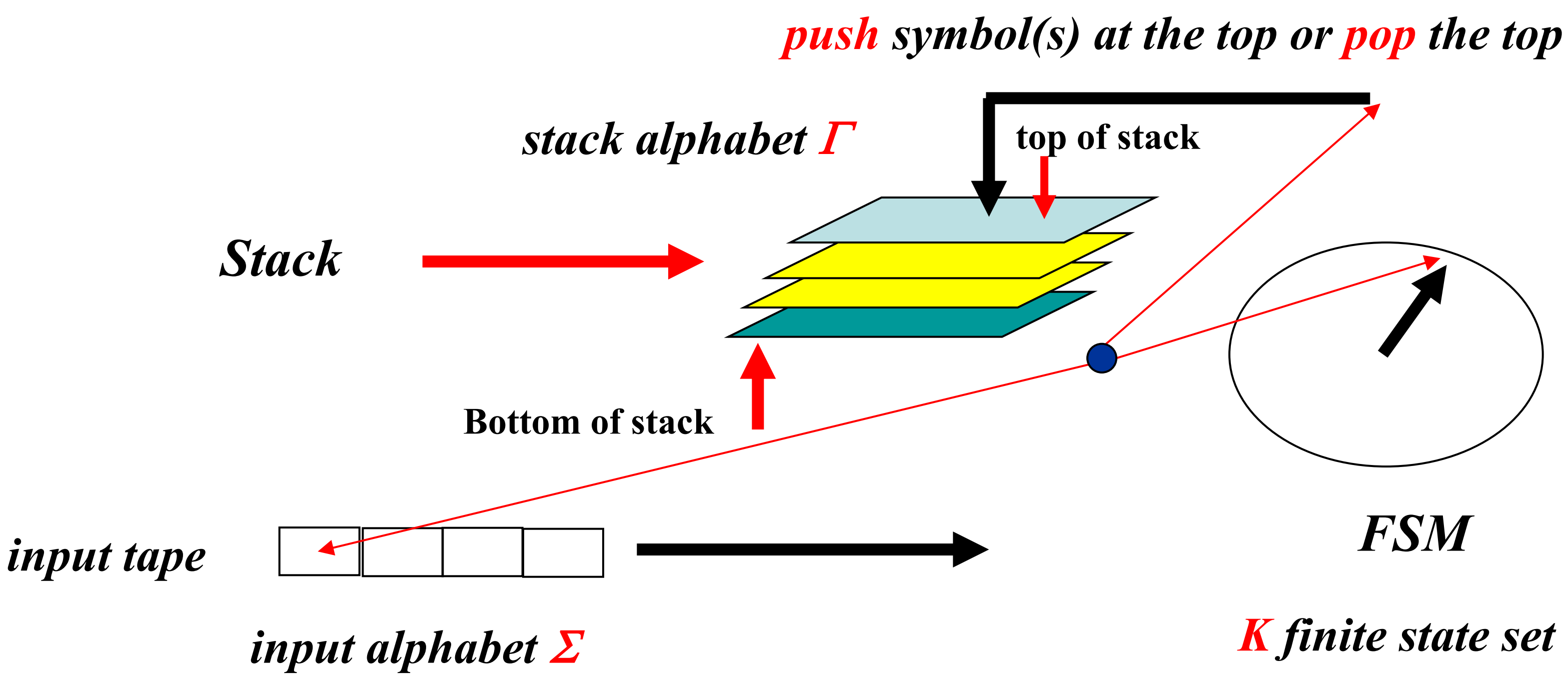
Pushdown Automata



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

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

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 either the symbol X at the top of the stack is removed and replaced by the string $\gamma' = v\,X$ with $v \in \Gamma^*$ of stack symbols (v is '**push'ed**) ; or $\gamma' = e$ in which case X is said to be **'pop'ped** .*

Instantaneous Description (ID) of a PDA

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

notation for computation

for the transition $(q,a,X) \rightarrow (p,\gamma)$
execution @ ID $(q,ax,X\lambda)$ |-- $(p,x,\gamma\lambda)$

