CS310M: Automata Theory (Minor)

Topic 5: Minimizing DFA

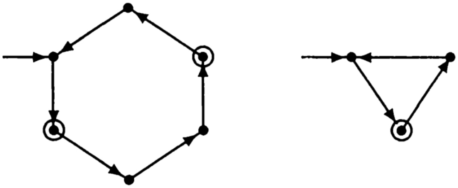
Paritosh Pandya

Indian Institute of Technology, Bombay
Course URL: https://cse.iitb.ac.in/∼pandya58/CS310M/automata.html

Autumn, 2021

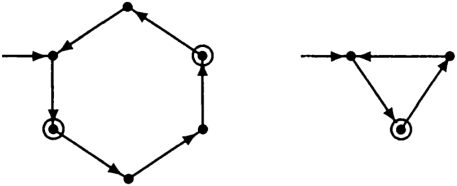
Motivation

There can be multiple DFAs recognizing the same language.



Motivation

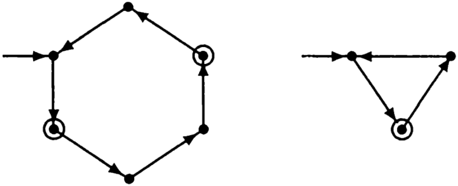
There can be multiple DFAs recognizing the same language.



- Which one is preferable? Why?

Motivation

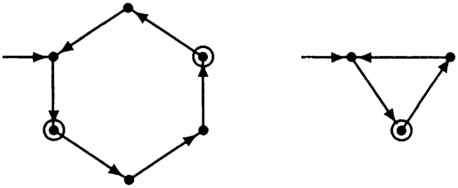
There can be multiple DFAs recognizing the same language.



- Which one is preferable? Why?
- ϵ-NFA to DFA conversion often gives rise to large automaton which may not be "optimal" in size.

Motivation

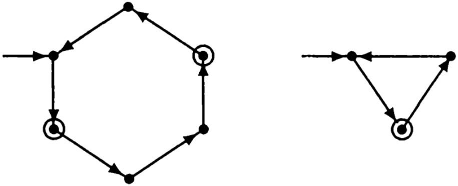
There can be multiple DFAs recognizing the same language.



- Which one is preferable? Why?
- ϵ-NFA to DFA conversion often gives rise to large automaton which may not be "optimal" in size.
- Conversion of Regular Expression to DFA gives rise to large automata which may not be "optimal" in size.

Motivation

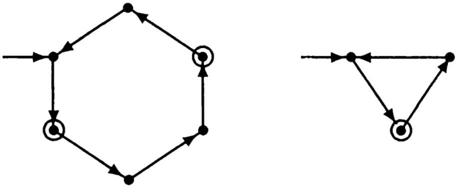
There can be multiple DFAs recognizing the same language.



- Which one is preferable? Why?
- ϵ-NFA to DFA conversion often gives rise to large automaton which may not be "optimal" in size.
- Conversion of Regular Expression to DFA gives rise to large automata which may not be "optimal" in size.
- Optimal : smallest in size possible for the same language – minimal sized DFA

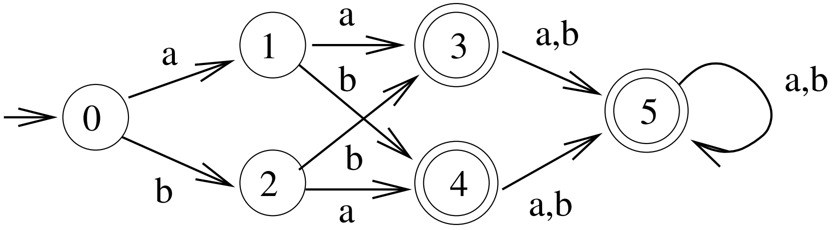
Motivation

There can be multiple DFAs recognizing the same language.

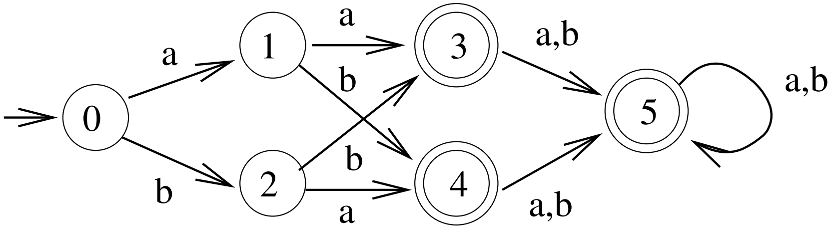


- Which one is preferable? Why?
- ϵ-NFA to DFA conversion often gives rise to large automaton which may not be "optimal" in size.
- Conversion of Regular Expression to DFA gives rise to large automata which may not be "optimal" in size.
- Optimal : smallest in size possible for the same language – minimal sized DFA
- Is there a unique minimal sized DFA? Minimal DFA

