Minimal State DFA

Theorem: The minimum state DFA for a regular language \( L \) is unique up to isomorphism (renaming of states).

This is a corollary of the Myhill-Nerode theorem and corresponds to the equivalence classes of \( R_L \).

Given any DFA \( M \) for \( L \), if it is completely contained with same equivalence class \( \equiv \) of \( M' \) (the machine for \( R_L \)), it may not possible.

For any machine \( M \), \( \# \text{states of } M' \leq \# \text{states of } M \)

Consider a machine \( M \) which the same no of states as \( M' \).
We want to map the states of $M$ to $M'$.

Let $q$ be a state of $M$, then clearly for some $x \in \Sigma^*$

$\delta(q_0, x) = q$ (otherwise $M$ is not min state)

$q \rightarrow [x]$ (corresponding to $R_c$)

for consistency verify that for any other $y$ s.t. $\delta(q_0, y) = q$

$[y] = [x]$

Since $x$ and $y$ belong to the same state they are in the same equiv

clan of $R_M$ $\Rightarrow$ they are in same equiv clan of $R_c$

To complete the proof show that

$\delta(q, a) = \delta([x], a) = [xa]$

(then in how $M'$ was constructed)
Constructing min state DFA

Since this is related to $R_L$, which is above string $x,y$ behaving identically with respect to acceptance / non-acceptance for any concatenate string $z$, no accepting state can be equal to non-accepting state because it must satisfy $R_L$ for $z = \varepsilon$

Consider a pair of state $p,q$

We cannot club $p$ and $q$ if for some string $z \in \Sigma^*$, $\delta(p,z) \in Q - F$ and $\delta(q,z) \in F$ or vice versa

\[ \Rightarrow \] If $(p',q')$ are not eqv then $p,q$ are not eqv if $\delta(p,a) = p'$ and $\delta(q,a) = q'$
Define a graph whose vertices correspond to $\mathbb{Q} \times \mathbb{Q}$ and edges correspond to

$$
\delta(p, a) = \Phi,
\delta(q, a) = \Psi,
$$

and the degree of a vertex equals $|E|=k$.

Initially 'mark' all pairs known to be not equivalent, viz $\exists F \times (\mathbb{Q}-F)$

A special case of "coarsest partitioning problem"
Try \# m-chip

Only \((A, B)\) can be clubbed

\[
\begin{align*}
(A, B) \xrightarrow{0} & (B, B) \\
(A, B) \xrightarrow{1} & (C, C)
\end{align*}
\]

Since the states are same in both cases, they cannot be distinguished by any thing.

Therefore, \(A, B\) can be clubbed.

The final Markov state automaton will be

The transitions of the new machine can be thought of as graph-contraction, i.e., once vertices are merged, the edges are between merged vertices. Argue why we cannot have a situation after merging vertices, i.e., no non-deterministic transitions.