COL 352 Lecture 8 Jan 17

Relationship between NFA and DFA

Q.1 Given a DFA M, can we design an NFA N so that $L(M) = L(N)$

Note every DFA is also an NFA by defn so trivially true

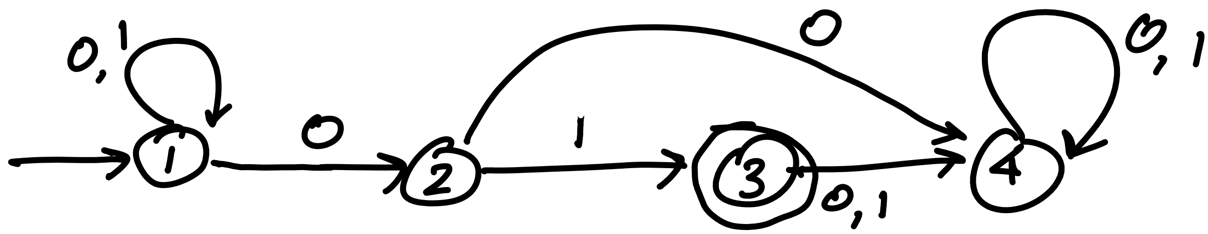
Q2 Given an NFA N, can we design a DFA M so that $L(M) = L(N)$?

This is not obvious, so let us consider an example

$$L_{01} = \{ w \in \{0,1\}^* \mid w \text{ ends with } 01 \}$$

It is easy to construct an NFA for L_{01}

NFA



DFA construction

{1}

	0	1
→ [1]	[1, 2]	[1]
[2]	[4]	[3]*
[3]*	[4]	[4]
[4]	[4]	[4]
[1, 2]	[1, 2, 4]	[1, 3]
[1, 3]*	[1, 2, 4]	[1, 4]
[1, 4]	[1, 2, 4]	[1, 4]
[2, 3]	[4]	[3, 4]

	0	1
[2, 4]	[4]	[3, 4]
[3, 4]	[4]	[4]
[1, 2, 3]*	[1, 2, 4]	[1, 3, 4]*
[2, 3 4]	[4]	[3, 4]
[1, 3, 4]*	[1, 2, 4]	[1, 4]
[1, 2, 4]	[1, 2, 4]	[1, 3, 4]
~~[1, 2, 3, 4]*~~	~~[1, 2, 4]~~	~~[1, 3, 4]~~

For any subset $P \subseteq Q$, we denote a state by $[P]$. Note that $[1, 2]$ is the same as $[2, 1]$ (ordering doesn't matter)

Since [1, 2, 3, 4] cannot be reached, we can eliminate it

The above construction can be generalized to any NFA. The new DFA can be represented as

$Q' = 2^Q$ where Q : set of states in NFA

$q_0' = [q_0']$

$F' = [P]$ s.t $P \cap F \neq \phi$ where F is the set of final states in NFA

$\delta' : Q' \times \Sigma \to Q'$ which is the same as $2^Q \times \Sigma \to 2^Q$

in particular $\delta'([P], a) = \hat{\delta}(P, a)$ of the NFA

$P \subseteq Q$, $a \in \Sigma$

Claim: $\delta'([P], w) = \hat{\delta}(P, w)$
$\forall$ $P \subseteq Q$ and $w \in \Sigma^*$

Proof: by induction on $|w|$

Try to write out a formal proof using the construction

Corollary: $\delta'([q_0'], w) \in F'$ in DFA M

iff $\hat{\delta}(q_0, w) \cap F$ in NFA N

from the previous claim and the definition of final states in DFA

$\Rightarrow$ For every NFA N, there exists a DFA M

The construction can be done using an algorithm that takes in NFA N as input and outputs the equivalent DFA

The representation of the NFA / DFA is exactly the tuple $M = (Q, q_0, F, \delta)$ where δ itself is a set of tuples (q, a, p) where $\delta(q, a) = p$