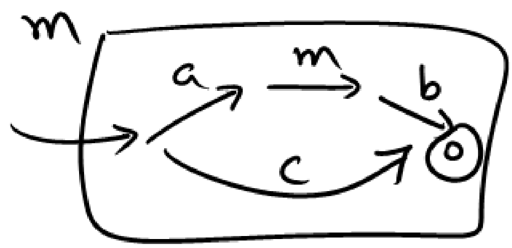
Recursive Automata



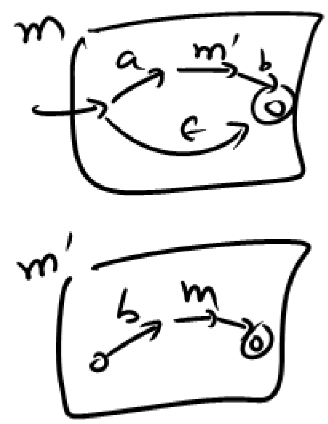
a b b a b

$S \rightarrow aSb \mid \epsilon$

aa . . . ab bb

Formal definition of recursive automata

Each module m
is an NFA
over $\Sigma \cup M$



A recursive automaton over Σ is a tuple $\left(M, m_0, \left\{ (Q_m, \Sigma \cup M, \delta_m, q_0^m, F_m) \right\}_{m \in M} \right)$

where :

- M is a finite set (called modules/procedures)
- $m_0 \in M$ (initial module)
- For every $m \in M$, $(Q_m, \Sigma \cup M, \delta_m, q_0^m, F_m)$ is an NFA.

Also $Q_m \cap Q_{m'} = \phi$ for any m, m', $m \neq m'$

[Q_m is a finite set ; $\delta_m : Q_m \times (\Sigma \cup \{\epsilon\} \cup M) \to \mathcal{P}(Q_m)$; $q_0^m \in Q_m$; $F_m \subseteq Q_m$]

A *configuration* of a RA is a pair (q, s)

where $q \in \bigcup_m Q_m$, $s \in Q^*$.

A recursive automaton $(M, m_0, \{(Q_m, \Sigma \cup M, \delta_m, q_0^m, F_m)\})$

accepts a word $w = a_1 a_2 a_3 \dots a_n$

if there exists $y_1 \dots y_t = a_1 \dots a_n$

and each $y_i \in \Sigma \cup \{\epsilon\}$ and there is

a sequence of configurations $C_0, C_1, \dots C_t$

- $C_0 = (q_0^{m_0}, \epsilon)$
- $C_t = (q, \epsilon)$ where $q \in F_{m_0}$

$\forall i = 0, \ldots t-1$

$C_i \xrightarrow{y_{i+1}} C_{i+1}$

$C_i = (q, s)$

$q \in Q_m$

return

$y_{i+1} = \epsilon$

$q \in F_m$

$C_{i+1} = (q', s')$

where $s = q's'$

internal

$q' \in \delta_m(q, y_{i+1})$

$C_{i+1} = (q', s)$

call

$q' \in \delta_m(q, m')$

$y_{i+1} = \epsilon$

$C_{i+1} = (q_0^{m'}, q's)$

The language accepted by an RA A,

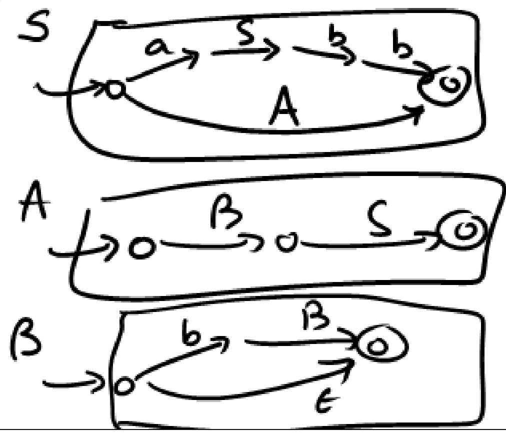
$\mathcal{L}(A) = \{ w \mid A \text{ accepts } w \}$

CFGs $\hookrightarrow$ RAs.

$$S \to a S b b \mid A$$

$$A \to B S$$

$$B \to b B \mid \epsilon$$



General construction:

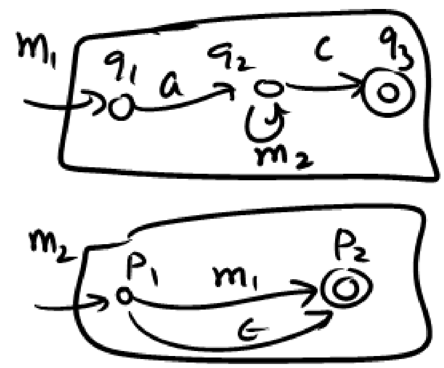
- Create a procedure for each variable.
- Create the state-machine for module V as any NFA accepting all the words w such that $V \to w$ is a rule.

