Non-Deterministic Finite Automata

NFAs accept the Regular Languages

Equivalence of Machines

Definition:

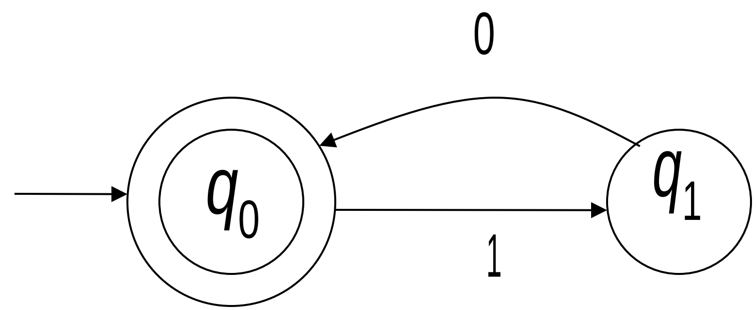
Machine M_1 is equivalent to machine M_2

if $L(M_1) = L(M_2)$

Example of equivalent machines

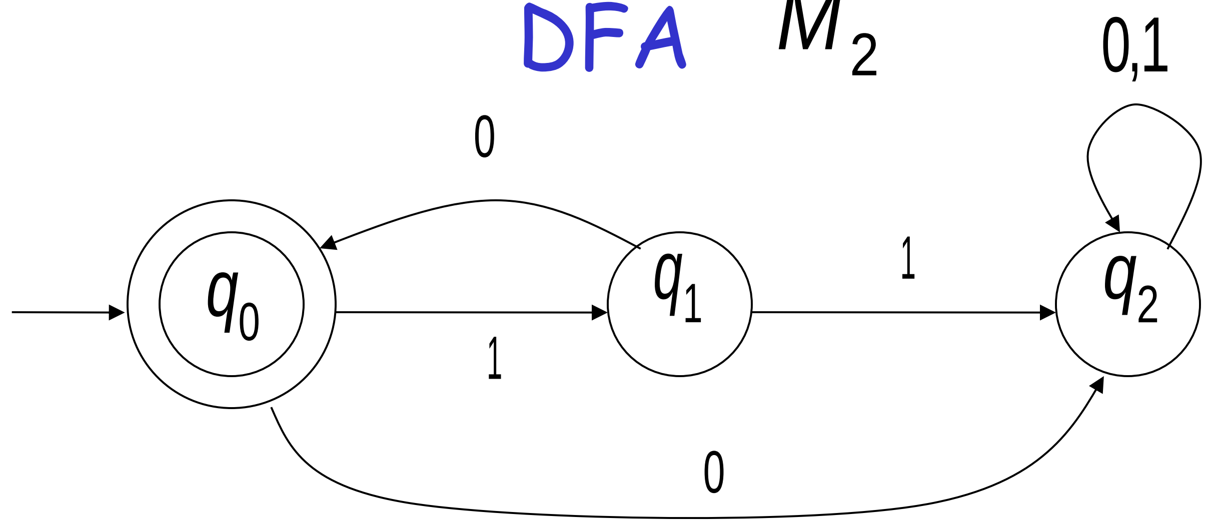
NFA M_1

$L(M_1) = ?$



DFA M_2

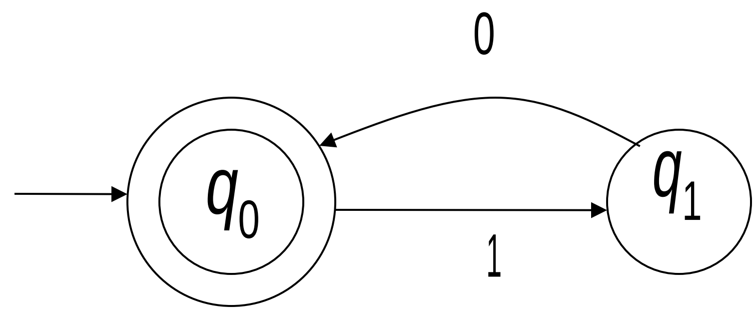
$L(M_2) = ?$



Example of equivalent machines

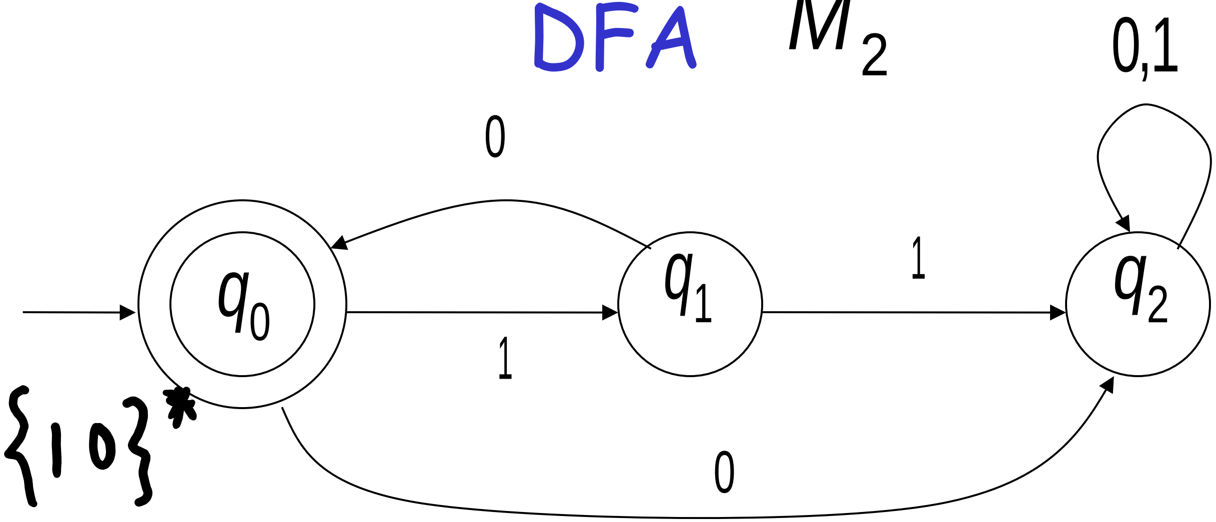
$L(M_1) = ?$

NFA M_1



$L(M_2) = ?$

DFA M_2



$L(M_1) = L(M_2) = \{10\}^*$

Theorem:

$$\left\{ \begin{array}{l} \text{Languages} \\ \text{accepted} \\ \text{by NFAs} \end{array} \right\} = \left\{ \begin{array}{l} \text{Regular} \\ \text{Languages} \end{array} \right\}$$

Languages
accepted
by DFAs

NFAs and DFAs have the same computation power,
accept the same set of languages

Proof: we only need to show

$$\left\{\begin{array}{l}\text{Languages}\\ \text{accepted}\\ \text{by NFAs}\end{array}\right\} \supseteq \left\{\begin{array}{l}\text{Regular}\\ \text{Languages}\end{array}\right\}$$

AND

$$\left\{\begin{array}{l}\text{Languages}\\ \text{accepted}\\ \text{by NFAs}\end{array}\right\} \subseteq \left\{\begin{array}{l}\text{Regular}\\ \text{Languages}\end{array}\right\}$$

Proof-Step 1

$$\left\{\begin{array}{l}\text{Languages}\\\text{accepted}\\\text{by NFAs}\end{array}\right\} \supseteq \left\{\begin{array}{l}\text{Regular}\\\text{Languages}\end{array}\right\}$$

Every DFA is trivially an NFA

⇓

Any language L accepted by a DFA is also accepted by an NFA

Proof-Step 2

$$\left\{\begin{array}{l}\text{Languages} \\ \text{accepted} \\ \text{by NFAs}\end{array}\right\} \subseteq \left\{\begin{array}{l}\text{Regular} \\ \text{Languages}\end{array}\right\}$$

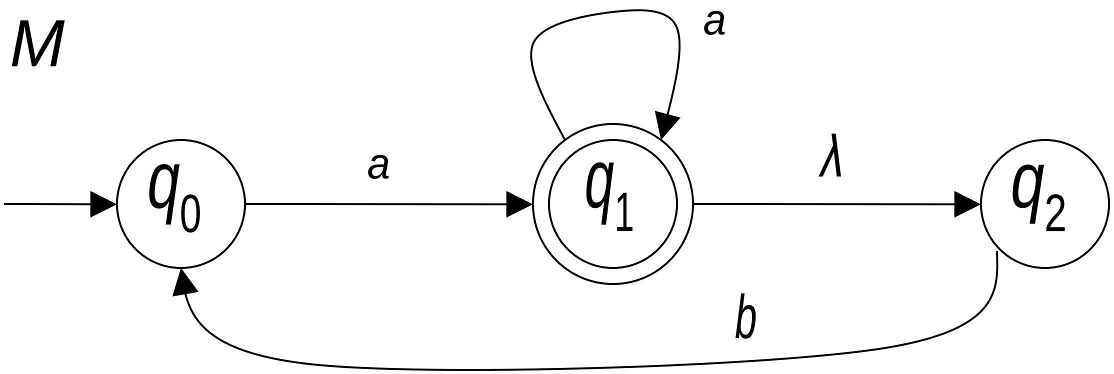
Any NFA can be converted to an equivalent DFA

$$\Downarrow$$

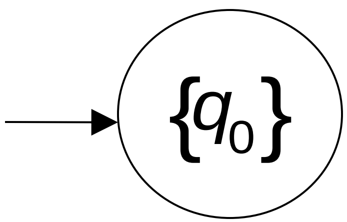
Any language L accepted by an NFA is also accepted by a DFA

