CS154

Finishing Minimization, The Myhill-Nerode Theorem, and Streaming Algorithms

DFA Minimization Theorem:

For every regular language L', there is a unique (up to re-labeling of states) minimal-state DFA M* such that L(M*) = L'.

Furthermore, there is an efficient algorithm which, given any DFA M, will output this unique M*.

Extending transition function δ to strings

Given M = (Q, Σ, δ, q_0, F), we can extend δ to a function $\Delta : Q \times \Sigma^* \rightarrow Q$ that works on strings:

$$\Delta(q, \varepsilon) = q$$

$$\Delta(q, \sigma) = \delta(q, \sigma)$$

$$\Delta(q, \sigma_1 \ldots \sigma_{k+1}) = \delta(\Delta(q, \sigma_1 \ldots \sigma_k), \sigma_{k+1})$$

$\Delta(q, w)$ = *the state of M reached after reading in w, starting from state q*

Note: $\Delta(q_0, w) \in F \Leftrightarrow$ M accepts w

Def. $w \in \Sigma^*$ distinguishes states q_1 and q_2 iff

$$\Delta(q_1, w) \in F \Leftrightarrow \Delta(q_2, w) \notin F$$

Extending transition function δ to strings

Given $M = (Q, \Sigma, \delta, q_0, F)$, we can extend δ to a function $\Delta : Q \times \Sigma^* \rightarrow Q$ that works on strings:

$$\Delta(q, \varepsilon) = q$$

$$\Delta(q, \sigma) = \delta(q, \sigma)$$

$$\Delta(q, \sigma_1 \ldots \sigma_{k+1}) = \delta(\Delta(q, \sigma_1 \ldots \sigma_k), \sigma_{k+1})$$

$\Delta(q, w)$ = *the state of M reached after reading in w, starting from state q*

Note: $\Delta(q_0, w) \in F \Leftrightarrow$ M accepts w

Def. $w \in \Sigma^*$ distinguishes states q_1 and q_2 iff *exactly one* of $\Delta(q_1, w)$, $\Delta(q_2, w)$ is a final state

Fix $M = (Q, \Sigma, \delta, q_0, F)$ and let $p, q \in Q$

Definition:

State p is *distinguishable* from state q

iff there is $w \in \Sigma^*$ that distinguishes p and q

iff there is $w \in \Sigma^*$ so that

exactly *one* of $\Delta(p, w)$, $\Delta(q, w)$ is a final state

State p is *indistinguishable* from state q

iff p is not distinguishable from q

iff for all $w \in \Sigma^*$, $\Delta(p, w) \in F \Leftrightarrow \Delta(q, w) \in F$

Pairs of indistinguishable states are redundant…

Fix $M = (Q, \Sigma, \delta, q_0, F)$ and let $p, q, r \in Q$

Define a binary relation $\sim$ on the states of M:

$p \sim q$ iff p is indistinguishable from q

$p \nsim q$ iff p is distinguishable from q

Proposition: $\sim$ is an equivalence relation

$p \sim p$ (reflexive)

$p \sim q \Rightarrow q \sim p$ (symmetric)

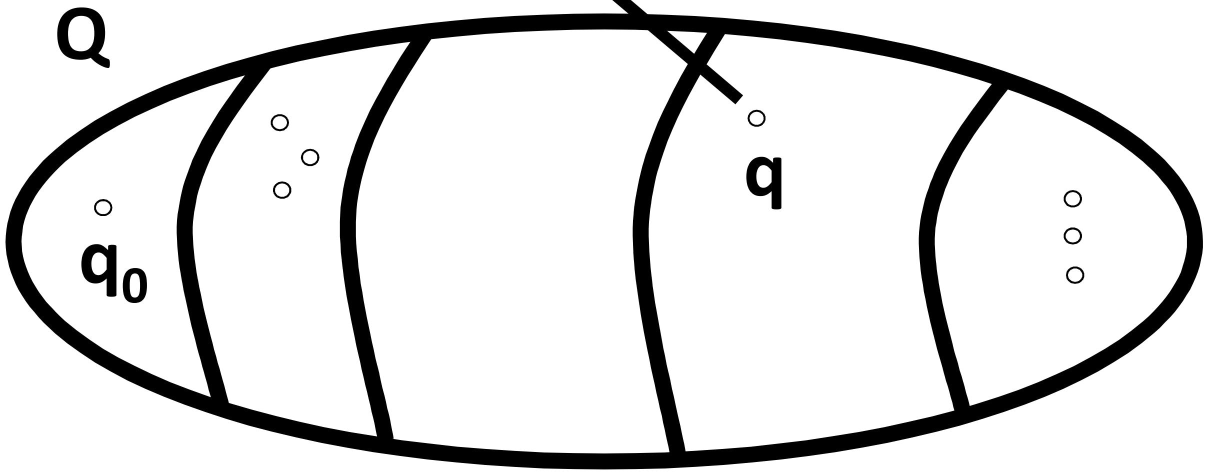
$p \sim q$ and $q \sim r \Rightarrow p \sim r$ (transitive)