RAs $\hookrightarrow$ CFGs

Variables: one for each state

$$X_q$$

Start variable: X_{q_1}



$$X_{q_1} \to a X_{q_2}$$

$$X_{q_2} \to X_{p_1} X_{q_2} \mid c X_{q_3}$$

$$X_{q_3} \to \epsilon$$

$$X_{p_1} \to X_{q_1} X_{p_2} \mid X_{p_2}$$

$$X_{p_2} \to \epsilon$$

RAs to CFGs.

Let $A = (M, m_0, \{(Q_m, \Sigma \cup M, \delta_m, q_0^m, F_m)\})$ be a RA.

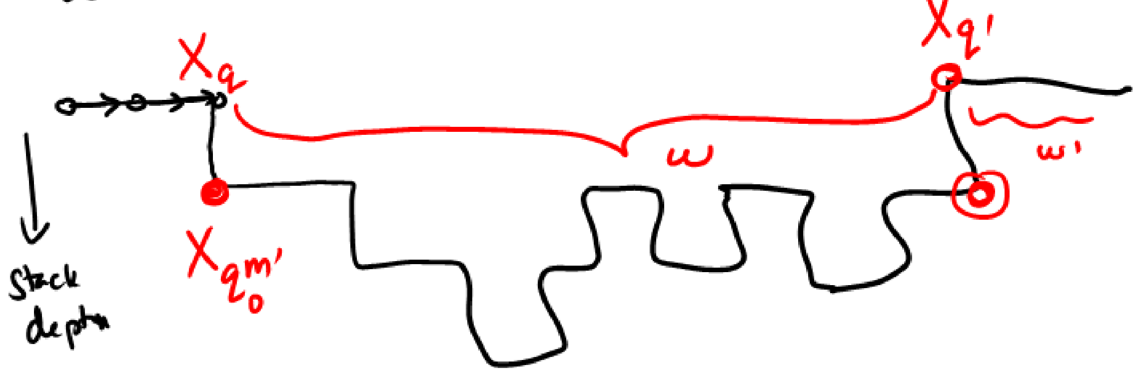
Let CFG $G = (V, \Sigma, R, S)$

where $V = \{X_q \mid q \in Q_m, m \in M\}$

$S = X_{q_0^{m_0}}$

$$R = \begin{cases} X_q \longrightarrow a\, X_{q'} & q' \in \delta_m(q, a), a \in \Sigma \cup \{\epsilon\} \\ X_q \longrightarrow X_{q_0^{m'}} X_{q'} & q' \in \delta_m(q, m') \\ X_q \longrightarrow \epsilon & q \in \bigcup_{m \in M} F_m. \end{cases}$$

$w \in L(\mathcal{A})$



$$X_q \rightarrow X_{q_0^{m'}} \, X_{q'}$$

CFLs $\equiv$ RA.

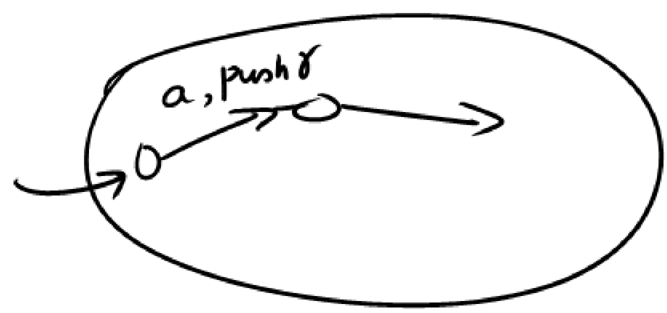
RA are not closed under intersection.

RA_1 and RA_2

You can't simulate both RA_1 and RA_2 together.

RAs can't accept $\{a^n b^n c^n \mid n \in \mathbb{N}\}$

Pushdown automata



Pushdown automata $\equiv$ CFLs $\equiv$ RAs.