Lecture 5: Minimizing DFAs, Myhill-Nerode Theorem
DFA Minimization Theorem:

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

Furthermore, there is an efficient algorithm which, given any DFA $M$, will output this unique $M^*$. 
Extending transition function $\delta$ to strings

Given DFA $M = (Q, \Sigma, \delta, q_0, F)$, we extend $\delta$ to a function $\Delta : Q \times \Sigma^* \rightarrow Q$ as follows:

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

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

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

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

Note: $\Delta(q_0, w) \in F \iff M \text{ accepts } w$

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

(EITHER both $\Delta(p, w), \Delta(q, w)$ are in $F$, OR both are not in $F$)

Pairs of indistinguishable states are redundant...
Fix $M = (Q, \Sigma, \delta, q_0, F)$ and let $p, q, r \in Q$

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

Proposition: $\sim$ is an equivalence relation

States of $M_{\text{MIN}} = \text{Equivalence classes of states of } M$
Algorithm: MINIMIZE-DFA

Input: DFA M

Output: DFA $M_{\text{MIN}}$ such that:

- $L(M) = L(M_{\text{MIN}})$
- $M_{\text{MIN}}$ has no inaccessible states
- $M_{\text{MIN}}$ is irreducible
- For all states $p \neq q$ of $M_{\text{MIN}}$, $p$ and $q$ are distinguishable

Theorem: $M_{\text{MIN}}$ 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. $D_M = \{ (p, q) \mid p, q \in Q \text{ and } p \sim q \}$
2. $EQUIV_M = \{ [q] \mid q \in Q \}$
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 $p \in F$ and $q \not\in F \implies$ mark $p \not\sim q$
The Table-Filling Algorithm

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

Output:  
(1) $D_M = \{ (p, q) \mid p, q \in Q \text{ and } p \sim q \}$
(2) $\text{EQUIV}_M = \{ [q] \mid q \in Q \}$

Base Case: For all $(p, q)$ such that $p$ in $F$ and $q$ not in $F$ $\Rightarrow$ mark $p \not\sim q$

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

\[ \delta (p, \sigma) = p' \quad \text{mark} \quad p \not\sim q \]
\[ \delta (q, \sigma) = q' \]

Repeat until no more D’s can be added
Can we mark \((q_1, q_2)\) as distinguishable?

Are \(q_0\) and \(q_1\) distinguishable?

Are \(q_0\) and \(q_2\) distinguishable?
Claim: If \((p, q)\) is marked D by the Table-Filling algorithm, then \(p \sim q\)

Proof: By induction on the number of iterations in the algorithm before \((p, q)\) is marked D.

If \((p, q)\) is marked D in the base case, then one state’s in \(F\) and the other isn’t, so \(\varepsilon\) distinguishes \(p\) and \(q\).

Suppose \((p, q)\) is marked D in a later iteration.

Then there are states \(p’, q’\) such that:

1. \((p’, q’)\) is marked D \(\Rightarrow p’ \sim q’\) (by induction)

So there’s a string \(w\) s.t. \(\Delta(p’, w) \in F \iff \Delta(q’, w) \notin F\)

2. \(p’ = \delta(p, \sigma)\) and \(q’ = \delta(q, \sigma)\), for some \(\sigma \in \Sigma\)

Then the string \(\sigma w\) distinguishes \(p\) and \(q\)!
Claim: If \((p, q)\) is not marked D by the Table-Filling algorithm, then \(p \sim q\)

Proof (by contradiction):
Suppose the pair \((p, q)\) is not marked D by the algorithm, yet \(p \sim q\) (call this a “bad pair”)

Then there is a string \(w\) such that \(|w| > 0\) and:

\[
\Delta(p, w) \in F \iff \Delta(q, w) \notin F
\]

(Why is \(|w| > 0\)?)

Of all such bad pairs, let \((p, q)\) be a bad pair with a minimum-length distinguishing string \(w\)
Claim: If \((p, q)\) is not marked D by the Table-Filling algorithm, then \(p \sim q\)

Proof (by contradiction):
Suppose the pair \((p, q)\) is not marked D by the algorithm, yet \(p \sim q\) (call this a “bad pair”)
Of all such bad pairs, let \((p, q)\) be a bad pair with a minimum-length distinguishing string \(w\)
\[\Delta(p, w) \in F \iff \Delta(q, w) \not\in F\]

We have \(w = \sigma w'\), for some string \(w'\) and some \(\sigma \in \Sigma\)
Let \(p' = \delta(p, \sigma)\) and \(q' = \delta(q, \sigma)\)
Then \((p', q')\) is also a bad pair!
But then \((p', q')\) has a SHORTER distinguishing string, \(w'\)
Contradiction!
Algorithm MINIMIZE

Input: DFA M

Output: Equivalent minimal-state DFA $M_{\text{MIN}}$

1. Remove all inaccessible states from M

2. Run Table-Filling algorithm on M to get:
   $EQUIV_M = \{ [q] \mid q \text{ is an accessible state of } M \}$

3. Define: $M_{\text{MIN}} = (Q_{\text{MIN}}, \Sigma, \delta_{\text{MIN}}, q_{0\text{MIN}}, F_{\text{MIN}})$
   
   $Q_{\text{MIN}} = EQUIV_M$, $q_{0\text{MIN}} = [q_0]$, $F_{\text{MIN}} = \{ [q] \mid q \in F \}$

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

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

Graphical representation of a minimized automaton with states $q_0$, $q_1$, and $q_2$. Transition arrows indicate the input symbols and the corresponding states.
Thm: $M_{\text{MIN}}$ is the unique minimal DFA equivalent to $M$

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

If $M'$ is a minimal DFA, then $M'$ has no inaccessible states and is irreducible. So the Claim implies:

*If $M'$ is a minimal DFA for $M$, then there is an isomorphism between $M'$ and $M_{\text{MIN}}$. So the Thm holds!*

Corollary: If $M$ has no inaccessible states and is irreducible, then $M$ is minimal.

