

Exercise 1 (Muller automata):

(2 points)

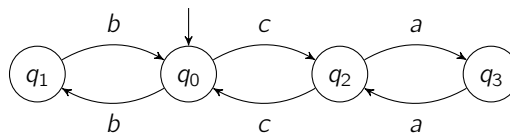
A nondeterministic Muller automaton is a quintuple $\mathcal{A} = (Q, \Sigma, \delta, Q_0, \mathcal{F})$ where Q , Σ , δ and Q_0 are as for NBA and $\mathcal{F} \subseteq 2^Q$. For an infinite run ρ of $\mathcal{A}$, let

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

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

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

- 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.

Solution: _____

- a) First, we observe that there is no run in $\mathcal{A}$ that visits $\{q_1, q_3\}$ infinitely often, but visits the states $\{q_0, q_2\}$ only finitely often. Hence, there is no run that is accepting because of $\{q_1, q_3\}$.

Since both accepting sets $\{q_0, q_2\}$ and $\{q_2, q_3\}$ ensure that from some point on, only c or a , respectively, are seen, the language of the automaton can be represented as

$$\mathcal{L}_\omega(\alpha_1.\beta_1^\omega + \alpha_2.\beta_2^\omega)$$

with

- $\alpha_1 = (bb + c(aa)^*c)^*c$
- $\beta_1 = a$
- $\alpha_2 = (bb + c(aa)^*c)^*$
- $\beta_2 = c$

- b) Let $\mathcal{A} = (Q, \Sigma, \delta, Q_0, \mathcal{F})$ be a GNBA with accepting sets $\mathcal{F} \subseteq 2^Q$. We can transform $\mathcal{A}$ to an NBA $\mathcal{A}' = (Q, \Sigma, \delta, Q_0, F)$ ($F \subseteq Q$) as seen in the lecture. Let $\mathcal{A}'' = (Q, \Sigma, \delta, Q_0, \mathcal{F}')$ with $\mathcal{F}' = \{F' \in 2^Q \mid F' \cap F \neq \emptyset\}$ be a nondeterministic Muller automaton. Then $\mathcal{A}''$ accepts the same language as $\mathcal{A}'$ (and therefore also $\mathcal{A}$) since we define all sets of states to be accepting as long as they contain at least one accepting state of the NBA, which effectively mimics the Büchi accepting condition.

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.

Solution:

Not all LT properties that are safety properties are A-recognizable since they are not necessarily ω -regular. For instance, take the LT property

$$E = \{\{a\}^n\{b\}^n\emptyset^\omega \mid n \in \mathbb{N}\} \cup \{\{a\}^\omega\}$$

over the atomic propositions $AP = \{a, b\}$.

Claim: E is a safety property. Let $\alpha \in cl(E)$. Then

$$\begin{aligned} \text{pref}(\alpha) &\subseteq \text{pref}(E) \\ &= \underbrace{\{\{a\}^n \mid n \in \mathbb{N}\}}_A \cup \underbrace{\{\{a\}^n\{b\}^m \mid n, m \in \mathbb{N}, n \geq m > 1\}}_B \cup \underbrace{\{\{a\}^n\{b\}^n\emptyset^m \mid n, m \in \mathbb{N}\}}_C \end{aligned}$$

As α has infinitely many prefixes, it must have infinitely many prefix in either

- (i) A ,
- (ii) B , or
- (iii) C

Analysis:

- Case (i): Then $\alpha = \{a\}^\omega \in E$
- Case (ii): No single word can have infinitely many prefixes in B since the number of b eventually exceeds the number of a .
- Case (iii): Then $\alpha = \{a\}^n\{b\}^n\emptyset^\omega$ for some n and therefore $\alpha \in E$

In all cases $\alpha \in E$, so $cl(E) \subseteq E$ and since $E \subseteq cl(E)$ for all LT properties E , we have $cl(E) = E$, so E is in fact a safety property. Now we go on to show that E is not A-recognizable.

