CSE396 Lecture Tue Feb 23 Spring 2016 ①

⓪ KWR ofc hrs today $4^{(?)}$–6pm, long Dept. meeting blocks 1–2pm

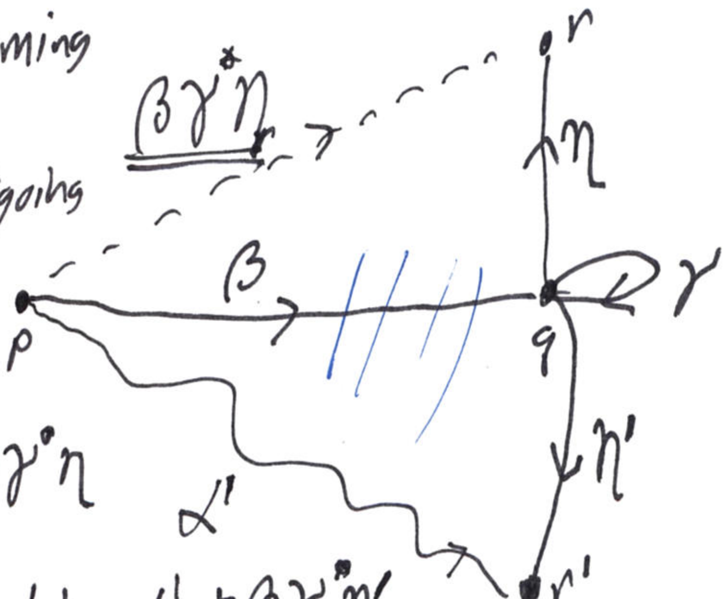
① **Theorem**: For every finite automaton (DFA, NFA, GNFA...) M, we can compute a regular expression R such that $\boxed{L(R) = L(M)}$.

Key Idea:

We can eliminate any nonaccepting state q different from the start state by bypassing every incoming edge (p, β, q) to every outgoing edge (q, η, r), thereby updating the entry (p, r).

Eliminate q:

For each incoming (p, β, q) {
  For each outgoing (q, η, r) {



$T(p,r) \mathrel{+}= \beta\gamma^*\eta$
  }
}

Update: $\alpha' + \beta\gamma^*\eta'$

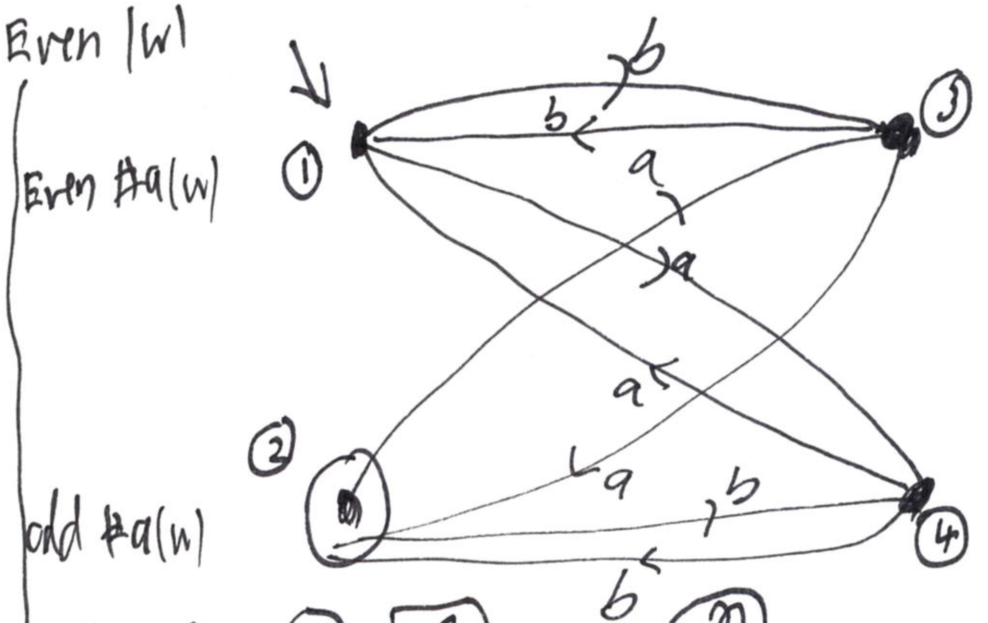
Since $q \notin F$, any processing that comes in to q must come out at r or r'... Hence we can provide the same processing capability directly. Then delete edge (p,q)

delete q;

- If M gets down to two states by repeated eliminations, then we can give $L(M) =: R$ via the two-state basis. (shown last Thu).
- If not, then add a new final state f with ε-arcs from all old final states

Example: Answer to HW2 1(g). First DFA M:

$L = \{x \in \{a,b\}^* : |x| \text{ is even} \wedge \#a(x) \text{ is odd}\}$.



odd |w|

Only one acc. state q ($\neq s$). Hence no need to add ε arcs.

