Homework 2: DFAs and NFAs

CSE 30151 Spring 2018

Due 2018/02/01 at 10:00pm

Instructions

- Create a PDF file (or files) containing your solutions. You can write your solutions by hand, but please scan them in the library or using a smartphone to get them into PDF.
- Please name your PDF file(s) as follows to ensure that the graders give you credit for all of your work:
  - If you're making a complete submission, name it `*netid*-hw2.pdf`, where *netid* is replaced with your NetID.
  - If you're submitting some problems now and want to submit other problems later, name it `*netid*-hw2-123.pdf`, where `123` is replaced with the problem numbers you are submitting at this time.
- Submit your PDF file(s) in Sakai. Don't forget to click the Submit button!

Problems (10 points each)

1. **Designing finite automata** Define a natural number to include 0, and define the base-10 representation of a natural number to allow leading `0`'s as well as ε (representing 0). Write a DFA (both a formal description **and** a state diagram) for base-10 representations of natural numbers that are:

   (a) divisible by 2

   (b) divisible by 3

   (c) divisible by 4

   As an alternative to (c), you can do the following more challenging problem:

   (c′) Show, for any $k > 0$, how to construct a DFA for base-10 representations of natural numbers divisible by k. Your answer should show how to write a formal description $M = (Q, \{\mathtt{0}, \ldots, \mathtt{9}\}, \delta, s, F)$ in terms of k.

2. **Nondeterminism** Consider the following language:

$$L_2 = \{uv \mid u, v \in \{\mathtt{a}, \mathtt{b}\}^*, u \text{ contains an even number of } \mathtt{a}\text{'s, and } v \text{ contains an even number of } \mathtt{b}\text{'s}\}$$

Note that as long as there is *some* way of cutting a string into u and v so as to satisfy the constraints, it's in L_2. So $\mathtt{ba} \in L_2$, because $u = \mathtt{b}$ has an even number (0) of a's and $v = \mathtt{a}$ has an even number (0) of b's.

(a) Write an NFA N_2 that recognizes L_2.

(b) What is the accepting path for $\mathtt{bab}^n$ through N_2? You can show the path for $n = 0, 1, 2, \ldots$ until the pattern is clear, or describe the general case. Either way, please write a few words about what you observe.

(c) Convert N_2 to a DFA M_2.

(d) What is the accepting path for $\mathtt{bab}^n$ through M_2? Again, you can show the path for $n = 0, 1, 2, \ldots$ until the pattern is clear, or describe the general case. Either way, please write a few words about what you observe.

3. **Regular/raluger** In the following, we'll use the language L_3 as an example (but the results must be proved for all L):

$$L_3 = \{\mathtt{deed}, \mathtt{deer}, \mathtt{red}, \mathtt{redder}, \mathtt{reed}\}.$$

(a) Recall that

$$L^R = \{w \mid w^R \in L\}.$$

For example, $L_3^R = \{\mathtt{deed}, \mathtt{reed}, \mathtt{der}, \mathtt{redder}, \mathtt{deer}\}$. Show that if L is regular, then L^R is also regular. (That is, given a formal description or state diagram of a NFA for L, show how to construct a formal description or state diagram of a NFA for L^R.)

(b) Define

$$\text{DOPPELGANGERS}(L) = \{w \mid w \in L \text{ and } w \in L^R\}.$$

For example, $\text{DOPPELGANGERS}(L_3) = \{\mathtt{deed}, \mathtt{reed}, \mathtt{deer}, \mathtt{redder}\}$. Show that if L is regular, then DOPPELGANGERS(L) is also regular.

(c) Define

$$\text{HALF}(L) = \{w \mid ww^R \in L\}.$$

For example, $\text{HALF}(L_3) = \{\mathtt{de}, \mathtt{red}\}$. Show that if L is regular, then HALF(L) is also regular. Hint: The NFA for HALF(L) differs from the NFA for DOPPELGANGERS(L) only in the choice of accept states.

Appendix: NFA intersection

For Problem 3, it is convenient (though not essential) to use the fact that the intersection algorithm (Sipser, page 46, footnote 3) also works on NFAs. The idea is the same: the new NFA can be in state (r_1, r_2) after reading a string w iff the first NFA can be in state r_1 after reading w and the second NFA can be in state r_2 after reading w.

I don't think you need it, but for completeness, here is the construction in more detail. Given two NFAs

$$
\begin{aligned}
N_1 &= (Q_1, \Sigma, \delta_1, q_1, F_1) \\
N_2 &= (Q_2, \Sigma, \delta_2, q_2, F_2)
\end{aligned}
$$

we can construct a new NFA N such that $\mathcal{L}(N) = \mathcal{L}(N_1) \cap \mathcal{L}(N_2)$. Let

$$N = (Q_1 \times Q_2, \Sigma, \delta, (q_1, q_2), F_1 \times F_2),$$

where δ is defined as follows.

$$
\begin{aligned}
\delta((r_1, r_2), a) &= \delta_1(r_1, a) \times \delta_2(r_2, a) && \text{if } a \neq \varepsilon \\
\delta((r_1, r_2), \varepsilon) &= (\delta_1(r_1, \varepsilon) \times \{r_2\}) \cup (\{r_1\} \times \delta_2(r_2, \varepsilon)).
\end{aligned}
$$

Using state diagrams, we can express the construction of the transitions as follows:

	N_1	N_2	N
For each pair of transitions	$r_1 \xrightarrow{a} s_1$	$r_2 \xrightarrow{a} s_2$	add $(r_1, r_2) \xrightarrow{a} (s_1, s_2)$
For each transition and state	$r_1 \xrightarrow{\varepsilon} s_1$	r_2	add $(r_1, r_2) \xrightarrow{\varepsilon} (s_1, r_2)$
For each state and transition	r_1	$r_2 \xrightarrow{\varepsilon} s_2$	add $(r_1, r_2) \xrightarrow{\varepsilon} (r_1, s_2)$

The first case is similar to the DFA intersection construction: Both N_1 and N_2 read an a symbol. But the handling of ε-transitions is new: In the second case, N_1 follows an ε-transition while N_2 does nothing; in the third case, N_2 follows an ε-transition while N_1 does nothing.