Proof by contradiction. Assume there exists a nondeterministic Büchi automaton $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ with $n_0 = |Q|$ states that A-recognizes E . Then there exists an accepting run

$$\begin{array}{ccccccc} \alpha & = & a & & a & \dots & a & & b & \dots \\ \rho & = & q_0 & \longrightarrow & q_1 & \longrightarrow & \dots & q_{n_0-1} & \longrightarrow & q_{n_0} & \longrightarrow & \dots \end{array}$$

of $\mathcal{A}$ on $\alpha = a^{n_0}b^{n_0}\emptyset^\omega$. Then the run visits some state $q \in Q$ at least twice until all a have been consumed, say at position i and j , $0 \leq i < j \leq n_0$, i.e. $q_i = q_j = q$. Then, we can create a run by repeating the fragment $q_i \dots q_j$ twice instead of once:

$$\rho' = q_0 q_1 \dots q_{i-1} \underbrace{q_i q_{i+1} \dots q_{j-1} q_i q_{i+1} \dots q_{j-1} q_j}_{\substack{= q_j \\ q \rightsquigarrow q \rightsquigarrow q}} q_{j+1} \dots$$

ρ' is (A-)accepting for the word $\alpha = a^{n+(j-i)}b^{n_0}\emptyset^\omega \notin E$ since the states appearing in ρ' are the same as the states appearing in ρ and ρ by definition only contains states in F . Contradiction to $\mathcal{A}$ A-recognizing E .

Exercise 3 (DBA):

(2 points)

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

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

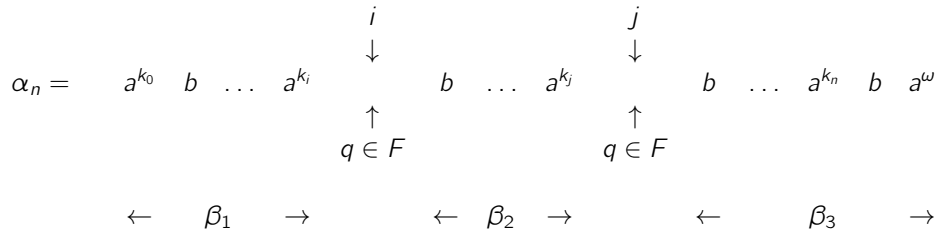
Solution:

Proof by contraposition: assume there is a DBA $\mathcal{A} = (Q, \Sigma, \delta, Q_0, F)$ that accepts $\mathcal{L}$ with $|F| = n$. For $\alpha \in \Sigma^\omega$, let $r(\alpha)$ denote the unique run of $\mathcal{A}$ on α . Recall that a word α is accepted if and only if $r(\alpha)$ visits F infinitely often. Let $\rho[i]$ denote the i th state of a run ρ .

As $\alpha_0 := a^\omega \in \mathcal{L}$, $r(\alpha_0)[i] \in F$ for infinitely many i . Let k_0 be the smallest such $i > 0$.

As $\alpha_1 := a^{k_0} b a^\omega \in \mathcal{L}$, $r(\alpha_1)[i] \in F$ for infinitely many i . Let k_1 be the smallest such i with $i > k_0$.

Doing this $n + 1$ times yields a word $\alpha_n := a^{k_0} b a^{k_1} b a^{k_2} \dots b a^{k_n} b a^\omega \in \mathcal{L}$ with $r(\alpha_n) \in F$ whenever a b is to be read next. Since $|F| \leq n$, $r(\alpha_n)$ must visit the same accepting state $q \in F$ at least twice when reading the prefix $a^{k_0} b a^{k_1} b a^{k_2} \dots b a^{k_n} b$, say at positions i and j ($0 \leq i < j \leq n$):



Then, however, there is an accepting run of $\mathcal{A}$ on

$$\beta_1 \beta_2^\omega \notin \mathcal{L}$$

which is a contradiction to $\mathcal{A}$ accepting $\mathcal{L}$.

Exercise 4 (Model Checking ω -regular Properties):

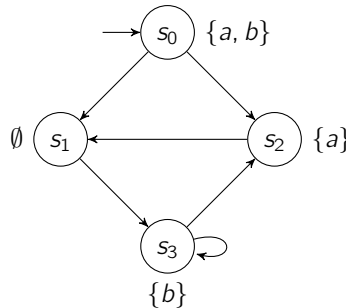
(4 points)

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 := "if a holds infinitely often, then b holds finitely often"

P_2 := " a holds infinitely often and b 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.

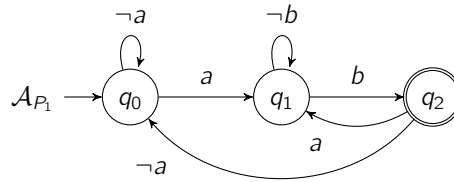
a) Derive suitable NBA $\mathcal{A}_{P_1}$, $\mathcal{A}_{P_2}$, where suitable means "appropriate for part b)-d)".

Hint: For P_1 you can find an automaton with 3 states and for P_2 4 states suffice. Derive the automata directly.

