CSE322 Theory of Computation (L6)

Recap of last lecture

L is regular if ∃ DFA D s.t.
L = L(D)

$\hat{\delta}(q, xy) = \hat{\delta}(\hat{\delta}(q, x), y)$

Today

Regular Operations

Closure Properties

Closure

Regular language is closed under operation "o":
if for all regular L, o(L) is regular.

Regular language is closed under operation "o":
if for all regular L1 and L2, o(L1, L2) is regular.

if A is regular, B is regular, then A ∪ B is regular.

Are rationals closed under division?

Are real numbers closed under square-root?

L = { starting with 0 } ∪ { ending with 1 }

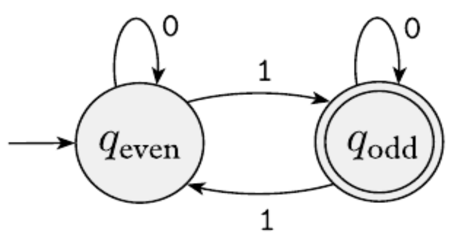
Closure under Union

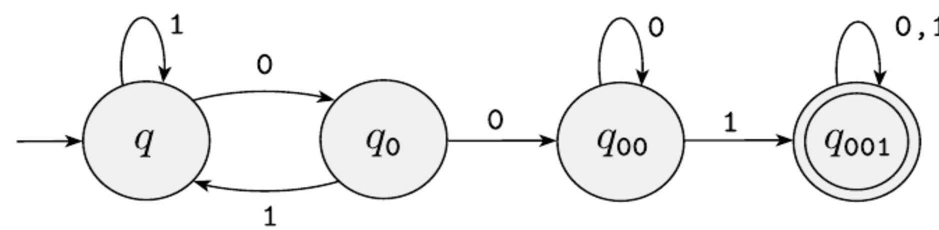
$x \in A \cup B \to x \in A$ or $x \in B$

If A and B are regular, then $A \cup B$ is regular.

Proof by constructing a product automaton.

$M1 = \langle Q1, \Sigma, \delta 1, q1, F1 \rangle$ $\qquad$ $M2 = \langle Q2, \Sigma, \delta 2, q2, F2 \rangle$





Claim:- $\hat{\delta}((r,s),w) = (\hat{\delta}_1(r,w), \hat{\delta}_2(s,w))$

Proof:- Exercise (induction on $|w|$)

$M = \langle Q, \Sigma, \delta, q, F \rangle$

$Q = Q_1 \times Q_2 = \{ (q_{even}, q), (q_{even}, q_0), (q_{even}, q_{00}), (q_{even}, q_{001}), (q_{odd}, q), (q_{odd}, q_0), \ldots (\quad), (\quad) \}$

Σ

$\delta : (Q_1 \times Q_2, \Sigma) \longrightarrow (Q_1 \times Q_2)$ $\qquad$ $\delta((r,s),b) = (\delta_1(r,b), \delta_2(s,b))$

$q = (q_{even}, q)$

$F = (F_1 \times Q_2) \cup (Q_1 \times F_2)$

use to prove

To show that $\hat{\delta}((r,s),w) \in F$ iff $\hat{\delta}_1(r,w) \in F_1$ or $\hat{\delta}_2(s,w) \in F_2$

$\equiv$ $w \in L(M)$ iff $w \in L(M_1)$ or $w \in L(M_2)$ $\equiv w \in A$ or $w \in B \equiv w \in A \cup B$

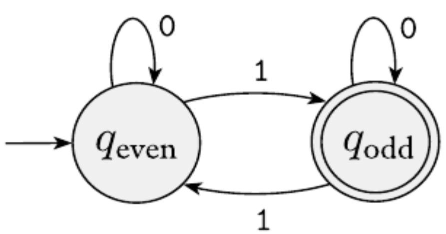
$\therefore L(M) = A \cup B$

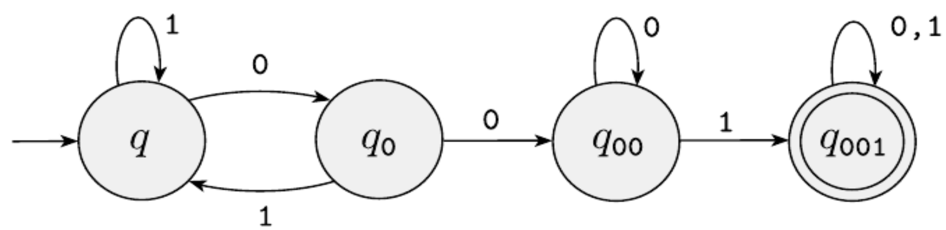
Closure under Intersection

If A and B are regular, then $A \cap B$ is regular.