Conversion NFA to DFA

NFA M

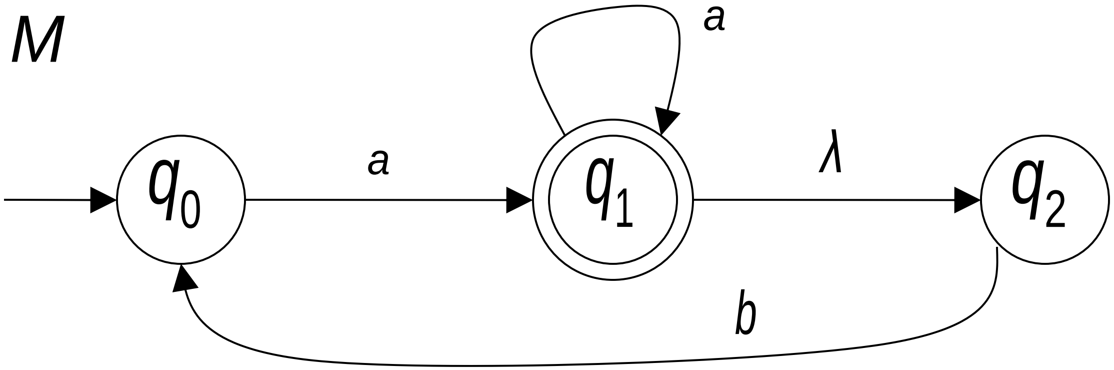


DFA M'

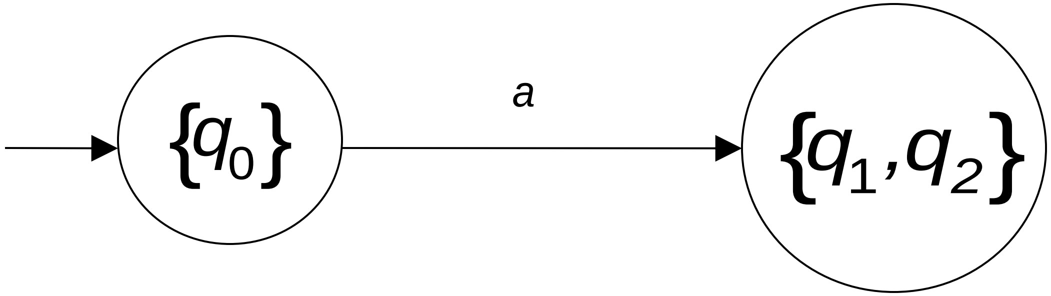


$$\delta(q_0, a) = \{q_1, q_2\}$$

NFA M

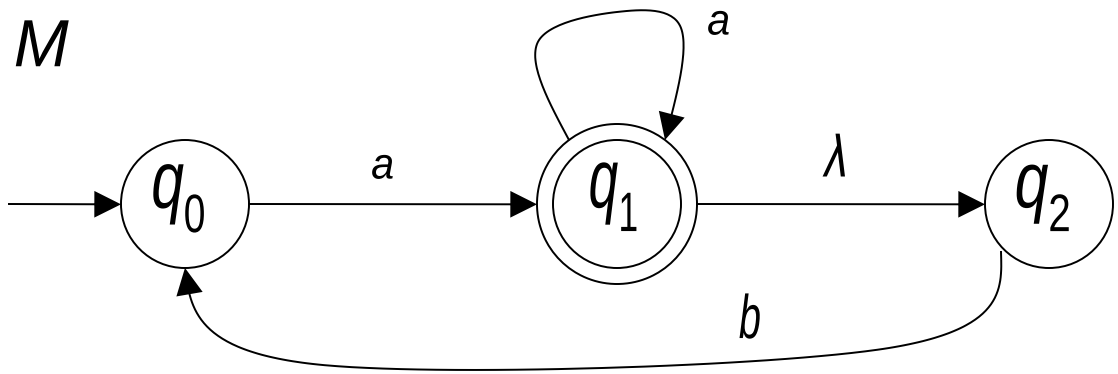


DFA M'

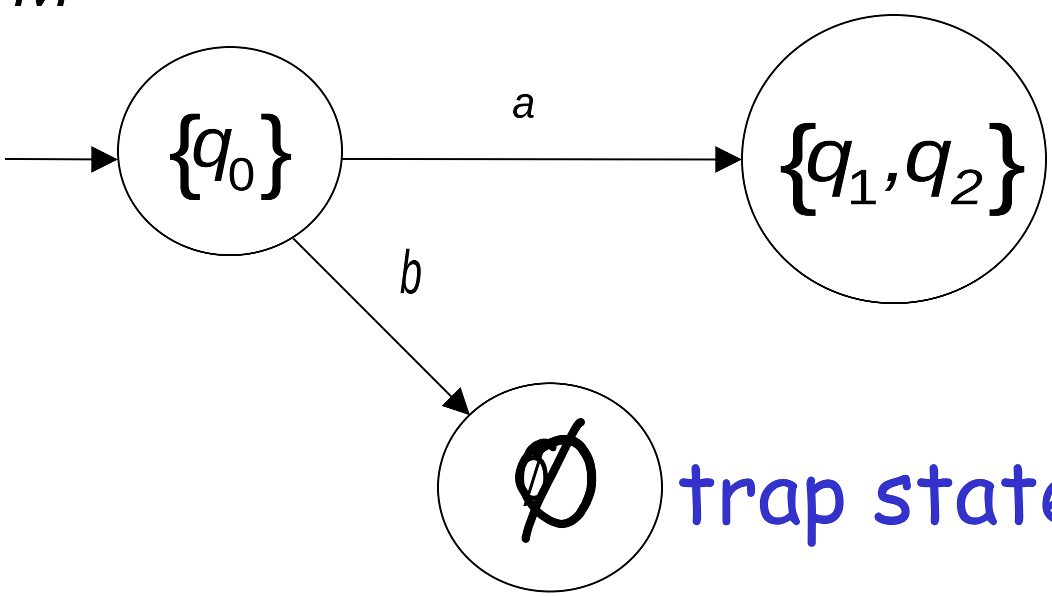


$\delta(q_0, b) = \emptyset$ empty set

NFA M

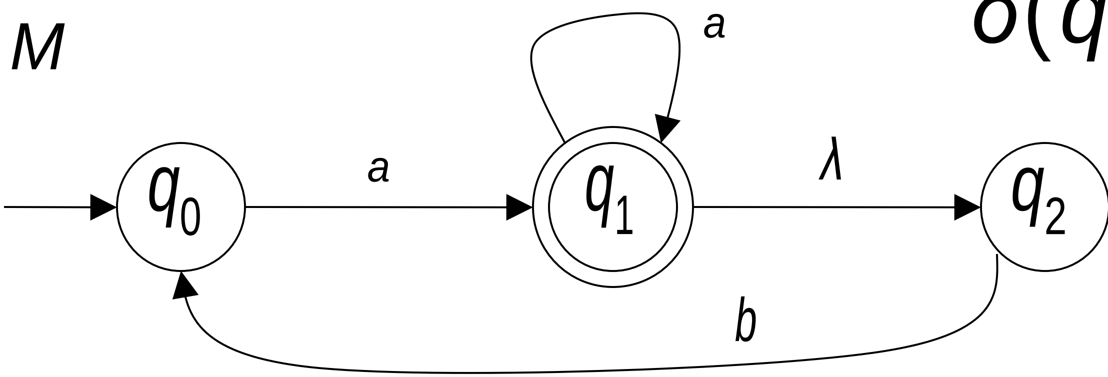


DFA M'