Fix $M = (Q, \Sigma, \delta, q_0, F)$ and let $p, q, r \in Q$

Proposition: $\sim$ is an equivalence relation

As a consequence, the relation $\sim$ partitions Q into disjoint equivalence classes

$[q] := \{ p \mid p \sim q \}$



Algorithm: MINIMIZE-DFA

Input: DFA M

Output: DFA M_{MIN} such that:

$L(M) = L(M_{MIN})$

M_{MIN} has no *inaccessible* states

M_{MIN} is *irreducible*

||

For all states $p \neq q$ of M_{MIN}, p and q are distinguishable

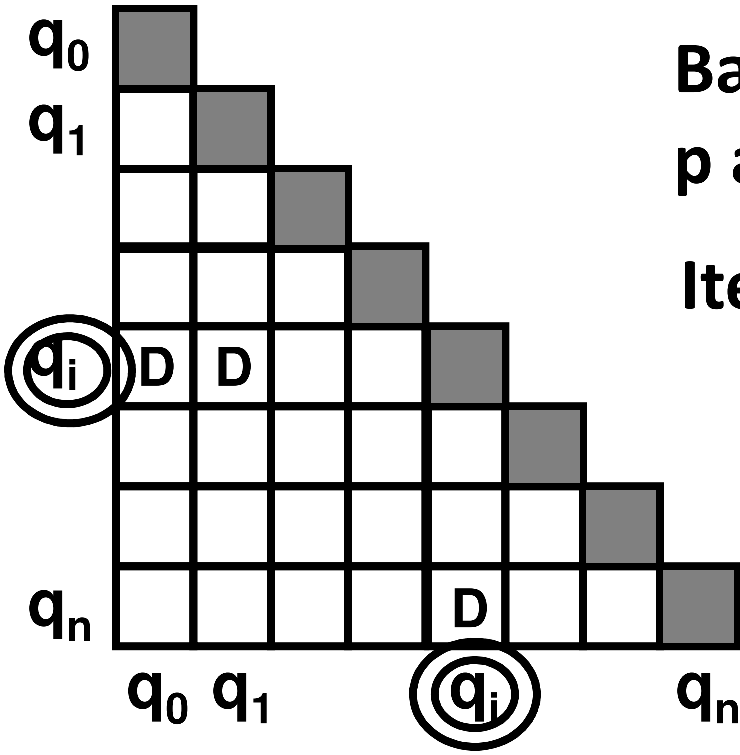
$$\boxed{\text{Theorem: } M_{MIN} \text{ is the unique minimal DFA that is equivalent to M}}$$

The Table-Filling Algorithm

Input: DFA $M = (Q, \Sigma, \delta, q_0, F)$

Output: (1) $D_M = \{ (p, q) \mid p, q \in Q \text{ and } p \not\sim q \}$

(2) $EQUIV_M = \{ [q] \mid q \in Q \}$



Base Case: For all (p, q) such that p accepts and q rejects $\Rightarrow p \not\sim q$

Iterate: If there are states p, q and symbol $\sigma \in \Sigma$ satisfying:

$$\begin{array}{l} \delta(p, \sigma) = p' \\ \qquad\qquad\quad \not\sim \quad \Rightarrow \ p \not\sim q \\ \delta(q, \sigma) = q' \end{array}$$