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

$\beta \in \Gamma^*$ *(current stack contents)*

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)* 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)*

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

(i) Transitions (X = generic variable)

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

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

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

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

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

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

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

(ii) Transitions (X = generic variable)

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

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

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

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

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

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

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

Examples of a computation sequences for a specific input $\omega = abcba$

$(q_0, abcba, Z_0) \vdash (q_0, bcba, aZ_0) \vdash (q_0, cba, baZ_0) \vdash (q, ba, baZ_0) \vdash (q, a, aZ_0)$

$(q_0, a, Z_0) \rightarrow (q_0, aZ_0)$ $\quad (q_0, b, a) \rightarrow (q_0, ba)$ $\quad (q_0, c, b) \rightarrow (q, b)$ $\quad (q, b, b) \rightarrow (q, e)$

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

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

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

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)

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

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

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

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

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

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

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

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

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

Example of a computation sequence for a specific input $\omega = abaabb$

$(q_0, abaabb, Z_0)$ |-- $(q_0, baabb, aZ_0)$ |-- $(q_0, aabb, Z_0)$ |-- (q_0, abb, aZ_0) |-- (q_0, bb, aaZ_0)

$(q_0, a, Z_0) \rightarrow (q_0, aZ_0)$ $(q_0, b, a) \rightarrow (q_0, e)$ $(q_0, a, Z_0) \rightarrow (q_0, aZ_0)$ $(q_0, a, a) \rightarrow (q_0, aa)$

|-- (q_0, b, aZ_0) |-- (q_0, e, Z_0) |-- (f, e, Z_0)

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

Acceptance by final state :

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

Conversion looks simple !

L to N : Whenever any final state f is entered first move to a special state p and then at p keep on popping the stack until stack is empty.

N to L : Whenever the stack is empty move to a final state ! (in this case initially introduce a new bottom stack symbol Z_{00} and state q_{00} in N and replace all Z_0' s in N by Z_{00} and all q_0's by q_{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 :

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

More precisely :

(q_0,w,Z_0)--|* (f,e,γ) ; f in F, γ in Γ^*

1 - Consider a PDA ***P*** *that accepts a language in* ***L(P) sense*** *; i.e by* ***final state*** *.*
For all final states ***f*** *and all stack symbols* ***X*** *in* ***P****,* ***add a transition (f, e, X) → (p,X)*** *in* ***P'*** *where* ***p*** *is a new state of* ***P'****. Then at* ***p*** *add the transitions* ***(p, e, Y) → (p,e)*** *for all* ***Y*** *in* Γ *in* ***P'*** *which allows eventually the stack to empty.*

(q_0,w,Z_0)--|* (q,e,e) ; q in Q

2 - Consider a PDA , ***P'*** *that accepts a language in* ***N(P') sense*** *; i.e by* ***empty stack*** *.*
Replace ***each*** *occurrence of* Z_0 *and* q_0 *in the* ***transitions*** *of* ***P'*** *by a new stack symbol* Z_{00} *and a new state* q_{00} *in* ***P*** *. Add a new transition* $(q_0, e, Z_0) \rightarrow (q_{00}, Z_{00} Z_0)$ to ***P*** *; add for* ***each state*** $q \in Q' \cup \{q_{00}\}$ *where* $q \neq q_0$ *a transition* $(q, e, Z_0) \rightarrow (f, Z_0)$ *; then* ***P*** *obtained from* ***P'*** *with these newly added transitions and with :*

$Q = Q' \cup \{q_{00}, f\}$; $\Gamma = \Gamma' \cup \{Z_{00}\}$; $F = \{f\}$ *, accepts* ***L*** *in* ***L(P) sense,*** *i.e. by* ***final state*** *.*

*Recall for all **f** in **F** and **X** in **Γ***
(f, e, X) → (p,X)

Proof of 1-

***P** accepts **L** in **L(P)** sense ⇒ **P'** accepts **L** in **N(P')** sense (trivial by construction)*

***P'** accepts **L** in **N(P')** sense ⇒ **P** accepts **L** in **L(P)** sense .*

*Move backwards in transitions from an accepting ID of **P'** with the empty stack until the state is different from **p**.*

By construction we must have $(f, u, X\gamma) \vdash (p, u, X\gamma) \vdash^* (p, u, e)$ where $u \in \Sigma^*$ does not change since each transition is of the type $(p,e,Y) \rightarrow (p,e)$ and is non input consuming.

