Problem Set 3

This problem set is due on **Friday February 13, by 4:00pm.**

Use the CS172 drop box.

Write **your name and your student ID number** on your solution. Write legibly. The description of your proofs should be as *clear* as possible (which does not mean *long* – in fact, typically, good clear explanations are also short.) Be sure to be familiar with the collaboration policy, and read the overview in the class homepage `www.cs.berkeley.edu/~luca/cs172`.

This version was posted on Feb 11, 2004, 3:10pm

1. Give a family of languages A_k, where each A_k can be recognized by a $O(k)$-state DFA whereas A_k^R requires $2^{\Omega(k)}$ states on a DFA. Prove that your languages have this property. (For a language L, the language L^R is defined as in Problem Set 1.)

2. A **two-way finite automaton** is like a deterministic finite automaton, except that the reading head can go backwards as well as forwards on the input tape. If it tries to back up off the left end of the tape, it stops operating without accepting the input. If it tries to move past the right end of the tape, it stops operating, and accepts the input if it is in an accepting state, and rejects otherwise. Formally, a two-way finite automaton M is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$, where Q, Σ, q_0 and F are as defined for deterministic finite automaton (refer to Sipser definition 1.1 on page 35), and $\delta : Q \times \Sigma \to Q \times \{\leftarrow, \rightarrow\}$ is a transition function, where $\leftarrow$ or $\rightarrow$ indicates the direction of head movement.

   As a two-way finite automaton computes, changes occur in the current state and the current head location. A **configuration** is a member of $\Sigma^* \times Q \times \Sigma^*$; configuration (u, q, v) indicates that the machine is in state q with the head on the first symbol of v and u to the left of the head. For instance, $1011q_701111$ represents the configuration when the input is 101101111, the current state is q_7, and the head is currently on the second zero (of the input). If $u = \epsilon$, then the machine head is at the first position of its input. If $v = \epsilon$, the configuration (u, q, ϵ) means that M has completed its operation on u, and ended up in state q.

   We write $(u_1, q_1, v_1) \vdash_M (u_2, q_2, v_2)$ if M can move from the configuration (u_1, q_1, v_1) to the configuration (u_2, q_2, v_2) in a single transition. Formally, the relation $(u_1, q_1, v_1) \vdash_M (u_2, q_2, v_2)$ holds iff:

   (i) $v_1 = \sigma v$ for some $\sigma \in \Sigma$;

   (ii) $\delta(q_1, \sigma) = (q_2, D)$; and

   (iii) $D = \rightarrow$ and $u_2 = u_1\sigma, v_2 = v$, or
   $D = \leftarrow$, $u_1 = u\sigma'$ for some $u \in \Sigma^*$, $\sigma' \in \Sigma$ and $v_2 = \sigma' v_1$.

As usual, we use $\vdash_M^*$ to denote the reflexive, transitive closure of $\vdash_M$ (that is, $(u_1, q_1, v_1) \vdash_M^* (u_2, q_2, v_2)$ holds if M can transition from configuration (u_1, q_1, v_1) to (u_2, q_2, v_2) in 0 or more steps), and $\vdash_M^+$ to denote the transitive (not reflexive) closure of $\vdash_M$ (that is, $(u_1, q_1, v_1) \vdash_M^+ (u_2, q_2, v_2)$ holds if M can transition from configuration (u_1, q_1, v_1) to (u_2, q_2, v_2) in 1 or more steps). Therefore, M accepts w iff $(\epsilon, q_0, w) \vdash_M^* (w, q, \epsilon)$ for some $q \in F$.

In this problem, we will use the Myhill-Nerode Theorem to show that a language accepted by a two-way finite automaton is accepted by a one-way finite automaton. Thus, the apparent power to move the head to the left does not enhance the power of finite automata.

(a) Design a two-way finite automaton with $O(k)$ states accepting the language $L_k = \{0,1\}^*1\{0,1\}^k\$$, where $\Sigma = \{0, 1, \$\}$. (Recall problem 4(a) on problem set 1, where you were asked to construct a DFA for L_k with 2^{k+1} states.)

(b) Let M be a two-way finite automaton as just defined. Let $q \in Q$ and $w \in \Sigma^*$. Observe that by definition, there is at most one[1] $p \in Q$ such that $(\epsilon, q, w) \vdash_M^* (w, p, \epsilon)$. Next, let t be some fixed "special state" not in Q, and for any $w \in \Sigma^*$, we define a function $\chi_w : Q \to (Q \cup t)$ as follows:

$$\chi_w(q) = \begin{cases} p, & \text{if } (\epsilon, q, w) \vdash_M^* (w, p, \epsilon) \\ t, & \text{otherwise} \end{cases}$$

Also, for any $w \in \Sigma^*$, define $\theta_w : Q \times \Sigma \to Q \cup \{t\}$ as follows:

$$\theta_w(q, a) = \begin{cases} p, & \text{if } (w, q, a) \vdash_M^+ (w, p, a) \text{ but it is not the case that} \\ & \quad (w, q, a) \vdash_M^+ (w, r, a) \vdash_M^+ (w, p, a) \text{ for any } r \neq p \\ t, & \text{if there is no } p \in Q \text{ such that } (w, q, a) \vdash_M^+ (w, p, a) \end{cases}$$

Informally, if M starting at configuration (w, q, a) goes (possibly more than once) to some configuration of the form (w, r, a), then $\theta_w(q, a)$ is the last such r; and $\theta_w(q, a)$ is t if M never goes to a configuration of that form. Note that $a \in \Sigma$ is a character, not a string.

Now, suppose that $w, v \in \Sigma^*, \chi_w = \chi_v$ and $\theta_w = \theta_v$. Show that, for any $u \in \Sigma^*$, M accepts wu iff M accepts vu.

(c) Show that if L is the language accepted by a deterministic two-way automaton, then L is accepted by some ordinary (one-way) DFA. (*Hint*: Use (b) to show that $\approx_L$ has finitely many equivalence classes.)

(d) Conclude that there is an exponential-time algorithm, which given a deterministic two-way automaton M, constructs an equivalent DFA. (*Hint*: How many different functions χ_w and θ_w can there be, as a function of $|Q|$ and $|\Sigma|$? Also, exponential-time here means a running time of $O(2^{\text{poly}(|\Sigma|,|Q|)})$.)

[1] It is possible that the machine tries to back off the left end of the tape and therefore stops operating, in which case no such p exists.