Proof by constructing a <u>product automaton.</u>

$M1 = \langle Q1, \Sigma, \delta 1, q1, F1 \rangle$ $\quad$ $M2 = \langle Q2, \Sigma, \delta 2, q2, F2 \rangle$





M accepts x iff
M1 accepts x <u>and</u>
M2 accepts x

$M = \langle Q, \Sigma, \delta, q, F \rangle$

$\hat{\delta}((r,s),w) \in F$ iff $\hat{\delta}_1(r,w) \in F_1$ and $\hat{\delta}_1(s,w) \in F_2$

Q

Σ

δ

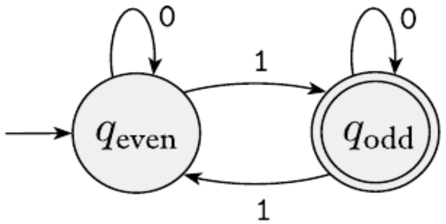
q

$F = F_1 \times F_2$

Closure under Complement

L

$\bar{L} = \{w \mid w \notin L\}$

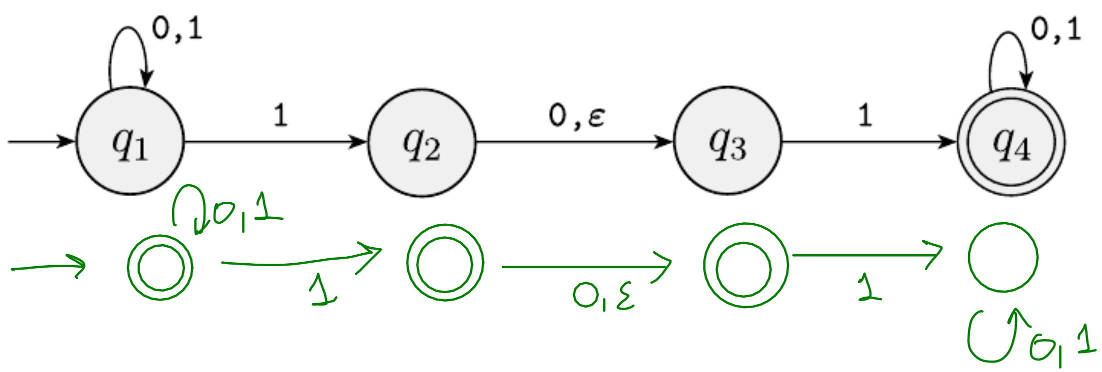


M, $L = \{\#(1,w) \text{ is odd}\}$

N $\bar{L} = \{\#(1,w) \text{ is even}\}$

N accepts x iff M doesn't accepts x

↳ copy of M with final states reversed

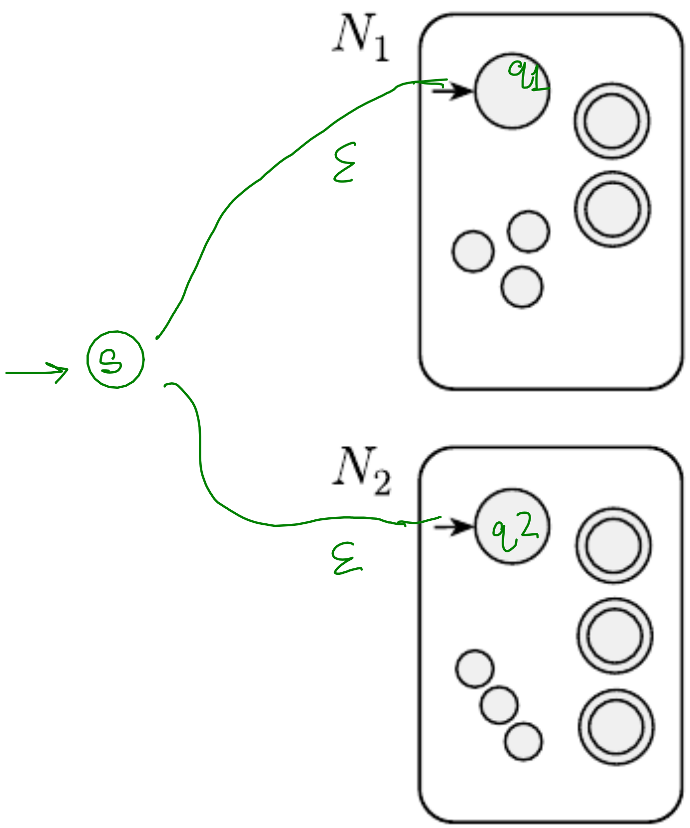


$L = \{w : w \text{ contains } 101 \text{ or } 11\}$

Think!

