Advanced Topics in Automata
Exercise 5

Submission: May 20, 2003

Exercise

1. Give a lower bound on the conversion of ABW to NBW.

2. The first step in checking the emptiness of an NBW was to forget about the alphabet. That is, when given an NBW we construct an NBW whose input alphabet contains only 1 letter, and check the emptiness of the new automaton.

   What is the complexity of deciding the emptiness of an ABW with 1-letter input alphabet ?

   Note that you have to give both a lower bound and an upper bound. The complexity is not PSPACE and the bounds are tight.

   Hint. We have to find the maximal subset F' of the accepting states such that from every state in F' we can create a prefix of a run tree such that all the leaves are in F'. A prefix of a run tree starting from $f \in F'$ is a finite run tree whose root is labeled by f, it obeys the consecution requirement until it reaches some state in F' (in a node different from the root). Given this maximal set F', we have to check whether there is a prefix of a run tree from s_0 whose leaves are all in F'. So the problem reduces to finding the maximal subset F'.

   In order to find F' try to eliminate elements.

Food for thought

1. We have seen that in order to complement an AFW, we have to take the dual automaton. That is, take the automaton whose transition is the dual of the first and whose set of accepting states is complemented.

   I sketched the proof that this automaton is indeed the complement. However, you have not noticed that I showed only soundness of the construction and not its completeness. That is, I showed that $\overline{A}$ and A cannot accept the same word. However, I didn't show that every word not accepted by A is accepted by $\overline{A}$ (or equivalently, that every word is either accepted by A or by $\overline{A}$).

   Prove completeness of the construction.

2. The same question for ABW. I claimed that given an ABW the dual ACW accepts the complement language. The soundness proof is just like the proof for the case of finite words. What about completeness ?

3. We have seen that the emptiness problem is simple for nondeterministic automata and complicated for alternating automata. I mentioned the the other disadvantage of alternating automata is *homomorphisms*.

   Given two alphabets Σ and Δ and a function $h : \Sigma \to \Delta$ we extend h into a homomorphism from Σ^* to Δ^* in the obvious way:

   - $h(\epsilon) = \epsilon$
   - $h(w_1 w_2 \cdots w_n) = h(w_1)h(w_2)\cdots h(w_n)$

   Similarly, given a language $L \subseteq \Sigma^*$ we define $h(L) = \{h(w) \mid w \in L\} \subseteq \Delta^*$.

   Given a nondeterministic automaton $A = \langle \Sigma, Q, q_0, \delta, F \rangle$ that recognizes some language L it is straight forward to construct an automaton for $h(L)$ (show!).

   If we are given an alternating automaton A for some language L, we cannot give an automaton for $h(L)$ without turning A first into a nondeterministic automaton.

   Find a set of languages L_n and a function h such that the best alternating automaton for $h(L_n)$ is exponentially larger than the alternating automaton for L_n.

   The question is formulated without distinction between automata on finite and infinite words.