Proof: Let $M^{\text{min}}$ be minimal for $M$. Then $L(M) = L(M^{\text{min}})$, no inaccessible states in $M$, and $M$ is irreducible. By Claim, both $M^{\text{min}}$ and $M$ are isomorphic to $M_{\text{MIN}}$!
Suppose we have proved the Claim is true. Assuming the Claim we can prove the Thm:

Proof of Thm: Let M’ be any minimal DFA for M. Since M’ is minimal, M’ has no inaccessible states and is irreducible (why?) By the Claim, there is an isomorphism between M’ and the DFA $M_{\text{MIN}}$ that is output by MINIMIZE(M). That is, $M_{\text{MIN}}$ is isomorphic to every minimal M’.

**Thm:** $M_{\text{MIN}}$ is the **unique** minimal DFA equivalent to M

**Claim:** Let M’ be a DFA where $L(M’)=L(M_{\text{MIN}})$ and M’ has no inaccessible states and M’ is irreducible. Then there is an *isomorphism* between M’ and $M_{\text{MIN}}$.
Thm: $M_{\text{MIN}}$ is the unique minimal DFA equivalent to $M$

Claim: Let $M'$ be a DFA where $L(M')=L(M_{\text{MIN}})$ and $M'$ has no inaccessible states and $M'$ is irreducible. Then there is an isomorphism between $M'$ and $M_{\text{MIN}}$

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

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

Recursive Step: If $p \mapsto p'$

Then $q \mapsto q'$
Base Case: $q_{0 \text{MIN}} \rightarrow q_0'$

Recursive Step: If $p \rightarrow p'$

\[ \sigma \quad \sigma \]

Then $q \rightarrow q'$
Base Case: \( q_{0_{\text{MIN}}} \mapsto q_0' \)

Recursive Step: If \( p \mapsto p' \)

\[
\begin{align*}
\sigma & \quad \sigma \\
\downarrow & \quad \downarrow \\
q & \quad q'
\end{align*}
\]

Then \( q \mapsto q' \)

Claim: Map 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 \( \delta_{\text{MIN}}(p, \sigma) \mapsto \delta'(p', \sigma) \)

(*this follows from the definition of the map*)
Base Case: $q_{0, \text{MIN}} \mapsto q_0'$

Recursive Step: If $p \mapsto p'$

\[
\begin{array}{c}
\sigma \\
\downarrow \\
q
\end{array}
\quad
\begin{array}{c}
\sigma \\
\downarrow \\
q'
\end{array}
\quad \text{Then } q \mapsto q'
\]

The map is defined everywhere

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

If $q \in M_{\text{MIN}}$, there is a string $w$ such that

$\Delta_{\text{MIN}}(q_{0, \text{MIN}}, w) = q$

Let $q' = \Delta'(q_0', w)$. Then we claim $q \mapsto q'$

*(prove by induction on $|w|$)*
Suppose there are states $q'$ and $q''$ such that $q \rightarrow q'$ and $q \rightarrow q''$

Suppose $q'$ and $q''$ are distinguishable

Contradiction!
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 \( q' = q'' \) (why?)
Base Case: $q_{0, \text{MIN}} \mapsto q_0'$

Recursive Step: If $p \mapsto p'$

\[
\begin{array}{c}
\sigma \\
\downarrow \\
q
\end{array}
\quad
\begin{array}{c}
\sigma \\
\downarrow \\
q'
\end{array}
\]

Then $q \mapsto q'$

The map is onto

Want to show: For all states $q'$ of $M'$ there is a state $q$ of $M_{\text{MIN}}$ 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 $M_{\text{MIN}}$ after reading in $w$

Claim: $q \mapsto q'$ \textit{(prove 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

![Diagram](image-url)
How can we prove that two regular expressions are equivalent?
The Myhill-Nerode Theorem
In DFA Minimization, we defined an equivalence relation between states of a DFA. We can also define a similar equivalence relation over strings for a language:

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


\[
x \equiv_L y \iff \text{for all } z \in \Sigma^*,xz \in L \iff yz \in L
\]

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

Claim: $\equiv_L$ is an equivalence relation

Proof? Same as before!
Let $L \subseteq \Sigma^*$ and $x, y \in \Sigma^*$

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

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 DFA for $L$.

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

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 $xz \in L \iff yz \in L$, 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 $z \in \Sigma^*$, $xz \in L \iff yz \in L$

$(\iff)$ 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 whose states are the equiv classes of $\equiv_L$

Define a DFA $M$ where:

$Q$ is the set of equivalence classes of $\equiv_L$

$q_0 = \left[\epsilon\right] = \{y \mid y \equiv_L \epsilon\}$

$\delta([x], \sigma) = [x \sigma]$

$F = \{[x] \mid x \in L\}$

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 $\equiv_L$

L is not regular if and only if
There are infinitely many strings $w_1, w_2, \ldots$ 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 $w_iz$ and $w_jz$ 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 $L$
So all pairs of strings in $S$ are distinguishable to $L$

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