Deterministic F.A.

\[ q_i \xrightarrow{a} q_j \]

Non-Det. F.A.

\[ \delta(q_i, a) = q_j \]

Serial Computation

\[ \delta(q_i, \varepsilon) = \varepsilon \]

\[ \delta(q_i, \varepsilon) = \{ q_j, q_k, \ldots \} \]

Parallel or "Multi-threaded" Computation

\[ q_0 \to q_1 \to q_2 \to q_3 \to \text{ACC/REJ} \]

Accept if any of the final states accepts.

At every level of the tree, the machine is simultaneously in all states.
$$S = \{0, 1\}$$

$$L = \{w \mid \text{second to last digit} = 0\}$$

$$= \{w \mid w = x1a, \quad x \in \Sigma^*, \quad a \in \Sigma\}$$

**DFA**

What are the possible combinations for last 2 digits?

$q = \{00, 01, 10, 11\}$

Make states for each possibility

$$f: \Gamma \times \Sigma \rightarrow 2 \quad \text{(output is some subset of}\ \{q\})$$

**NFA**

Every time you get a 1, "guess" that it's the next to last digit.

**Example** $w = 10$

**Example** $w = 101$

$$b_2 \text{ has no transitions out means on any input it goes to the empty set. (if "dies")}$$

$G = 10$

**We accept 10**

**We reject 101**
Formal definition of NFA:

$NFA \ N = (Q, \Sigma, \delta, q_0, F)$

- $Q =$ Finite set of states
- $\Sigma =$ Finite alphabet
- $\delta: Q \times \Sigma \cup \{ \varepsilon \} \rightarrow 2^Q$ (output is some subset of $Q$)
- $q_0 \in Q$ Start state
- $F \subseteq Q$ Accept states

Example:

$NFA \ N_1$

$\delta$

$\begin{array}{c|c|c|c}
0 & 1 & \varepsilon \\
\hline
q_0 & q_0, q_2 & q_0, q_2 \\
q_1 & q_2 & q_2 \\
q_2 & \emptyset & \emptyset & \emptyset \\
\end{array}$

$q_0 = q_0$

$F = \{ q_2 \}$
Class: DFA = \{ L \mid L = L(M) \text{ for some DFA } M \}

Class: NFA = \{ L \mid L = L(N) \text{ for some NFA } N \}

Question: \text{ Class: DFA } \leq \text{ Class: NFA }

i.e. Can you rewrite any DFA as an NFA?

TM: \text{ Class: DFA } \leq \text{ Class: NFA }

Proof: Let \( L \in \text{ DFA } \Rightarrow L = L(M) \text{ for some DFA } M \)

\( M = (Q, \Sigma, \delta, q_0, F) \)

Define NFA \( N = (Q, \Sigma, \delta', q_0, F) \)

If \( \delta(q_i, a) = q_j \)

Define:

\( \delta'(q_i, a) = \epsilon q_j \)

\( \delta'(q_j, \epsilon) = \emptyset \)

Question: NFA \leq DFA ?

i.e. Can you rewrite any NFA as a DFA?

If you can turn each level of tree into states of DFA, then yes.

Ponder this.