:*

$(p_1 , w , X_1 X_2 \ldots X_n) \mid\!-\!-^k_P (p_2 , u , X_2 \ldots X_n)$ *where throughout* ***k-1*** *steps of computation the stack contents were of the form* $\gamma . (X_1 \ldots . X_n)$ *; except before the last step at which it was* $(X_1 \ldots . X_n)$ *and the next action was the popping of* X_1 *. Hence for* $w = w_1 u$ *:*

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

$(p_1 , w_1 , X_1) \mid\!-\!-^k_P (p_2 , e , e)$ *since* X_2 *never became visible at the top and had no influence on prior steps of computation. This statement can be made more precise by using induction on* ***k*** *. The final result follows by using the induction hypothesis applied to the problem of size* ***n-1*** *below. :*

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

u *and* $X_2 \ldots X_n$ *has no influence on the first* ***k*** *steps of computation*

Proof of the ***'only if '**, that is the* **'⇐'** *part of the* ***Lemma***

Given states $\boldsymbol{p_2 , \ldots , p_n}$ *and inputs* $\boldsymbol{w_1 , w_2 , \ldots , w_n}$ *with* $\boldsymbol{w := w_1 w_2 \ldots w_n}$ **:**

$\boldsymbol{(p_i , w_i , X_i) \mid\!\!-\!\!-^* (p_{i+1} , e , e) , i = 1, 2, \ldots, n}$ *; show* $\boldsymbol{(p_1 , w , X_1 X_2 \ldots X_n) \mid\!\!-\!\!-_P^{*} (p_{n+1} , e , e)}$.

Set $\boldsymbol{(p_1 , w , X_1 X_2 \ldots X_n) = (p_1 , w_1 w_2 \ldots w_n , X_1 X_2 \ldots X_n)}$ *and use induction on* $\boldsymbol{n}$ **.**

First show that under the given premiss : $\boldsymbol{(p_1 , w_1 , X_1) \mid\!\!-\!\!-^k (p_2 , e , e)}$ **,**

$\boldsymbol{(p_1 , w_1 u , X_1 \gamma) \mid\!\!-\!\!-^k (p_2 , u , \gamma)}$ *for any* $\boldsymbol{u \in \Sigma^*}$ *and* $\boldsymbol{\gamma \in \Gamma^*}$ **.** *To prove this we use induction on* $\boldsymbol{k}$. *For* $\boldsymbol{k=1}$ *it is obvious since the only possibility is* $\boldsymbol{w_1 = a}$ *or* $\boldsymbol{= e}$ *and* $\boldsymbol{X_1}$ **is** *popped. Else* $\boldsymbol{w_1 = a . v}$ *and* $\boldsymbol{(p_1 , w_1 , X_1) \mid\!\!-\!\!- (p'_1 , v , \phi) \mid\!\!-\!\!-^{k-1} (p_2 , e , e)}$

and by the induction hypothesis $\boldsymbol{(p_1 , w_1 u , X_1 \gamma) \mid\!\!-\!\!- (p'_1 , v u, \phi \gamma) \mid\!\!-\!\!-^{k-1} (p_2 , u , \gamma)}$

and so letting $\boldsymbol{u = w_1 w_2 \ldots w_n}$ *and* $\boldsymbol{\gamma = X_2 X_3 \ldots X_n}$

$\boldsymbol{(p_1 , w_1 w_2 \ldots w_n , X_1 X_2 \ldots X_n) \mid\!\!-\!\!-_P^{k} (p_2 , w_2 \ldots w_n , X_2 X_3 \ldots X_n)}$ *and by the induction hypo.*

$\boldsymbol{(p_1 , (w_2 \ldots w_n), (X_2 \ldots X_n)) \mid\!\!-\!\!-^* (p_{n+1} , e , e)}$

Proving the main result

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

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

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

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

Now apply the ***Lemma (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*

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

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 :* $r(i) < k$

$(q_i, u_i, Y_i) \mid\!\!--_P{}^{r(i)} (q_{i+1}, e, e)$ *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 , $(q\, ,u,\, X) \mid\!\!--_P {}^* (q_{n+1}\, , e\, , e)$ *; let* $u = av$ *hence*

