Lecture #4: DFA Design and Verification — Part Two

What To Do Before the Lecture

1. Complete the preparatory viewing or reading for Lecture #4 — nothing that each video will probably be understandable if you play it at double speed.

2. ***Print*** and read through the rest of this document and — if you have time — try to solve the problems! These should help you to complete applications of the design process that is introduced in the preliminary material for this and the previous lectures.

   Note that a ***supplemental document*** includes completions of design examples that were started in material for the previous lecture. This might be helpful if you are having trouble understanding the design process that you are to use.

   This also includes material that is for interest only, namely, a proof of the technical result, which can be used to prove the correctness of deterministic finite automata, that is introduced at the end of the preparatory material for this lecture.

Problems To Be Solved

DFA Design — The Language To Be Considered

Let $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ and let $L \subseteq \Sigma^\star$ be the following language:

$$L = \{w \in \Sigma^\star \mid \omega \text{ ends with } \mathtt{abb}\}.$$

The problem to be solved — which was also considered in the previous lecture presentation — is to design a ***deterministic finite automaton*** with alphabet Σ, whose language is L, and to prove that this DFA is correct.

Designing a DFA

A First Attempt

We first tried to design a DFA for this language, under the assumption that the only information that the DFA would need to remember, about the string that has been processed so far, is ***whether it belongs to the above language L***.

This would correspond to a deterministic finite automaton with two states, q_0 and q_1, corresponding to the set

$$S_0 = \{\omega \in \Sigma^\star \mid \omega \textit{ does not} \text{ end with "}\mathtt{abb}\text{"}\}$$

— which corresponds to the state q_0 — and the set

$$S_1 = \{\omega \in \Sigma^\star \mid \omega \textit{ does} \text{ end with "}\mathtt{abb}\text{"}\}$$

— which corresponds to the state q_1. Since $\lambda \in S_0$, q_0 is the start state. Since $S_1 = L$ (so that $S_1 \subseteq L$) and $S_0 = \Sigma^\star \setminus L$ (so that $S_0 \cap L = \emptyset$) q_1 should be the only accepting state. That is, $F = \{q_1\}$.

This *first* attempt to design a deterministic finite automaton, for L, ***failed***.

How This Failed:

What Could Be Learned From This:

A Second Attempt

What Must the DFA Remember?

Representation Using Subsets of $\Sigma^\star$:

Initial Sanity Checks:

Identifying Transitions:

A Third Attempt

What Went Wrong — Which Transition was not Well Defined? Why?

What Must the DFA Remember?

Representation Using Subsets of $\Sigma^{\star}$:

Initial Sanity Checks:

Identifying Transitions:

What We Have Now

Writing a Proof Correctness

What Result, from the Lecture Notes, is Useful?

What Things Must Be Established? A detailed list of the properties that should be established is as follows.

Other Things to Think About, or To Do:

Verification vs. Testing