Advanced Automata Theory (Summer 2017) Problem Set #3 (Due 31.05.2017)

Problem 1. Consider the regular language $L = (ab^*a + ba^*b)^*$ over the alphabet $\Sigma = \{a, b\}$.

(a) Specify all Myhill-Nerode equivalence classes of L. For each pair of L-equivalence classes $[\![u]\!]_L \neq [\![v]\!]_L$, provide a word $w \in \Sigma^*$ such that $uw \in L \Leftrightarrow vw \notin L$.

(b) Draw the DFA $\mathcal{A}_L$.

Problem 2. Show that the following languages over the alphabet $\Sigma = \{a, b\}$ are not regular by showing that the number of L-equivalences classes is infinite.

(a) $L_1 = \{u \in \Sigma^* \mid |u|_a = |u|_b\}$, where $|u|_x$ denotes the number of times the symbols x appears in u

(b) $L_2 = \{a^n \mid n \text{ is a power of 2 (i.e., } \exists j \in \mathbb{N}\colon n = 2^j)\}$

Problem 3. Let $\mathcal{A} = (Q, \Sigma, \delta, q_{in}, F)$ be an NFA and $\sim_\mathcal{A}$ the equivalence relation over states of $\mathcal{A}$ defined by

$$p \sim_\mathcal{A} q \text{ if and only if } L(\mathcal{A}_p) = L(\mathcal{A}_q).$$

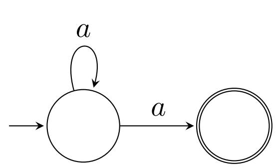
(Recall that $\mathcal{A}_p$ is the same NFA as $\mathcal{A}$ only with initial state p.)

We have seen in the lecture that $L(\mathcal{A}/_{\sim_\mathcal{A}}) = L(\mathcal{A})$ holds if $\mathcal{A}$ is a DFA. Show that this is also true for NFAs.

Hint: Use an accepting run of $\mathcal{A}/_{\sim_\mathcal{A}}$ on a word u to construct an accepting run of $\mathcal{A}$ on u. One way to do so is by an induction "backwards" over the run on u.

Problem 4.

(a) Consider the NFA $\mathcal{A}$

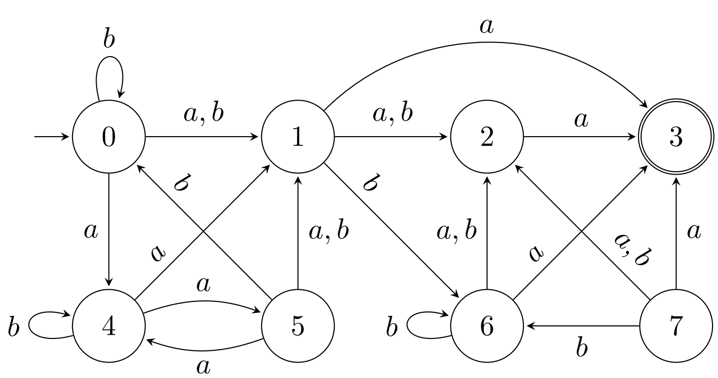


Provide two additional, pairwise distinct DFAs $\mathcal{A}_1$ and $\mathcal{A}_2$ with two states such that $L(\mathcal{A}_1) = L(\mathcal{A}_2) = L(\mathcal{A})$.

(b) Specify a family of regular languages L_n and for each $n \in \mathbb{N}$ NFAs $\mathcal{A}_1^{(n)}, \ldots, \mathcal{A}_n^{(n)}$ such that

- $L(\mathcal{A}_i^{(n)}) = L_n$ for $i \in \{1, \ldots, n\}$; and
- $\mathcal{A}_1^{(n)}, \ldots, \mathcal{A}_n^{(n)}$ all have the same number of states but are pairwise non-isomorphic; and
- each NFA with strictly less states than $\mathcal{A}_i^{(n)}$ for an arbitrary $i \in \{1, \ldots, n\}$ does not accept L_n.

Problem 5. Consider the NFA $\mathcal{A}$ given by the following pointed LTS:



(a) Construct the bisimulation quotient automaton $\mathcal{B}$ of $\mathcal{A}$ using the marking algorithm presented in the lecture.

(b) Prove that $\mathcal{B}$ is a minimal NFA accepting $L(\mathcal{A})$ or give a counterexample to disprove this claim.

Problem 6. Consider the sample $\mathcal{S} = (S_+, S_-)$ with

$$S_+ = \{abbab\}$$

and

$$S_- = \{\varepsilon, ab, ba\}$$

over the alphabet $\Sigma = \{a, b\}$.

(a) Run the RPNI algorithm on $\mathcal{S}$. Draw the resulting DFA after each *successful* merge.

(b) Is the DFA resulting in Part (a) a minimal DFA that is consistent with $\mathcal{S}$? If not, provide a consistent DFA that is minimal.

Problem 7. Run Angluin's algorithm for the target language

$$L = \Sigma^* b(aa)^* + (aa)^+$$

over the alphabet $\Sigma = \{a, b\}$.

Provide all intermediate observation tables that are constructed in the course of the algorithm and briefly sketch what operation was performed to go from one table to the next. Use canonically smallest counterexamples to answer equivalence queries.

Problem 8. The goal of this exercise is to illustrate that maintaining a prefix-closed set R in Angluin's algorithm is in fact necessary. To this end, provide a closed and consistent observation table $O = (R, S, T)$ such that $\varepsilon \in R$, R is *not* prefix-closed, and $\mathcal{A}_O$ is not correct on words of R (i.e., there exists a word $u \in R$ such that $T(u) = 1$ if and only if $u \notin L(\mathcal{A}_O)$).