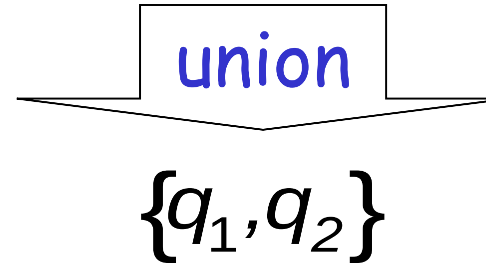


trap state

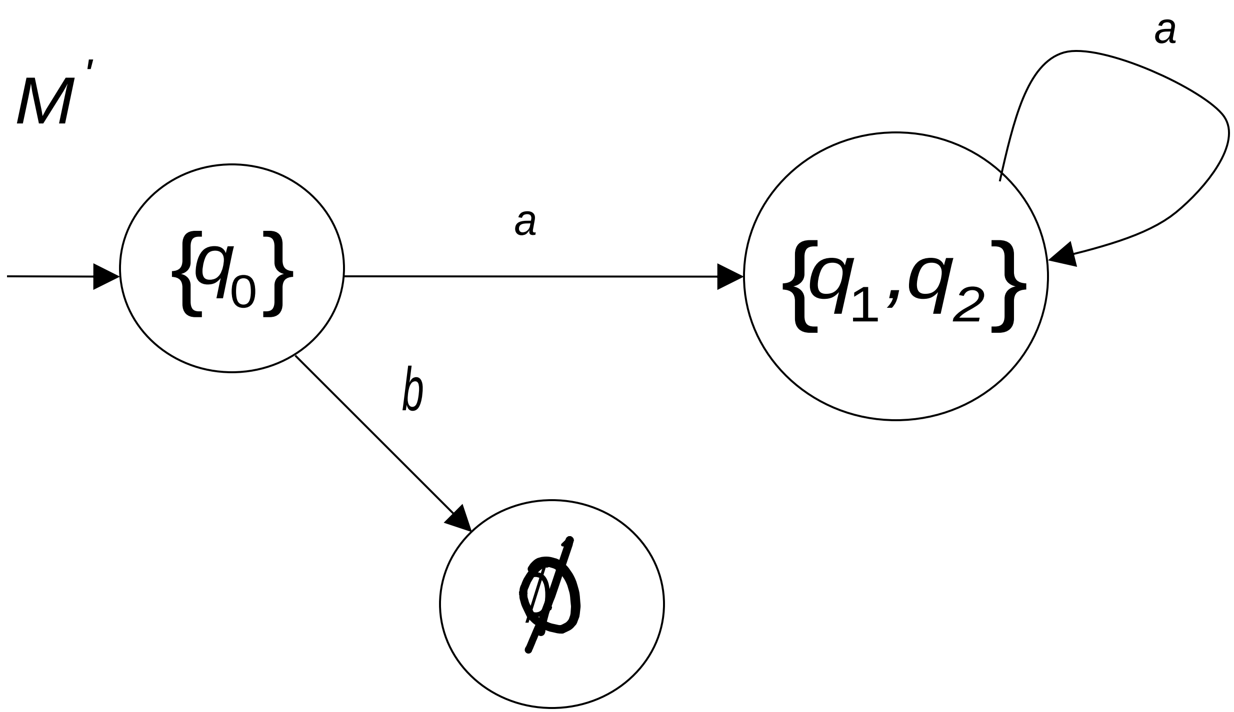
NFA M



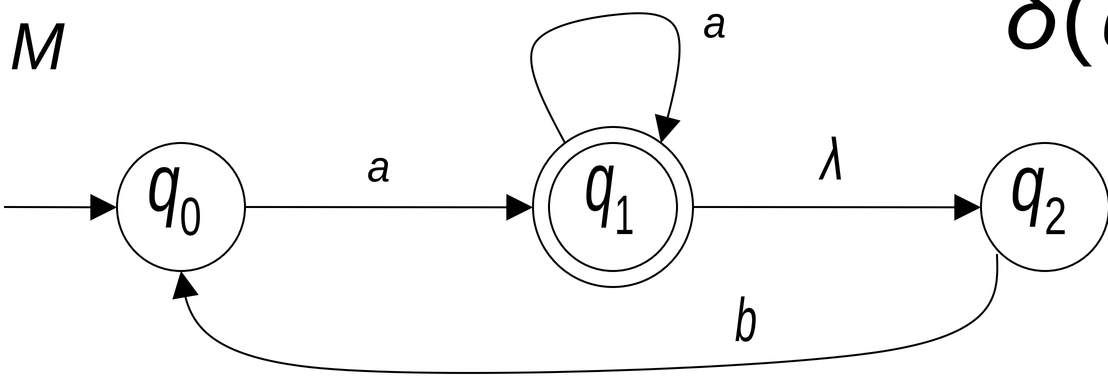
$$\delta(q_1, a) = \{q_1, q_2\}$$
$$\delta(q_2, a) = \emptyset$$



DFA M'