*But since **P'** accepts we must have **u=e** and therefore the computation of **P** :*

*$(q_0, w, Z_0) \vdash^* (f, e, X\gamma)$ is also accepting in **L(P)** sense.*

Proof of 2-

P' *accepts* ***L*** *in* ***N(P')*** *sense* $\Rightarrow$ ***P*** *accepts* ***L*** *in* ***L(P)*** *sense .*

After the injected transition $(q_0, e, Z_0) \rightarrow (q_{00}, Z_{00} Z_0)$ *in* ***P***, ***P*** *computes with* q_{00} *and* Z_{00} *replacing the original* q_0 *and* Z_0 ***of P'***. *Hence the original* ***final*** *transition* $(r, e, Z_0) \rightarrow (t, e)$ *and* ***final*** *ID* (t, e, e) *with* ***empty stack*** *and possible states* ***r*** *and* ***t*** *with* $r,t \neq q_0$ *in* ***P'*** *correspond to :* $(r, e, Z_{00}) \rightarrow (t, Z_0)$ *and* (t, e, Z_0) *in* ***P*** *respectively.*

Hence adding the transition $(q, e, Z_0) \rightarrow (f, Z_0)$ *with* $q=t$ *ensures that* ***P*** *accepts* ***L*** *in* ***L(P)*** *sense .*

P *accepts* ***L*** *in* ***L(P)*** *sense* $\Rightarrow$ ***P'*** *accepts* ***L*** *in* ***N(P')*** *sense .*

By construction the ***only*** *final state* ***f*** *of* ***P*** *corresponds to the transitions :*

$(q, e, Z_0) \rightarrow (f, Z_0)$ *for all* $q \neq q_0$ *, hence an accepting computation of* ***P*** *must look like*

$(q_0, w, Z_0) \vdash (q_{00}, w, Z_{00} Z_0) \vdash^* (q, e, Z_{00} Z_0) \vdash (q, e, Z_0) \vdash (f, e, Z_0)$

But the ID (q, e, Z_0) *of* ***P*** *above corresponds to ID* (q, e, e) *in* ***P'*** *hence* ***P'*** *accepts* ***L .***

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

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

$\delta(q, e, A) := (\,(q, \beta) \mid A \to \beta$ 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

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

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

Proof relies on relating a leftmost derivation of G

to an accepting computation of P using induction

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

$\Updownarrow$

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

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

initialization ***final state step***

$S \Rightarrow_{lm} \gamma_1 \ldots \Rightarrow_{lm} \gamma_m \ldots \Rightarrow_{lm} \gamma_n = w \in L_G$;

From $\gamma_m = w_m A_m \beta_m$ *to* $\gamma_{m+1} = w_{m+1} A_{m+1} \beta_{m+1}$:

leftmost nonterminal

leftmost production $A_m \rightarrow \Psi_m$ *hence*

$\gamma_m \Rightarrow w_m (\Psi_m) \beta_m = w_m (\Psi_m \beta_m) = w_m (u_{m+1} A_{m+1} \beta_{m+1})$

$w_m u_{m+1} A_{m+1} \beta_{m+1} = \gamma_{m+1} = w_{m+1} A_{m+1} \beta_{m+1}$

therefore $w_{m+1} = w_m u_{m+1}$ $, m=0,1,\ldots,n-1$; *with* $w_0 := e$

e=null

$w_1 = u_1$; $w_2 = w_1 u_2 = u_1 u_2$; $w_3 = w_2 u_3 = u_1 u_2 u_3$

$w_k = u_1 u_2 \ldots u_k$; *and* $w_n = w = u_1 u_2 \ldots u_n$ (*using* $\gamma_n = w_n A_n \beta_n = w$)

