Prof. Dr. Ir. Joost-Pieter Katoen
Christian Dehnert, Sebastian Junges

Exercise 1 (Muller automata): (2 points)

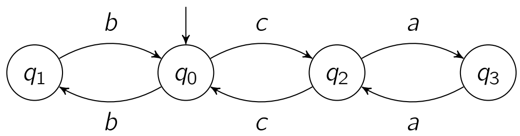
A nondeterministic Muller automaton is a quintuple $\mathcal{A} = (Q, \Sigma, \delta, Q_0, \mathcal{F})$ where Q, Σ, δ and Q_0 are as for NBA and $\mathcal{F} \subseteq 2^Q$. For an infinite run ρ of $\mathcal{A}$, let

$$inf(\rho) := \{q \in Q \mid \exists^\infty i \geq 0.\ \rho[i] = q\}.$$

Let $\alpha \in \Sigma^\omega$.

$$\mathcal{A} \text{ accepts } \alpha \iff \text{ex. inf. run } \rho \text{ of } \mathcal{A} \text{ on } \alpha \text{ s.t. } inf(\rho) \in \mathcal{F}$$

a) Consider the following Muller automaton $\mathcal{A}$ with $\mathcal{F} = \{\{q_2, q_3\}, \{q_1, q_3\}, \{q_0, q_2\}\}$:



Give the language accepted by $\mathcal{A}$ by means of an ω-regular expression.

b) Show that every GNBA $\mathcal{G}$ can be transformed into a nondeterministic Muller automaton $\mathcal{A}$ such that $\mathcal{L}_\omega(\mathcal{A}) = \mathcal{L}_\omega(\mathcal{G})$ by defining the corresponding transformation.

Exercise 2 (A-recognizable): (2 points)

A language $\mathcal{L} \subseteq \Sigma^\omega$ is said to be A-recognized by a (nondeterministic) Büchi automaton $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ if

$$\alpha \in \mathcal{L} \iff \text{ex. a run } \rho \text{ of } \mathcal{A} \text{ on } \alpha \text{ s.t. } \forall i.\ \rho[i] \in F.$$

$\mathcal{L}$ is called A-recognizable if there exists an automaton $\mathcal{A}$ that A-recognizes $\mathcal{L}$.

Prove or disprove that an LT property E is a safety property if and only if E is A-recognizable.

Exercise 3 (DBA): (2 points)

Formally prove that there is no DBA $\mathcal{A}$ over the alphabet $\Sigma = \{a, b\}$ that accepts the language

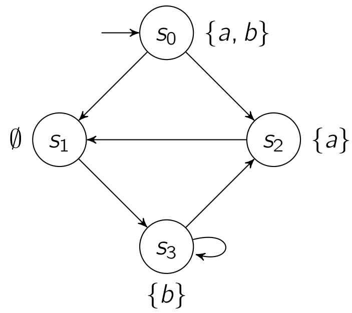
$$\mathcal{L} := \mathcal{L}_\omega((a + b)^*.a^\omega).$$

Exercise 4 (Model Checking ω-regular Properties): (4 points)

Let the ω–regular LT properties P_1 and P_2 over the set of atomic propositions AP $= \{a, b\}$ be given by

$$P_1 := \text{"if } a \text{ holds infinitely often, then } b \text{ holds finitely often"}$$
$$P_2 := \text{"} a \text{ holds infinitely often and } b \text{ holds infinitely often"}$$

The model is given by the transition system TS as follows:



Algorithmically check whether $TS \models P_1$ and $TS \models P_2$. For this, proceed as follows.

a) Derive *suitable* NBA $\mathcal{A}_{P_1}$, $\mathcal{A}_{P_2}$, where suitable means "appropriate for part b)-d)".
Hint: For P_1 you can find an automaton with 3 states and for P_2 4 states suffice. Derive the automata directly.

b) Outline the reachable fragments of the product transition systems $\mathrm{TS} \otimes \mathcal{A}_{P_1}$ and $\mathrm{TS} \otimes \mathcal{A}_{P_2}$.

c) Decide whether $\mathrm{TS} \models P_1$ by checking an appropriate persistence property via nested depth-first search on $\mathrm{TS} \otimes \mathcal{A}_{P_1}$. Document *all* changes to the contents of U, V, π and ξ (the state sets and stacks of the nested depth-first search, see lecture). If the property is violated, provide a counterexample *based on the execution of the algorithm*.

d) Decide whether $\mathrm{TS} \models P_2$ by checking an appropriate persistence property via SCC analysis on $\mathrm{TS} \otimes \mathcal{A}_{P_2}$. If the property is violated, provide a counterexample *based on your analysis*.