CSE 322 - Introduction to Formal Methods in Computer Science
Minimizing DFAs

Dave Bacon
Department of Computer Science & Engineering, University of Washington

Last time we discussed the Myhill-Nerode theorem:

> **Myhill-Nerode Theorem**
> A is regular if and only if $\equiv_A$ has a finite number of equivalence classes. In addition there is a DFA M with $A = L(M)$ having precisely one state for each equivalence class of $\equiv_A$.

Further we also discussed how the DFA constructed from this theorem, i.e. using the equivalence classes of $\equiv_A$, was the smallest (fewest number of states) DFA which accepted A. Now this is great if you are given a nice description of $\equiv_A$. But what if you aren't given a nice description of A, but instead are given a DFA which recognizes A. Is there a nice way to construct the minimal sized DFA accepting the A?

One way to proceed would be to look at individual states and then see whether they are equivalent under $\equiv_A$. But this is a real pain. So we proceed differently. Rather than starting with individual states and clumping them together, we start with all the states clumped together and then separate them as we go along.

Let $M = (Q, \Sigma, \delta, q_0, F)$ be the DFA we are going to minimize.

To do this we will use an equivalence relation on states. For each $i \geq 0$ define an equivalence relation $\equiv_i$ on the set of states Q via $p \equiv_i q$ if and only if for all strings of length less than i, $z \in \Sigma^*$, $|z| \leq i$, $\delta^*(p, z) \in F$ if and only if $\delta^*(q, z) \in F$.

How will we use this equivalence condition among states? The basic idea is that if this equivalence condition holds for all i, then the inputs which reach the states p and q should all go to the states in the minimal DFA for $L(M)$. Lets formalize this.

Let $A = L(M)$. Let $p, q \in Q$ and suppose that every state of M is reachable from the start state q_0. Then $p \equiv_i q$ for all $i \geq 0$ if and only if the equivalence class of $\equiv_M$ corrsponding to p and q lie in the same equivalence class of $\equiv_A$.

Now lets prove this. First of all if $p \equiv_i q$ for all $i \geq 0$, then for any $z \in \Sigma^*$, if $\delta^*(q_0, x) = q$ and $\delta^*(q_0, y) = p$, then we know that $\delta^*(q_0, xz) \in F$ if and only if $\delta^*(q_0, yz) \in F$. But this means that $xz \in A$ if and only if $yz \in A$. Therefore any two states which reach p and q are in the same equivalence class of $\equiv_A$.

Next suppose that p and q are not equivalent under $\equiv_i$ for some i. Then these is some z with $|z| \leq i$, such that only one of $\delta^*(p, z)$ and $\delta^*(q, z)$ is in F. Since p and q are both reachable from the start state q_0, there are strings x and y such that $\delta(q_0, x) = q$ and $\delta(q_0, y) = p$. Therefore only one of $\delta^*(q_0, xz)$ and $\delta^*(q_0, yz)$ is in F. But this means that only one of xz and yz is in A. This means x and y are in different equivalence classes of $\equiv_A$.

I. MINIMZING DFAS

Okay so given the definition of $\equiv_i$ between states, how do we use this to minimize DFAs? The meta idea is that if we can establish which states are equivalent under $\equiv_i$ for all $i \geq 0$, then these states can be grouped together in the minimal DFA. But how do we show find equivalent states for this seemingly infinite series of $i \geq 0$ conditions? We do this inductively and show that after a finite number of states this induction terminates.

In particular lets begin by finding which states are equivalent under $\equiv_0$. p and q are equivalent under $\equiv_0$ if and only if for $z = \varepsilon$, $\delta * (p, z) \in F$ if and only if $\delta^*(q, z)$. This is equivalent to $\delta^*(p, \varepsilon) \in F$ if and only if $\delta^*(q, varepsilon) \in F$. But this means that p and q are equivalent under $\equiv_i$ if they either both lie in F or both lie in $Q - F$. Thus the two equivalence classes of $\equiv_0$ are F and $Q - F$.

Now lets show how, given the $\equiv_i$ it is easy to figure out $\equiv_{i+1}$. Begin by noting that states which are equivalent under $\equiv_{i+1}$ are equivalent under $\equiv_i$ since they must agree for all string z of length $|z| \leq i$. To also make sure that they agree on strings z' of length $|z'| = i + 1$, we just need to check that after reading the first character of z', we arrive at states that are equivalent under $\equiv_i$. In other words $p \equiv_{i+1} q$ if and only if $p \equiv_i q$ and for all $a \in \Sigma$, $\delta(p, a) \equiv_i \delta(q, a)$.