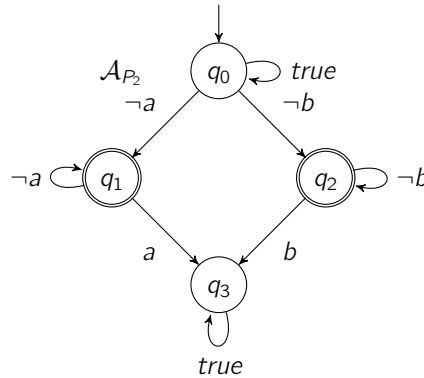
- b)** Outline the reachable fragments of the product transition systems $TS \otimes \mathcal{A}_{P_1}$ and $TS \otimes \mathcal{A}_{P_2}$.
- c)** Decide whether $TS \models P_1$ by checking an appropriate persistence property via nested depth-first search on $TS \otimes \mathcal{A}_{P_1}$. Document *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 $TS \models P_2$ by checking an appropriate persistence property via SCC analysis on $TS \otimes \mathcal{A}_{P_2}$. If the property is violated, provide a counterexample *based on your analysis*.

Solution: _____

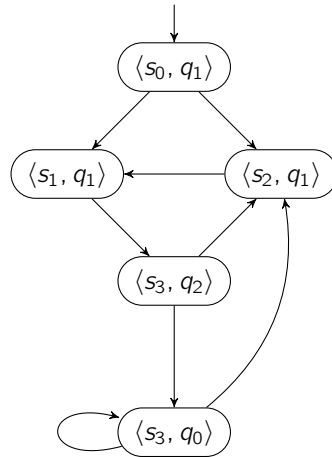
- a)** We require an NBA for the complement of P_1 , i.e. for the language “infinitely often a and infinitely often b ” ($= P_2$). The automaton can (for example) be chosen as follows.



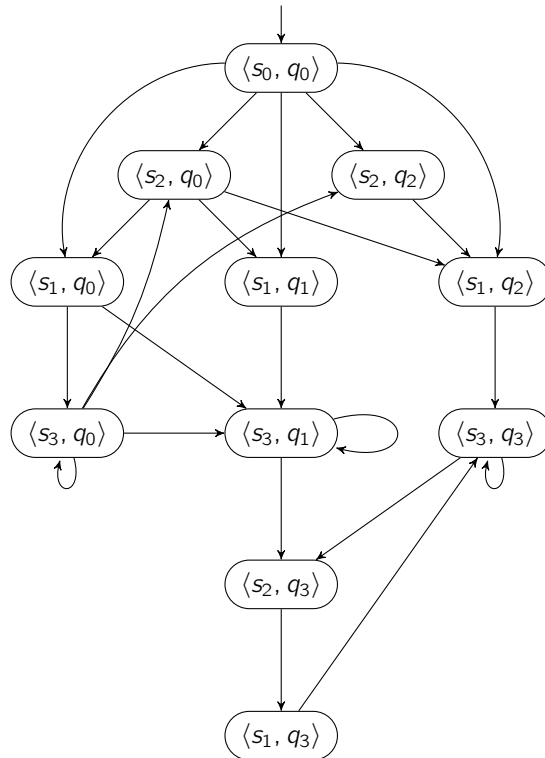
We also require an NBA for the complement of P_2 , which is P_1 . The automaton can (for example) be chosen as follows.



- b)** The reachable fragment of the product $TS \otimes \mathcal{A}_{P_1}$ is



and the reachable fragment of the product $TS \otimes \mathcal{A}_{P_2}$ is



c) The nested depth-first search works as follows

- Initialize
 - $U = \emptyset$
 - $\pi = \epsilon$
 - $V = \emptyset$
 - $\xi = \epsilon$
- insert $\langle s_0, q_1 \rangle$ in U and push $\langle s_0, q_1 \rangle$ to π :
 - $U = \{\langle s_0, q_1 \rangle\}$

