Non Deterministic Finite Automata

Beulah A.

AP/CSE

DFA vs NFA

In a *DFA,*

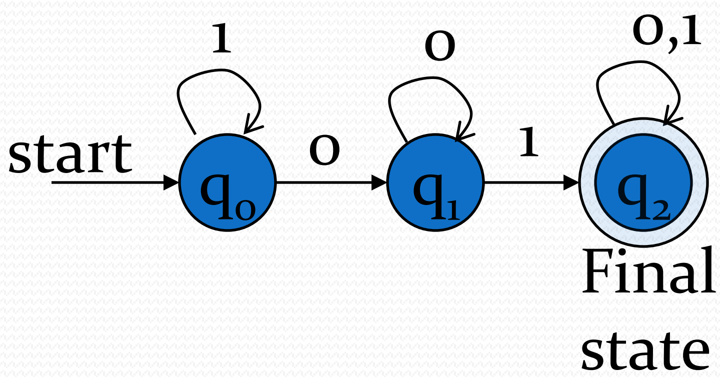
- Each symbol causes a move (eventhough the state of the machine remains unchanged after the move)
- The next state is completely determined by the current state and current symbol.

Where as in a *NFA*

- The machine can move without consuming any symbols and sometimes there is no possible moves and sometimes there are more than one possible moves.
- The state is only partially determined by the current state and input symbol.

Example

DFA for strings containing 01



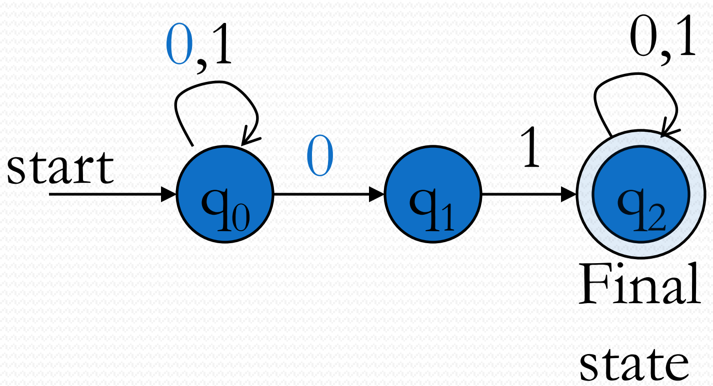
- $Q = \{q_0, q_1, q_2\}$
- $\Sigma = \{0, 1\}$
- $S = q_0$
- $F = \{q_2\}$
- Transition table

states

δ	0	1
→ q_0	q_1	q_0
q_1	q_1	q_2
$*q_2$	q_2	q_2

Example

NFA for strings containing 01



- $Q = \{q_0,q_1,q_2\}$
- $\Sigma = \{0,1\}$
- $S = q_0$
- $F = \{q_2\}$
- Transition table

symbols

δ	0	1
→q_0	$\{q_0,q_1\}$	$\{q_0\}$
q_1	Φ	$\{q_2\}$
$*q_2$	$\{q_2\}$	$\{q_2\}$

states

NFA Specification

- A Non Deterministic finite automata (NFA) is a 5-tuple $(\mathcal{Q}, \Sigma, S, F, \delta)$ where
  - $\mathcal{Q}$ is a finite set of states
  - Σ is a set of alphabets
  - $\mathcal{S}$: $q_0 \in \mathcal{Q}$ is the initial state
  - $F \subseteq \mathcal{Q}$ is a set of accepting states (or final states)
  - $\delta: \mathcal{Q} \times \Sigma \rightarrow 2^{\mathcal{Q}}$ is a transition function

    $2^{\mathcal{Q}}$ is power set of $\mathcal{Q}$

Extended Transition Function (δ)

- Basis : $\bar{\delta}(q, \varepsilon) = \{q\}$

- Induction : $\bar{\delta}(q, wa) = \bigcup_{P \in \bar{\delta}(q,w)} \delta(P, a)$

for each $w \in \Sigma^*$, $a \in \Sigma$ and $P \in (q, w)$

Language of a NFA

- Language accepted by NFA is

$$L(A) = \{w : \bar{\delta}(q_0, w) \cap F \neq \varphi\}$$

Equivalence of DFA and NFA

- As every DFA is an NFA, the class of languages accepted by NFA's includes the class of languages accepted by DFA's.
- DFA can simulate NFA.
- For every NFA, there exist an equivalent DFA.

