CS310

Converting NFA to DFA

Sections:1.2 Page 54

September 13, 2006

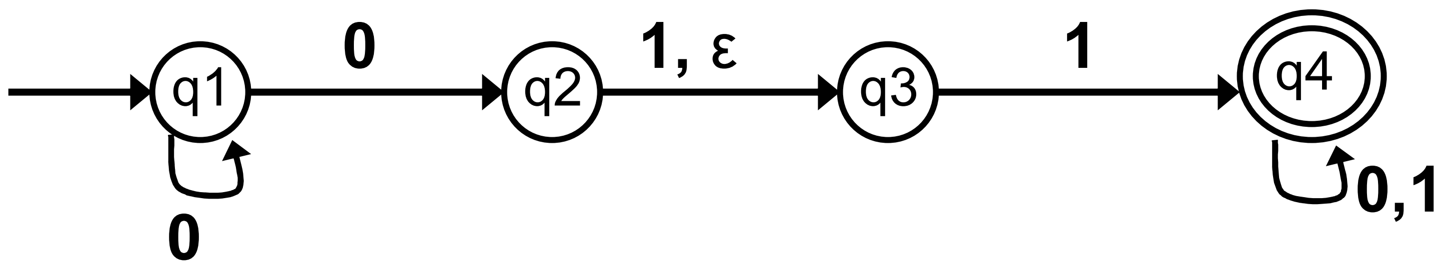
Quick Review

- 5 tuple ($Q, \Sigma, \delta, q_0, F$)

  $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$

  $\delta : Q \times \Sigma_\varepsilon \to P(Q)$ (DFA: $\delta : Q \times \Sigma \to Q$)

  $P(Q)$ is the power set of Q.



Convert NFA to DFA

- Two machines are equivalent if they recognize the same language
- Every NFA has an equivalent DFA (Th 1.39)
- $\delta_{nfa} : Q \times \Sigma_\varepsilon \rightarrow P(Q)$
- The DFA will need to represent all subsets in $P(Q)$ (how many?)
  - let's assume no ε-transitions initially

Convert NFA to DFA

- NFA is $N = (Q, \sum, \delta, q_0, F)$
- DFA is $M = (Q`, \sum`, \delta`, q_0`, F`)$

$Q` = P(Q)$

$q_0` = \{q_0\}$

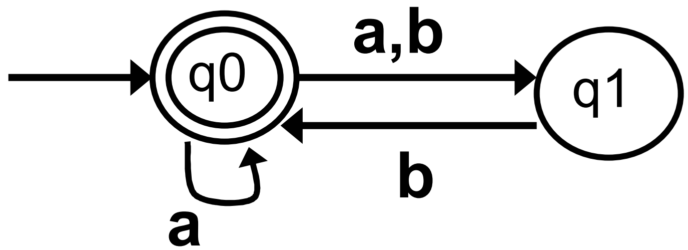
$F` = \{R \in Q` \mid R$ contains an accept state of NFA$\}$

$\delta` : Q` x \sum` \texttt{->} Q`$

$(P(Q) \text{ x } \sum \texttt{->} Q)$

Example (without δ)

NFA



$Q=\{q0,q1\}$

$\Sigma=\{a,b\}$

$Q_0 = q0$

$F = \{q0\}$

δ	a	b
q0	{q0,q1}	{q1}
q1	Ø	{q0}

DFA

$Q` = \{\varnothing,$

$\Sigma` = \{a,b\}$

$Q_0` =$

$F` = \{$

Let's define the $\delta`$

$\delta : Q \times \sum \rightarrow P(Q)$ in NFA

$\delta` : Q` \times \sum \rightarrow Q`$ in DFA

$R \in Q`$, $a \in \sum$

$\delta`(R, a) = \{ q \in Q \mid q \in \delta(r, a) \text{ some } r \in R \}$

Union of all sets that can be reached from a state in set R using the δ with input a

$\delta`(R, a) = \bigcup_{r \in R} \delta(r, a)$

Converting NFA to DFA - ε Transitions

- Define start state and δ\` to include all states that can be reached from a given state by 0 or more ε transitions

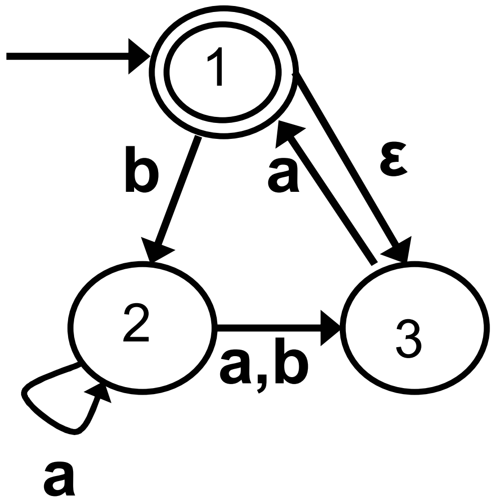
E(R) = { q | q can be reached from R by using 0 or more ε transitions }

$\delta`(R,a) = \{ q \in Q \mid q \in E(\delta(r, a)) \text{ for some } r \in R \}$

$\delta`(R, a) = \{ q \in Q \mid q \in \delta(r, a) \text{ some } r \in R \}$

Conversion Example (with δ)

NFA



$Q=\{1,2,3\}$

$\sum=\{a,b\}$

$Q_0 = 1$

$F = \{1\}$

DFA

$Q`=\{\emptyset,$

$\sum`=\{a,b\}$

$Q_0`=$

$F` = \{$