Repeat until no more D's can be added

Algorithm MINIMIZE

Input: DFA M

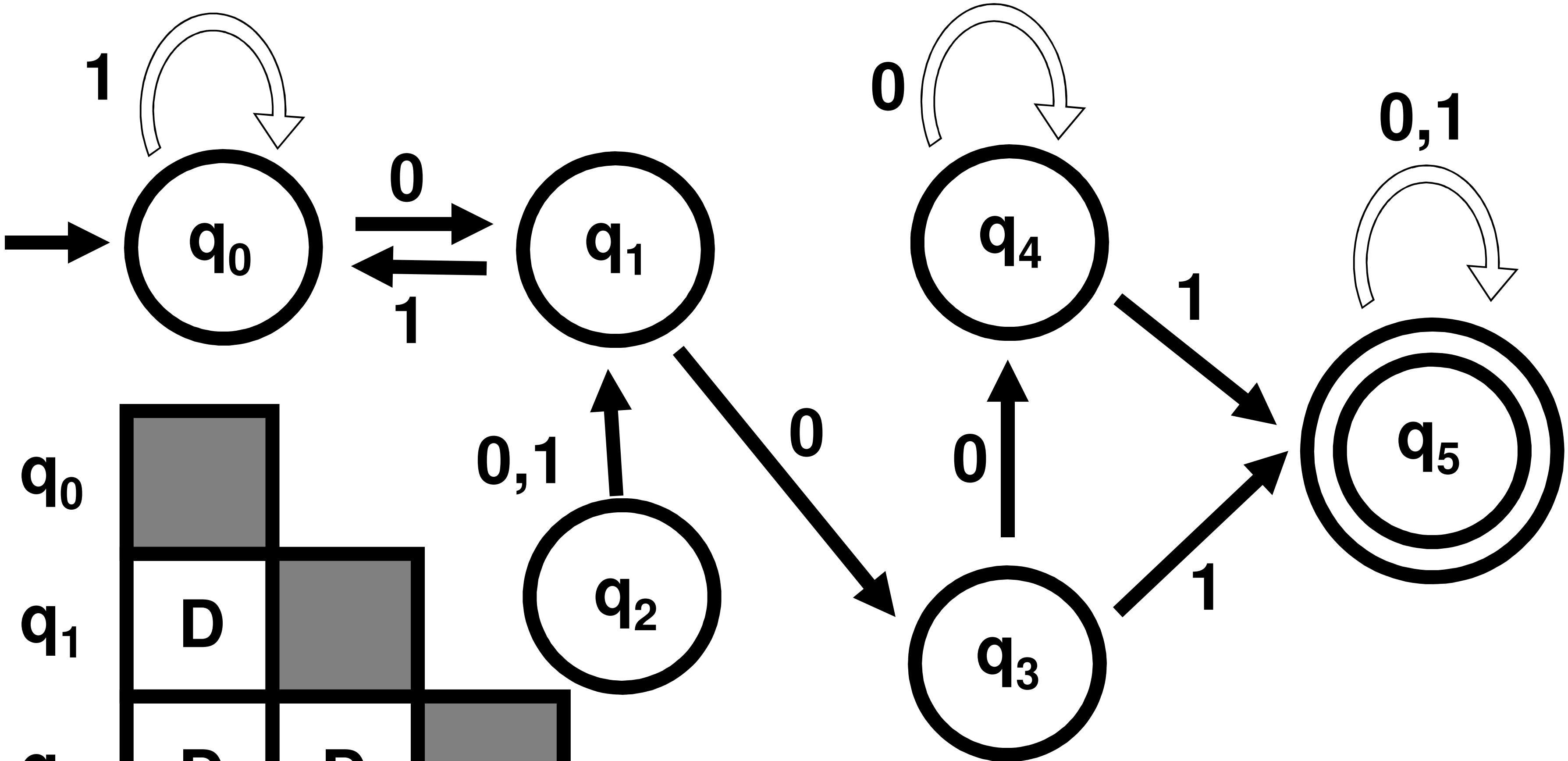
Output: Equivalent minimal-state DFA M_{MIN}

