CS481F01 Proving L = L(M)

A. Demers

14 Sept

Here are some hints on proving a NFA recognizes a language L. Doing this for a DFA is similar, and usually easier.

In general there can be a lot of creativity involved in defining a machine to recognize a given language L and then proving it correct. The machine definition and proof are not independent – som machines recognizing L may be much easier to prove correct than other machines.

Suppose you have a language L and a NFA

$$N \quad = \quad (Q, \Sigma, \Delta, S, F)$$

and you need to show $L \ = \ L(N)$. Of course that means you must show

$$\begin{array}{l} x \in L \quad \Rightarrow \quad (\exists s \in S, f \in F)(f \in \hat{\Delta}(\{s\}, x)) \qquad \text{and} \\ (\exists s \in S, f \in F)(f \in \hat{\Delta}(\{s\}, x)) \quad \Rightarrow \quad x \in L \end{array}$$

Intuitively, you usually do this by associating each state $q \in Q$ with some property P_q of the input strings that can cause N to reach state q. You then show that L is exactly the set of strings for which P_f is true for some final state f.

More formally, define a predicate

$$P \ \subseteq \ Q \times \Sigma^*$$

and show the following two things:

$$\begin{array}{lc} (a) & q \in \hat{\Delta}(S, x) \quad \Leftrightarrow \quad P(q, x) \\ (b) & x \in L \quad \Leftrightarrow \quad (\exists q \in F) P(q, x) \end{array}$$

Clearly this shows that $L \ = \ L(N)$, with the predicate P used to characterize the strings that can reach the various states of Q. The proof of (a) generally uses induction on x and possibly cases on q. The proof of (b) is more *ad hoc*, but is usually easy, assuming you have designed Q and Δ carefully.

So here is an example. Let $\Sigma = \{0,1\}$, and let L_k be strings over Σ^* that end with at least k consecutive 1's. That is,

$$L_k \quad = \quad \{\ x1^k \mid x \in \Sigma^*\}.$$

A simple NFA recognizing L_k is

$$\begin{array}{lcll} N_k & = & (\{q_i | 0 \leq i \leq k\}, \Sigma, \Delta_k, \{q_0\}, \{q_k\}) & \\ \Delta_k(q_i, 0) & = & \{q_0\} & 0 \leq i \leq k \\ \Delta_k(q_j, 1) & = & \{q_j, q_{j+1}\} & 0 \leq j < k \\ \Delta_k(q_k, 1) & = & \{q_k\} & \end{array}$$

Drawing the state diagram is left as an exercise, since it's much easier than typesetting the state diagram. Note that some of the edges in the state diagram are not absolutely necessary – for example

$$q_1 \ \longrightarrow_1 \ q_1$$

does not change the recognized language, but it simplifies the formal description of the machine.

Intuitively, the machine is correct because the index i of state q_i is just the minimum number of consecutive 1's that must be seen in order to reach the state. We capture this by the predicate

$$P(q_i, x) \quad \equiv \quad (\exists y)(x \ = \ y1^i)$$

Then our proof obligation (a) from above becomes

$$(a_k) \qquad q_i \in \hat{\Delta}(\{q_0, x\}) \quad \Leftrightarrow \quad (\exists y)(x \ = \ y1^i)$$

and (b) becomes

$$\begin{array}{ll} (b_k) & x \in L_k \quad \Leftrightarrow \quad (\exists q \in F) P(q, x) \\ & \Leftrightarrow \quad P(q_k, x) \\ & \Leftrightarrow \quad (\exists y)(x \ = \ y1^k) \end{array}$$

We need to prove both of these. Of course, (b_k) is easy – if we just substitute in the definition of L_k we get

$$x \in \{\ y1^k \mid y \in \Sigma^*\} \quad \Leftrightarrow \quad (\exists y)(x \ = \ y1^k)$$

which is immediate.

A formal proof of (a_k) uses induction on $|x|$ and, if this machine were more complex, could require explicit cases on q in the basis and inductive steps. Here is a proof:

Basis: $(x = \epsilon)$. For the basis, $P(q_i, \epsilon)$ is clearly true *iff* $i = 0$. Since

$$\hat{\Delta}(S, \epsilon) \quad = \quad S \quad = \quad \{q_0\}$$

the result is immediate.

Inductive Step: $(x = x'0)$. In this case,

$$\begin{aligned} \hat{\Delta}(S, x'0) \quad &= \quad \bigcup\{\ \Delta(q, 0) \mid q \in \hat{\Delta}(S, x')\ \} \quad = \quad \{\ q_0\ \} \\ &= \quad \{\ q_i \mid \exists y\ x'0 = y1^i\ \} \\ &= \quad \{\ q \mid P(q, x)\ \} \end{aligned}$$

as required.

Inductive Step: $(x = x'1)$. For this case, define $\oplus$ by

$$m \oplus n \quad = \quad \min(m + n, k)$$

so that

$$\Delta(q_i, 1) \quad = \quad \{\ q_i, q_{i \oplus 1}\ \}$$

Now

$$\begin{aligned} \hat{\Delta}(S, x'1) \quad &= \quad \bigcup\{\ \Delta(q_i, 1) \mid q_i \in \hat{\Delta}(S, x')\ \} \\ &= \quad \bigcup\{\ \{q_i, q_{i \oplus 1}\} \mid q_i \in \hat{\Delta}(S, x')\ \} \end{aligned}$$

By the induction hypothesis, this is

$$\begin{aligned} &= \quad \bigcup\{\ \{q_i, q_{i \oplus 1}\} \mid \exists y'\ x' \ = \ y'1^i\ \} \\ &= \quad \bigcup\{\ \{q_i, q_{i \oplus 1}\} \mid \exists y'\ x'1 \ = \ y'1^{i \oplus 1}\ \} \end{aligned}$$

from which the result is immediate. □