Comp 330 - Lecture 6 - September 19th 2023

Silly announcements
- → Granola bar OH
- → Italian expression : Basta la pasta
  - → Enough pasta!
  - → Stop being silly!

Serious announcements
- → Note $L_{DFA} = L_{NFA} = L_{NFA+\varepsilon}$
- → Releasing A2 early : Today

Closure properties of regular languages

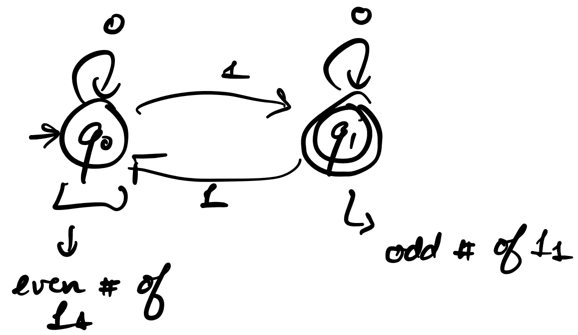
$$
\begin{array}{c}
\text{REG} \\
\updownarrow \\
\text{DFA} \leftrightarrow \text{NFA} \leftrightarrow \text{NFA}+\varepsilon
\end{array}
$$

NEW! Closure properties :
- → Show that L' is regular
- → Apply operator Θ to L s.t. $L' = \Theta(L)$
- → If Θ "preserves regularity", then L' is regular

Ex Recall language $L = \{ w \in \{0,1\}^* :$
$w \text{ has odd \# 1s} \}$

We know L is regular since

M

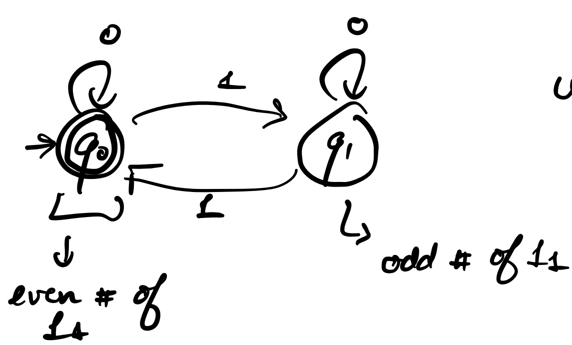


$w_0 = 100$
accepted
$w_1 = 101$
rejected

Prove that $L' = \{ w \in \{0,1\}^* : w \text{ has even \# of 1s} \}$

Notice that $L' = \overline{L}$

M'



s.t.
$L(M') = \overline{L}$

$w_0 = 100$
rejected
$w_1 = 101$
accepted

Exercise (For you) This is not true for NFA.

Claim Given some alphabet Σ, if $L \subseteq \Sigma^*$ is regular, then $\overline{L}$ is also regular.
Reg languages are closed under the complement.

Proof Given $L \subseteq \Sigma^*$, L is regular,

$\Rightarrow$ $\exists$ DFA $M = (Q, \Sigma, \delta, s_0, F)$

s.t. $L(M) = L$.

(A) Construct DFA $M' = (Q', \Sigma, \delta', s', F')$

s.t. $L(M') = \overline{L}$

$$\begin{cases} Q' := Q \\ \delta' := \delta \\ s' := s_0 \\ F' := \overline{F} = Q - F \end{cases}$$

(B) Show correctness : $L(M') = \overline{L} = \overline{L(M)}$

Suppose $x \in \Sigma^*$.

$x \in L(M') \iff \delta'^*(s', x) \in F'$

Replacing $\iff \delta^*(s_0, x) \in \underline{Q - F}$ not an accept state

$\iff \delta^*(s_0, x) \notin F$

$\iff x \in \overline{L(M)}$

Def (Closure properties of REG) Given Σ & a binary language operator $\Theta : 2^{\Sigma^*} \times 2^{\Sigma^*} \to 2^{\Sigma^*}$

We say that regular languages are <u>closed</u> under Θ if

$\forall L_1, L_2 \subseteq \Sigma^*$, if L_1 is REG AND L_2 is REG then $L_1 \ominus L_2$ is REG.

This generalizes to n-ary operators.