Doesn't reject 101.

Closure under Union (using NFA)



L1 N1 = <Q1, Σ1, δ1, q1, F1>

L2 N2 = <Q2, Σ2, δ2, q2, F2>

M = <Q, Σ, δ, q, F> accepting L1 ∪ L2

Q

Σ

δ

q

F

Exercise: L(M) = L1 ∪ L2

L = w starts with 0 and has odd length, or starts with 1 and has even length

Prove that L is regular.

$L = (L1 \cap L2) \cup (L3 \cap L4)$

L1 = {w starting with 0} Construct M1 & show L(M1) = L1

L2 = { w has odd length} Construct M2 & show L(M2) = L2

$L3 = \overline{L1}$

$L4 = \overline{L2}$

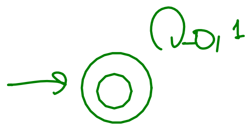
Closure under Intersection (alt.)

Proof using other closure properties.

$$L1 \cap L2 = \overline{\overline{L1} \cup \overline{L2}}$$

$$L1 \cap L2 \subseteq L1 \cup L2$$

Σ^* is regular.

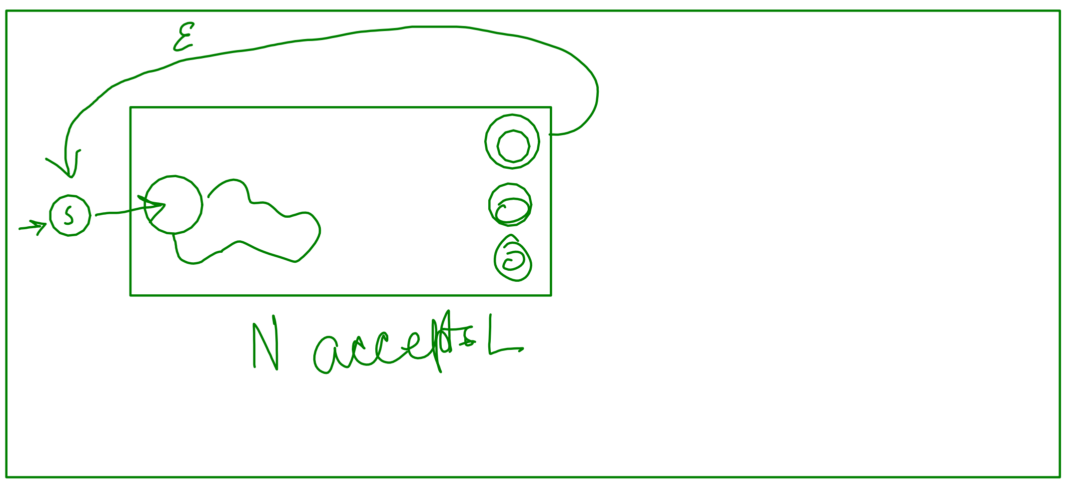


~~w contains exactly 2 A and at least 2 B~~

Closure under Kleene Star & Kleene Plus

$$L^* = \bigcup_{k \geq 0} L^{(k)} = \{ w \mid w = w_1 \cdot w_2 \cdots w_j \text{ where } w_i \in L \ \forall i = 1 \ldots j \text{ for some } j \geq 0 \}$$

$$L^+ = \bigcup_{k \geq 1} L^{(k)} = L \cdot L^*$$

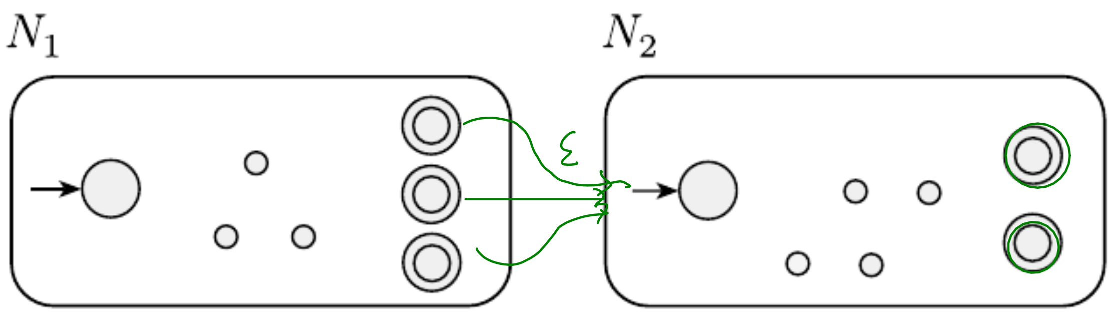


N' accepts L*

Closure under Concatenation

If L1 is regular, L2 is regular, then L1·L2 is regular.