1. **Remove all inaccessible states from M**
2. **Run Table-Filling algorithm on M to get:**
   $EQUIV_M = \{ [q] \mid q$ is an accessible state of $M \}$
3. **Define: $M_{MIN} = (Q_{MIN}, \Sigma, \delta_{MIN}, q_{0\,MIN}, F_{MIN})$**

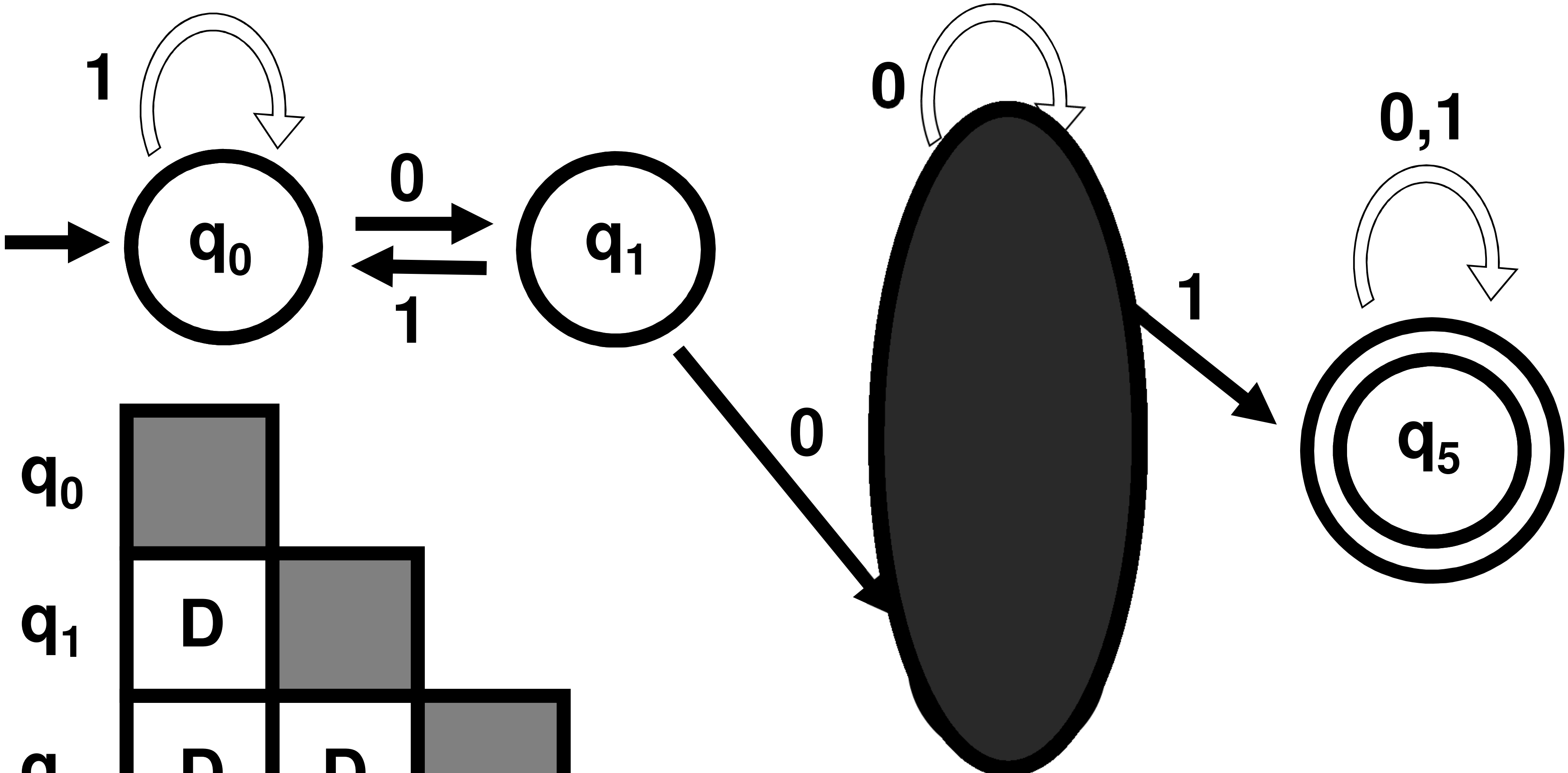
$$Q_{MIN} = EQUIV_M, \quad q_{0\,MIN} = [q_0], \quad F_{MIN} = \{ [q] \mid q \in F \}$$