Set v_k *such that* $w_k v_k = w$ *for all* $k = 1,\ldots n$; *or* $v_k := u_{k+1} \ldots u_n$

$w_0 A_0 \beta_0 = S \rightarrow w_1\, A_1\, \beta_1$

$$S \Rightarrow_{lm} \gamma_1 \ldots \Rightarrow_{lm} \gamma_m \ldots \Rightarrow_{lm} \gamma_n = w \in L_G\ ;$$

$u_1 = w_1\,;\ w = w_1\, v_1$

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

$$\alpha_1 = (q, u_1 v_1, u_1 A_1 \beta_1 Z_0) \mid\!\!-\!\!-^{|u1|} (q, v_1, A_1 \beta_1 Z_0) \mid\!\!-\!\!-$$

$(A_1 \rightarrow \Psi_1\,;\ \Psi_1 \beta_1 = u_2 A_2 \beta_2)$

$$\alpha_2 = (q, u_2 v_2, u_2 A_2 \beta_2 Z_0) \mid\!\!-\!\!-^{|u2|} (q, v_2, A_2 \beta_2 Z_0) \mid\!\!-\!\!- \ldots$$

$(A_2 \rightarrow \Psi_2\,;\ \Psi_2 \beta_2 = u_3 A_3 \beta_3)$

production transitions

$\vdots$

$$\alpha_k = (q, u_k v_k, u_k A_k \beta_k Z_0) \mid\!\!-\!\!-^{|uk|} (q, v_k, A_k \beta_k Z_0) \mid\!\!-\!\!- \ldots$$

$\vdots$

$(A_k \rightarrow \Psi_k\,;\ \Psi_k \beta_k = u_{k+1} A_{k+1} \beta_{k+1})$

$\vdots$

$$\alpha_n = (q, u_n v_n, u_n A_n \beta_n Z_0) \mid\!\!-\!\!-^{|un|} (q, e, Z_0) \mid\!\!-\!\!- (f, e, Z_0)$$

e

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

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

total number of production transitions $= n$

total number of transitions $= n + |w| + 2$

Example

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

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

$L_G = \{ 0^n 1^k ; k > n \geq 0 \}$

$S \Rightarrow AB \Rightarrow 0A1B \Rightarrow 0e1B \Rightarrow 011B \Rightarrow 0111$

$(q_0, 0111, Z_0) \vdash_P (q, 0111, SZ_0) \vdash_P (q, 0111, ABZ_0) \vdash_P$

$(q, 0111, 0A1BZ_0) \vdash_P (q, 111, A1BZ_0) \vdash_P$

$(q, 111, 1BZ_0) \vdash_P (q, 11, BZ_0)$ $\vdash_P (q, 11, 1Z_0) \vdash_P (q, 1, Z_0)$ wrong !

$(q, 11, 1BZ_0) \vdash_P (q, 1, BZ_0) \vdash_P (q, 1, 1Z_0)\ ) \vdash_P (q, e, Z_0) \vdash_P (f, e, Z_0)$

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*

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

$T = \Sigma,$

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

Productions in R :

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

(2) *For each transition component with :*

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

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

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* $(q\ ,\ w\ ,\ X) \mid\!\!-\!-_P^* (p\ ,e,\ e)$

(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 ∈ **{a,b} | #'**a**'s = #'**b**'s in w)*

Transitions for P

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

$(q, a, Z_0) \to (q, aZ_0)$; $Z \to aAZ$

$(q, b, Z_0) \to (q, bZ_0)$; $Z \to bBZ$

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

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

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

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

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

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

then

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

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

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

then

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

$S \to [q_0 Z_0 p] = [qZ_0 q] = Z$; *or take* Z *as start symbol*

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

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

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

Transitions for P

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

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

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

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

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

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

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

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

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

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

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

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

$B \rightarrow bBB$
$B \rightarrow a$

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

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

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

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

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

$(q, e, B) \rightarrow (q, bBB)$

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

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

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

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

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

Transitions for different PDA   Accept : abba

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

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

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

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

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

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

$(q, abba, Z_0)$ |-- $(q, abba, ZZ_0)$ |-- $(q, abba, aAZZ_0)$ |-- (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 *A PDA* $\boldsymbol{P}$ *has a* $\boldsymbol{k}$*-step computation :* $(p, w, \beta) \vdash_P^k (q, e, e)$