Eliminate 4
Eliminate 3
$R = L(M) = L_{12}$.

Eliminate (4):

β Incoming State	Regexp Char	γ Self	η Outgoing Regexp	State
1	a	ε	a	1
			b	2
2	b	ε	a	1
			b	2

Since we will use γ^*, and $\emptyset^* = \varepsilon^* = \varepsilon$ it doesn't matter if an empty "self" is called $\emptyset$ or ε.

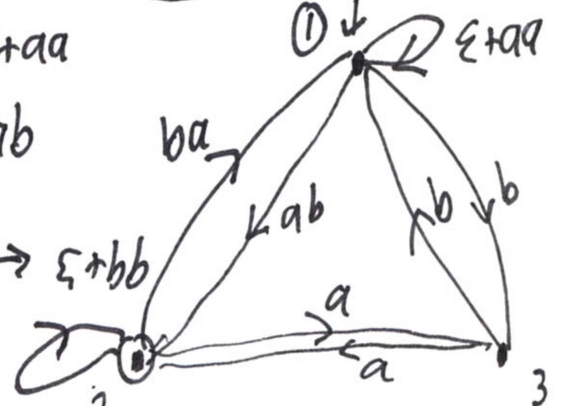
$\therefore$ Update $T(1,1)$, $T(1,2)$, $T(2,1)$ and $T(2,2)$.

Old $T(1,1) = \varepsilon$ New $T(1,1) + a\varepsilon^* a = \varepsilon + aa$

old $T(1,2) = \emptyset$ New: $T(1,2) + a\varepsilon^* b = ab$

old $T(2,1) = \emptyset$ New: $b\varepsilon^* a = ba$

old $T(2,2) = \varepsilon$ New: $T(2,2) += b\varepsilon^* b \to \varepsilon + bb$



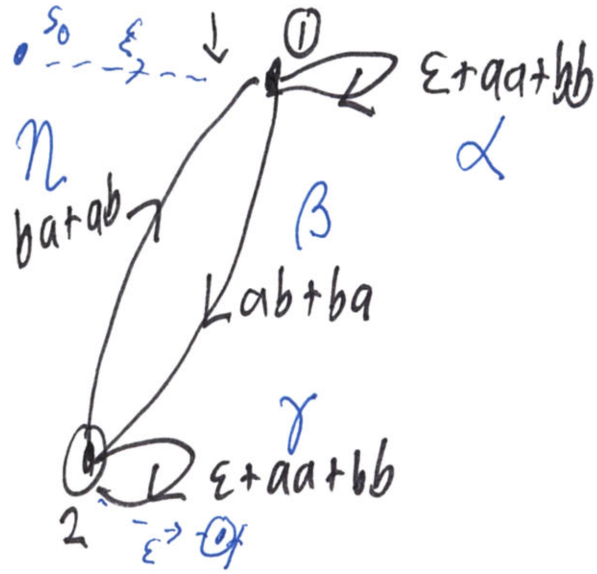
Eliminate 3: Incoming 1, 2 Outgoing 1, 2

$T(1,1) += \underset{b}{T(1,3)}\ \underset{\varepsilon}{T(3,3)^*}\ \underset{b}{T(3,1)}$

$\therefore$ new $T(1,1) = \text{old} + bb = \varepsilon + aa + bb$

$T(1,2) += \underset{b}{T(1,3)}\ \underset{\varepsilon}{T(3,3)^*}\ \underset{a}{T(3,2)}$

new $T(1,2) =$ $\underline{ab + ba}$ -



$$\text{new } T(2,1) = \text{old } T(2,1) + T(2,3)\,T(3,3)^*\,T(3,1)$$
$$= ba + a\cdot\varepsilon\cdot b$$
$$= ba+ab$$

$$\text{new } T(2,2) = \text{old } T(2,2) + T(2,3)\,T(3,3)^*\,T(3,2)$$
$$= \varepsilon+bb + a\cdot\varepsilon\cdot a$$
$$= \varepsilon + bb + aa.$$

From the Two-State Base Case

$$L(M) = L_{12} = (\alpha + \beta\gamma^*\eta)^*\beta\gamma^*$$

$\boxed{\text{ON HW, YOU MAY STOP HERE}}$

All laps until last time thru 1

$$= \text{New } T(1,2) = \big(T(1,1) + T(1,2)\,T(2,2)^*\,T(2,1)\big)^*\,\underbrace{T(1,2)\,T(2,2)^*}_{\text{go to finish.}}$$