$$\delta_{MIN}([q], \sigma) = [\delta(q, \sigma)]$$

Claim: $L(M_{MIN}) = L(M)$



q_0					
q_1	D				
q_3	D	D			
q_4	D	D			
q_5	D	D	D	D	
	q_0	q_1	q_3	q_4	q_5



	q_0	q_1	q_3	q_4	q_5
q_0					
q_1	D				
q_3	D	D			
q_4	D	D			
q_5	D	D	D	D	

Thm: M_{MIN} is the unique minimal DFA equivalent to M

Claim: Suppose $L(M') = L(M_{MIN})$ and M′ has no inaccessible states and M’ is irreducible.
Then there is an *isomorphism* between M′ and M_{MIN}

Suppose for now the Claim is true.
If M’ is a minimal DFA, then M’ has no inaccessible states and is irreducible *(why?)*
So the Claim implies:

Let M’ be a minimal DFA for M.
Then, there is an isomorphism between M’ and the DFA M_{MIN} that is output by MINIMIZE(M).
Therefore the Thm holds!

Thm: M_{MIN} is the unique minimal DFA equivalent to M

Claim: Suppose $L(M') = L(M_{MIN})$ and M' has no inaccessible states and M' is irreducible.
Then there is an *isomorphism* between M' and M_{MIN}

Proof: We recursively construct a map from the states of M_{MIN} to the states of M'

Base Case: $q_{0\,MIN} \mapsto q_0'$

Recursive Step: If

$$\begin{tikzcd} p \arrow[r, mapsto] \arrow[d, "\sigma"] & p' \arrow[d, "\sigma"] \\ q & q' \end{tikzcd}$$

Then $q \mapsto q'$

Base Case: $\mathbf{q_{0\,MIN} \mapsto q_0'}$

Recursive Step: **If**
$$\begin{array}{ccc} \mathbf{p} & \mapsto & \mathbf{p'} \\ \big\downarrow \sigma & & \big\downarrow \sigma \\ \mathbf{q} & & \mathbf{q'} \end{array}$$
Then $\mathbf{q \mapsto q'}$

Base Case: $q_{0\,MIN} \mapsto q_0'$

Recursive Step:

$$\begin{tikzcd} \text{If } p \arrow[d, "\sigma"] \arrow[r, mapsto] & p' \arrow[d, "\sigma"] \\ q & q' \end{tikzcd}$$

Then $q \mapsto q'$

Goal: Show this is an isomorphism. Need to prove:

The map is defined everywhere

The map is well defined

The map is a bijection

The map preserves all transitions:

If $p \mapsto p'$ then $\delta_{MIN}(p, \sigma) \mapsto \delta'(p', \sigma)$

(this follows from the definition of the map!)

Base Case: $q_{0\,MIN} \mapsto q_0'$

Recursive Step: **If** $p \mapsto p'$

$$\begin{tikzcd} p \arrow[r, mapsto] \arrow[d, "\sigma"] & p' \arrow[d, "\sigma"] \\ q & q' \end{tikzcd}$$

Then $q \mapsto q'$

The map is defined everywhere

That is, for all states q of M_{MIN}
there is *some* state q' of M' such that $q \mapsto q'$

If $q \in M_{MIN}$, there is a string w such that
$\Delta_{MIN}(q_{0\,MIN}, w) = q$ **(Why?)**

Let $q' = \Delta'(q_0', w)$. Then $q \mapsto q'$
(proof by induction on $|w|$)

Base Case: $q_{0\ MIN} \mapsto q_0'$

Recursive Step: If $p \mapsto p'$, $p \xrightarrow{\sigma} q$, $p' \xrightarrow{\sigma} q'$. Then $q \mapsto q'$

$$\begin{array}{ccc} p & \mapsto & p' \\ \downarrow\sigma & & \downarrow\sigma \\ q & & q' \end{array}$$

