Mehryar Mohri
Speech Recognition
Courant Institute of Mathematical Sciences
Homework assignment 1 - Solution

1. Generalities.

   (a) Give an example of an unambiguous automaton that is not deterministic.

   Figure 1 shows an example of such an automaton.

   (b) Give an example of a non-deterministic automaton over the alphabet $\{a, b\}$ whose determinization results in an exponentially larger automaton.

   Example: non-deterministic automaton directly derived from the regular expression $(a + b)^* ab^{n-1}$. The number of states of the automaton after determinization is 2^n. Figure 1 shows that automaton for $n = 5$.

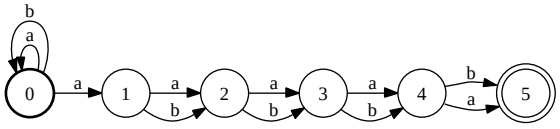

Figure 1: Finite automaton A_5.

2. String-matching automata.

   (a) Give a regular expression for the set of strings over the alphabet $\{a, b\}$ ending with aba.

   $r = (a + b)^* aba$.

   (b) Use the FSM library or OpenFst to create a binary representation of a non-deterministic automaton representing that expression.

   `fsmcompile aut.txt -ilab aut.txt > aut.fsm`.

   (c) Show a graphical representation of that automaton.

   `fsmdraw -p -ilab aut.fsm | dot -Tps > aut.ps`.
   See Figure 2.

   (d) Indicate how this automaton can be used to find in a text the occurrences of string aba.

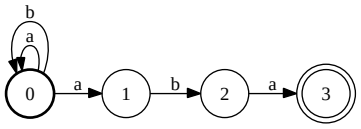

Figure 2: Finite automaton A_5.

A text over the alphabet $\{a, b\}$ contains aba iff t belongs to the language $(a+b)^*aba$. To find occurrences of aba, read t with the automaton, every time the set of states reached contains a final state, an occurrence of aba is found.

(e) Determinize and minimize that automaton (use software library) and show the graphical representation of the result.

`fsmdeterminize aut.fsm | fsmminimize | fsmdraw -p -ilab | dot -Tps > aut.ps`.

The result is shown on Figure 3.

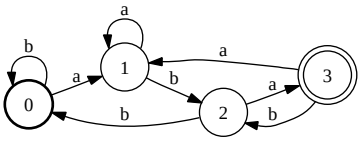

Figure 3: Finite automaton A_5.

3. Division with finite-state transducers. The transducer of Figure 4 can take as input a binary sequence representing an integer n_1 and return a binary sequence representing the integer n_2.

(a) Verify that the transducer is deterministic (or sequential) and complete. Is the inverse transducer sequential?

The transducer is deterministic but the inverse is not since at state 0 for example, there are two transitions with output label 0.

(b) How is n_2 related to n_1? (hint: think division modulo some integer n that you should specify).

n_2 is the result of the division of n_1 by 4.

(c) What does the number of the destination state of an accepting path indicate about the integer represented by the string labeling that path?

That number gives the rest of the division of n_1 by 4.

(d) Give an automaton representing the set of binary sequences representing numbers equal to 1 modulo 4.

It suffices to keep only state 1 as a final state and project the transducer on the input.

(e) Show how these results can be generalized: give a finite-state transducer that takes as input strings representing numbers in base x and that returns in the same base a string representing the result of the division by some integer p. Comment as before on the destination state of an accepting path. Note: you should describe the transducer in detail and prove its correctness.

The following is the definition of the transducer $T_{x,p} = (\Sigma, \Delta, Q, I, F, E)$ in base $x > 1$.

$\Sigma = \Delta = \{0, 1, \ldots, x-1\}$
$Q = F = \{0, 1, \ldots, p-1\}$
$I = \{0\}$
$E = \left\{(r, a, b, q) : r, q \in Q, a \in \Sigma, \left(\lfloor (rx + a)/p \rfloor = b\right) \wedge \left(rx + a \equiv q \pmod{p}\right)\right\}$.

The output label b of each transition is in Δ since

$$(rx + a)/p \leq [(p-1)x + (x-1)]/p = [p(x-1)]/p = (x-1). \quad (1)$$

By construction, the transducer is deterministic.

