CS520: Lecture 7 handout
State Minimization in Finite Automata (contd.)

February 14, 2008

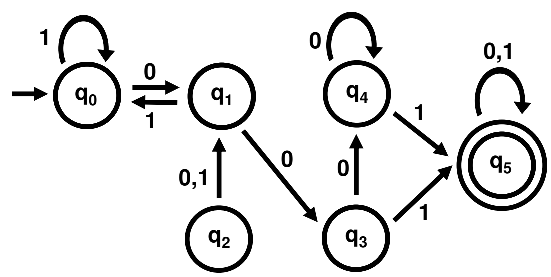
Last time we proved any regular language has a unique minimal (smallest) DFA. We also discussed how to remove redundant states from any given DFA. At the heart of that construction is the algorithmic problem of partitioning the states of a DFA into equivalence classes based on the relation $\sim$.

In this lecture we will design an algorithm for constructing such a partition. At first this seems like a difficult task, given that the distinguishability of two states depends on their behavior on *every* string in Σ^*. Our simple and efficient algorithm is based on the following observations:

- The empty string ϵ distinguishes any accept state from any reject state.
- If a string w distinguishes two states q_1 and q_2, and there exist states p_1 and p_2 with $\delta(p_1, a) = q_1$ and $\delta(p_2, a) = q_2$, then the string aw distinguishes p_1 and p_2.

These facts suggest the following algorithm: we start with noting that every accept state is distinguishable from every reject state; then we "propagate" this distinguishability to other pairs of states by following the transition function backwards from pairs that are already marked as distinguishable.

The algorithm is illustrated as a "table-filling" algorithm for the DFA below.



We first start by removing any states in the DFA that are unreachable from q_0 — in the above DFA, q_2 is unreachable. Then we form a table for the rest of the states. We enter a "d" into the table for any pair of states that are found to be distinguishable. Following the first rule above, the state q_5 is distinguishable from all the other states. We therefore initialize the table as shown in the figure on the left below.

	q_0	q_1	q_3	q_4	q_5
q_0					
q_1					
q_3					
q_4					
q_5	d	d	d	d	

	q_0	q_1	q_3	q_4	q_5
q_0					
q_1					
q_3	d	d			
q_4	d	d			
q_5	d	d	d	d	

	q_0	q_1	q_3	q_4	q_5
q_0					
q_1	d				
q_3	d	d			
q_4	d	d			
q_5	d	d	d	d	

Next we pick any pair of states already found to be distinguishable, say q_0 and q_5. We follow the arrows pointing into these two states backwards labeled by a particular symbol, say 1. We find that three states q_3, q_4 and q_5 have arrows labeled 1 pointing into q_5, and two states q_0 and q_1 have arrows labeled 1 pointing to q_0. Therefore, following the second rule above, all of q_3, q_4 and q_5 are distinguishable from all of q_0 and q_1. We fill in these entries as shown in the figure above in the center.

Next we follow "0" arrows back from q_0 and q_5, but these do not lead to a pair of states that we haven't considered before. We then move on to another pair of states labeled a "d" in the table, say q_1 and q_3. Following the "0" arrows back from these states we find that q_1 and q_0 are also distinguishable. Continuing in this manner for all the entries marked d in the table, we find that all pairs except for (q_3, q_4) are distinguishable. The final table is given above on the right.

Finally, we can construct the minimal DFA equivalent to the one above as given below.