Ex Common reg. lang. closure properties.
Given Σ, $L_1, L_2 \subseteq \Sigma^*$, L_1, L_2 reg Then

$$\left.\begin{array}{ll} ① & L_1 \cup L_2 \\ ② & L_1 \cap L_2 \\ ③ & \overline{L_1} \\ ④ & L_1 \cdot L_2 \\ ⑤ & L_1^* \end{array}\right\} \text{REG.}$$

Proof technique: Proving that Σ REG languages are closed under some binary operator $\ominus$

① Assume $L_1, L_2 \subseteq \Sigma^*$ are regular.
$\Rightarrow \exists$ DFA M_1, M_2 s.t. $L(M_1) = L_1$ & $L(M_2) = L_2$

② Use M_1, M_2 to create DFA/NFA/NFA+ε M' s.t. $L(M') = L_1 \ominus L_2$

③ Show correctness of M' [Good way of checking M''s logic]

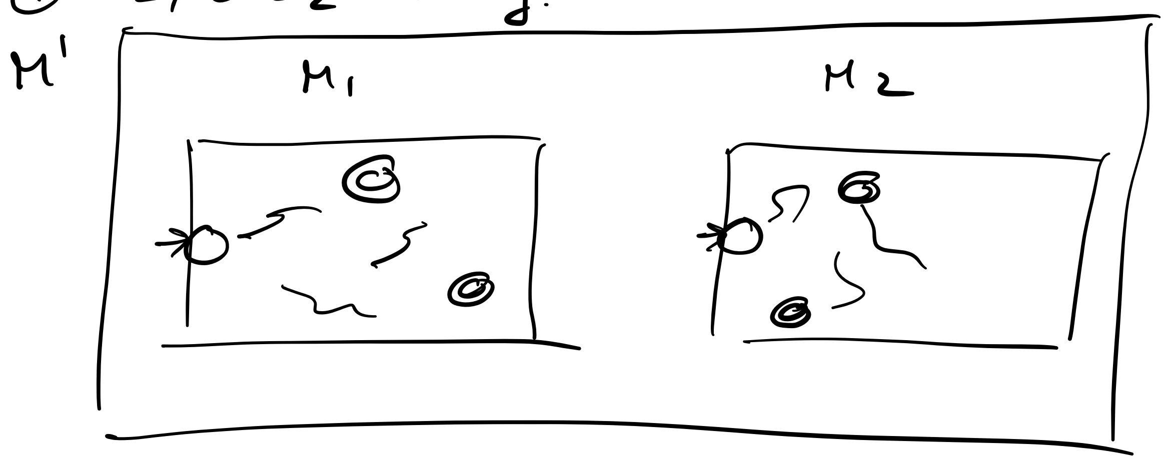
<u>Proofs</u> (Sketch) Suppose $L_1, L_2 \subseteq \Sigma^*$ are regular

$\Rightarrow \underset{\text{DFA}}{\exists}$ $M_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$

$M_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$

$\Rightarrow L(M_1) = L_1$ & $L(M_2) = L_2$

① $L_1 \cup L_2$ is reg.



s.t. $L(M') = L_1 \cup L_2$

Build NFA $M' = (Q', \Sigma, \Delta', S', F')$

$Q' = Q_1 \cup Q_2$

$S' = \{s_1, s_2\}$

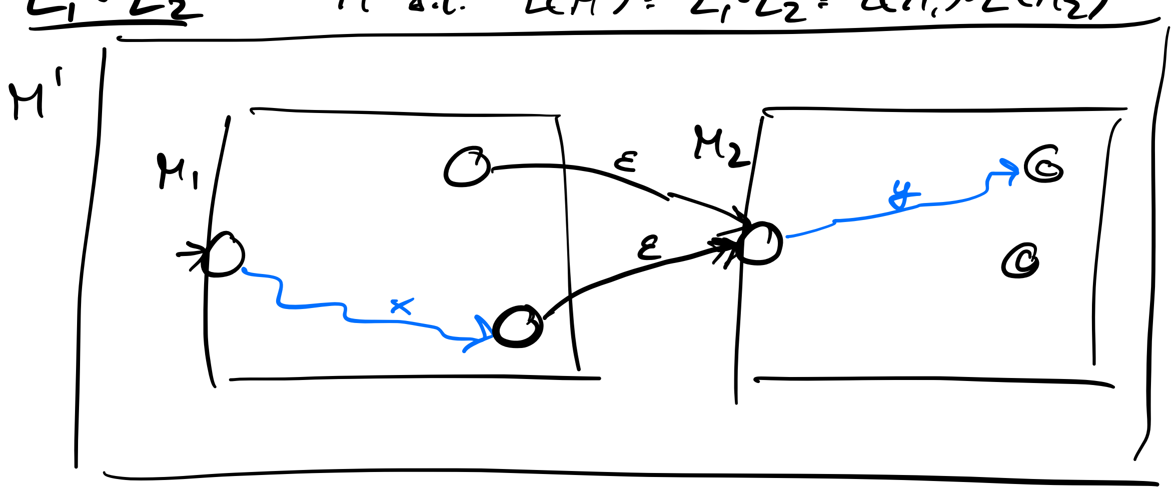
$F' = F_1 \cup F_2$

$$\Delta'(q, a) = \begin{cases} \{\delta_1(q,a)\} & \text{if } q \in Q_1 \\ \{\delta_2(q,a)\} & \text{if } q \in Q_2 \end{cases}$$

