**The Theory of Computation**
Bernard M.E. Moret
Addison-Wesley-Longman, 1998

*Solution Manual*
*Chapter 3, Problem 3.28*

Since $L$ is regular, it is accepted by some deterministic FA with fully specified transition, say $M = (\Sigma, Q, S, F, \delta)$. Given $M$, we construct a new FA $M'$ that accepts exactly $NP(L)$. $M'$ has the same alphabet, same same start state, and same accepting states as $M$; its state set is $Q \cup \{T\}$; and its transition function $\delta'$ is constructed from that of $M$ as follows:

$$\delta'(q, a) = \begin{cases} \delta(q, a) & \text{if } q \in Q - F \\ T & \text{if } q \in F \text{ or } q = T \end{cases}$$

That is, $M'$ behaves exactly like $M$ on any string until it first reaches an accepting state of $M$; any further input at that point moves $M'$ to the rejecting trap state $T$. Thus any string accepted by $M'$ is also accepted by $M$; moreover, any string $x$ accepted by $M'$ cannot be the prefix of any other string accepted by $M'$, since any input beyond the last character of $x$ will cause $M'$ to enter the rejecting trap. Note that $M'$ as constructed may have many inaccessible states, including inaccessible accepting states.