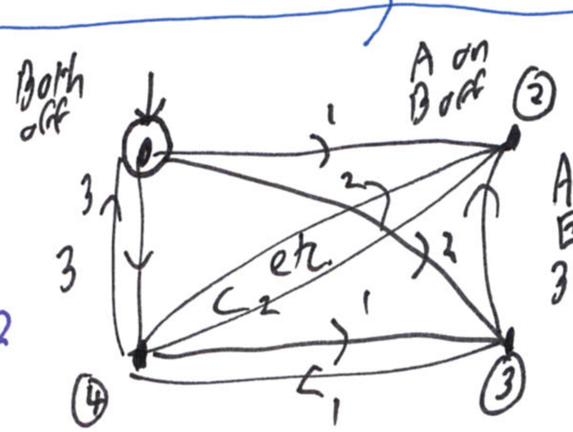
$$T(2,2)^* = (\varepsilon+aa+bb)^* = (aa+bb)^*$$

$$\big[\varepsilon+aa+bb+(ab+ba)(aa+bb)^*(ab+ba)\big]^*(ab+ba)(aa+bb)^*.$$

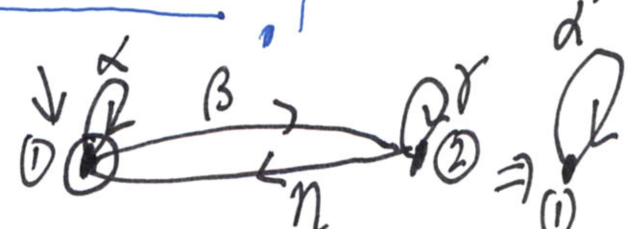
$\boxed{\text{THERE IS NO OBLIGATION TO SIMPLIFY THESE MONSTROSITIES}}$

$$= \big(aa+bb+(ab+ba)(aa+bb)^*(ab+ba)\big)^*(ab+ba)(aa+bb)^*$$

Sketch Example 2



After Elim 3 & 4 :



$$L(R) = \text{final } L_{11} =$$
$$\alpha' = \big(T(1,1) + T(1,2)\,T(2,2)^*\,T(2,1)\big)^*.$$

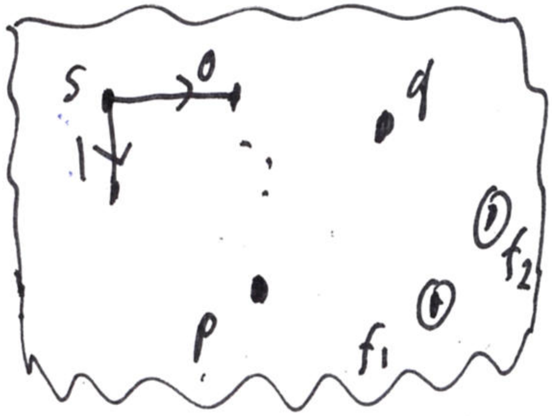
$L = L_{11} \cup L_{12}$ ← $\boxed{\text{ON HW, YOU MAY STOP HERE}}$

Answer $= (\alpha')^*$.

III §1.4: Now we say things that only work for DFAs.

Key Idea: Suppose we have a DFA M that processes x from s to some state p, and processes $y \neq x$ from s to some state q. (When) Can $p = q$?

M



Suppose there is some string $z \in \Sigma^*$ such that $xz \in L(M)$ but $yz \notin L(M)$ or vice-versa $xz \notin L(M) \wedge yz \in L(M)$. Then $p \neq q$.

I.e. Suppose L is a language and strings x and y $(x \neq y)$ are such that for some $z \in \Sigma^*$: $(xz \in L \wedge yz \notin L) \vee (xz \notin L \wedge yz \in L)$

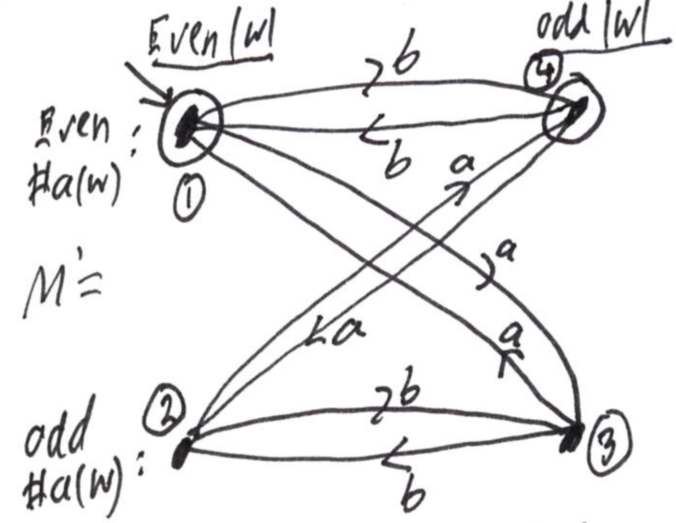
$\equiv$ $xz \in L$ XOR $yz \in L$

i.e. $L(xz) \neq L(yz)$

Then any DFA M such that $L(M) = L$ (if M exists at all) must process x and y to different states p and q.

