Universität Leipzig
Institut für Informatik
Winter 2026/2027

Prof. Dr. Karin Quaas

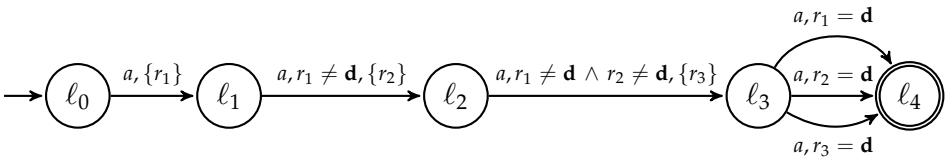
Exercises

Verification for Infinite-State Systems

Solutions for Exercise Sheet 2 (for 29.10.2026)

Exercise 2.1

Is the following register automaton deterministic and/or complete?



SOLUTION: It is not deterministic: $r_1 = \mathbf{d} \wedge r_2 = \mathbf{d}$ is satisfiable and we have $(\ell_3, a, r_1 = \mathbf{d}, \emptyset, \ell_4) \in E$ and $(\ell_3, a, r_2 = \mathbf{d}, \emptyset, \ell_4) \in E$. It is not complete, because for ℓ_1, a, and the register test $\phi = (r_1 = \mathbf{d} \wedge r_2 = \mathbf{d}) \in \Phi^{\mathrm{cpl}}(\mathcal{R})$ there does not exist any edge $(\ell_1, a, \phi', \lambda, \ell) \in E$ such that $\phi \models \phi'$.

Exercise 2.2

Prove that the decision problem is in **P**: given a finite set of registers $\mathcal{R}$ and two register tests ϕ, ϕ' in $\Phi(\mathcal{R})$, does $\phi \models \phi'$ hold?

SOLUTION: We are given a finite set of registers $\mathcal{R} = \{r_1, \ldots, r_n\}$, and two register tests ϕ, ϕ' in $\Phi(\mathcal{R})$. We aim to decide whether $\phi \models \phi'$, that is, whether ϕ entails ϕ'. Let us first prove that $\phi \models \phi'$ iff $\phi \wedge \neg\phi'$ is not satisfiable. (Note that $\neg\phi'$ is actually not a valid register test, as it may contain disjunction. However, we can easily adapt the notion of $\models$ to such formulas.)

"if" (By contraposition.) Suppose not $\phi \models \phi'$, that is $\nu, d \models \phi$ and not $\nu, d \models \phi'$ for some valuation ν and datum d. Then $\nu, d \models \phi$ and $\nu, d \models \neg\phi'$. Then $\nu, d \models \phi \wedge \neg\phi'$. Then $\phi \wedge \neg\phi'$ is satisfiable.

"only if" Suppose $\phi \models \phi'$, that is, for every valuation ν and datum d we have $\nu, d \models \phi$ implies $\nu, d \models \phi'$. Towards contradiction, suppose $\phi \wedge \neg\phi'$ is satisfiable. Then there exists valuation ν and datum d such that $\nu, d \models \phi \wedge \neg\phi'$. Hence $\nu, d \models \phi$ and $\nu, d \models \neg\phi'$. Hence $\nu, d \models \phi$ and not $\nu, d \models \phi'$, contradiction.

For deciding $\phi \models \phi'$, we exploit the above claim. Let $\phi' = \psi'_1 \wedge \cdots \wedge \psi'_m$, where ψ'_i is atomic for all $1 \leq i \leq m$. Clearly, $\phi \wedge \neg\phi'$ is satisfiable iff there exists $1 \leq i \leq m$ such that $\phi \wedge \neg\psi'_i$ is satisfiable. Hence check for each $1 \leq i \leq m$, whether $\phi \wedge \neg\psi'_i$ is satisfiable (in **P**, by Exercise 1.4). If yes, then return NO (ϕ does not entail ϕ' by the above claim), and if no for all $1 \leq i \leq m$, then return YES ($\phi \models \phi'$ because $\phi \wedge \neg\phi'$ is not satisfiable). This can clearly be done in **P**.

Exercise 2.3

Prove that the problem of deciding whether a given register automaton $\mathcal{A}$ is **not** complete is in **NP**.