$(q\, , av,\, X) \mid\!\!--_P (q_1\, , v,\, Y_1\, Y_2 \ldots Y_n\,)$

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

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

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 (only 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* $\boldsymbol{P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)}$ *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* $|\delta(q, a, X)| = 1$ *for some* $a \in \Sigma$ *then* $|\delta(q, e, X)| = 0$

(equivalently: if $|\delta(q, e, X)| = 1$ *then* $|\delta(q, a, X)| = 0$ *for all* $a \in \Sigma$*)*

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 **y = x . u** for some **u** (i.e **x** is a prefix of **y**)*

*e.g. : **10011001***

$L = \{w.c.w^R \mid w \in (0+1)^*\}$ *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'***

*Note that if $e \in L$ then **L** does **NOT** have the prefix property unless $L = \{e\}$ since **e** is a strict prefix of any string $u \neq e$*

($\Leftarrow$)

cannot allow $|\delta(f,a,X)|=1$

since $|\delta(f,e,X)|=1$ for all X

*Let **P'** accept language **L** as **L(P')** (by final state **f**)*

Let $(q_0, u, Z_0) \mid\!\!-\!\!-_{P'} (q_1, u_1, \alpha_1) \mid\!\!-\!\!-_{P'} \ldots (q_n, u_n, \alpha_n) \mid\!\!-\!\!-_{P'} (f, e, \alpha_{n+1})$

*be any accepting computation of **P'** .*

*By adding the transitions **(f,e,X) → (f,e)** for ALL $X \in \Gamma$ solves the problem*

*provided that this version of **P'** ,namely **P**, is a **DPDA** and accepts **u** by **N(P)**.*

To justify this step we show that for any step in the computation above, that is :

$(q_j, u_j, \alpha_j) \mid\!\!-\!\!-_{P'} (q_{j+1}, u_{j+1}, \alpha_{j+1})$ *; the transition used cannot be :*

$(f, a, Y) \to (q_{j+1}, Y')$ *for some* $a \in \Sigma$; *for if so* $q_j = f$ *and* $u_j \neq e$ *hence for some **w***

$u = w\, u_j$ *and **w** is accepted by final state **f** and **prefix property** is violated by **L***

*contrary to assumption **(1)** ! Hence the transitions **(f,e,X) → (f,e)** do not violate*

*the assumption that **P** is a **DPDA** .*

($\Rightarrow$)

*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 !) : $(q, a, Z_0) \mid\!\!-\!\!-_P (p, e, e)$ which corresponds to the computation : $(q, a, Z_0 Z) \mid\!\!-\!\!-_{P'} (p, e, Z)$, 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'**.*

Exercise : *Show that **L** has the prefix property if it is **N(P)** accepted by a DPDA **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 $(w \in \{a,b\}^* \mid \#'a's = \#'b's \text{ in } w)$

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

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

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

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

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

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

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

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 a 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*

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

Next we prove the above statement by using induction on the steps of computation !

Proof (continued) : *Consider the first transition of* $\boldsymbol{P}$:

$$(q_0, au, Z_0) \mid\text{--} (q_1, u, X_1 X_2 \ldots X_m) \mid\text{--} \ldots \mid\text{--} (q_k, e, e)$$

where acceptance is assumed to be by $\boldsymbol{N(P)}$

By a previous lemma applied to : $(q_1, u, X_1 X_2 \ldots X_m) \mid\text{--} \ldots \mid\text{--} (q_k, e, e)$

there exists $w_1, w_2, \ldots, w_m$ *and* $p_1, p_2, p_m, \ldots, p_{m+1}$ *with* $u = w_1 w_2 \ldots w_m$; $p_1 = q_1$ *and* $p_{m+1} = q_k$ *such that* :

$$(p_j, w_j, X_j) \mid\text{--}^{kj} \ldots \mid\text{--} (p_{j+1}, e, e), \quad j=1,\ldots, m$$

where each $k_j < k$ *and this corresponds to the derivation*

$$S \Rightarrow_G [q_0 Z_0 q_k] \Rightarrow_G a [p_1 X_1 p_2] [p_2 X_2 p_3] \ldots [p_m X_m p_{m+1}]$$

where $[p_j X_j p_{j+1}] \Rightarrow_G w_j$ *and since* $k_j < k$ *and the computation sequence is unique by induction hypothesis, parse tree of each* $[p_j X_j p_{j+1}]$ *is unique.*