for $p,q \in Q$, $w \in \Sigma^*$ *and* $\beta \in \Gamma^+$

if and only if

for the same $p,q \in Q$, $w \in \Sigma^*$ *and* $\beta \in \Gamma^+$ *, and the* ***same transitions*** *as above* $\boldsymbol{P}$ *has a* $\boldsymbol{k}$*-step computation :* $(p, w\,u, \beta\gamma) \vdash_P^k (q, u, \gamma)$ *for any* $u \in \Sigma^*$ *and any* $\gamma \in \Gamma^*$

Proof : *(by using induction on* $\boldsymbol{k}$*)*

for $k=1$

$(p, w, \beta) \vdash_P (q, e, e)$ ***iff*** $w=a$ $(a \in \Sigma$ **or** $a = e)$; $\beta = X$ *and there is a transition :* $(p, a, X) \rightarrow (q, e)$ ***iff*** $(p, a\,u, X\gamma) \vdash_P (q, u, \gamma)$ *for any* $u \in \Sigma^*$ *and any* $\gamma \in \Gamma^*$

Hence for any $\mathbf{k > 0}$ *let* $\mathbf{w = av}$ *and* $\boldsymbol{\beta = Y\alpha}$ *and for any transition*

$(p, a, Y) \rightarrow (p_1, \eta)$

$(p, av, Y\alpha) \mid\!\!-\!-_P (p_1, v, \eta\alpha) \mid\!\!-\!-_P^{k-1} (q, e, e)$

Thus by induction hypothesis for the **(k-1)***-step computation*

$(p_1, v, \eta\alpha) \mid\!\!-\!-_P^{k-1} (q, e, e)$ *holds* ***iff***

$(p_1, v\,u, \eta\alpha\gamma) \mid\!\!-\!-_P^{k-1} (q, u, \gamma)$

holds for ***any*** $\mathbf{u} \in \Sigma^*$ *and any* $\gamma \in \Gamma^*$ *using the* ***same*** *transitions as in*

$(p_1, v, \eta\alpha) \mid\!\!-\!-_P^{k-1} (q, e, e)$

Thus using the transition $(p, a, Y) \rightarrow (p_1, \eta) : (p, w, \beta) \mid\!\!-\!-_P^{k} (q, e, e)$

iff $(p, wu, \beta\gamma) \mid\!\!-\!-_P (p_1, vu, \eta\alpha\gamma) \mid\!\!-\!-_P^{k-1} (q, u, \gamma)$

Corollary to 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*

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

if and only if

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

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

Proof : *Repeatedly use* **Lemma** *with* $\boldsymbol{w u = w_j (w_{j+1} w_{j+2} \ldots w_n)}$ ***and*** $\boldsymbol{\beta \gamma = X_j (X_{j+1} X_{j+2} \ldots X_n)}$ *noting that each pair* $\boldsymbol{(w_j , p_{j+1})}$ *is the part of the input string consumed ; and the state reached, precisely at the instance* $\boldsymbol{X_j}$ *is popped !*

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

$k=1$

$(q, a, X) \rightarrow (q_{n+1}, e)$

$\boxed{[q\, X\, q_{n+1}] \rightarrow a \text{ hence}}$

$[q\, X\, q_{n+1}] \Rightarrow_G^* a$

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

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

Now apply ***Corollary to 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*

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

By definition of the grammar **G** *for the transition above 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

$k=1$, $[q\ X\ q_{n+1}] \rightarrow a$

$(q,a,X) \rightarrow (q_{n+1}, e)$

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* ***Corollary to 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