The map is well defined

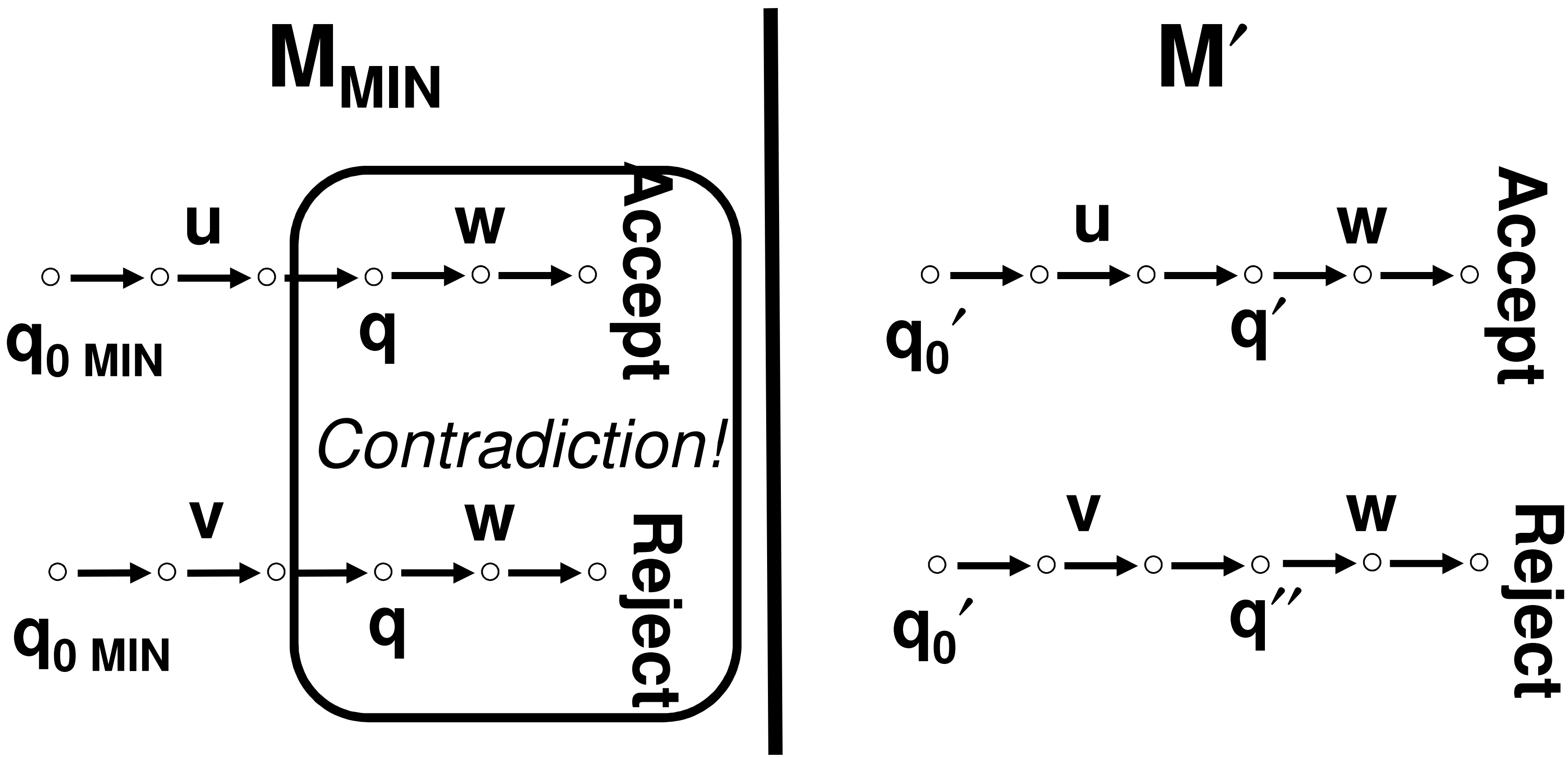
Proof by contradiction.

