3

May 11, 2016

1 Non-deterministic Finite Automata (NFA)

Def: an NFA is a 5-tuple $N = (Q, \Sigma, \delta, q_0, F)$ where:

- Q is a finite, nonempty set of states
- Σ is an alphabet
- $q_0 \in Q$ is the start state
- $F \in Q$ is the set of accepted states
- δ is the transition functino which has the form: $\delta : Q \times (\Sigma \cup \{\epsilon\}) \to P(Q)$

Def: an NFA $N = (Q, \Sigma, \delta, q_0, F)$ *accepts* a string $\omega \in \Sigma^*$ if there exists 1. $m \in \mathbb{N}$ 2. $r_0, ...r_m \in Q$ 3. $\sigma_1, ...\sigma_m \in \Sigma \cup \{\epsilon\}$
such that

1. $r_0 = q_0$
2. $r_m \in F$
3. $\omega = \sigma_1 ... \sigma_m$
4. $r_{k+1} \in \delta(r_k, \sigma_{k+1})$

for $k = 0...m - 1$

1.1 ϵ-closure

$R \subseteq Q$

The ϵ-closure of R is $\epsilon(R) =$ set of all states that can be reached from any state in R by following zero or more ϵ-transitions

$\delta^*(q, \epsilon) = \epsilon(\{q\})$

$\delta^*(q, \omega\sigma) = \epsilon(\cup_{r \in \delta^*(q,\omega)} \delta(r, \sigma))$

for all q, ω, σ

$L(N) = \{\omega \in \Sigma^* : N \text{ accepts } \omega\}$

$= \{\omega \in \Sigma^* : \delta^*(q_0, \omega) \cap F \neq \emptyset\}$

Thm: Let Σ be an alphabet and let $A \subseteq \Sigma^*$. The language A is regular iff $A = L(N)$ for some NFA N.

Proof pt 1: regular $\implies A = L(N)$ for an NFA N.
$A = L(m)$ for a **DFA** $m = (Q, \Sigma, \delta, q_0, F)$
Then let $N = (Q, \Sigma, \eta, q_0, F)$ where
$\eta(q, \sigma) = \{\delta(Q, \sigma)\}$
$\eta(q, \epsilon) = \emptyset$
for all $q \in Q, \sigma \in \Sigma$.

Proof pt 2: $A = L(N)$ for an NFA $N \implies$ regular.
$N = (Q, \Sigma, \eta, q_0, F)$
Define $m = (P(Q), \Sigma, \delta, S, G)$ as:
$\delta(R, \sigma) = \epsilon(\cup_{q \in R} \eta(a, \sigma))$ (subset construction)
$S = \epsilon(\{q_0\}), G = \{R \in P(Q) : R \cap F \neq \emptyset\}$

Goal: find NFA k with 2 states such that $L(K) = $ complement of $L(N)$
Also: show an NFA that takes strings (3rd from the end is a 1) must have >= 8 states to represent it in a DFA.