- $\pi = \langle s_0, q_1 \rangle$
- $V = \emptyset$
- $\xi = \epsilon$
- insert $\langle s_1, q_1 \rangle$ in U and push $\langle s_1, q_1 \rangle$ to π :
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle$
 - $V = \emptyset$
 - $\xi = \epsilon$
- insert $\langle s_3, q_2 \rangle$ in U and push $\langle s_3, q_2 \rangle$ to π :
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle, \langle s_3, q_2 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle \langle s_3, q_2 \rangle$
 - $V = \emptyset$
 - $\xi = \epsilon$
- insert $\langle s_2, q_1 \rangle$ in U and push $\langle s_2, q_1 \rangle$ to π :
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle, \langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle \langle s_3, q_2 \rangle \langle s_2, q_1 \rangle$
 - $V = \emptyset$
 - $\xi = \epsilon$
- pop $\langle s_2, q_1 \rangle$ from π and — since $\langle s_2, q_1 \rangle \models \neg F$ — do *not* call CYCLE_CHECK($\langle s_2, q_1 \rangle$):
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle, \langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle \langle s_3, q_2 \rangle$
 - $V = \emptyset$
 - $\xi = \epsilon$
- insert $\langle s_3, q_0 \rangle$ in U and push $\langle s_3, q_0 \rangle$ to π :
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle, \langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle, \langle s_3, q_0 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle \langle s_3, q_2 \rangle \langle s_3, q_0 \rangle$
 - $V = \emptyset$
 - $\xi = \epsilon$
- pop $\langle s_3, q_0 \rangle$ from π and — since $\langle s_3, q_0 \rangle \models \neg F$ — do *not* call CYCLE_CHECK($\langle s_3, q_0 \rangle$):
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle, \langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle, \langle s_3, q_0 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle \langle s_3, q_2 \rangle$
 - $V = \emptyset$
 - $\xi = \epsilon$
- pop $\langle s_3, q_2 \rangle$ from π and — since $\langle s_3, q_2 \rangle \not\models \neg F$ — do call CYLCE_CHECK($\langle s_3, q_2 \rangle$):
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle, \langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle, \langle s_3, q_0 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle$
 - $V = \{\langle s_3, q_2 \rangle\}$
 - $\xi = \langle s_3, q_2 \rangle$
- insert $\langle s_2, q_1 \rangle$ in V and push $\langle s_2, q_1 \rangle$ to ξ :
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle, \langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle, \langle s_3, q_0 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle$

- $V = \{\langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle\}$
- $\xi = \langle s_3, q_2 \rangle \langle s_2, q_1 \rangle$
- insert $\langle s_1, q_1 \rangle$ in V and push $\langle s_1, q_1 \rangle$ to ξ :
 - $U = \{\langle s_0, q_1 \rangle, \langle s_1, q_1 \rangle, \langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle, \langle s_3, q_0 \rangle\}$
 - $\pi = \langle s_0, q_1 \rangle \langle s_1, q_1 \rangle$
 - $V = \{\langle s_3, q_2 \rangle, \langle s_2, q_1 \rangle, \langle s_1, q_1 \rangle\}$
 - $\xi = \langle s_3, q_2 \rangle \langle s_2, q_1 \rangle \langle s_1, q_1 \rangle$
- since $\langle s_3, q_2 \rangle \in \text{Post}(\langle s_1, q_1 \rangle)$, $\text{CYCLE_CHECK}(\langle s_3, q_2 \rangle)$ returns “true”
- nested DFS returns “no” + $\underbrace{(\langle s_0, q_1 \rangle \langle s_1, q_1 \rangle)}_{\alpha}, \underbrace{\langle s_3, q_2 \rangle \langle s_2, q_1 \rangle \langle s_1, q_1 \rangle}_{\beta}$

Since $\text{TS} \otimes \mathcal{A}_{P_1}$ does not satisfy the persistence property, we have $\text{TS} \not\models P_1$. A counterexample can be obtained from the result value of the algorithm by taking $\alpha\beta^\omega$.

d) There are three reachable, non-trivial SCCs in $\text{TS} \otimes \mathcal{A}_{P_2}$:

- $C_1 = \{\langle s_1, q_0 \rangle, \langle s_2, q_0 \rangle, \langle s_3, q_0 \rangle\}$
- $C_2 = \{\langle s_3, q_1 \rangle\}$
- $C_3 = \{\langle s_3, q_3 \rangle, \langle s_2, q_3 \rangle, \langle s_1, q_3 \rangle\}$

The intersection of C_2 with the set of states whose second component is an accepting state in $\mathcal{A}_{P_2}$ ($\{q_1, q_2\}$) is non-empty. Therefore, we have $\text{TS} \otimes \mathcal{A}_{P_2} \not\models \text{“eventually always } \neg F\text{”}$ where F is the set of accepting states of $\mathcal{A}_{P_2}$. Since $\text{TS} \otimes \mathcal{A}_{P_2}$ does not satisfy the persistence property, we have $\text{TS} \not\models P_2$. A counterexample can be obtained by taking an initial path fragment (*) to $\langle s_3, q_1 \rangle$ in the SCC C_2 and then construct a cycle (**) back to $\langle s_3, q_1 \rangle$. Therefore,

$$\underbrace{\langle s_0, q_0 \rangle \langle s_2, q_0 \rangle \langle s_1, q_1 \rangle}_{(*)} \underbrace{(\langle s_3, q_1 \rangle)^\omega}_{(**)}$$

is a suitable counterexample.