Algorithm for DFA minimization

Algorithm takes as input a DFA $M = (Q, \Sigma, \delta, s, F)$ and marks pairs of states (unordered)

A pair $\{p, q\}$ will be marked when the procedure detects that p and q are not equivalent.

Algorithm for DFA minimization

Algorithm takes as input a DFA $M = (Q, \Sigma, \delta, s, F)$ and marks pairs of states (unordered)

A pair $\{p, q\}$ will be marked when the procedure detects that p and q are not equivalent.

Algorithm.

1. All pairs of states $\{p, q\}$ are initially unmarked.
2. if $p \in F$ and $q \notin F$ or vice versa, mark $\{p, q\}$.
3. Repeat until no change in marking
4. if $\exists \{p, q\}$ - unmarked such that $\{\delta(p, a), \delta(q, a)\}$ is marked for some $a \in \Sigma$ then mark $\{p, q\}$.
5. $p \approx q$ iff $\{p, q\}$ is not marked.

$$\boxed{p \approx q \text{ iff } \forall x \in \Sigma^* \left(\hat{\delta}(p, x) \in F \text{ iff } \hat{\delta}(q, x) \in F\right)}$$

Algorithm.

1. All pairs of states $\{p,q\}$ are initially unmarked.
2. if $p \in F$ and $q \notin F$ or vice versa, mark $\{p,q\}$.
3. Repeat until no change in marking
4. if $\exists \{p,q\}$ - unmarked such that $\{\delta(p,a), \delta(q,a)\}$ is marked for some $a \in \Sigma$ then mark $\{p,q\}$.
5. $p \approx q$ iff $\{p,q\}$ is not marked.

Termination. There are atmost $\binom{n}{2}$ pairs.

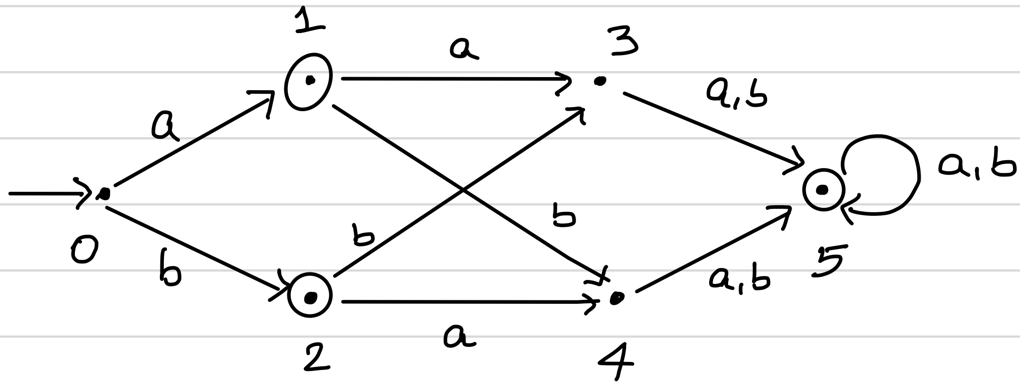
Step 4 marks at least one pair in each iteration

$\therefore$ Repeat step [line 3] can have atmost $\binom{n}{2}$ iterations.

Algorithm.

1. All pairs of states $\{p,q\}$ are initially unmarked.
2. if $p \in F$ and $q \notin F$ or vice versa, mark $\{p,q\}$.
3. Repeat until no change in marking
4. if $\exists \{p,q\}$ - unmarked such that $\{\delta(p,a), \delta(q,a)\}$ is marked for some $a \in \Sigma$ then mark $\{p,q\}$.
5. $p \approx q$ iff $\{p,q\}$ is not marked.

Example.



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

$\{0,3\} \xrightarrow{a} \{1,5\}$.
$\{0,3\} \xrightarrow{b} \{2,5\}$
$\{1,5\} \xrightarrow{a} \{3,5\}$.
$\{2,5\} \xrightarrow{a} \{4,5\}$.

Theorem. The pair $\{p, q\}$ is marked by the Algorithm iff $\exists x \in \Sigma^*$, s.t $\hat{\delta}(p, x) \in F$ and $\hat{\delta}(q, x) \notin F$ or vice versa. That is, iff $p \not\approx q$.

Proof. By induction.

Theorem. The pair $\{p,q\}$ is marked by the Algorithm iff $\exists x \in \Sigma^*$, s.t $\hat{\delta}(p,x) \in F$ and $\hat{\delta}(q,x) \notin F$ or vice versa. That is, iff $p \not\approx q$.

Proof. By induction.

Isomorphism. Two DFAs

$M = (Q_M, \Sigma, \delta_M, s_M, F_M)$ and $N = (Q_N, \Sigma, \delta_N, s_N, F_N)$

are isomorphic if there is a bijection $f: Q_M \to Q_N$ such that

1. $f(s_M) = s_N$.

2. $f(\delta_M(p,a)) = \delta_N(f(p),a) \quad \forall p \in Q_M, \forall a \in \Sigma$

3. $p \in F_M$ iff $f(p) \in F_N$.

Myhill-Nerode Theorem.

If M and N are any two DFAs with no inaccessible states s.t $L(M) = L(N)$, then $M/\approx$ and $N/\approx$ are isomorphic.

Corollary.
The output of the collapsing algorithm is the minimal DFA for the set and it is unique up to isomorphism.