SOLUTION: Let $\mathcal{A} = (\mathcal{L}, \Sigma, \mathcal{R}, \mathcal{L}_{\text{in}}, E, \mathcal{L}_{\text{acc}})$ be the input register automaton. $\mathcal{A}$ is not complete if there exists $\ell \in \mathcal{L}$, $a \in \Sigma$, and some register test $\phi \in \Phi^{\text{cpl}}(\mathcal{R})$ such that there exists no edge $(\ell, a, \phi', \lambda, \ell') \in E$ with $\phi \models \phi'$. So guess the witness ℓ, a and ϕ. The size of the witness is polynomial in the size of $\mathcal{A}$. Deciding whether there exists no edge $(\ell, a, \phi', \lambda, \ell') \in E$ with $\phi \models \phi'$ can be done in polynomial time: go through all edges $(\ell, a, \phi', ., .) \in E$ (at most $|E|$ many) and for each of them decide in **P** whether $\phi \models \phi'$ (by previous assignment).

Exercise 2.4

Give a polynomial-time-reduction from the problem to decide whether a given propositional formula in conjunctive normal form is unsatisfiable to the problem of deciding whether a given register automaton is complete.

SOLUTION: Let $\varphi = \psi_1 \wedge \psi_2 \wedge \cdots \wedge \psi_m$ be a propositional formula in CNF, consisting of m clauses and over n variables $x_1, \ldots, x_n$. We define a register automaton $\mathcal{A}_\varphi = (\mathcal{L}, \{a\}, \mathcal{R}, \mathcal{L}_{\text{in}}, E, \mathcal{L}_{\text{acc}})$ as follows:

- $\mathcal{L} = \mathcal{L}_{\text{in}} = \mathcal{L}_{\text{acc}} = \{\ell\}$ ... a single location
- $\Sigma = \{a\}$ ... a single letter a
- $\mathcal{R} = \{r_1, \ldots, r_n\}$ ... for each variable x_i a register r_i
- For each clause ψ_j, define an edge of the form $(\ell, a, \phi'_j, \emptyset, \ell)$, where ϕ'_j corresponds to the negation of clause ψ_j in the following sense: ϕ'_j is a conjunction of atomic register tests, and this conjunction contains the atomic register test $r_i = \mathbf{d}$ if x_i occurs negatively in ψ_j, and it contains $r_i \neq \mathbf{d}$ if x_i occurs positively in ψ_j. For example, if $\psi_1 = (\neg x_1 \vee x_3 \vee \neg x_4)$, then $\phi'_1 = (r_1 = \mathbf{d} \wedge r_3 \neq \mathbf{d} \wedge r_4 = \mathbf{d})$.

Let us prove that φ is unsatisfiable iff $\mathcal{A}_\varphi$ is complete.

Suppose φ is unsatisfiable. That is, for every assignment $\sigma : \{x_1, \ldots, x_n\} \to \{0, 1\}$ there exists $1 \leq i \leq m$ such that ψ_i is evaluated to 0 under σ.

Let $\phi \in \Phi^{\text{cpl}}(\mathcal{R})$ be arbitrary. We prove that there exists ϕ' such that $(\ell, a, \phi', \emptyset, \ell) \in E$ and $\phi \models \phi'$. Define the assignment $\sigma_\phi : \{x_1, \ldots, x_n\} \to \{0, 1\}$ by

$$\sigma_\phi(x_i) = \begin{cases} 1 & \text{if } r_i = \mathbf{d} \in \phi \\ 0 & \text{if } r_i \neq \mathbf{d} \in \phi \end{cases}.$$

Let $1 \leq i \leq m$ be such that σ_ϕ evaluates ψ_i to 0. Note that ψ_i is a disjunction, for example, of the form $x_k \vee \neg x_p$. In order to be evaluated to 0 under σ_ϕ, we must have $\sigma_\phi(x_k) = 0$ and $\sigma_\phi(x_p) = 1$. By definition of σ_ϕ this implies $r_k \neq \mathbf{d} \in \phi$ and $r_p = \mathbf{d} \in \phi$.

By definition of E, the edge corresponding to ψ_i is of the form $r_k \neq \mathbf{d} \wedge r_p = \mathbf{d}$. Hence $\phi \models \phi'_j$.

