Diskrete Modellierung

Wintersemester 2016/2017

Mario Holldack, M. Sc.
Prof. Dr. Georg Schnitger
Hannes Seiwert, M. Sc.

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

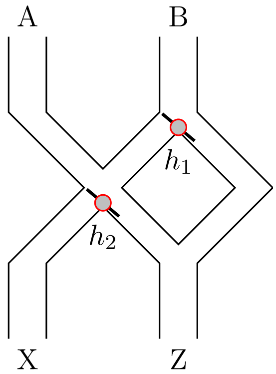
Institut für Informatik
AG Theoretische Informatik

Übungsblatt 12

Ausgabe: 19.01.17
Abgabe: 26.01.17

Aufgabe 12.1 *Modellierung mit DFAs* $(13 + 3 + 6 + 3 = 25$ Punkte)

Die nachfolgende Abbildung zeigt ein Spiel, in dem Murmeln in Öffnungen A oder B in die Spielbahn fallen gelassen werden.



Je nach Stellung der Hebel h_1 und h_2 rollen die eingeworfenen Murmeln an der jeweiligen Kreuzung nach links unten oder rechts unten. Sobald eine Murmel auf einen dieser Hebel trifft, wird dieser nach dem Passieren der Murmel umgestellt, sodass die nächste Murmel, welche die Kreuzung passiert, in die andere Richtung rollt. Zu Beginn sind beide Hebel so eingestellt, dass die nächste Murmel, die auf den Hebel trifft, nach rechts rollt (vgl. obige Abbildung).

Wenn beispielsweise nacheinander drei Murmeln fallen gelassen werden, wobei die erste und dritte Murmel bei B und die zweite Murmel bei A fallen gelassen wird, dann kommen die ersten beiden Murmeln an der Öffnung Z und die letzte Murmel an der Öffnung X heraus.

Wir wollen das Spiel mithilfe eines DFAs modellieren. Dazu kodieren wir Folgen eingeworfener Murmeln durch Wörter über $\Sigma := \{A, B\}$, wobei beispielsweise das Wort $AABBA$ beschreibt, dass zwei Murmeln in Öffnung A, danach zwei Murmeln in Öffnung B und eine weitere Murmel in Öffnung A eingeworfen wurde. Der DFA soll genau dann akzeptieren, wenn die letzte eingeworfene Murmel aus der Öffnung Z herauskommt.

a) Geben Sie den DFA in grafischer Darstellung an. Ein Zustand sollte sowohl Information über die Positionen der beiden Hebel h_1 und h_2 enthalten als auch angeben, aus welcher Öffnung die vorherige Murmel herausgerollt ist.

b) Welche der folgenden Eingaben werden akzeptiert, welche nicht? (Eine Begründung ist nicht nötig.)

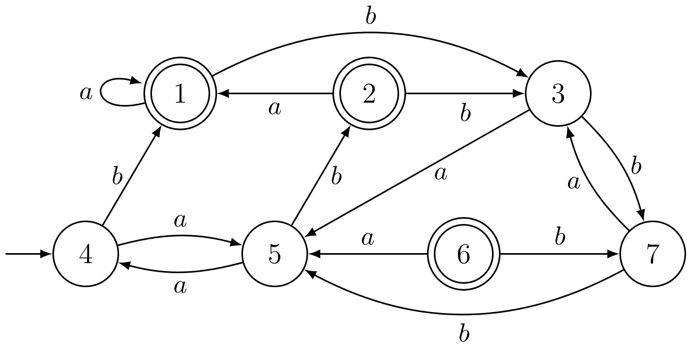
$$BBB \qquad AAABBAAAB \qquad ABABABAB$$

c) Ist es für irgendein $n \in \mathbb{N}_{>0}$ möglich, n Murmeln so einzuwerfen, dass mindestens die (aufgerundete) Hälfte der eingeworfenen Murmeln aus Öffnung X herauskommt?
Ist es für alle $n \in \mathbb{N}_{>0}$ möglich, n Murmeln so einzuwerfen, dass alle aus Öffnung Z herauskommen?
Begründen Sie anhand Ihres DFAs!

d) Ist Ihr DFA minimal? (Eine Begründung ist nicht nötig.)

Aufgabe 12.2 *Minimierung von DFAs* (25 Punkte)

Der DFA $A = (\Sigma, Q, \delta, q_0, F)$ mit $\Sigma = \{a, b\}$, $Q = \{1, 2, 3, 4, 5, 6, 7\}$, $q_0 = 4$ und $F = \{1, 2, 6\}$ sei durch folgende Grafik gegeben:



Minimieren Sie A, d. h. bestimmen Sie den Äquivalenzklassenautomaten A'. Verwenden Sie dazu den Algorithmus aus der Vorlesung und geben Sie auch die Tabelle an, in der Paare inäquivalenter Zustände mit den Mengen M_i markiert sind.

Aufgabe 12.3 *Nerode-Relation und Nerode-Automaten* ((4+4+4) + 8 + 10 = 30 Punkte)

a) Bestimmen Sie für die Sprache

$$F = \Sigma^* \cdot \{aaba\} \cdot \Sigma^* \text{ über dem Alphabet } \Sigma = \{a, b, c\}$$

den Nerode-Automaten in folgenden Schritten:

i) Geben Sie alle Äquivalenzklassen der Nerode-Relation an und beschreiben Sie jede der Klassen durch Angabe aller in ihr enthaltenen Elemente. (Z. B. $[\varepsilon]_F = \{\dots\}$)
ii) Wählen Sie für jede Klasse einen (möglichst kurzen) Vertreter und trennen Sie die Vertreter verschiedener Klassen jeweils durch einen Zeugen.
iii) Geben Sie den Nerode-Automaten in grafischer Darstellung an.

Hinweis: Um alle Äquivalenzklassen zu bestimmen, beginnen Sie mit der Äquivalenzklasse $[\varepsilon]_F$ des leeren Wortes. Was passiert im Nerode-Automaten, wenn nun ein a, b oder c gelesen wird? Sind die Wörter a, b, c äquivalent zu ε? Welche weiteren Äquivalenzklassen treten auf und welche Elemente enthalten sie?

b) Konstruieren Sie (ohne Begründung) den Nerode-Automaten für die Sprache $S = \Sigma^* \cdot \{aaba\}$ über dem Alphabet $\Sigma = \{a, b, c\}$.

c) Geben Sie (ohne Begründung) einen DFA mit möglichst wenigen Zuständen für die Sprache $L = \Sigma^* \cdot \{aaa, ba, bb\}$ über dem Alphabet $\Sigma = \{a, b\}$ an, z. B. den Nerode-Automaten.

Aufgabe 12.4 *Schwierige Sprachen benötigen riesige Automaten!* (20 + 15* Punkte)

Sei $n \in \mathbb{N}$ und $\Sigma = \{a, b\}$. Betrachten Sie die Sprache $L_n := \{xy : x, y \in \Sigma^n, x \neq y\}$.

a) Zeigen Sie: Jeder DFA für L_n benötigt mindestens 2^n Zustände.

Hinweis: Zeigen Sie $\text{Index}(L_n) \geq 2^n$.

b$*$) Konstruieren Sie einen DFA A_n für die Sprache L_n mit möglichst wenigen Zuständen.