NFA M

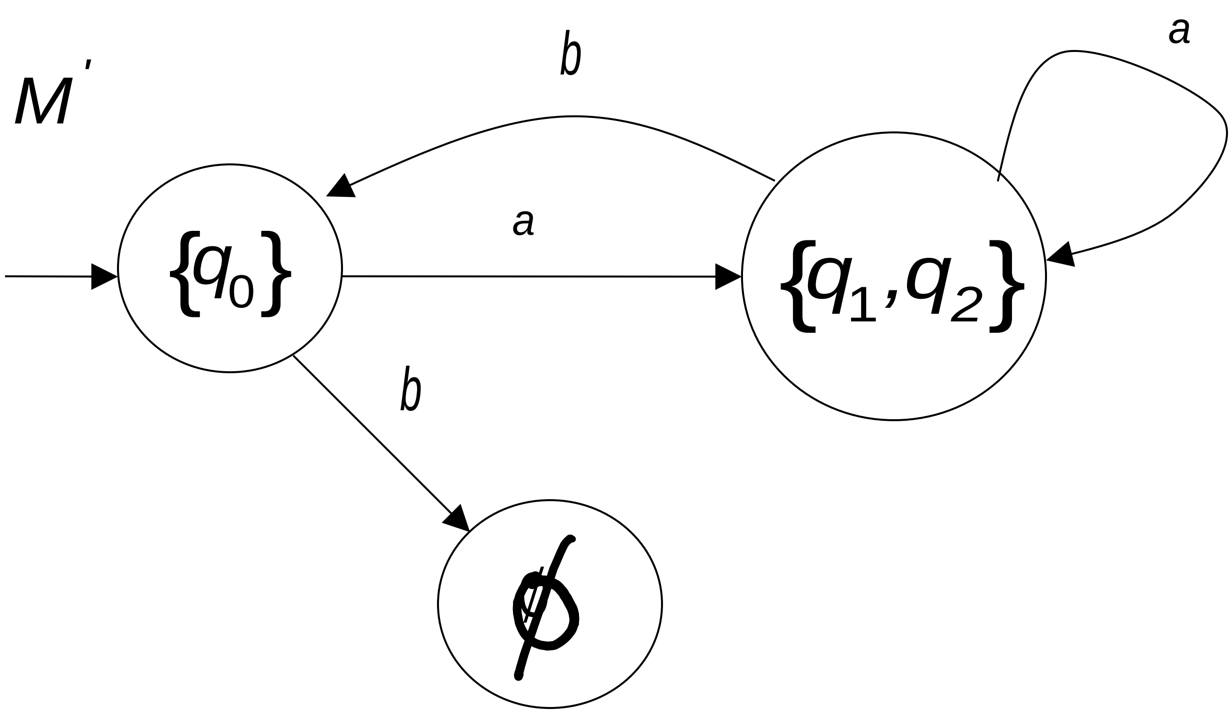


$$\delta(q_1,b) = \{q_0\}$$
$$\delta(q_2,b) = \{q_0\}$$

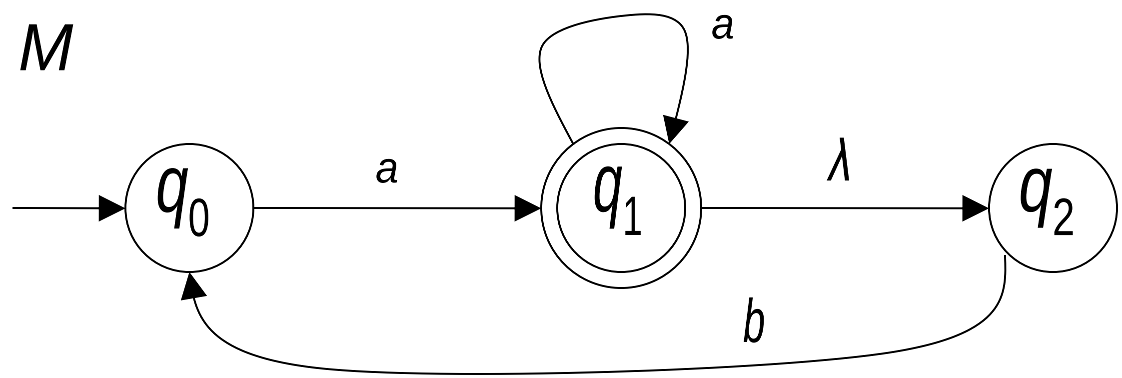
union

$\{q_0\}$

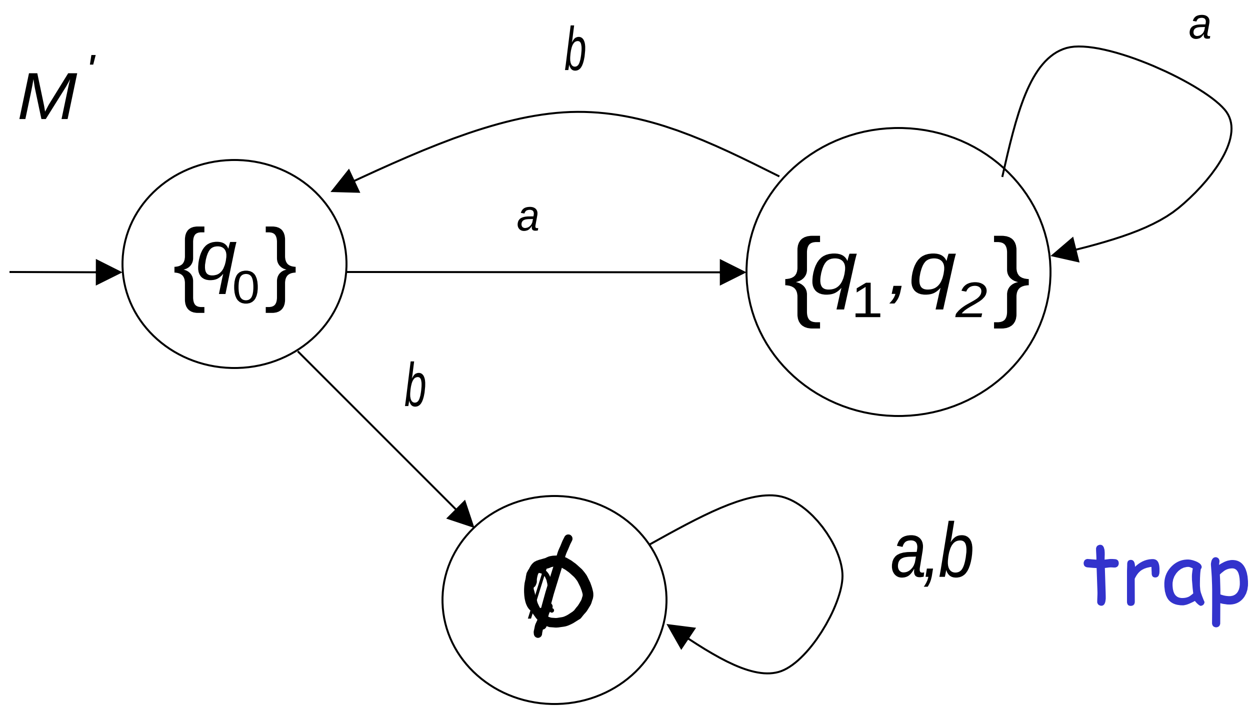
DFA M'



NFA M



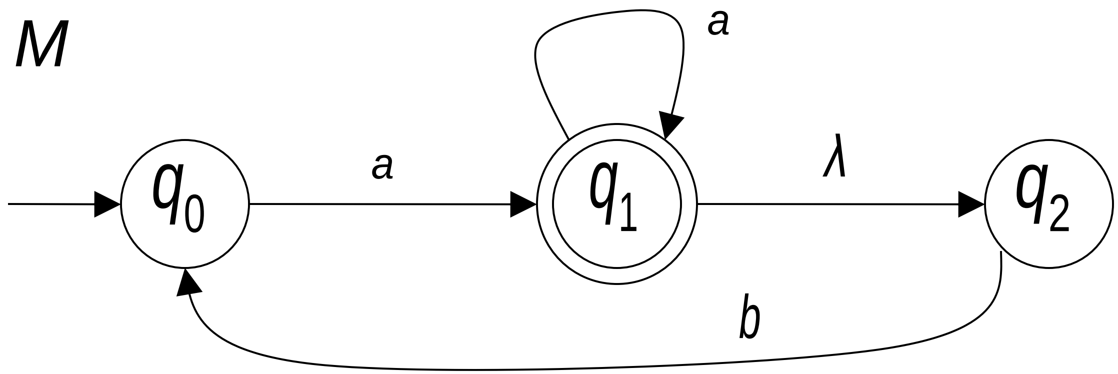
DFA M'



trap state

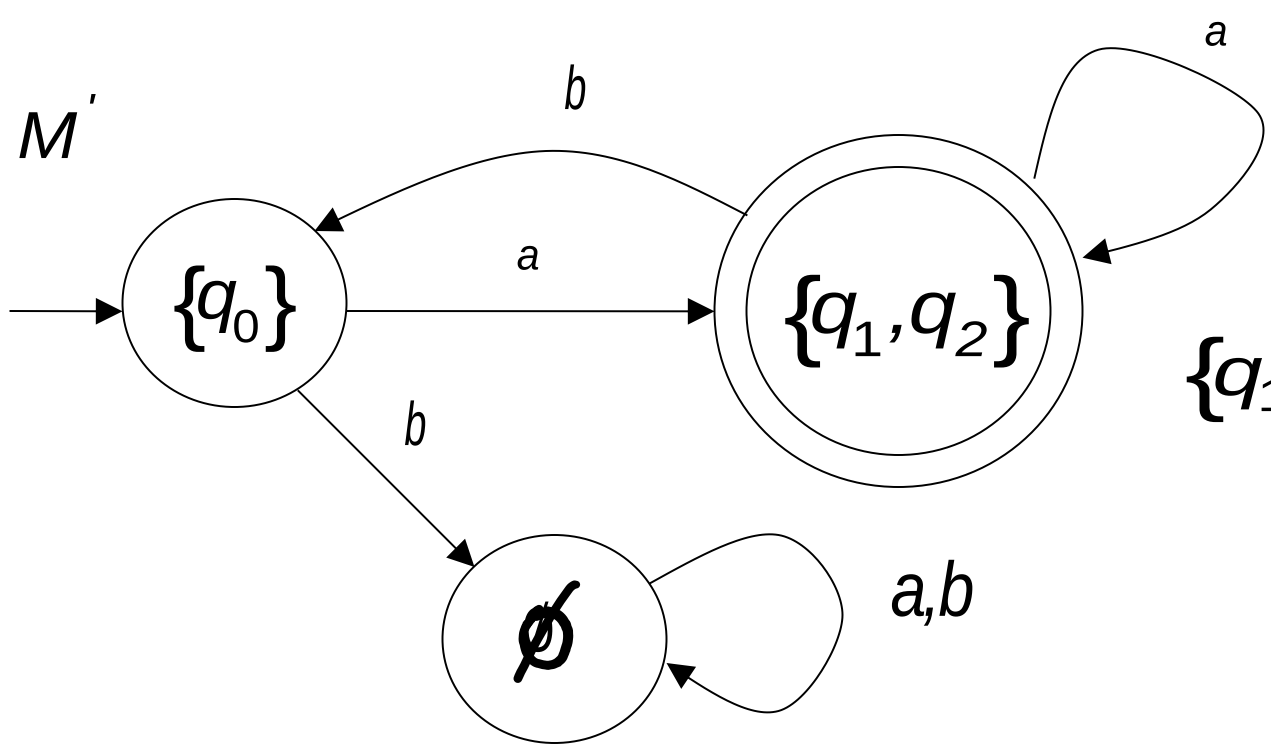
END OF CONSTRUCTION

NFA M



$q_1 \in F$

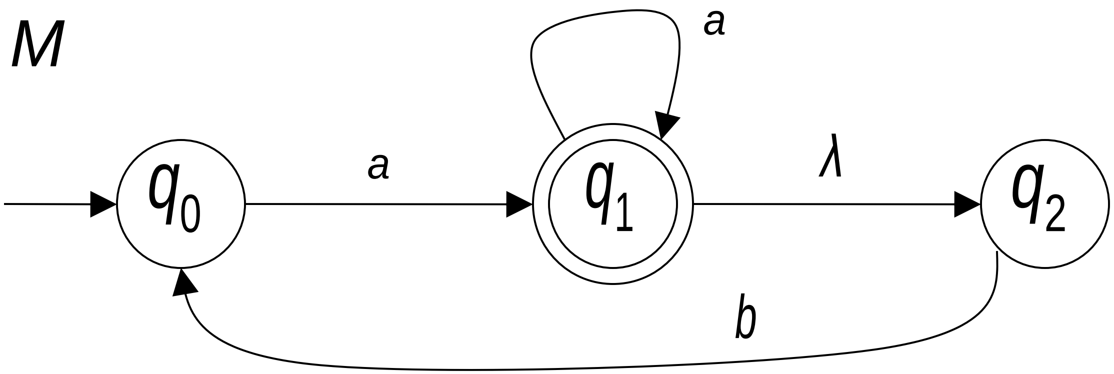
DFA M'