$w \in L_G$

iff

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

for any p , S → [q₀ Z₀ p] is a production

iff

$(q_0 , w , Z_0) \vdash_P^* (p , e, e)$

iff

$w \in L(P)$

if $[q\, X\, q_{n+1}] \Rightarrow_G^* u$ *then*

$(q , u, X) \vdash_P^* (q_{n+1} , e , e)$

By Part 1 and Part 2 above

if $(q , u, X) \vdash_P^* (q_{n+1} , e , e)$ *then*

$[q\, X\, q_{n+1}] \Rightarrow_G^* u$

$q \rightarrow q_0$

$u \rightarrow w$

$X \rightarrow Z_0$

$q_{n+1} \rightarrow p$

Deterministic Pushdown Automata (DPDA)

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

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

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

e.g. : 10011001

$\boldsymbol{L = \{w.c.w^R \mid w \in (0+1)^*\}}$ *has the prefix property whereas* $\boldsymbol{L' = 0^*}$ *or ;*

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

Theorem *A language* $\boldsymbol{L}$ *is* $\boldsymbol{N(P')}$ *for some DPDA* $\boldsymbol{P'}$

if and only if :

(1) $\boldsymbol{L}$ *has the prefix property*

(2) $\boldsymbol{L}$ *is* $\boldsymbol{L(P)}$ *for some DPDA* $\boldsymbol{P}$

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

($\Leftarrow$)

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

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

Let the DPDA **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 $(f,e,X) \to (f,e)$ *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 :*

$(f, a, Y) \to (q_{j+1}, \alpha')$ *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* $(f,e,X) \to (f,e)$ *do not violate*

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

(⇒)

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.*

In ***P*** *replace in* ***each*** *transition of* ***P'*** *the occurrence of* q_0 *by a new state* q_{00}, *and that of* Z_0 *by* Z_{00} *and insert the transition* $(q_0, e, Z_0) \rightarrow (q_{00}, Z_{00}Z_0)$.

The ***last*** *computation of* ***P'*** *in accepting any word* ***w*** *will correspond in* ***P*** *to* $(q, a, Z_{00} Z_0) \vdash_{P'} (q, e, Z_0)$ *for some* $q \neq q_0$; a *in* Σ *or* $a=e$; *hence adding the transition* $(q, e, Z_0) \rightarrow (f, Z_0)$ *for all possible* $q \neq q_0$ *solves the problem where* ***f*** *is the only final state of* ***P*** *and* ***P*** *is a DPDA.*

Exercise : *Show that* ***L*** *has the prefix property if it is* ***N(P')*** *accepted by the DPDA* ***P'*** *!*

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

$X = a$, b or Z_0

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

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

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

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

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

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

Is this a DPDA ?

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

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

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

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

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

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

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

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

Is this a DPDA ?

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

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

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

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

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

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

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

How about this ?

Ambiguous Grammars and DPDA

Theorem *If a language* $\mathbf{L}$ *is accepted by a DPDA* $\mathbf{P}$ *then it has a non-ambiguous CFG.*

Proof : For a DPDA $\mathbf{P}$ *and* $\mathbf{w}$ *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 derivation :*

$S \Rightarrow_G [q_0 Z_0 f] \Rightarrow_G \ldots \Rightarrow_G w$; *which corresponds to a unique parse tree for* **w** *hence* **G is** *non-ambiguous.*

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

Proof *: For* $\mathbf{k=1}$ *using N(P) acceptance*

$(q_0, w, Z_0) \mid\text{--} (q_k, e, e)$ ***iff*** $w=a$ *or* $w=e$ **;** *and*

$(q_0, a, Z_0) \rightarrow (q_k, e)$ **or** $(q_0, e, Z_0) \rightarrow (q_k, e)$ *is a transition and*

$[q_0 Z_0 q_k] \rightarrow a$ *or* $[q_0 Z_0 q_k] \rightarrow e$ *is a production and so*

$[q_0 Z_0 q_k] \Rightarrow_G a$ *or* $[q_0 Z_0 q_k] \Rightarrow_G e$ *is a derivation with a unique parse tree trivially.*

Proof (continued) : *Now 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 **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 :*

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

where each $k_j < k$ *and by the first transition above this corresponds to the derivation*

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

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 the parse tree of each* $[p_j X_j p_{j+1}]$ *and overall parse tree is unique.*