$= \{ w : w$ can be written as $w = w_1 \cdot w_2$ for $w_1 \in L_1, w_2 \in L_2 \}$



$N1 = \langle Q1, \Sigma 1, \delta 1, q1, F1 \rangle$

$N2 = \langle Q2, \Sigma 2, \delta 2, q2, F2 \rangle$

$M = \langle Q, \Sigma, \delta, q, F \rangle$

Q

Σ

δ

q

F

Take DFA D. Let $X = \{ q : \text{there exists some } x \in \Sigma^* \text{ such that } d'(q0,x) = q\}$

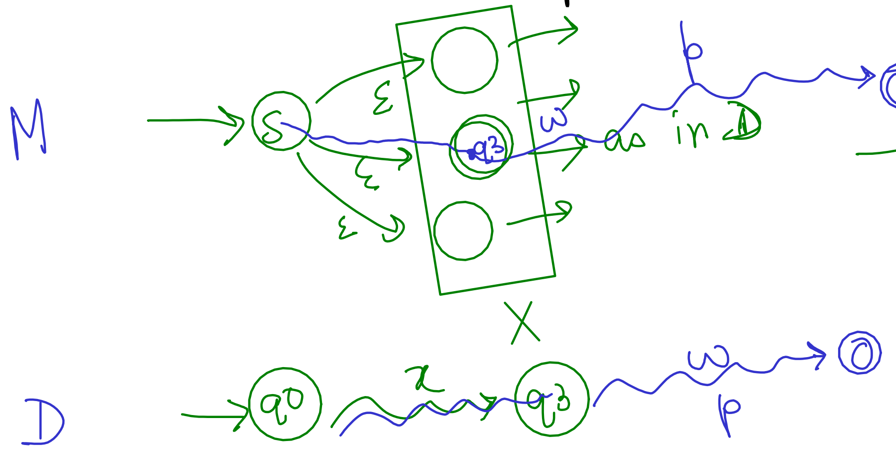
(D $\rightarrow L$; M is an NFA)

Construct $M = \langle Q \cup \{s\}, S, d1, s, F\rangle$

$q0 \overset{x}{\rightsquigarrow} q$

where $d1(q,a) = \{d(q,a)\}$ & $d1(s,e) = X$

Show that: if M accepts x then there exists y s.t. D accepts yx.



If w is accepted by M then $\exists x$ s.t. xw is accepted by D & vice versa.

$L(M) = \{ w \mid \exists x \text{ s.t. } xw \in L(D)\}$

Exercise: Show that if D accepts yx then M accepts x.

given: w accepted by M $\equiv \hat{\delta}_1(s,w) \cap F \neq \emptyset$

To show: $\exists x$ s.t. $\hat{\delta}(q0,xw) \in F$

$\hat{\delta}_1(s,w) = F'$

$\hat{\delta}_1(s,w) = \hat{\delta}_1(s,\varepsilon w)$

$q \overset{x}{\rightsquigarrow} q_1 \overset{y}{\rightsquigarrow} q_2$

$\hat{\delta}(\hat{\delta}(q,x),y) = \hat{\delta}(q,xy)$ true for DFA

NFA: $\hat{\delta}(q,xy) = \bigcup_{r \in \hat{\delta}(q,x)} \hat{\delta}(r,y)$

Given a regular language L, show that HL is a regular language.

HL = { x : xx is in L }

On any input string w, N would try to simulate D on ww. N does so by first guessing r - the state D would be at if given input w (N uses non-determinism to guess r). Then N parallely runs D on w starting at D's starting state, and running D on w starting at r. It may be helpful to let N "remember" r in its state after guessing it.

Let DFA D=<Q,S, ,q0,F> accept L.

Construct NFA N=<Q1, S, d1, q01, F1> where

Q1=QxQxQ U {s} where s is a new state not in Q,

q01=s,

F1={(r,r,t) : r in Q, t in F}

d1(s,e) = {(q0,r,r) : r in Q }

d1(s, symbol a) = { } // no-op

d1((r1, r2, t), e) = { } // no-op (since DFA D has no e-moves)

d1((r1, r2, t), a) = {(d(r1,a), r2, d(t,a)) }

Claim: If D accepts u=ww then N accepts u.

Claim: If N accepts w then D accepts ww.