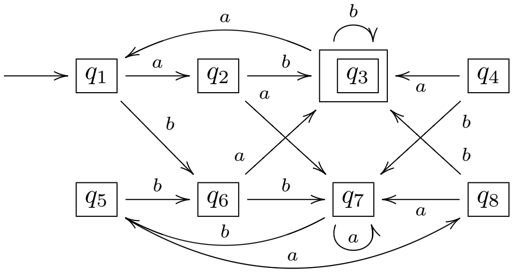
Thus we see how to inductively get $\equiv_{i+1}$ equivalence classes from $\equiv_i$ equivalence classes. But does this process stop? First note that if the equivalence classes of $\equiv_i$ are equal to the equivalence classes of $\equiv_{i+1}$, then the equivalence classes of $\equiv_{i+2}$ will be the same as those of $\equiv_{i+1}$. In other words, once we hit a place where going to the next $\equiv_{i+1}$ yields the same set of equivalent states, then we are done.

Thus we have a way to construct the minimal DFA. We start with $\equiv_0$ and then proceed to construct $\equiv_{i+1}$ from $\equiv_i$ until this repeats. At this point, we stop. We then collapse all of the states which are in the same equivalence class to the same state in our new DFA. Outgoing links are then guaranteed to be correct during this collapse.

Finally we see that this process must always terminate. Why? Because at each step we either increase the number of equivalence classes or we leave the number of equivalence classes the same. Since we start with all of the states in the same equivalence class, this means that after $|Q|-1$ states we must eventually terminate.

II. EXAMPLE

Lets do an example. Consider the DFA

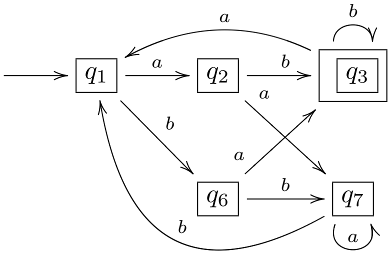


To start with we remove q_4 since it is not reachable. Then after applying the equivalence of states for $\equiv_0$ the two equivalence clases are $\{q_1, q_2, q_5, q_6, q_7, q_8\}$ and $\{q_3\}$.

Now what happens? Well lets look at states in the first set and see where they go. Under an a, $q_1 \to q_2$, $q_2 \to q_7$, $q_5 \to q_8$, $q_6 \to q_3$, $q_7 \to q_7$, and $q_8 \to q_7$. Thus only q_6 is distinguishable by reading a from the other states in $\{q_1, q_2, q_5, q_6, q_7, q_8\}$. What about upon reading b? Well if we go through the transitions out of $\{q_1, q_2, q_5, q_6, q_7, q_8\}$, q_2 and q_8 both have transitions to q_3. Thus the sets of distinguishable states under $\equiv_1$ are $\{q_1, q_5, q_7\}$, $\{q_2, q_8\}$, $\{q_3\}$ $\{q_6\}$

Continuing onward, consider $\{q_1, q_5, q_7\}$. Under b, q_1 and q_5 go to q_6, while q_7 goes to q_5. q_1 and q_5 have a transitions to q_2 and q_8 respectively, and q_7 has a transition to q_5. Therefore, $\{q_1, q_5\}$ are indistinguishable at this point, but q_7 is distinguishable. Now consider $\{q_2, q_8\}$. Both have a transitions to elements of $\{q_1, q_5, q_7\}$, and b transitions to q_3. Thus they are still indistinguishable. Note we do not need to check sets with only one state. Thus under $\equiv_2$ the equivalence classes are $\{q_1, q_5\}$, $\{q_7\}$, $\{q_2, q_8\}$, $\{q_3\}$, $\{q_6\}$.

Finally if we examine the sets $\{q_1, q_5\}$ and $\{q_2, q_8\}$ we see that they have no transitions which distinguish them. This means that under $\equiv_3$ the equivalence classes are the same as those in $\equiv_2$. Thus we halt the procedure. In the new minimized DFA we construct merge $\{q_1, q_5\}$ into one state as well as $\{q_2, q_8\}$. The resulting minimized DFA is


q_1
q_2
q_3
q_6
q_7
a
b