$\therefore$ If we have a set S of K strings such that for all distinct $x, y \in S$ there is a z s.t. $L(xz) \neq L(yz)$, then any DFA M such that $L(M) = L$ must process them all to different states. $\therefore$ M must have at least K states. Thu: $K \to \infty$!

Extra: One More Example. Same arcs as PS2 1(g) answer from lecture, but different accepting states. (5)



Since there is only one accepting state different from start, we can get away with avoiding the text's step of adding a new final state with ε-arcs from old ones. Eliminating ② & ③ will be enough.

Elim ③:

	Inc.	Self	Out
	① a	ε	a ①
			b ②
	② b	ε	a ①
			b ②

$T(1,1)_{new} = T(1,1)_{old} + a\varepsilon^* a = \varepsilon + aa$ (Since self-loop, or $= aa$ is OK)

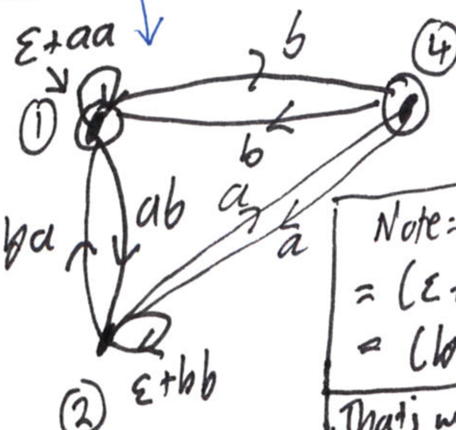
$T(1,2)_{new} = T(1,2)_{old} + a\varepsilon^* b = \emptyset + ab = ab$

$T(2,1)_{new} = T(2,1)_{old} + b\varepsilon^* a = \emptyset + ba = ba$

why not $\varepsilon + ba$? Because ① → to → ② is not a self-loop.

$T(2,2)_{new} = \varepsilon + bb$

Next step is case with self ≠ ε



Elim ③:

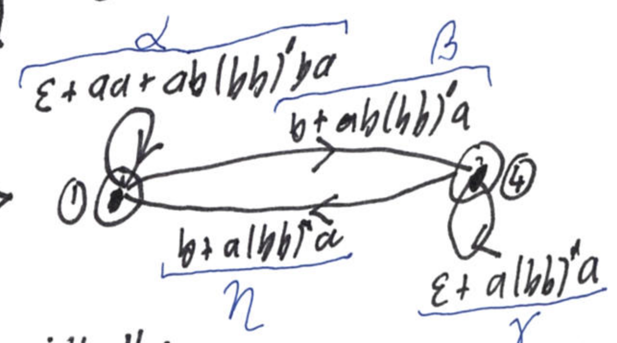
	Inc	Self	Out
	① ab	ε+bb	ba ①
			a ④
	④ a	ε+bb	ba ①
			a ④

∴ update all four of $T(1,1)$, $T(1,4)$, $T(4,1)$ and $T(4,4)$.

Note: $\text{self}^* = (\varepsilon + bb)^* = (bb)^*$

That's why OK to omit "ε+" in self loops

$T(1,1)_{new} = T(1,1)_{old} + T(1,2)\,T(2,2)^*\,T(2,1)$
$= \boxed{\varepsilon +} aa + ab(bb)^* ba$ ⇨



$T(1,4)_{new} = T(1,4)_{old} + T(1,2)\,T(2,2)^*\,T(2,4)$
$= b + ab(bb)^* a$

$T(4,4)_{new} = T(4,1)_{old} + T(4,2)\,T(2,2)^*\,T(2,1)$
$= b + a(bb)^* ba$

$T(4,4)_{new} = T(4,4)_{old} + T(4,2)\,T(2,2)^*\,T(2,4)$
$= \boxed{\varepsilon \quad +} \; a(bb)^* a$

OK to omit ε+ here on self-loops. Thus the algorithm often doesn't get the simplest answer!!

Resist the temptation to say "all states are accepting so $L = \Sigma^* = (a+b)^*$"! Instead

$L(M') = L_{11} \cup L_{14}$
$= L_{11} \cdot (\varepsilon + T(1,4)\,T(4,4)^*)$
$\boxed{= (\alpha + \beta\gamma^*\eta)^* \cdot (\varepsilon + \beta\gamma^*)}$

Written out in full the answer is gory long. The joke is on us, because in fact $L(M') = \{x : \#a(x) \text{ is even}\} = (b^*ab^*ab^*)^*$.