Suppose $\mathcal{A}_\varphi$ is complete. We prove that φ is unsatisfiable. So let $\sigma : \{x_1, \ldots, x_n\} \to \{0,1\}$ be an arbitrary variable assignment. Let $\phi_\sigma \in \Phi^{\text{cpl}}(\mathcal{R})$ be a corresponding complete register test, that is, ϕ contains $r_i = \mathbf{d}$ if $\sigma(x_i) = 1$ and ϕ contains $r_i \neq \mathbf{d}$ if $\sigma(x_i) = 0$. By assumption, there exists some ϕ'_j such that $(\ell, a, \phi'_j, \emptyset, \ell) \in E$ and $\phi \models \phi'_j$. By construction, the clause ψ_j in φ is the negation of ϕ'_j, and hence σ evaluates ψ_j to 0. Hence φ is evaluated to 0 under σ. Since σ was arbitrary, we have proven that φ is unsatisfiable.

Note that the problem of deciding whether a given register automaton is complete is **coNP**-complete by what we proved in the preceding two assignments.

Exercise 2.5

Which of the following data languages over $\Sigma = \{a, b\}$ are data regular, that is, can be accepted by a register automaton? Prove your answer!

(a) $L_{11} = \{(a, d_1) \ldots (a, d_n) \mid d_i \neq d_n \text{ for all } 1 \leq i < n, n \geq 1\}$

(b) $L_{12} = \{(a_1, d_1) \ldots (a_n, d_n) \mid$ there exists some $1 \leq i \leq n$ such that $a_i = a$, and $a_j \neq b$ or $d_j \neq d_i$ for all $i < j \leq n\}$

SOLUTION: (a) We show that $L_{11} = \{(a, d_1) \ldots (a, d_n) \mid d_i \neq d_n \text{ for all } 1 \leq i < n, n \geq 1\}$ cannot be accepted by any register automaton. Towards contradiction suppose there exists some register automaton $\mathcal{A}$ with $L(\mathcal{A}) = L_{11}$. Suppose $\mathcal{A}$ uses k registers. Consider the data word w of the form

$$(a, d_1)(a, d_2) \ldots (a, d_k)(a, d_{k+1})(a, d_{k+2}),$$

where $d_i \neq d_j$ for all $1 \leq i < j \leq k+2$ (all data are pairwise distinct). Hence $w \in L_{11}$ and thus, there must exist some initialized accepting run of $\mathcal{A}$ on w of the form

$$(\ell_0, \nu_0) \xrightarrow{(a,d_1)} \ldots \xrightarrow{(a,d_k)} (\ell_k, \nu_k) \xrightarrow{(a,d_{k+1})} (\ell_{k+1}, \nu_{k+1}) \xrightarrow{(a,d_{k+2})} (\ell_{k+2}, \nu_{k+2})$$

There exists some $1 \leq i \leq k+1$ such that $d_i \notin \nu_{k+1}(r)$ for all registers r in $\mathcal{A}$ (because we have $k+1$ data but only k registers), that is, $\mathcal{A}$ has "forgotten" the datum d_i after processing the first $k+1$ letters of w. But then, $\nu_{k+1}, d_{k+2} \models \phi$ if, and only if, $\nu_{k+1}, d_i \models \phi$, where ϕ is the register test at the edge used in the last transition of the run above. Hence

$$(\ell_0, \nu_0) \xrightarrow{(a,d_1)} \ldots \xrightarrow{(a,d_k)} (\ell_k, \nu_k) \xrightarrow{(a,d_{k+1})} (\ell_{k+1}, \nu_{k+1}) \xrightarrow{(a,d_i)} (\ell_{k+2}, \nu_{k+2})$$

is an initialized accepting run of $\mathcal{A}$ on

$$(a, d_1)(a, d_2) \ldots (a, d_k)(a, d_{k+1})(a, d_i).$$

But this word is not in L as $d_i = d_{k+2}$. Contradiction to $L_{11} = L(\mathcal{A})$.

(b) This data language can be accepted by the following register automaton: in the first location, it nondeterministically guesses the position where a is read and for there will in future never be any b with the same datum.

