ECE374 SP23 HW3

Contributors

Zhirong Chen (zhirong4)

Ziyuan Chen (ziyuanc3)

Problem 1

A *finite-state transducer* (FST) gives an output based on the transition instead of the current state. It is defined by a 5-tuple:

$$(\Sigma, \Gamma, Q, \delta, s)$$

The output alphabet of a FST_{AR} consists of two signals, namely accept and reject ($\Gamma = \{\mathrm{A}, \mathrm{R}\}$). We say that $L(FST_{AR})$ represents the language consisting of all strings that end with an accept (A) output signal.

Prove that $L(FST_{AR})$ represents the class of regular languages.

Solution

Proof. For the given FST, we construct a DFA M' such that $L(M') = L(FST_{AR})$:

$$M' = (\Sigma, Q', \delta', s', A')$$

where

- $Q' = Q \times \Gamma$
- $\delta'((q, b), a) = \delta(q, a), \forall q \in Q, a \in \Sigma, b \in \Gamma$
- $s' = (s, \mathrm{A})$
- $A' = \{(q, \mathrm{A}) \mid q \in Q\}$

Here we assume that in a FST, $\delta : Q \times \Sigma \to Q \times \Gamma$

The core idea is to explicitly encode the last output signal in the states. Note that there is a clear boundary between *next-state logic* and *output logic* -- thus, the expression of δ' is unrelated to b.