NFA to DFA blowup

Chris Calabro

February 27, 2005

1 Basic definitions

A *deterministic finite state automaton* (DFA) is a machine $D = (\Sigma, Q, q_0, F, \delta)$ where Σ is a finite non-empty alphabet, Q is a finite set of states, $q_0 \in Q$ is the start state, $F \subseteq Q$ is the set of accepting states, $\delta : Q \times \Sigma \to Q$ is the transition function; and that a *nondeterministic finite state automaton* (NFA) is a machine $N = (\Sigma, Q, q_0, F, \delta)$ defined in the same way except that $\delta : Q \times \Sigma \to \mathcal{P}(Q)$.

D *accepts* $x \in \Sigma^*$ iff $\exists p_0, \ldots, p_{|x|} \in Q$ such that

$$\begin{aligned} & p_0 = q_0 \\ & \forall i \in \{1, \ldots, |x|\}\ p_i = \delta(p_{i-1}, x_i) \\ & p_{|x|} \in F, \end{aligned}$$

and similarly, N *accepts* x iff $\exists p_0, \ldots, p_{|x|} \in Q$ such that

$$\begin{aligned} & p_0 = q_0 \\ & \forall i \in \{1, \ldots, |x|\}\ p_i \in \delta(p_{i-1}, x_i) \\ & p_{|x|} \in F. \end{aligned}$$

The *language* $\mathcal{L}(D)$ ($\mathcal{L}(N)$) of D (N) is the set of strings that D (N) accepts. 2 machines are *equivalent* iff they have the same language. We can say that D is *minimal* iff every DFA D' equivalent to D has at least as many states as D has.

2 NFAs are exponentially more powerful than DFAs

It is sometimes said that NFAs are no more powerful than DFAs, but this is in reference to computability. With respect to complexity, NFAs and DFAs are quite different.

Lemma 1. *For each $n \in \mathbb{N}$, there is an $n + 1$ state NFA whose minimal equivalent DFA has $\geq 2^n$ states.*

Proof. Let $\Sigma = \{0, 1\}$ and let $L = \{x1y \mid x \in \Sigma^*, y \in \Sigma^{n-1}\}$. Then L is the language of the NFA N that we now describe graphically. N has the states $0, \ldots, n$, where 0 is the start state, and an edge from $i-1$ to i for $i \in \{1, \ldots, n\}$ and an edge from 0 to 0. The edge from 0 to 0 is labeled $0, 1$, the edge from 0 to 1 is labeled 1, and the edge from $i-1$ to i is labeled $0, 1$ for $i \in \{2, \ldots, n\}$. The only accept state is n.

Now suppose indirectly that some DFA D with $k < 2^n$ states has language L. Then by the pigeonhole principle, there are some 2 distinct strings $x, y \in \Sigma^n$ such that D ends up in the same state when run on x as when run on y. x, y are different, so there is some $i \in \{1, \ldots, n\}$ such that $x_i \neq y_i$ and wlog suppose $x_i = 0, y_i = 1$. Let z be any string of length $i - 1$, say $z = 1^{i-1}$. Then $xz \notin L, yz \in L$ and yet $D(xz) = D(yz)$, a contradiction.

Note also that the standard construction transforming an NFA into a DFA will produce a DFA D equivalent to N and with 2^{n+1} states. □

3 NFAs might only reject long strings

Fix $\Sigma = \{0, 1\}$ and let

$$\text{All}_{\text{NFA}} = \{\langle N \rangle \mid N \in \text{NFA}, N \text{ has alphabet } \Sigma, \mathcal{L}(N) = \Sigma^*\}.$$

Also let $A_{\text{NFA}} = \{\langle N, x \rangle \mid N \in \text{NFA}, N(x) = acc\}$.

One might think that $\text{All}_{\text{NFA}} \in \text{coNP}$ since $A_{\text{NFA}} \in \text{P}$ and to show that $\langle N \rangle \notin \text{All}_{\text{NFA}}$ it suffices to demonstrate a string that N rejects. The problem with this argument is that it is not obvious how long such a string might be.

The next result shows that an NFA might reject only very long strings.

Lemma 2. *There is a sequence $N_1, N_2, \cdots \in \text{NFA}$ such that N_n has $n^{2+o(1)}$ states and the shortest string that N_n rejects has length $\geq 2^n$.*

Proof. We describe N_n graphically. N will have alphabet $\Sigma = \{1\}$, a start state q_0, and n "loops". The ith loop will consist of a cycle of p_i states where p_i is the ith smallest prime. The edges of the loop will all be labeled 1. A single node in each loop will be a reject state and the others will be accept states. There is an edge labeled ϵ from q_0 to the reject state of each loop. Also q_0 is an accept state.

So $\mathcal{L}(N_n) = \{1^j \mid j = 0 \vee p_1 \nmid j \vee \cdots \vee p_n \nmid j\}$ and $\overline{\mathcal{L}(N_n)} = \{1^j \mid p_1, \ldots, p_n \mid j > 0\}$. But then the shortest string that N_n rejects has length $p_1 \cdots p_n \geq 2^n$.

We now bound the number of states that N_n has. The prime number theorem says that the number of primes $\leq n$ is $\pi(n) \in \Theta(\frac{n}{\ln n})$. So the nth smallest prime $p_n \leq n^{1+o(1)}$ So the number of states that N_n has is $1 + p_1 + \cdots + p_n \leq 1 + np_n \leq n^{2+o(1)}$. □