$\{q_1, q_2\} \in F'$

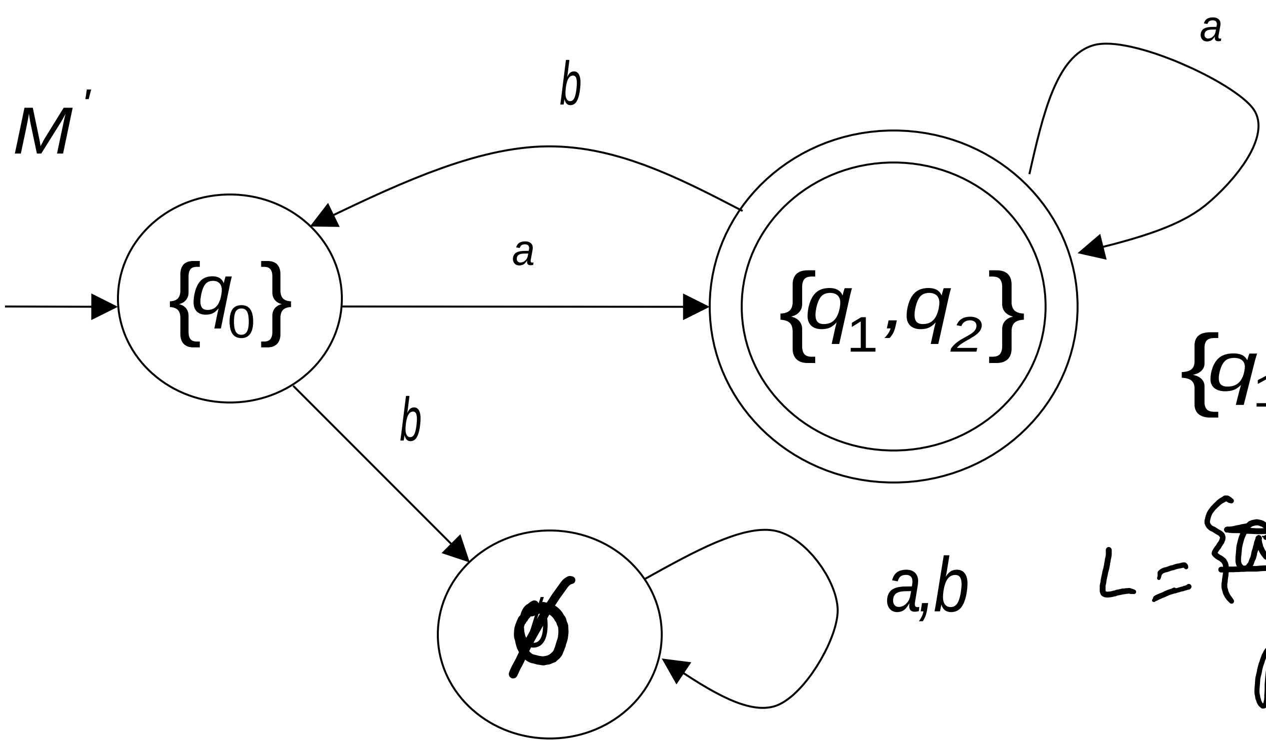
END OF CONSTRUCTION

NFA M



$q_1 \in F$

DFA M'



$\{q_1, q_2\} \in F'$

$L = \cancel{\{a^+\} \cup \{a^+ba\}^*}$

$a^+(ba^+)^*$

General Conversion Procedure

Input: an NFA M

Output: an equivalent DFA M'

with $L(M) = L(M')$

The NFA has states $q_0, q_1, q_2, \ldots$

The DFA has states from the power set

$$\emptyset, \{q_0\}, \{q_1\}, \{q_0, q_1\}, \{q_1, q_2, q_3\}, \ldots$$

Conversion Procedure Steps

step

1. Initial state of NFA: q_0

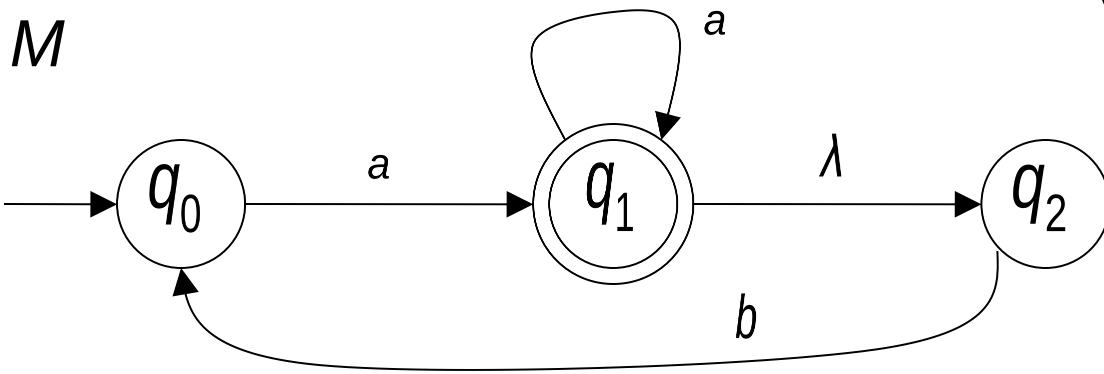
$$\delta(q_0, \lambda) = \{q_0, \ldots\}$$

Initial state of DFA: $\{q_0, \ldots\}$

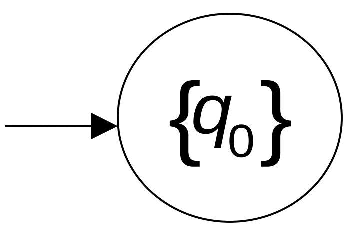
Example

$$\delta(q_0, \lambda) = \{q_0\}$$

NFA M



DFA M'



step

2. For every DFA's state $\{q_i, q_j, \ldots, q_m\}$

compute in the NFA

$$\left.\begin{array}{l} \delta^*(q_i, a) \\ \cup\, \delta^*(q_j, a) \\ \ldots \\ \cup\, \delta^*(q_m, a) \end{array}\right\} = \{q'_k, q'_l, \ldots, q'_n\}$$

Union

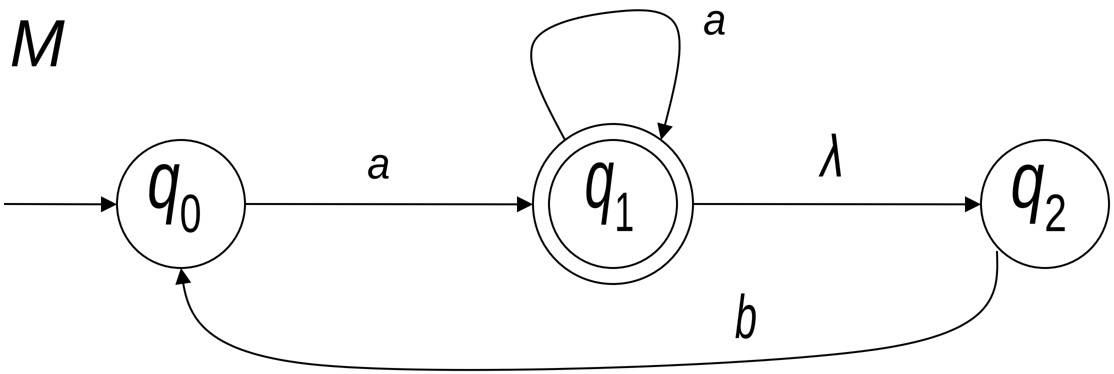
add transition to DFA

$$\delta(\{q_i, q_j, \ldots, q_m\}, a) = \{q'_k, q'_l, \ldots, q'_n\}$$

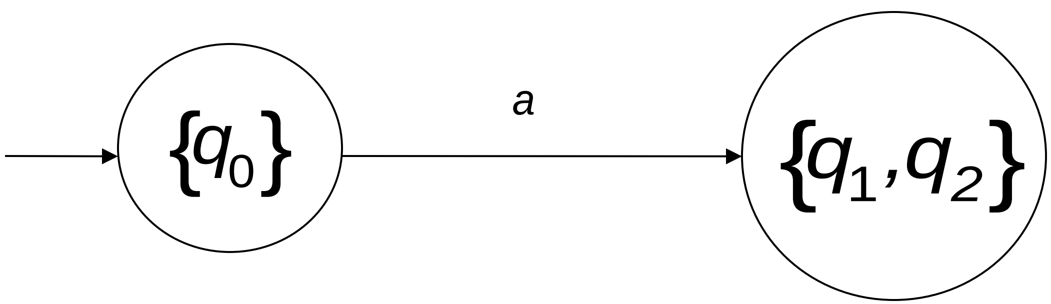
Example

$$\delta^*(q_0, a) = \{q_1, q_2\}$$

NFA M



DFA M'