DFA Minimization



DFA Minimization



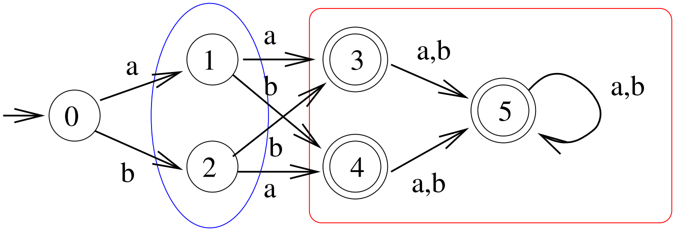
word x separates state p from state q iff

$\hat{\delta}(p, x) \in F \wedge \hat{\delta}(q, x) \notin F$ or vice versa.

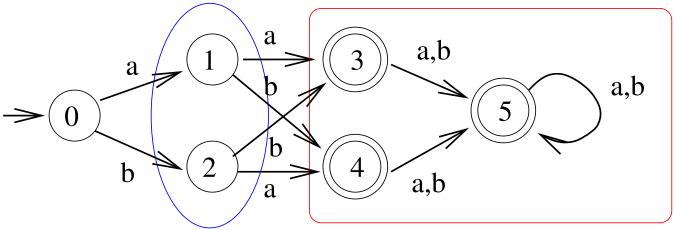
- ϵ separates (1,3)
- a separates (0,1)
- 3,4,5 cannot be separated

i.e. $3 \approx 4$, $3 \approx 5$, $4 \approx 5$

DFA Minimization



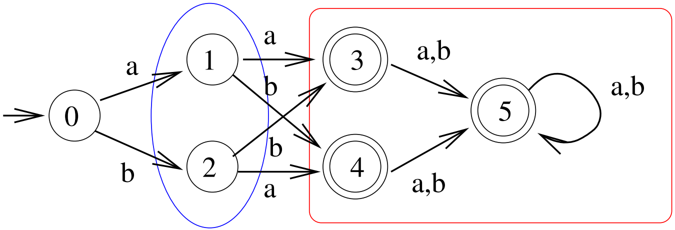
DFA Minimization



Equivalent States in a DFA

$$p \approx q \stackrel{\text{def}}{=} \forall x \in \Sigma^*. \ (\hat{\delta}(p, x) \in F \Leftrightarrow \hat{\delta}(q, x) \in F)$$

DFA Minimization

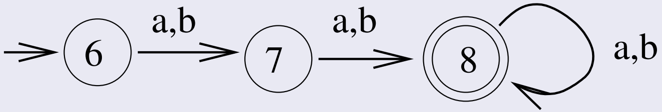


$\{0\}$
$\{1,2\}$
$\{3,4,5\}$

Equivalent States in a DFA

$$p \approx q \overset{\text{def}}{=} \forall x \in \Sigma^*.\ \ (\hat{\delta}(p,x) \in F \Leftrightarrow \hat{\delta}(q,x) \in F)$$

Equivalent Automaton



DFA Minimization (2)

$$p \approx q \quad \stackrel{\text{def}}{=} \quad \forall x \in \Sigma^*. \ \ (\hat{\delta}(p, x) \in F \Leftrightarrow \hat{\delta}(q, x) \in F)$$

DFA Minimization (2)

$$p \approx q \quad \stackrel{\text{def}}{=} \quad \forall x \in \Sigma^*. \; (\hat{\delta}(p,x) \in F \Leftrightarrow \hat{\delta}(q,x) \in F)$$

Proposition $\approx$ is an equivalence relation, i.e.
(a) $\forall p.\ p \approx p$, (b) $\forall p,q.\ p \approx q \;\Rightarrow\; q \approx p$
(c) $\forall p,q,r.\ p \approx q \wedge q \approx r \;\Rightarrow\; p \approx r$
(Exercise: Check that above properties are true.)

DFA Minimization (2)

$p \approx q \stackrel{\text{def}}{=} \forall x \in \Sigma^*. \ (\hat{\delta}(p,x) \in F \Leftrightarrow \hat{\delta}(q,x) \in F)$

Proposition $\approx$ is an equivalence relation, i.e.
(a) $\forall p.\ p \approx p$, (b) $\forall p, q.\ p \approx q \Rightarrow q \approx p$
(c) $\forall p, q, r.\ p \approx q \wedge q \approx r \Rightarrow p \approx r$
(Exercise: Check that above properties are true.)

