ECE374 Assignment 3

Due 02/13/2023

Group & netid

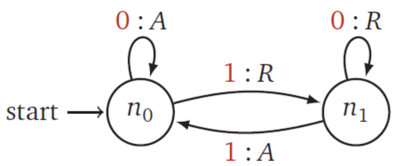
Chen Si *chensi3*

Jie Wang *jiew5*

Shitian Yang *sy39*

Problem 1

1. In a previous lab/homework we talked about a new machine called a *finite-state transducer* (FST). The special part thing about this type of machine is that it gives an output on the transition instead of the state that it is in. An example of a finite state transducer is as follows:



defined by the five tuple: $(\Sigma, \Gamma, Q, \delta, s)$. Let's constrain this machine (call is FST_{AR}) a bit and say the output alphabet consists of two signals: accept or reject ($\Gamma = \{A, R\}$). We say that $L(FST_{AR})$ represents the language consisting of all strings that end with a accept (A) output signal.

Prove that $L(FST_{AR})$ represents the class of regular languages. *Note: We are referring to all possible FST_{AR} not just the one shown above. This is a language transformation task.*

Solution:

I. $FST_{AR} \rightarrow DFA$ = Regular

For a FST_{AR}, we have the following definition:

$FST_{AR} = (\Sigma, \Gamma, Q, \delta, s)$

in which

Σ is the set of input symbols in this language;

$\Gamma = \{A, R\}$ represents whether entering this state via a certain transition is considered accepting;

Q is the set of states;

s is the starting state;

δ is the set of all transitions, where:

$\delta(q_1, a) = (q_2, b), q_1, q_2 \in Q, a \in \Sigma, b \in \Gamma$, which indicates that this FST_{AR} takes in a symbol *a* at state q_1 and transitions to the new state q_2 while outputting a signal indicating whether it's accepting or rejecting.

Therefore, we could construct a DFA $M = (\Sigma', Q', \delta', s', A')$ based on this FST_{AR} that
$\Sigma' = \Sigma$ is the set of input symbols in this language;
$Q' = Q \times \{Accept, Reject\}$ is the set of states that represents both the current in the FST_{AR} and also whether this state is considered accepting or not.
$s' = (s, Reject)$, marking the start state to be "Rejecting" when the FST_{AR} hasn't taken in any input symbols.
$A' = \{q' = (q, Accept) | q \in Q, q' \in Q'\}$, marking the accepting state of this DFA to be all states with the ***Accept*** flag, which indicates that in the original FST_{AR}, the input string finally arrives at the last state via a final transition outputting "Accept".
δ'=
For arbitrary $q_1, q_2 \in Q, a \in \Sigma$
(1) $\delta'((q_1, Accept), a) = (q_2, Accept)$
If in the transition rules of FST_{AR} we have $\delta(q_1, a) = (q_2, A)$;
(2) $\delta'((q_1, Reject), a) = (q_2, Accept)$
If in the transition rules of FST_{AR} we have $\delta(q_1, a) = (q_2, A)$;
(3) $\delta'((q_1, Accept), a) = (q_2, Reject)$
If in the transition rules of FST_{AR} we have $\delta(q_1, a) = (q_2, R)$;
(4) $\delta'((q_1, Reject), a) = (q_2, Reject)$
If in the transition rules of FST_{AR} we have $\delta(q_1, a) = (q_2, R)$;

Therefore, we could determine that $L(FST_{AR}) = L(M)$ is regular, that is, given a language that is represented with a FST_{AR}, we could prove that it's regular.

II. Regular = $DFA \to FST_{AR}$

Reversely thinking, we could also transform an arbitrary regular language, in the form of a DFA, to a FST_{AR} with the following method.

Given an arbitrary regular language L in the form of a DFA $M = (\Sigma, Q, \delta, s, A)$
We have
Σ is the set of input symbols in this language;
Q is the set of states;
s is the starting state;
A is the set of all accepting states;
δ is the set of all transitions, where:
$\delta(q_1, a) = q_2, q_1, q_2 \in Q, a \in \Sigma$, which indicates that M takes in a symbol *a* at state q_1 and transitions to the new state q_2.

We could construct a $FST_{AR} = (\Sigma', Q', \delta', s', \Gamma')$ based on M that

$\Sigma' = \Sigma$ is the set of input symbols;
$Q' = Q$ is the set of states in the FST_{AR};
$s' = s$ is the starting state;
$\Gamma' = \{Accept, Reject\}$, marking whether this transition, if as the last transition of the input string, would take it to an accepting state or a rejecting state.
δ'=
For $q_1, q_2 \in Q, a \in \Sigma$
(1) $\delta'(q_1, a) = (q_2, Accept)$, if $q_2 \in A$
    If the next state lead by this transition is accepting, output accepting;
(1) $\delta'(q_1, a) = (q_2, Reject)$, if $q_2 \notin A$
    If the next state lead by this transition is not accepting, output rejecting;

Therefore, we could determine that $L(M) = L(FST_{AR})$ is regular, that is, given an arbitrary regular language, we could prove that it could be represented in the form of a FST_{AR}.

In a nutshell, we could prove that $L(FST_{AR})$ represents the class of regular languages.