Übungen zu Model Checking

Abgabe bis zum

Aufgabe 2.1 ω-reguläre Sprachen vs. reguläre Sprachen

Zeigen Sie, dass für jeden Büchi-Automaten $\mathcal{B}$ über dem Alphabet Σ reguläre Sprachen $U_i, V_i \subseteq \Sigma^*$ für $i \in \{1, \ldots, n\}$ existieren mit

$$L(\mathcal{B}) = \bigcup_{i=1}^{n} U_i V_i^\omega.$$

Dabei ist n eine Konstante, welche allein von der Anzahl der akzeptierenden Zustände von $\mathcal{B}$ abhängt.

Betrachten Sie hierzu für ein Wort $w = w_0 w_1 w_2 \ldots \in L(\mathcal{B})$ eine akzeptierende Berechnung $q = q_0 q_1 \ldots$ von $\mathcal{B}$ mit $q_{i+1} \in \delta(q_i, w_i)$. Es muss somit ein akzeptierender Zustand q_a unendlich oft in q auftreten. Zerlegen Sie dann mit Hilfe von q_a den Büchi-Automaten $\mathcal{B}$ geeignet in endliche Automaten.

Aufgabe 2.2

Sei Σ ein Alphabet, d.h., eine beliebige endliche, nicht-leere Menge. Dann ist $\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$ die Menge aller abzählbaren Wörter über Σ. Die Sprache $\mathcal{L}_{\text{reg}}$ der regulären Ausdrücke über Σ ist, wie bekannt, folgendermaßen definiert:

$$\begin{array}{lcl}
\mathcal{L}^0_{\text{reg}} & := & \{\emptyset\} \cup \Sigma \cup \{\varepsilon\} \\
\mathcal{L}^{i+1}_{\text{reg}} & := & \mathcal{L}^i_{\text{reg}} \cup \{(\phi + \psi), \phi\psi, (\phi)^* \mid \phi, \psi \in \mathcal{L}^i_{\text{reg}}\} \\
\mathcal{L}_{\text{reg}} & := & \bigcup_{i \in \mathbb{N}} \mathcal{L}^i_{\text{reg}}.
\end{array}$$

Entsprechend ist die Sprache der ω-regulären Ausdrücke über Σ gegeben durch

$$\begin{array}{lcl}
\mathcal{L}^0_{\omega-\text{reg}} & := & \mathcal{L}_{\text{reg}} \cup \{(\phi)^\omega \mid \phi \in \mathcal{L}_{\text{reg}}\} \\
\mathcal{L}^{i+1}_{\omega-\text{reg}} & := & \mathcal{L}^i_{\omega-\text{reg}} \cup \{(\phi + \psi) \mid \phi, \psi \in \mathcal{L}^i_{\omega-\text{reg}}\} \cup \{\phi\psi \mid \phi \in \mathcal{L}_{\text{reg}}, \psi \in \mathcal{L}^i_{\omega-\text{reg}}\} \ . \\
\mathcal{L}_{\omega-\text{reg}} & := & \bigcup_{i \in \mathbb{N}} \mathcal{L}^i_{\omega-\text{reg}}
\end{array}$$

Die Semantik $L(\phi) \subseteq \Sigma^\infty$ eines ω-regulären Ausdrucks $\phi \in \mathcal{L}_{\omega-\text{reg}}$ ist dann induktiv über den Formelaufbau definiert:

$$\begin{array}{lcl}
L(\emptyset) & := & \emptyset \\
L(\varepsilon) & := & \{\varepsilon\} \\
L(a) & := & \{a\} \, (\forall a \in \Sigma) \\
L((\phi + \psi)) & := & L(\phi) \cup L(\psi) \\
L((\phi)^*) & := & \{w_0 w_1 \ldots w_n \mid \forall n \in \mathbb{N} \, \forall i \leq n : w_i \in L(\phi)\} \\
L(\phi\psi) & := & \{uw \mid u \in L(\phi), w \in L(\psi)\} \\
L((\phi)^\omega) & := & \{w_0 w_1 w_2 \ldots \mid \forall n \in \mathbb{N} \, w_n \in L(\phi)\}
\end{array}$$

Zeigen Sie, dass es für jeden ω-regulären Ausdruck $\phi \in \mathcal{L}_{\omega-\text{reg}}$ einen nichtdeterministischen Büchi-Automaten $\mathcal{B}_\phi$ mit $L(\mathcal{B}_\phi) = L(\phi)$ gibt. Ihre Büchi-Automaten dürfen auch ε-Übergänge besitzen.

Verwenden Sie die bekannten Sätze über die Äquivalenz von regulären Ausdrücken und endlichen Automaten.