$\approx$ partitions Q into equivalence classes.
Let $[p]$ denote the equivalence class of p.

Example The classes are $\{0\}$, $\{1,2\}$ and $\{3,4,5\}$.

Given DFA $M = (Q, \Sigma, \delta, [q_0], F)$ and $\approx$ as before, the Quotient automaton is $M/\approx \stackrel{\text{def}}{=} (Q', \Sigma, \delta', [q_0], F')$, where

$$Q' = \{[p] \mid p \in Q\}$$
$$\delta'([p], a) = [\delta(p, a)]$$
$$F' = \{[f] \mid f \in F\}$$

Quotient Automaton

Given DFA $M = (Q, \Sigma, \delta, [q_0], F)$ and $\approx$ as before, the Quotient automaton is $M/\approx \stackrel{\text{def}}{=} (Q', \Sigma, \delta', [q_0], F')$, where

$$Q' = \{[p] \mid p \in Q\}$$
$$\delta'([p], a) = [\delta(p, a)]$$
$$F' = \{[f] \mid f \in F\}$$

Well-formedness

Lemma(Congruence) $p \approx q \quad \Rightarrow \quad \forall a \in \Sigma.\ \delta(p, a) \approx \delta(q, a)$.
Lemma $p \in F \quad \Leftrightarrow \quad [p] \in F'$

Given DFA $M = (Q, \Sigma, \delta, [q_0], F)$ and $\approx$ as before, the Quotient automaton is $M/\approx \stackrel{\text{def}}{=} (Q', \Sigma, \delta', [q_0], F')$, where

$$
\begin{aligned}
Q' &= \{[p] \mid p \in Q\} \\
\delta'([p], a) &= [\delta(p, a)] \\
F' &= \{[f] \mid f \in F\}
\end{aligned}
$$

Well-formedness

Lemma(Congruence) $p \approx q \quad \Rightarrow \quad \forall a \in \Sigma.\ \ \delta(p, a) \approx \delta(q, a)$.
Lemma $p \in F \quad \Leftrightarrow \quad [p] \in F'$

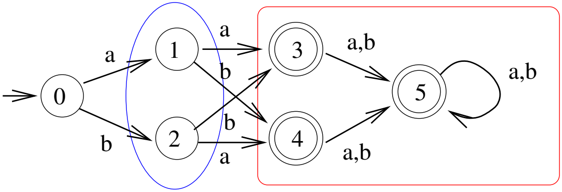
Lemma $\hat{\delta}'([p], x) \quad = \quad [\hat{\delta}(p, x)]$.
Proof: By structural induction on x,

Theorem $L(M/\approx) \quad = \quad L(M)$.

Proof:

Data Structure for ≈ relation

{0} {1,2}, {3,4,5}



	0	1	2	3	4	5
0						
1	✓					
2	✓					
3	✓	✓	✓			
4	✓	✓	✓			
5	✓	✓	✓			

Minimization algorithm

Partition Refinement Algorithm.

Algorithm [Hopcroft 1971]

1. Make pairs table with $(p, q) \in Q \times Q$ and $p \leq q$.
2. Mark (p, q) if $p \in F \wedge q \notin F$ or vice versa.
3. Repeat following steps until no change occurs.
    1. Pick each unmarked state (p, q).
    2. If $(\delta(p, a), \delta(q, a))$ is marked for some $a \in \Sigma$ then mark (p, q).
4. For each pair, $p \approx q$ **iff** (p, q) is unmarked.

Minimization algorithm

Algorithm [Hopcroft 1971]

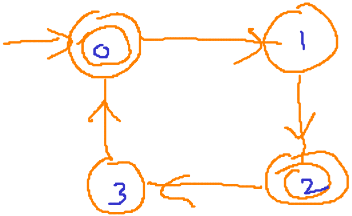
1. Make pairs table with $(p, q) \in Q \times Q$ and $p \leq q$. ✓
2. Mark (p, q) if $p \in F \wedge q \notin F$ or vice versa. ✓ Phase
3. Repeat following steps until no change occurs.
   1. Pick each unmarked state (p, q).
   2. If $(\delta(p, a), \delta(q, a))$ is marked for some $a \in \Sigma$ then mark (p, q).
4. For each pair, $p \approx q$ **iff** (p, q) is unmarked.

