CS310

Converting NFA to DFA

Sections:1.2 Page 54

September 15, 2006

Quick Review

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

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

  $\delta : Q \times \Sigma_\varepsilon \rightarrow \mathcal{P}(Q)$

  Every NFA has an equivalent DFA (Th 1.39)

  We can convert from NFA to DFA

  - NFAs are often easier to build
  - DFAs are easier to code

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

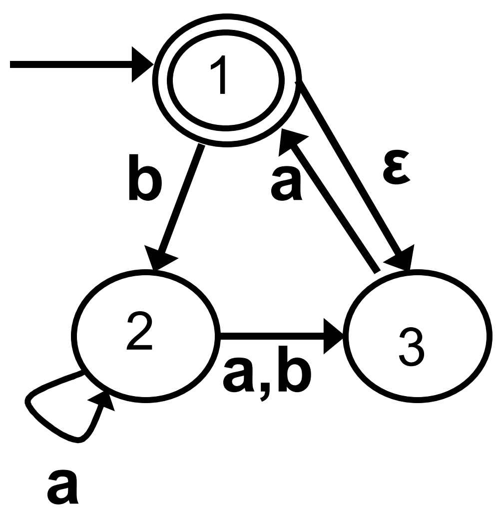
$E(R) = \{ q \mid q$ can be reached from R by using 0 or more ε transitions $\}$

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

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

Conversion Example (with δ)

NFA



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

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

$Q_0 = 1$

$F = \{1\}$

DFA

$Q`=\{Ø,$

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

$Q_0` =$

$F` = \{$

NFA-DFA equivalence

- Th 1.25: Every NFA has an equivalent DFA

  Corollary: A language is regular if and only if there exists an NFA that recognizes it

Proof:

If the language is regular, there exists a DFA that recognizes it. Each DFA is an NFA. Conversely, if there exists an NFA that recognizes the language, convert the NFA to a DFA.

Proof with NFAs

- Theorem 1.25: The class of regular languages is closed under the union operation.
  - We proved this using DFAs
    - What was the computation the new DFA simulated?
  - Is it any easier to prove using NFAs?

Proof with NFAs

- Regular languages are closed under concatenation

  this is where we stopped using DFAs

    what made this hard for DFAs?

Practice

- Construct an NFA to recognize concatenation of DFAs

  $A = \{ w \mid w \text{ contains at least three1s}\}$

  $B = \{ \varepsilon \}$

Proof with NFAs

- Regular languages are closed under Kleene star

  What is Kleene star?

Practice

- Construct an NFA to recognize Kleene star of A if $A = \{ w \mid w$ contains at least two 0s and at most one 1$\}$