Suppose there are states q' and q'' such that $q \mapsto q'$ and $q \mapsto q''$

We show that q' and q'' are *indistinguishable*, so it must be that $q' = q''$

Suppose there are states q′ and q″ such that $q \mapsto q'$ and $q \mapsto q''$

Now suppose q′ and q″ are distinguishable…



Base Case: $q_{0\,MIN} \mapsto q_0'$

Recursive Step: If $p \mapsto p'$

$$\begin{tikzcd} p \arrow[r, mapsto] \arrow[d, "\sigma"] & p' \arrow[d, "\sigma"] \\ q & q' \end{tikzcd}$$

Then $q \mapsto q'$

The map is onto

Want to show: For all states q' of M' there is a state q of M_{MIN} such that $q \mapsto q'$

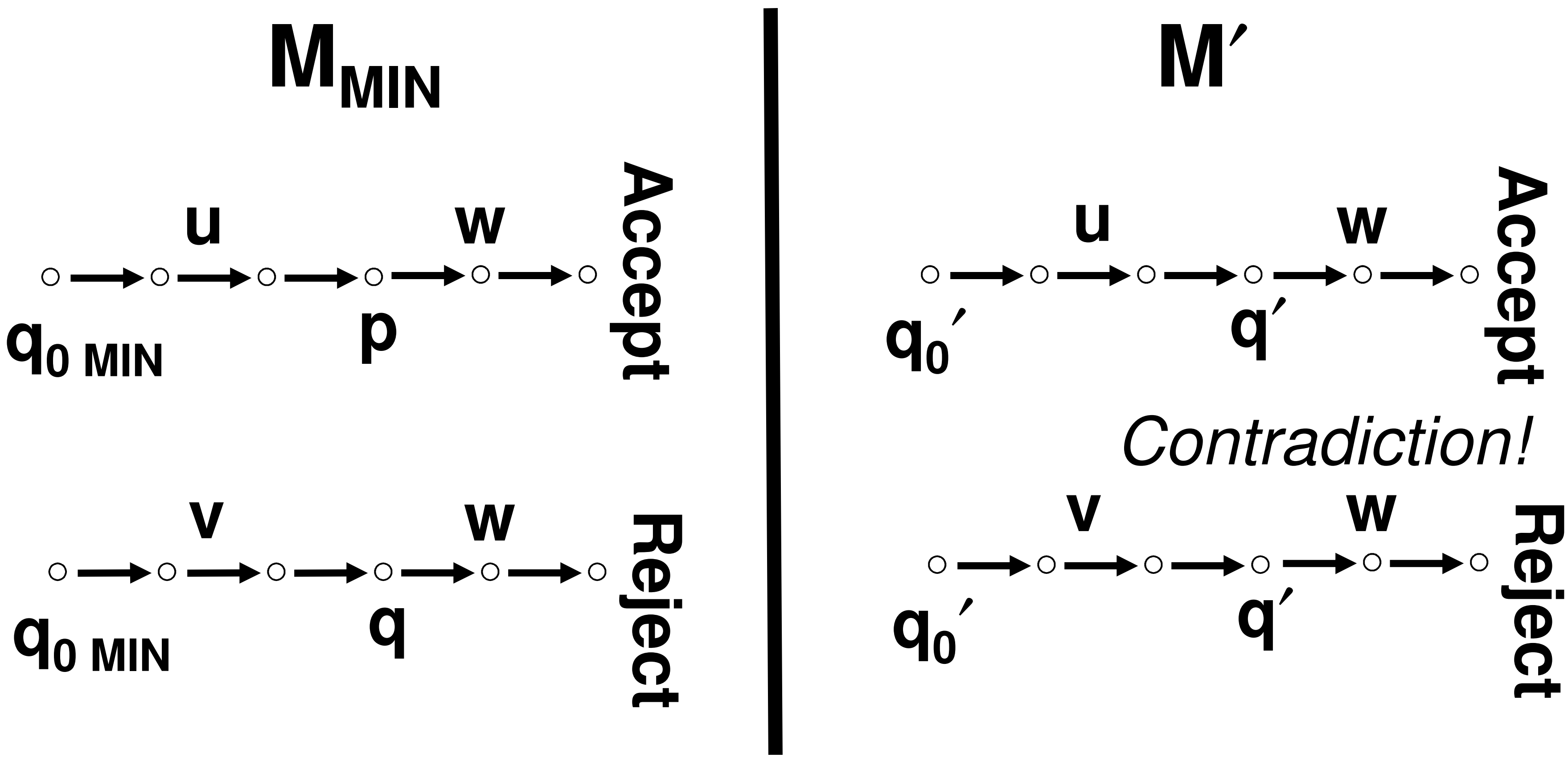
For every q' there is a string w such that
M' reaches state q' after reading in w

