CIS 425/520 Automata Theory
Fall 2014

Assignment 2

due October 17, 2014

On the following pages you will find a series of proofs. For each proof do the following:

1. Give each proof a score from $1 - 10$ with 10 being a perfect proof.
2. If you take off points, give clear indication of why the proof lost points.
3. Give helpful feedback on the proof. This can includes positive feedback on what parts of the proof are effective as well as constructive feedback on how the proof might be improved.
4. Note how long it took you grade the proof.[1]

Once you have finished correcting the proofs, write your own version of the proof. Feel free to borrow things that you liked from the proofs you corrected.

[1]If a proof takes a long time to asses, it is probably not well written. Sometimes a proof takes a long time to understand because the topic is challenging. However, that is not the case for these proofs.

AN NFA and an all-NFA only differe on their input. an aNFA (all-NFA) only takes input if that input will lead to an accepting state every time, whereas a NFA only takes input if one string gets to a final state.

* Let $\overline{M}$ be the all-NFA, and $\overline{L}$ be the language. If there is an $\overline{M}$ that accepts all $\overline{L}$, then there is an NFA $\overline{N}$ that accepts $\overline{L}$. This ~~proves~~ will show that $\overline{L}$ is regular, and if $\overline{L}$ is regular, then L is regular under closed compliment.

* $\overline{L} = \overline{L}(\overline{M})$, $\overline{N} = (Q, \Sigma, \delta, q_0, F' = Q - F)$ and accepts $\overline{L}$.
Now if $x \in \overline{L}$, then all computations of x in $\overline{M}$ lead to ~~good~~ final states (F), so none end up in (F'), Thus $\overline{N}$ doesn't accept x, ~~$x \in \overline{N}(\overline{L})$~~ $x \in \overline{L}(\overline{N})$.

* If $x \notin \overline{L}$, then its accepted by $\overline{N}$, and therefore $x \in \overline{L}(\overline{N})$.
I showed $x \in \overline{L}(\overline{M})$ iff $x \notin \overline{L}(\overline{N})$, so N accepts $\overline{L}$

Problem 4

Let us show that all-NFAs can be converted to DFAs. Let $N = (Q, \Sigma, \delta, q_0, F)$ be an all-NFA. Consider a DFA $M = (Q', \Sigma, \delta', q_0', F')$, where

$$\begin{aligned} Q' &= 2^Q, \\ F' &= 2^F \setminus \{\varnothing\}, \\ q_0' &= E(\{q_0\}), \\ \delta'(r, a) &= \bigcup_{q \in r} E(\delta(q, a)), \quad r \in Q', a \in \Sigma, \end{aligned}$$

and $E(r)$ denotes the ε-closure of a set of states $r \subseteq Q$ (as in the proof of equivalence of DFAs and NFAs).

It is easy to see from the above construction that

$$N \text{ accepts } w \;\Leftrightarrow\; \widehat{\delta}(q_0, w) \subseteq F \;\Leftrightarrow\; \widehat{\delta'}(q_0', w) \in F' \;\Leftrightarrow\; M \text{ accepts } w.$$

Indeed, by definition, the all-NFA N accepts a string w iff it can be written as $w = y_1 y_2 \cdots y_m$, $y_i \in \Sigma_\varepsilon$ and for all possible sequences of states from Q

$$s_0^{(1)}, s_1^{(1)}, \ldots, s_m^{(1)}, \quad s_0^{(2)}, s_1^{(2)}, \ldots, s_m^{(2)}, \quad \ldots, \quad s_0^{(n)}, s_1^{(n)}, \ldots, s_m^{(n)},$$

such that

$$s_0^{(k)} = q_0, \quad s_{i+1}^{(k)} \in \delta(s_i^{(k)}, y_{i+1}), \quad i = 0, \ldots, m-1,$$

it holds that $s_m^{(k)} \in F$ for all $k = 1, \ldots, n$.

But there is a one-to-one correspondence between all sets of such sequences and series of states $t_0, t_1, \ldots, t_m \in Q'$ such that

$$t_i = \bigcup_{k=1}^{n} r_i^{(k)}, \quad i = 0, \ldots, m,$$

and consequently, $t_0 = q_0'$ and $t_m \in F'$.

Since every DFA without any changes is an all-NFA, the opposite conversion is trivial. Thus, all-NFAs recognize the same class of (regular) languages as DFAs.

Proof. Let M be any **all**-NFA and $L(M)$ is the language recognized by M. Note that M can also be viewed as an NFA for the strings $\overline{L(M)} = \{x \mid x \notin L(M)\}$. The language $\overline{L(M)}$ is regular since it is recognized by an NFA and $L(M)$ is regular since regular languages are closed under complement. $\square$

4. Sipser 1.38

An *all*-NFA *M* is a 5-tuple (*Q, Σ, δ,* q_0, *F*) that accepts $x \in \Sigma^*$ if *every* possible state that *M* could be in after reading input x is a state from *F*. Note, in contrast, that an ordinary NFA accepts a string if *some* state among these possible states is an accept state. Prove that all-NFAs recognize the class of regular languages.
Hint: First observe that a DFA is an all-NFA (why?). For the opposite direction, from each all-NFA you should construct an equivalent DFA.

Answer:

Let M be an all-NFA. NTS for all L which are accepted by some all-NFA M ∃ an NFA N which accepts ¬L. This would imply that if L is accepted by some all-NFA, ¬L is regular. The class of regular languages is closed under complement, therefore L is regular as well.

Let L = L(M). NTS NFA N = (Q, Σ, δ, q_0, F' = Q - F) will accept L. Suppose $x \in L$, then all of the paths of M on x lead to a state in F, implying that none of the paths end in F'. Therefore, NFA N does not accept x, and $x \notin L(N)$. If x is not accepted by M($x \notin L$), then it is accepted by N($x \in L(N)$).
Since $x \in L(M)$ iff $x \notin L(N)$, ¬L is accepted by N.