Remark REG languages are closed under $\cap$

$$\overline{\overline{L_1} \cup \overline{L_2}} = L_1 \cap L_2$$

$\underline{L_1 \cdot L_2}$ $\quad M'$ s.t. $L(M') = L_1 \cdot L_2 = L(M_1) \cdot L(M_2)$



$$x \in L_1 \ , \ y \in L_2 \Rightarrow xy \in L_1 L_2$$

Construct NFA+ε $N' = (Q', \Sigma, \boxed{\varepsilon}, \Delta', S', F')$

$\varepsilon \notin \Sigma$

$Q' := Q_1 \cup Q_2$

$S' := \{s_1\}$

$F' := F_2$

$$\Delta'(q,a) = \begin{cases} \{\delta_1(q,a)\} & \text{if } q \in Q_1,\ a \in \Sigma \\ \{s_2\} & \text{if } q \in F_1,\ a = \varepsilon \\ \{\delta_2(q,a)\} & \text{if } q \in Q_2,\ a \in \Sigma \end{cases}$$

Exercise (For you) L_1^* $\rightarrow$ [Post extra video]

Remark: Closure properties are not bidirectional

Ex: if $L_1 \cup L_2$ is REG $\not\Rightarrow$ L_1, L_2 are REG

Why? $L_1 = \{a^n b^n : n \geq 0\} \rightarrow$ Not REG

$$\underbrace{L_1}_{\text{Not REG}} \cup \{a,b\}^* = \underbrace{\{a,b\}^*}_{\text{REG}}$$

In fact, if $L_1 \cup L_2$ REG $\not\Rightarrow$ L_1 is REG or L_2 is REG.

Ex: Prove, using closure properties, that L is reg.

$$L = \{(ab)^n (ba)^m : n, m \geq 0\} \cup \{(aba)^n : n \text{ is odd}\}$$

Non-standard closure properties

Ex: Suppose L is regular, $\Sigma = \{a, b\}$, show that following lang is REG

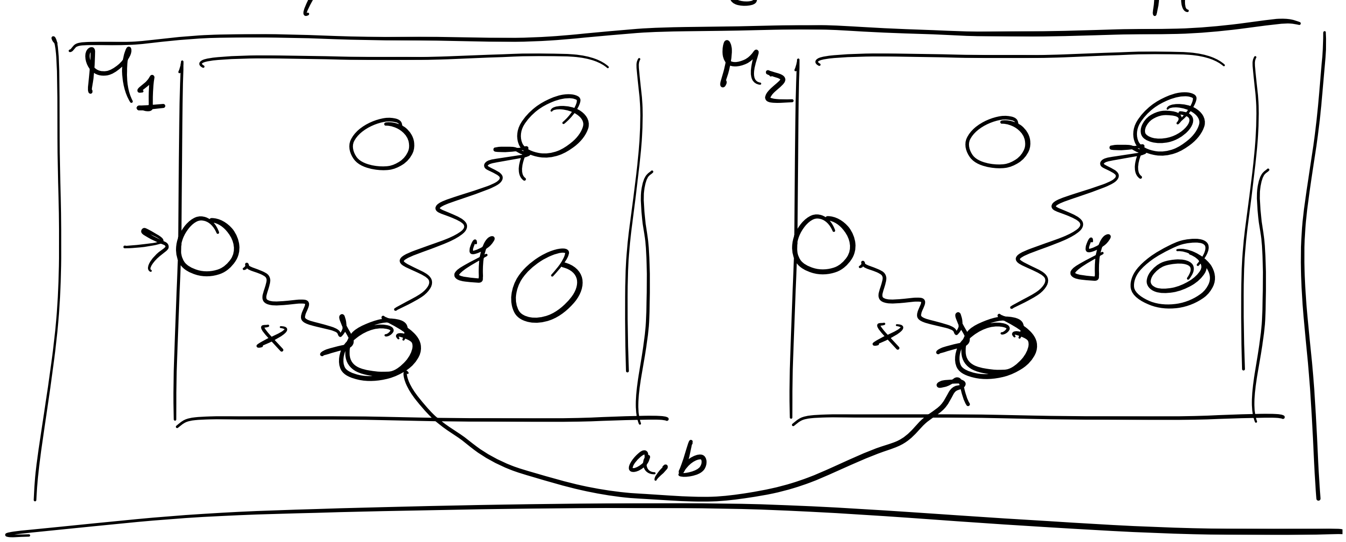
$$add1(L) = \{x\sigma y : \sigma \in \Sigma, \exists x, y \in \Sigma^*, xy \in L\}$$