Theorem

- For every NFA, there exists a DFA which simulates the behavior of NFA. If L is the set accepted by NFA, then there exists a DFA which also accepts L.

Proof

- Let M= ($\mathcal{Q}$, Σ, q_0, F, δ) be NFA accepting L we construct DFA M^1 = ($\mathcal{Q}^1$, Σ, q_0^1, F^1, δ^1), where

- $\mathcal{Q}^1 = 2^{\mathcal{Q}}$ (power set of $\mathcal{Q}$) (any state in $\mathcal{Q}^1$ is denoted by [q_1, q_2, ...q_i] where q_1, q_2, ... $q_i \in \mathcal{Q}$)
- $q_0^1 = [q_0]$
- F^1 is set of final states.

Proof Cont…

- As M (NFA) starts with initial state q_0. q_0^1 is defined as $[q_0]$.
- In M^1 (DFA) the final state (F^1) can be subset of Q containing all final states of F.
- Now we define

$$\delta^1([q_1, q_2, \ldots\ldots q_i], a) = \delta(q_1, a) \cup \delta(q_2, a) \cup \ldots \delta(q_i, a)$$

equivalently,

$$\delta^1([q_1, q_2, \ldots q_i], a) = [p_1, p_2, \ldots p_j]$$

if and only if

$$\delta(\{q_1, q_2, \ldots q_i\}, a) = \{p_1, p_2, \ldots p_j\}$$

Proof by Induction

Input string x

$\delta^1(q_0^{\,1}, x) = [p_1, p_2,\ldots.p_j]$

if and only if

$\delta(q_0, x) = \{p_1, p_2,\ldots.p_j\}$

Basis

- The result is trivial if string length is 0 i.e., $|x| = 0$
- since $q_0^{\,1} = [q_0]$. x must be ε

Proof by Induction

Induction

- Suppose the hypothesis is true for inputs of length m.
- Let xa be a string of length m +1 with a in Σ.

  Then $\delta^1(q_0^{\,1}, xa) = \delta^1(\delta^1(q_0^{\,1},x), a)$
- By induction hypothesis

  $$\delta^1(q_0^{\,1}, x) = [p_1,p_2\ldots\ldots\ldots\ldots p_j]$$
- if and only if

  $$\delta(q_0, x) = \{p_1,p_2\ldots\ldots\ldots\ldots p_j\}$$

Proof by Induction

- By definition of δ^1

  $$\delta^1([p_1,p_2 \ldots\ldots\ldots\ldots\ldots p_j], a) = [r_1,r_2 \ldots\ldots\ldots\ldots\ldots r_k]$$

- if and only if

  $$\delta(\{p_1,p_2 \ldots\ldots\ldots\ldots\ldots p_j\}, a) = \{r_1,r_2 \ldots\ldots\ldots\ldots\ldots r_k\}$$

- Thus

  $$\delta^1(q_0{}^1, xa) = [r_1,r_2 \ldots\ldots\ldots\ldots\ldots r_k]$$

- if and only if

  $$\delta(q_0, xa) = \{r_1,r_2 \ldots\ldots\ldots\ldots\ldots r_k\}$$

- which establishes the inductive hypothesis.
- Thus L(M) = L(M1)

Summary

- Definition of Non-deterministic Finite Automata
- Transition diagram, transition function and properties of transition function of NFA
- Equivalence of DFA and NFA

Test Your Knowledge

- Design a NFA that accepts input string 0's and 1's that ends with 11
- Design a NFA over {0,1} to accept strings with 3 consecutive 0's

Reference

- Hopcroft J.E., Motwani R. and Ullman J.D, “Introduction to Automata Theory, Languages and Computations”, Second Edition, Pearson Education, 2008