Termination: In each pass at least one new pair must get marked.

Theorem (p, q) is marked
iff $\exists x \in \Sigma^*. \hat{\delta}(p, x) \in F \wedge \hat{\delta}(q, x) \notin F$ or vice versa.
iff $p \not\approx q$.

Example

$\Sigma = \{a\}$ $(aa)^*$



	0	1	2	3
0	.			
1	$\checkmark_0$	.		
2		$\checkmark_0$	.	
3	$\checkmark_0$		$\checkmark_0$	.

Phase 0

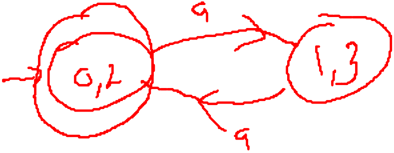
Phase 1

$(0,2) \rightarrow (1,3)$ unmark.

$(1,3) \xrightarrow{a} (0,2)$ unmarked.

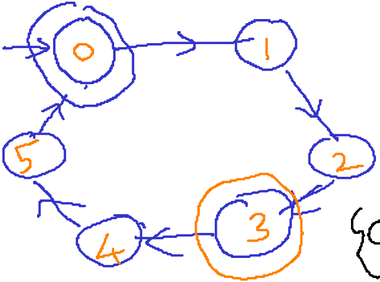
Termination.

$0 \approx 2, \quad 1 \approx 3$



Example

$L = (aaa)^*$



$\{0,3\}$
$\{1,4\}$
$\{2,5\}$

Phase 0

Phase 1

$(0,3) \xrightarrow{a} (1,4)$ unmark

$(1,4) \xrightarrow{a} (2,5)$ Unmark.

$(1,5) \xrightarrow{a} (2,0)$ marked

$(2,4) \xrightarrow{a} (3,5)$ marked

$(1,2) \xrightarrow{a} (2,3)$ marked

	0	1	2	3	4	5
0						
1	$\checkmark_0$					
2	$\checkmark_0$	$\checkmark_1$				
3		$\checkmark_0$	$\checkmark_0$			
4	$\checkmark_0$		$\checkmark_1$	$\checkmark_0$		
5	$\checkmark_0$	$\checkmark_1$		$\checkmark_0$	$\checkmark_1$	

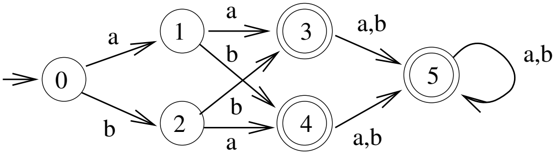
Phase 2.

$(0,3) \xrightarrow{a} (1,4)$ Unmarked.

$(1,5) \xrightarrow{a} (0,3)$ Unmarked.

$(2,5) \xrightarrow{a} (3,0)$ Unm,

Example



Minimize using the algorithm.

	0	1	2	3	4	5
0						
1						
2						
3						
4						
5						

We prove (p, q) is marked implies $\exists x \in \Sigma^*.\hat{\delta}(p, x) \in F \wedge \hat{\delta}(q, x) \notin F$ or vice versa.

Proof: By Ind. on phase number

Base step: $p \not\approx_0 q \Rightarrow \epsilon$ separates (p,q)

Ind. step $p \not\approx_{k+1} q \Rightarrow \exists a, p', q'.\ (p,q) \xrightarrow{a} (p',q')$ and $p' \not\approx_k q'$

$\exists y.\ \hat{\delta}(p', y) \in F \ \&\ \hat{\delta}(q', y) \notin F$ or V.V.

$\Rightarrow \hat{\delta}(p, ay) \in F \ \&\ \hat{\delta}(q, ay) \notin F$ or V.V.

We prove $\exists x \in \Sigma^*. \hat{\delta}(p, x) \in F \wedge \hat{\delta}(q, x) \notin F$ or vice versa.
implies (p, q) is marked

Assume x separates (p, q) & x is shortest
Proof by Ind. on $|x|$
Base step $x = \epsilon \Rightarrow p V_0 q$
Ind. step $(x = ay)$
x sep. $(p, q) \Rightarrow \hat{\delta}(p, ay) \in F$ & $\hat{\delta}(q, ay) \notin F$ or VV

$\Rightarrow \hat{\delta}(\delta(p, a), y) \in F$ & $\hat{\delta}(\delta(q, a), y) \notin F$ VV

$\Rightarrow \delta(p, a), \delta(q, a)$ is marked {Ind. Hyp}

$\Rightarrow (p, q)$ gets marked in next phase.