Let q be the state of M_{MIN} after reading in w
Claim: $q \mapsto q'$ *(proof by induction on $|w|$)*

The map is one-to-one

Proof by contradiction. Suppose there are states $p \neq q$ such that $p \mapsto q'$ and $q \mapsto q'$

If $p \neq q$, then p and q are distinguishable



How can we prove that two regular expressions are equivalent?

The Myhill-Nerode Theorem

In DFA Minimization, we defined an equivalence relation between states.

We can also define a similar equivalence relation over *strings* and *languages*:

Let $L \subseteq \Sigma^*$ and $x, y \in \Sigma^*$

$x \equiv_L y$ iff for all $z \in \Sigma^*$, $[xz \in L \Leftrightarrow yz \in L]$

Define: x and y are indistinguishable to L iff $x \equiv_L y$

Claim: $\equiv_L$ is an equivalence relation

Proof?

Let $L \subseteq \Sigma^*$ and $x, y \in \Sigma^*$

$x \equiv_L y$ iff for all $z \in \Sigma^*$, $[xz \in L \Leftrightarrow yz \in L]$

The Myhill-Nerode Theorem:
A language L is regular *if and only if* the number of equivalence classes of $\equiv_L$ is *finite*.

Proof ($\Rightarrow$) Let $M = (Q, \Sigma, \delta, q_0, F)$ be a min DFA for L.

Define the relation: $x \sim_M y \Leftrightarrow \Delta(q_0,x) = \Delta(q_0,y)$

Claim: $\sim_M$ is an equivalence relation with $|Q|$ classes

Claim: If $x \sim_M y$ then $x \equiv_L y$

Proof: $x \sim_M y$ implies for all $z \in \Sigma^*$, xz and yz reach the *same state* of M. So $xz \in L \Leftrightarrow yz \in L$, and $x \equiv_L y$

Corollary: Number of equiv. classes of $\equiv_L$ is *at most* the number of equiv. classes of $\sim_M$ (which is $|Q|$)

Let $L \subseteq \Sigma^*$ and $x, y \in \Sigma^*$

$x \equiv_L y$ iff for all $z \in \Sigma^*$, $[xz \in L \Leftrightarrow yz \in L]$

($\Leftarrow$) If the number of equivalence classes of $\equiv_L$ is *k* then there is a DFA for L with *k* states

Idea: Build a DFA using equivalence classes of $\equiv_L$!

Define a DFA M where

- **Q is the set of equivalence classes of $\equiv_L$**
- **$q_0 = [\varepsilon] = \{y \mid y \equiv_L \varepsilon\}$**
- **$\delta([x], \sigma) = [x\sigma]$**
- **$F = \{[x] \mid x \in L\}$**

Claim: M accepts x if and only if $x \in L$

The Myhill-Nerode Theorem gives us a *new* way to prove that a given language is not regular:

L is not regular
if and only if
there are infinitely many equiv. classes of $\equiv_L$

Distinguishing set for L

L is not regular
if and only if
There are infinitely many strings w_1, w_2, … so that for all $w_i \neq w_j$, w_i and w_j are distinguishable to L:

there is a $z \in \Sigma^*$ such that
exactly one of $w_i z$ and $w_j z$ is in L

The Myhill-Nerode Theorem gives us a *new* way to prove that a given language is not regular:

Theorem: $L = \{0^n 1^n \mid n \geq 0\}$ is not regular.

Proof: Consider the infinite set of strings

$$S = \{0, 00, 000, \ldots, 0^n, \ldots\}$$

Take any pair $(0^m, 0^n)$ of distinct strings in S

Let $z = 1^m$

Then $0^m 1^m$ is in L, but $0^n 1^m$ is *not* in L

That is, all pairs of strings in S are distinguishable

Hence there are infinitely many equivalence classes of $\equiv_L$, and L is not regular.