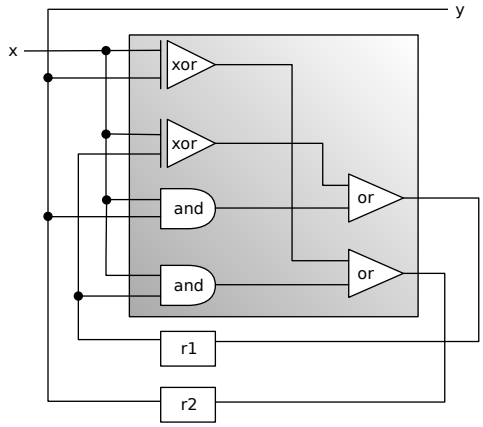
Applied Automata Theory (WS 2012/2013) Technische Universität Kaiserslautern

Exercise Sheet 7

Jun.-Prof. Roland Meyer, Reiner Hüchting, Georgel Călin Due: Tue, Dec 4 **(noon)**

Exercise 7.1 Circuit Verification

Consider a circuit[1] that continuously receives inputs x and generates outputs y:



The circuit uses registers r_1 and r_2, which are initially $r_1 = 0$ and $r_2 = 1$.

(a) Construct a Büchi automaton over the alphabet $\{0,1\}^2$ that accepts all sequences of input/output pairs which describe the possible runs of the circuit.

Hint: The states are determined by r_1 and r_2 and the transitions only depend on x.

(b) Use the automaton to determine whether the circuit satisfies the properties ...

P_{fair}: whenever x is infinitely often high, then y is infinitely often high.

P_{safe}: always $x = y = 1$ or $x = y = 0$.

$P_{\text{persistent}}$: starting from some point, y will always be high.

(c) Give words (finite if possible) that satisfy P_i and $\neg P_i$ for each $i \in \{\text{fair}, \text{safe}, \text{persistent}\}$.

Exercise 7.2 Verifying Operating Systems

Our goal is to verify an operating systems OS that runs k processes and has a scheduler. This means we are given the following Büchi automata:

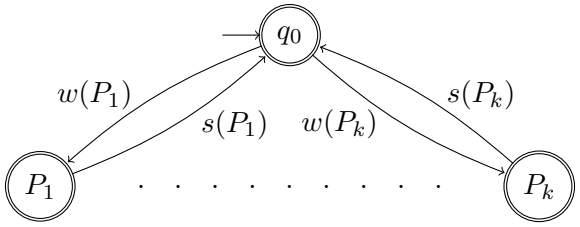
$A_{\text{OS}} := A_{P_1} \parallel \ldots \parallel A_{P_k}$:	Describes the behaviour of the operating system, where A_{P_i} represents the behavior of process P_i.
A_{Sched}:	Describes the scheduling strategy.
A_{Prop}:	Describes a property to be checked.

[1] Inspired by C. Baier & J.P. Katoen: Principles of Model Checking

Our verification task amounts to solving the following model checking problem:

$$\mathsf{L}(A_{\mathrm{OS}}) \cap \mathsf{L}(A_{\mathrm{Sched}}) \subseteq \mathsf{L}(A_{\mathrm{Prop}}).$$

However, we do not want to solve this problem separately for every scheduling strategy, but in a general way. Therefore, we introduce a *most general* scheduling Büchi automaton A_{MG}, that allows for arbitrary behaviors of the scheduler:



The alphabet contains letters $w(P_i), s(P_i)$ for all processes P_i, meaning *wake up* or *suspend* the respective process. Thus, the scheduler can wake up and suspend processes at will, and the processes only work when awake. The problem with this general scheduler is that it is not *fair*: it does not necessarily wake up each process infinitely often.

(a) Modify A_{MG} to a Büchi automaton A_{MGF} that is a most general *fair* scheduler for k processes. This means your automaton has to wake up every process infinitely often and the behavior of your Büchi automaton must be as general as possible. In particular, do not implement a concrete scheduling strategy.

(b) Present an automaton A_{RR} that describes the *Round Robin* scheduling strategy. What is the relationship between $\mathsf{L}(A_{\mathrm{RR}})$ and $\mathsf{L}(A_{\mathrm{MG}})$ respectively $\mathsf{L}(A_{MGF})$?

(c) Why can you conclude $\mathsf{L}(A_{\mathrm{OS}}) \cap \mathsf{L}(A_{\mathrm{RR}}) \subseteq \mathsf{L}(A_{\mathrm{Prop}})$ from $\mathsf{L}(A_{\mathrm{OS}}) \cap \mathsf{L}(A_{\mathrm{MGF}}) \subseteq \mathsf{L}(A_{\mathrm{Prop}})$?

Exercise 7.3 On NBA Complementation

Let A be a Büchi automaton and $U, V \subseteq \Sigma^*$ be equivalence classes with respect to $\sim_A$.

(a) Let $w \in \mathsf{L}(A)$ and assume $w \in UV^\omega$. Prove $UV^\omega \subseteq \mathsf{L}(A)$.

(b) Suppose $w \in \overline{\mathsf{L}(A)}$ and $w \in UV^\omega$. Prove $UV^\omega \subseteq \overline{\mathsf{L}(A)}$.

Exercise 7.4 Disjunctive Well-Foundedness

A partially ordered set $(A, \leq)$ is said to be *well-founded* if for every sequence

$$a_1 \geq a_2 \geq a_3 \geq \cdots,$$

$a_i \in A$, $i \in \mathbb{N}$, there is an $n \in \mathbb{N}$ such that $a_m = a_n$ for any $m \geq n$.

Let $T_1, \ldots, T_n \subseteq A \times A$ be well-founded partial orders and $R \subseteq A \times A$ be a partial order such that $R \subseteq T_1 \cup \cdots \cup T_n$. Show that R is well-founded, too.

Hint: Use Ramsey's Theorem.