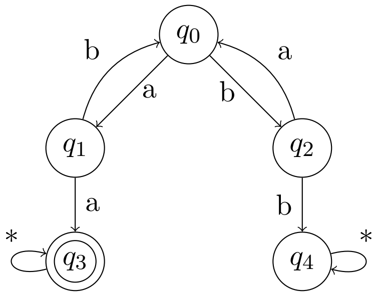
GandALF — Exercise Sheet 12

Exercise 1. Let $\mathcal{A}$ be a deterministic ω-PDA with Büchi acceptance conditions. Show that the language $\mathcal{L}(\mathcal{A})$ is a Borel set.

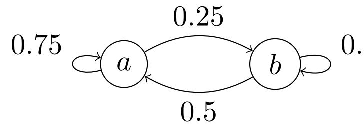
In the following two assignments we consider the Büchi automaton $\mathcal{A}_{aa}$, the Markov chain $\mathcal{M}_1$ and the MDP $\mathcal{M}_2$ depicted below.

Exercise 2. Compute the probability of $\mathcal{L}(\mathcal{A})$ under the probability measure given by $\mathcal{M}_1$.

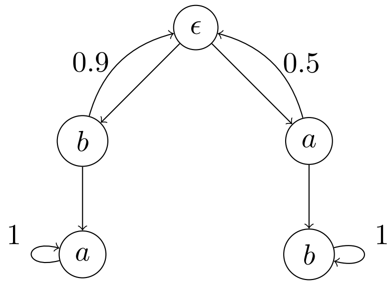
Exercise 3. Compute the maximal probability of $\mathcal{L}(\mathcal{A})$ w.r.t. $\mathcal{M}_2$ and construct a scheduler attaining this maximum.



$\mathcal{A}_{aa}$



$\mathcal{M}_1$



$\mathcal{M}_2$

Exercise 4. Consider input Boolean variables $P_I = \{r_1, r_2\}$ and output variables $P_O = \{g_1, g_2\}$. Present all steps of the synthesis procedure for the specification $(G(\neg g_1 \vee \neg g_2)) \wedge (G(r_1 \to F g_2)) \wedge (G(r_2 \to F g_2))$ to obtain a transducer with the input 2^{P_I} and output 2^{P_O} that satisfies the specification.

Exercises 5–8 are about a moder approach to synthesis.

Exercise 5. Given an LTL specification φ over variables $P_I \cup P_O$, construct a universal co-Büchi tree automaton (UCT) recognizing full infinite 2^{P_O}-labeled 2^{P_I}-trees t such that every branch of t satisfies φ. *Recall that "universal automata" are the dual to non-deterministic automata, i.e., in an accepting run all possible paths of computation need to be accepting.*

Hint: Construct a universal co-Büchi word automaton in an intermediate step.

Exercise 6. k-UTC (universal k-co-Büchi tree automata) are defined similarly to UTC, but the acceptance condition is stronger; it states that along each path, every state from F is visited at most k times. Show that for every UCT $\mathcal{A}$, there exists k such that $\mathcal{L}(\mathcal{A})$ is non-empty if and only if the language of $\mathcal{A}$ considered as k-UCT is non-empty. Give a bound on k w.r.t. the size of $\mathcal{A}$?

Exercise 7. Construct an algorithm checking emptiness of the language of k-UTC. Discuss its complexity w.r.t. k and the size of the input automaton.

Exercise 8. Combine Exercises 5, 6, and 7 to give an alternative synthesis algorithm.

Exercise 9. Prove undecidability of the realizability from components problem.