Institut für Informatik I
Prof. Dr. Heiko Röglin
Tobias Brunsch

Musterlösung Übungsblatt 10

Aufgabe 10.4

(c) Es gilt $L(M_1) = L(M_2)$ genau dann, wenn $L(M_1) \,\Delta\, L(M_2) := (L(M_1) \cap \overline{L(M_2)}) \cup (L(M_2) \cap \overline{L(M_1)}) = \emptyset$ gilt. Dies können wir uns auf verschiedene Arten zu Nutze machen.

Ansatz 1

Eine Möglichkeit besteht darin, die Aufgabenteile (a) und (b) zu verwenden. Nach Aufgabenteil (b) können wir Automaten M_3 und M_4 mit $|Q_1|$ bzw. $|Q_2|$ Zuständen konstruieren, die die Sprachen $\overline{L(M_1)}$ bzw. $\overline{L(M_2)}$ entscheiden. Nach Aufgabenteil (a) können wir dann Automaten M_5 und M_6 mit je $|Q_1| \cdot |Q_2|$ Zuständen konstruieren, die die Sprachen $L(M_1) \cap L(M_4)$ bzw. $L(M_2) \cap L(M_3)$ entscheiden.

Nun können wir wie folgt vorgehen: Konstruiere den Automaten M_5 und teste, ob $L(M_5) = \emptyset$ gilt. (Wie das geht, diskutieren wir später.) Ist das nicht der Fall, dann gilt $L(M_1) \neq L(M_2)$. Ansonsten konstruieren wir den Automaten M_6 und testen, ob $L(M_6) = \emptyset$ gilt. Ist das nicht der Fall, dann gilt $L(M_1) \neq L(M_2)$, ansonsten gilt Gleichheit.

Natürlich hätte man nach Aufgabenteil (a) auch einen Automaten M bauen können, der die Sprache $L(M_5) \cup L(M_6)$ entscheidet, um dann zu testen, ob $L(M) = \emptyset$ gilt. Da dieser aber $|Q_1|^2 \cdot |Q_2|^2$ Zustände hat, wäre dadurch unsere Laufzeitrestriktion von $O(|Q_1| \cdot |Q_2| \cdot |\Sigma|)$ verletzt.

Ansatz 2

Der zweite Ansatz verläuft analog zu Aufgabenteil (a), indem wir einen Automaten $M = (Q, \Sigma, \delta, q_0, F)$ konstruieren, der die Sprache $L(M_1) \,\Delta\, L(M_2)$ entscheidet. Dazu setzen wir $Q := Q_1 \times Q_2$, $q_0 := (q_0^1, q_0^2)$ und $\delta((q_1, q_2), a) := (\delta_1(q_1, a), \delta_2(q_2, a))$ für alle Zustände $(q_1, q_2) \in Q$ und alle Zeichen $a \in \Sigma$. Dieser simuliert beide Automaten M_1 und M_2 gleichzeitig. Das entscheidende ist die Wahl der akzeptierenden Zustände. Wir wollen wissen, ob es ein Wort gibt, das einen der beiden Automaten in einen akzeptierenden Zustand und den anderen in einen nichtakzeptierenden Zustand überführt. Als Menge akzeptierender Zustände wählen wir daher $F := \{(q_1, q_2) \in Q_1 \times Q_2 : q_1 \in F_1 \iff q_2 \notin F_2\}$. Dann gilt $L(M_1) = L(M_2) \iff L(M) = \emptyset$, d.h. wir testen, ob M die leere Sprache entscheidet.

Wie testet man für einen Automaten $M = (Q, \Sigma, \delta, q_0, F)$, ob $L(M) = \emptyset$ gilt?

Ein Automat entscheidet genau dann die leere Sprache, wenn alle seine akzeptierenden Zustände überflüssige Zustände sind. Es genügt also, den Automaten M als gerichteten Graphen auf der Knotenmenge Q aufzufassen und darauf eine Tiefensuche[1] durchzuführen, beginnend mit Knoten q_0. Dies ist in Zeit $O(|Q| \cdot |\Sigma|)$ möglich. Es gilt $L(M) = \emptyset$ genau dann, wenn bei der Suche kein Knoten/Zustand $q \in F$ besucht wurde.

[1] Natürlich beenden wir die Tiefensuche, sobald wir Knoten q_0 abgearbeitet haben.