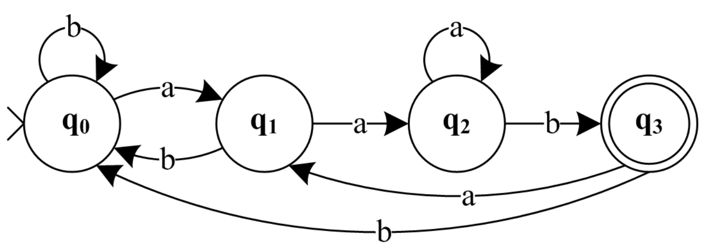
ECS120 Introduction to the Theory of Computation
Fall Quarter 2007

Discussion Notes
Wednesday, October 3, 2007

Minimal DFA Proof

Before trying to prove that your DFA is minimal, you may want to check that it is minimal! You can do this in JFLAP (as shown in discussion).

To prove that it is minimal, we will be using the proof by contradiction and the pigeonhole principle. For example, lets provide and prove a minimal DFA for the language $L = \{w \in \Sigma^* \mid w \text{ ends with } aab\}$ where $\Sigma = \{a, b\}$. We are given the DFA $M = \{Q, \Sigma, \delta, q_0, F\}$ represented by the following state diagram:



Claim: M is a minimal DFA for $L(M)$.

Suppose: For the sake of contradiction, suppose that there exists a smaller DFA $M' = \{Q', \Sigma', \delta', q_0', F'\}$ for L such that $\mid Q' \mid \leq 3$.

Consider the following strings:

$$\begin{aligned} x_1 &= \epsilon \\ x_2 &= a \\ x_3 &= aa \\ x_4 &= aab \end{aligned}$$

These strings are chosen such that the computation of these strings takes us into each of the four states in M. Observe:

$$\begin{aligned} \widehat{\delta}(q_0, \epsilon) &= q_0 \\ \widehat{\delta}(q_0, a) &= q_1 \\ \widehat{\delta}(q_0, aa) &= q_2 \\ \widehat{\delta}(q_0, aab) &= q_3 \end{aligned}$$

By the pigeonhole principle, two of these computations $\widehat{\delta}$ on strings x_1 to x_4 must yield the same state in M'. Therefore, we must show for each pair of computations $\left(\widehat{\delta}(q_0, x_i), \widehat{\delta}(q_0, x_j)\right)$ that:

$$\widehat{\delta}(q_0, x_i) \quad \neq \quad \widehat{\delta}(q_0, x_j)$$

There are $\binom{4}{2}$ cases we must show contradict our assumption. One of these cases is illustrated below:

Case 1: Show contradiction for x_1 and x_2. We start with the following statement:

$$\widehat{\delta}(q_0, \epsilon) \quad = \quad \widehat{\delta}(q_0, a)$$

Notice that by the definition of $\hat{\delta}$ we can pad both sides with the same string without affecting the equality. We pad each string with ab to get:

$$\widehat{\delta}(q_0, ab) \quad = \quad \widehat{\delta}(q_0, aab)$$

However, our language should accept the string aab but not the string ab. Therefore $\widehat{\delta}(q_0, ab) \notin F$ but $\widehat{\delta}(q_0, aab) \in F$. Thus these two computations can not result in the same state, giving us a contradiction. (A state cannot be both accepting and rejecting at the same time!)

By proving each of the cases results in a contradiction, we prove that our DFA is indeed minimal. To recap, the steps to do this are:

1. Create a minimal DFA M.
2. Assume for sake of contradiction that a DFA M' exists with fewer states.
3. Choose strings x_i such that the computation of each string in M results in a different state. Try to keep these strings as simple as possible, and keep in mind which strings the language accepts and rejects.
4. By the pigeonhole principle, at least two of those strings must result in the same state when computed on M'.
5. For each pair of strings x_i and x_j show that (using padding where necessary), $\widehat{\delta}'(q_0', x_i) \in F'$ but $\widehat{\delta}'(q_0', x_j) \notin F'$. Therefore the computation of x_i and x_j cannot result in the same state in M'.

If you show that every case is impossible, then M' cannot exist. Therefore, by proof by contradiction, your original DFA M is minimal.