The proof of correctness is based on a recurrence. We denote by $[u]$ the integer value of a sequence u in base x.

Assume that any string u of length at most n leads to state r such that $[u] \equiv r \pmod{p}$ and that the output label of the unique path from 0 labeled with u is the binary sequence u' representing $\lfloor [u]/p \rfloor$.

Consider now the string $u0$. By definition of the transducer, the transition with input label 0 leaving r leads to q with $rx \equiv q \pmod{p}$ and its output label is $b = \lfloor rx/p \rfloor$.

By definition, $[u0] = x[u]$ and $\lfloor [u0]/p \rfloor = \lfloor x[u]/p \rfloor = x\lfloor [u]/p \rfloor + \lfloor rx/p \rfloor = x[u'] + \lfloor rx/p \rfloor = x[u'] + b = [u'b]$. This shows that the output label of the path labeled with $u0$ is correctly giving $\lfloor [u0]/p \rfloor$. Also, $[u0] = x[u] \equiv x[u] \equiv rx \pmod{p} \equiv q \pmod{p}$, thus the state q reached by $u0$ is correctly giving the rest of the division of $u0$ by p.

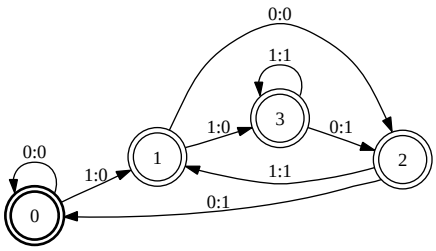

Figure 4: Finite-state transducer.

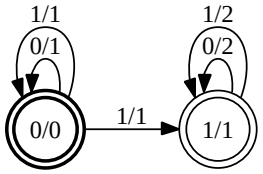

Figure 5: Weighted automaton computing integer values.

Proceeding similarly with other labels $a \in \Sigma$, $a \neq 0$, we can show that the output label of the path labeled with ua gives the correct quotient and the state reached the rest of the division of $[ua]$ by p, which shows the correctness of the transducer for strings of length $n+1$. The correctness for strings of length exactly 1 is straightforward.

4. Computing integer values with weighted automata. The previous exercise showed how transducers can be used for division, but the result was given as a sequence. Here, we wish to further *compute* the integer value of that sequence.

   (a) Give a *weighted regular expression* describing the weighted automaton of Figure 5 defined over the semiring $(\mathbb{R}, +, \times, 0, 1)$.

   The following is the equivalent *weighted regular expression* (rational power series):

   $$((\mathbf{0} + \mathbf{1})/1)^*(\mathbf{1}/1)((\mathbf{0} + \mathbf{1})/2)^*,$$

   where to distinguish weights and symbols, symbols are written in boldface, and weights are indicated after the slash symbol.

   (b) Show that it associates to any binary sequence its integer value (give a proof).

Let $z = z_0 \dots z_n$ be a binary sequence. In view of the rational power series given in the previous questions, the weight automaton A associates to z is

$$A(z) = \sum_{i=0}^{n} \sum_{z=uz_iv} 1 \cdot z_i \cdot 2^{|v|} = \sum_{i=0}^{n} z_i 2^{n-i},$$

which is precisely the integer value of z.

(c) Give a weighted automaton with a similar property for sequences in an arbitrary base x.

The automaton A_x can be constructed straightforwardly from the following rational power series:

$$\sum_{i=1}^{x-1} (\mathbf{\Sigma}/1)^* (\mathbf{i}/i)(\mathbf{\Sigma}/i)^*,$$

where $\Sigma = \{0, 1, \dots, x-1\}$.

(d) Give a weighted automaton that takes as input a sequence in an arbitrary base x and returns the integer value of the division of the number it represents by some integer p (hint: use previous exercise and composition).

Transducer $T_{x,p}$ associates to a sequence z in base x a sequence z' in base x whose integer value $n_{z'}$ is the rest of the division of the integer value of z by p. Since $A_x(z')$ is by definition the integer value of z' in base x, the result can be obtained by composition:

$$(T_{x,p} \circ A_x)(z, z') = n_{z'}.$$

Thus, the weighted automaton sought can be constructed by projection of $(T_{x,p} \circ A_x)$ on the output: $\Pi_O(T_{x,p} \circ A_x)$, where Π_O is the output projection operator.