↳ strings in add1(L) are strings from L with an added letter at any position

$\Sigma = \{a, b\}$, $aa \in L$, $baa, aba, aab \in add1(L)$

Sol Since L is reg, we know $\exists$ DFA $M = (Q, \Sigma, \delta, s, F)$ s.t. $L(M) = L$

$xy \in L$, $x\sigma y \in add1(L)$



NFA $M' = (Q', \Sigma, \Delta', S_0, F')$

$Q' := Q \times \{1, 2\}$ → 1,2 will act as an index to M_1 or M_2

$S_0 := \{(s, 1)\}$

$\sigma \in \Sigma$

$F' := F \times \{2\}$

$$\Delta'((q, n), \sigma) = \begin{cases} \{(\delta(q,\sigma), 1), (q, 2)\} & \text{if } q \in Q, n = 1 \\ \{(\delta(q,\sigma), 2)\} & \text{if } n = 2 \end{cases}$$

$q \in Q$, $n \in \{1,2\}$

<u>Ex</u> Suppose L_1, L_2 are regular, $\Sigma = \{a, b\}$, show the following is regular

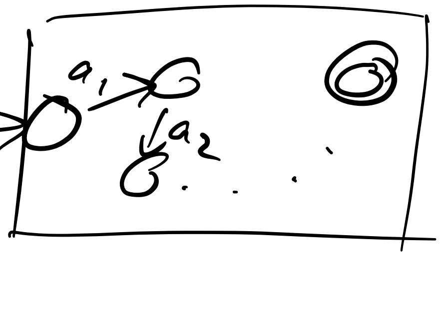
$$\text{shuffle}(L_1, L_2) = \{a_1 b_1 a_2 b_2 \cdots a_n b_n : a_1 \ldots a_n \in L_1,\ b_1 \ldots b_n \in L_2,\ a_i, b_i \in \Sigma\}$$

$aaa \in L_1$ $\qquad$ $bbb \in L_2$

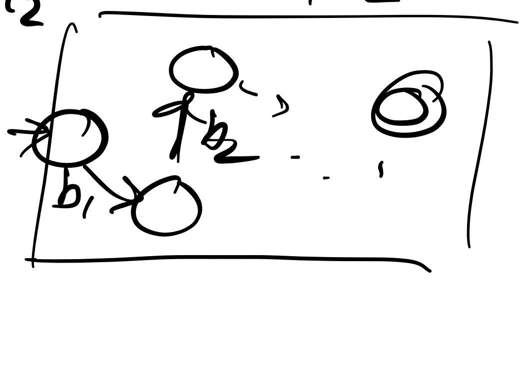
$a\ b\ a\ b\ a\ b \in \text{shuffle}(L_1, L_2)$

$M_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$ $\qquad$ $M_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$

M_1



M_2



$x = a_1 \dots a_n \in L_1$ $\qquad$ $y = b_1 \dots b_n \in L_2$

M': Desiderata $\Rightarrow$ Given

shuffle$(xy) = a_1 b_1 a_2 b_2 \dots a_n b_n$

$M' = (Q', \Sigma, \Delta', S_0, F')$

$$Q' : (\underset{\in Q_1}{p}, \underset{\in Q_2}{q}, \underset{\substack{\text{index} \\ \{1,2\}}}{n})$$

$Q' = Q_1 \times Q_2 \times \{1,2\}$

$S_0 = \{(s_1, s_2, 1)\}$

$$\Delta((p,q,n), \sigma) = \begin{cases} \{(\delta_1(p,\sigma), q, 2)\} & \text{if } n=1 \\ \{(p, \delta_2(q,\sigma), 1)\} & \text{if } n=2 \end{cases}$$

$F' = F_1 \times F_2 \times \underbrace{\{1\}}$

Check why!

(if $a \in L_1$, $b \in L_2$

Then

$$(s_1, s_2, 1) \xrightarrow{a} (q_1, s_2, 2) \xrightarrow{b} (q_1, q_2, 1)$$

$q_1 = \delta(s_1, a)$

$q_2 = \delta(s_2, b)$

q_1 & q_2 are both accept states, end in M_1.