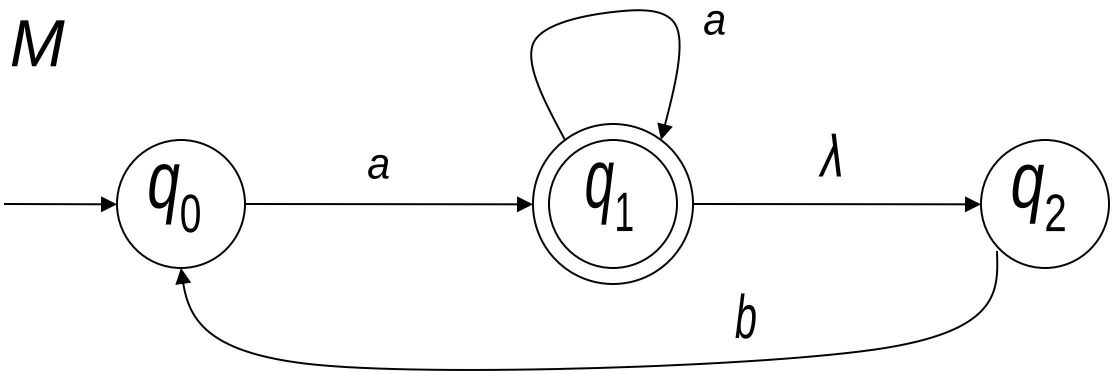
$$\delta(\{q_0\}, a) = \{q_1, q_2\}$$

step

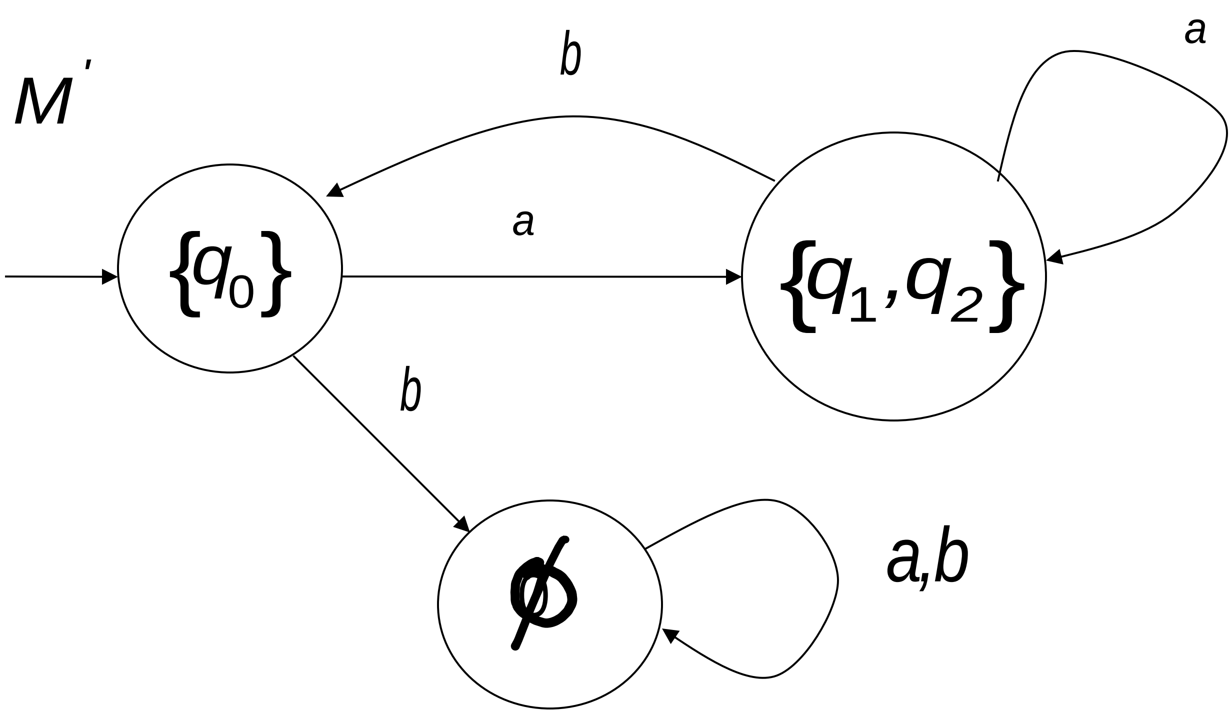
3. Repeat Step 2 for every state in DFA and symbols in alphabet until no more states can be added in the DFA

Example

NFA M



DFA M'



step

4. For any DFA state $\{q_i, q_j, \ldots, q_m\}$

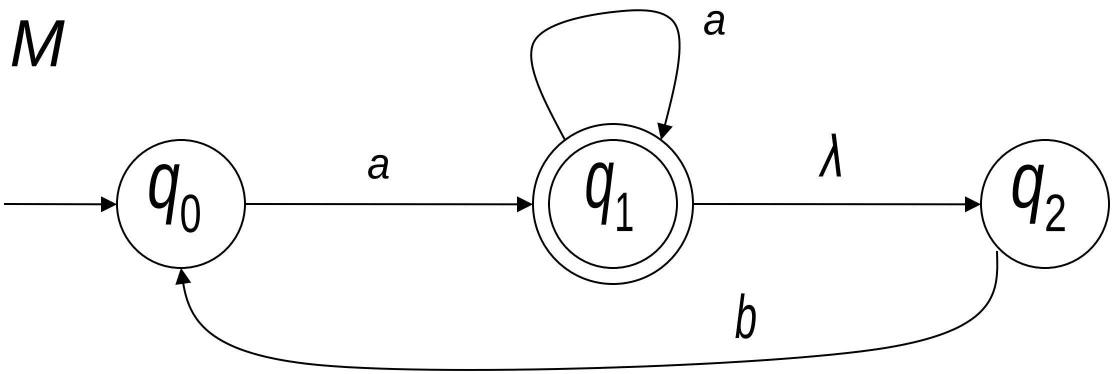
if some q_j is accepting state in NFA

Then, $\{q_i, q_j, \ldots, q_m\}$

is accepting state in DFA

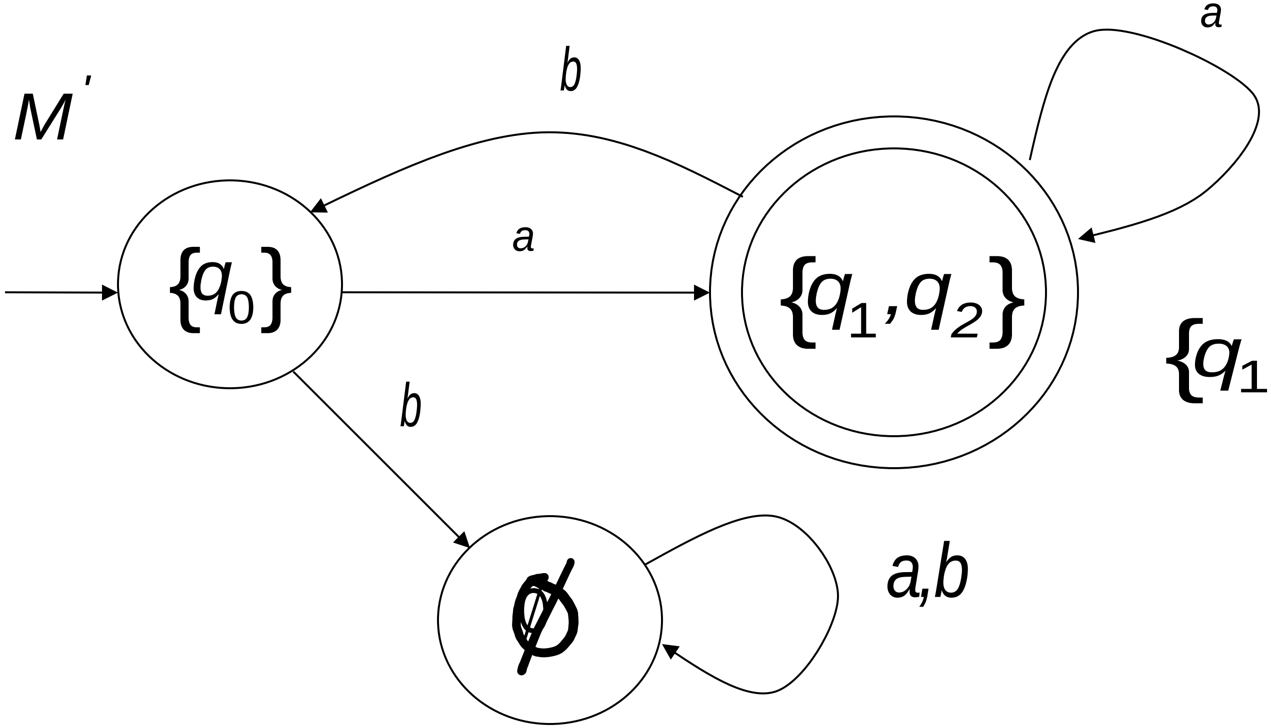
Example

NFA M



$q_1 \in F$

DFA M'



$\{q_1, q_2\} \in F'$

Lemma:

If we convert NFA M to DFA M'

then the two automata are equivalent:

$$L(M) = L(M')$$

Proof:

We only need to show: $L(M) \subseteq L(M')$

AND

$$L(M) \supseteq L(M')$$

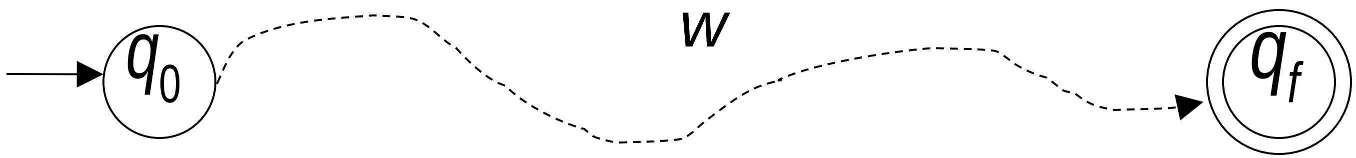
First we show: $L(M) \subseteq L(M')$

We only need to prove:

$$w \in L(M) \quad \Rightarrow \quad w \in L(M')$$

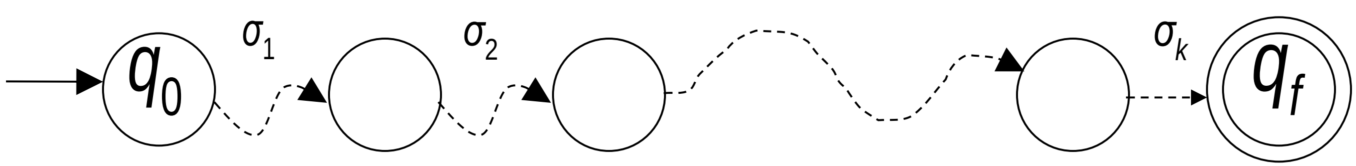
NFA

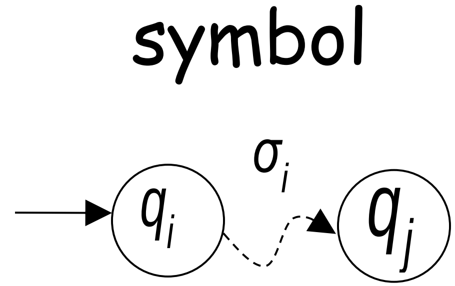
Consider $w \in L(M)$



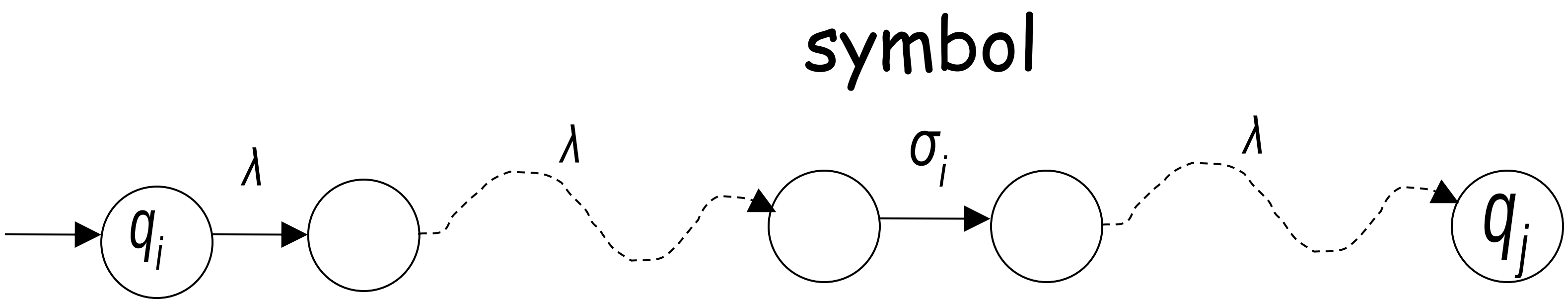
symbols

$$w = \sigma_1 \sigma_2 \cdots \sigma_k$$



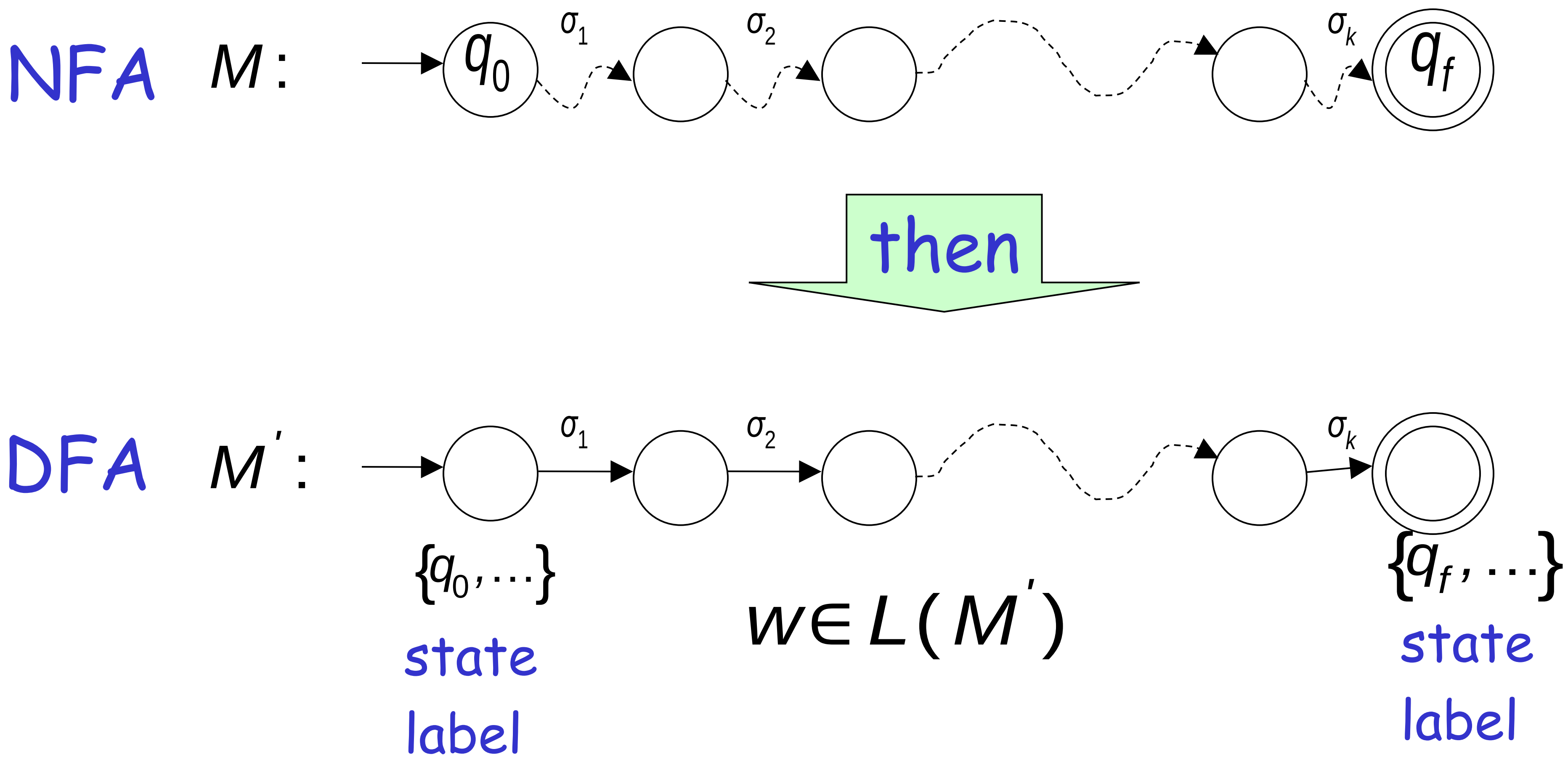


denotes a possible sub-path like



We will show that if $w \in L(M)$

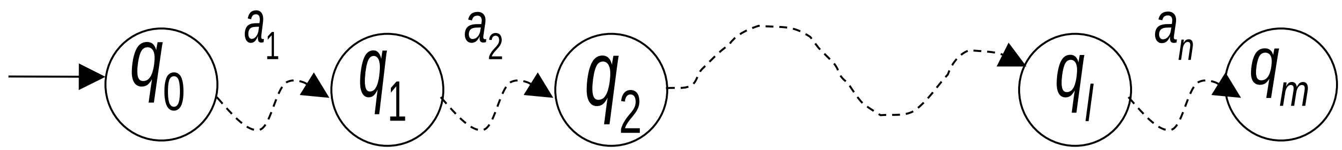
$$w = \sigma_1 \sigma_2 \cdots \sigma_k$$



More generally, we will show that if in M:

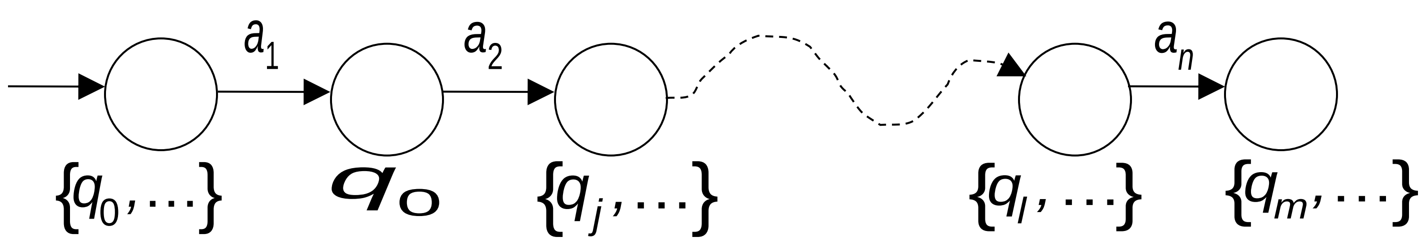
(arbitrary string) $v = a_1 a_2 \cdots a_n$

NFA M:



then

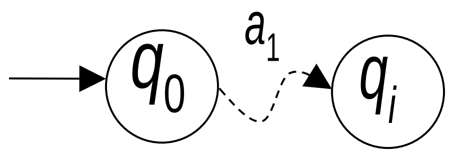
DFA M':



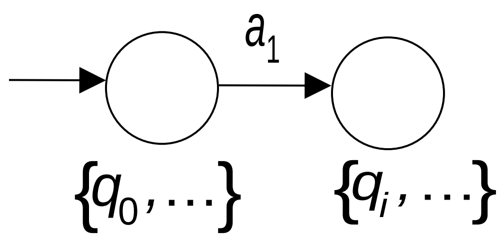
Proof by induction on $|v|$

Induction Basis: $|v| = 1$ $\qquad v = a_1$

NFA M:



DFA M':



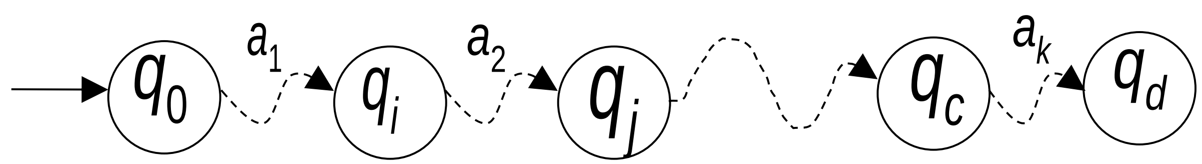
is true by construction of M'

Induction hypothesis: $1 \leq |v| \leq k$

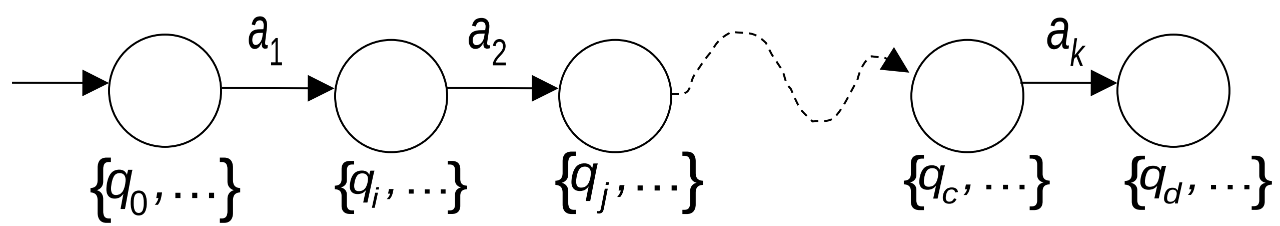
$$v = a_1 a_2 \cdots a_k$$

Suppose that the following hold

NFA $M:$



DFA $M':$

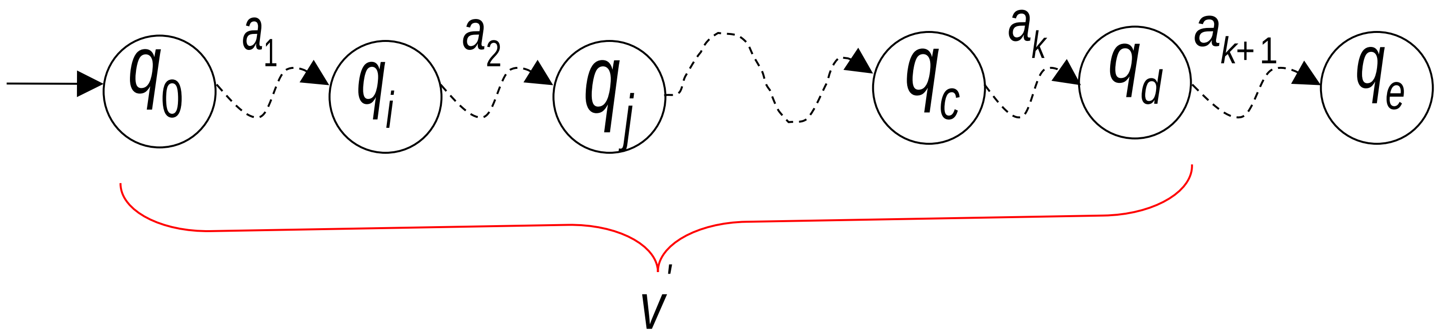


Induction Step: $|v| = k+1$

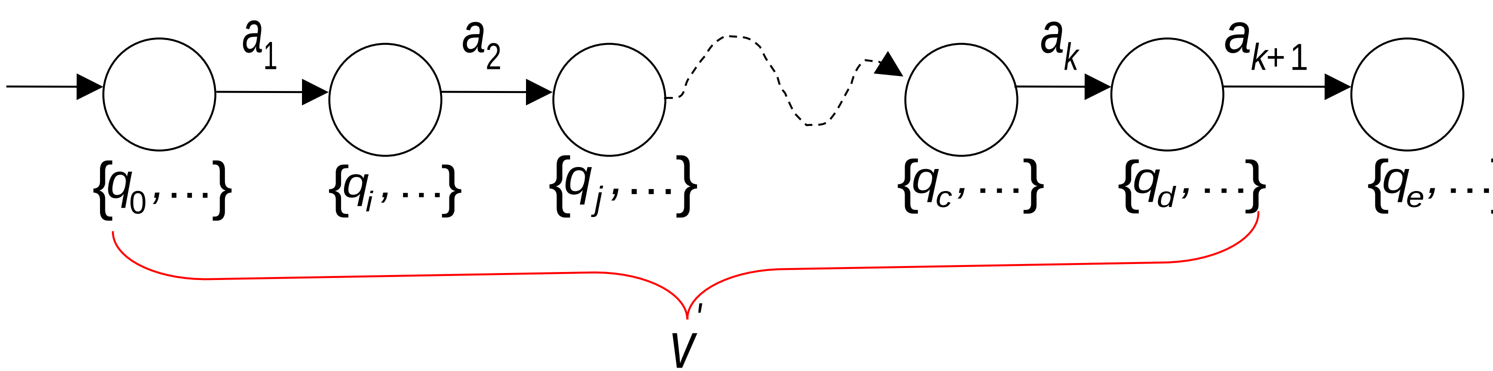
$$v = \underbrace{a_1 a_2 \cdots a_k}_{v'} a_{k+1} = v' a_{k+1}$$

Then this is true by construction of M'

NFA M:

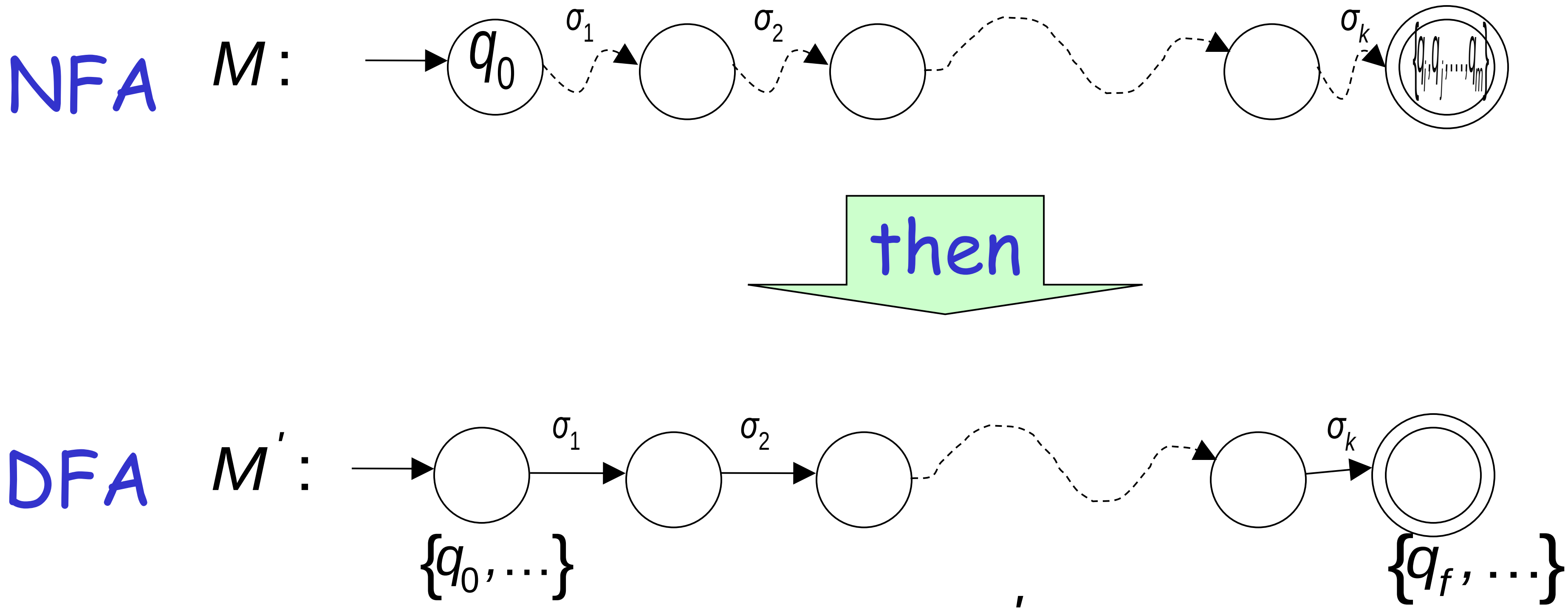


DFA M':



Therefore if $w \in L(M)$

$w = \sigma_1 \sigma_2 \cdots \sigma_k$



$w \in L(M')$

We have shown: $L(M) \subseteq L(M')$

With a similar proof

we can show: $L(M) \supseteq L(M')$

Therefore: $L(M) = L(M')$

END OF LEMMA PROOF