Aufgabe 2.3 Büchi-Automaten: deterministisch vs. nichtdeterministisch

(a) Geben Sie einen nichtdeterministischen Büchi-Automaten $\mathcal{B}$ für die Sprache

$$L := L((a+b)^*(b)^\omega) = \{w \in \{a,b\}^\omega \mid \text{ In } w \text{ kommen nur endlich viele } a \text{ vor.}\}$$

an.

(b) Determinisieren Sie den Büchi-Automaten $\mathcal{B}$ aus der letzten Teilaufgabe mit Hilfe des Verfahrens für endliche Automaten (Potenzmengenkonstruktion). Der so erhaltene Büchi-Automat heiße $\mathcal{B}'$. Geben Sie dann $L(\mathcal{B}')$ an.

(c) Zeigen Sie, dass es keinen deterministischen Büchi-Automaten gibt, welcher L akzeptiert.

Nehmen Sie hierfür an, dass $\mathcal{B}'' = (Q, \{a,b\}, \delta, q_o, F)$ ein *deterministischer* Büchi-Automat mit $L(\mathcal{B}'') = L$ ist, welcher $n = |F|$ akzeptierende Zustände besitzt.

Betrachten Sie dann das Verhalten von $\mathcal{B}''$ bezüglich der Familie von Eingabewörtern:

$$\begin{array}{lcl} w_1 & = & b^\omega \\ w_2 & = & b^{k_1}ab^\omega \\ w_3 & = & b^{k_1}ab^{k_2}ab^\omega \\ w_4 & = & \dots \end{array} .$$

Die $k_i \in \mathbb{N}$ und $w_i \in \{a,b\}^\omega$ sind dabei folgendermaßen induktiv definiert beginnend mit $w_1 = b^\omega$, $k_0 = 0$ und $i = 1$:

k_i ist die kleinste natürliche Zahl größer $k_{i-1}+1$, so dass sich $\mathcal{B}''$ nach Lesen der ersten k_i Zeichen von w_i in einem akzeptierenden Zustand befindet. w_{i+1} ist dann als $(w_i)_0(w_i)_1\dots(w_i)_{k_i}ab^\omega$ definiert.

Unter den Annahme, dass $L(\mathcal{B}'') = L$ gilt und $\mathcal{B}''$ deterministisch ist, existiert diese Folge.

Leiten Sie mit Hilfe dieser Wörter $w_1, w_2, \dots$ einen Widerspruch zu der obigen Annahme ($L(\mathcal{B}'') = L$ und $\mathcal{B}''$ deterministisch) her.

Aufgabe 2.4 LTL zu Büchi-Automat

(a) Auf dem letzten Aufgabenblatt wurden NF-**LTL**-Formeln definiert. Diese verwenden $\mathbf{G}$ anstatt $\mathbf{R}$, um die Negation von $\phi\,\mathbf{U}\,\psi$ darstellen zu können. (Erinnerung: $\neg(\phi_1\,\mathbf{U}\,\phi_2) \equiv (\neg\phi_1)\,\mathbf{R}(\neg\phi_2) \equiv (\mathbf{G}\,\neg\phi_2)\vee((\neg\phi_2)\,\mathbf{U}(\neg\phi_1\wedge\neg\phi_2)))$

Welchen Vorteil bietet die Darstellung mittels $\mathbf{R}$ in Bezug auf das in der Vorlesung vorgestellte Verfahren zur Konstruktion eines entsprechenden Büchi-Automaten?

(b) Konstruieren Sie mit dem Verfahren aus der Vorlesung einen Büchi-Automaten für die Formel $p\,\mathbf{R}\,q$ über dem Alphabet $2^{\mathbf{AP}}$ mit $\mathbf{AP} = \{p,q\}$.

(c) Geben Sie einen möglichst kleinen Büchi-Automaten für die Formel $(\mathbf{G}\,p) \to (p\,\mathbf{U}\,q)$ an. Sie müssen *nicht* den Algorithmus aus der Vorlesung verwenden.

Aufgabe 2.5

Geben Sie jeweils eine Kripkestruktur $\mathcal{K}$ und eine **LTL**-Formel ϕ an mit

$$\begin{array}{ll} (a) & (\mathcal{K} \models \phi \vee \psi) \wedge (\mathcal{K} \not\models \phi) \wedge (\mathcal{K} \not\models \psi) \\ (b) & (\mathcal{K} \not\models \phi) \wedge (\mathcal{K} \not\models \neg\phi) \end{array}$$

Erinnerung:

- Ein Zustand s einer Kripkestruktur $\mathcal{K}$ erfüllt die **LTL**-Formel ϕ (kurz: $\mathcal{K}, s \models \phi$), falls *jede* in s beginnende Berechnung ϕ erfüllt.
- $\mathcal{K} \models \phi$ gilt genau dann, falls der Anfangszustand r von $\mathcal{K}$ die Formel ϕ erfüllt.