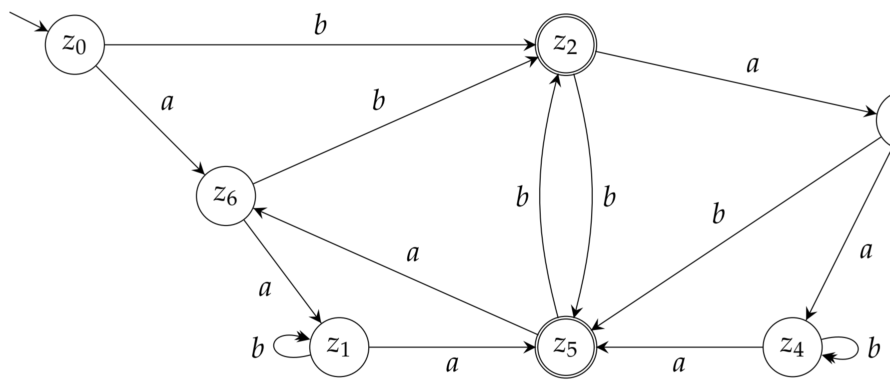
Prof. Dr. Jasmin Blanchette
Jannis Limperg
Lydia Kondylidou

Ludwig-Maximilians-Universität München
Institut für Informatik
Abgabe bis 28. Juni 2023, 10:00 Uhr

Lösungsvorschlag zur Übung 10 zur Vorlesung
Theoretische Informatik für Medieninformatiker

TIMI10-1 ***Automaten minimieren*** (2 Punkte)

Gegeben sei der DFA A:



Berechnen Sie einen Minimalautomaten B mit $L(A) = L(B)$. Geben Sie die Minimierungstabelle und den Zustandsgraph von B an.

LÖSUNGSVORSCHLAG:

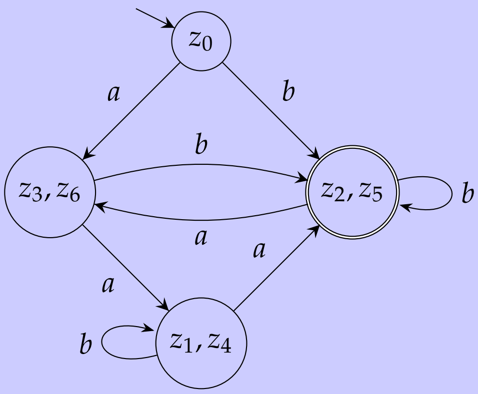
Minimierungstabelle:

	z_0	z_1	z_2	z_3	z_4	z_5	z_6
z_0	—	—	—	—	—	—	—
z_1	$1(z_1, z_2)$	—	—	—	—	—	—
z_2	$*$	$*$	—	—	—	—	—
z_3	$2(z_4, z_6)$	$1(z_5, z_1)$	$*$	—	—	—	—
z_4	$1(z_4, z_2)$		$*$	$1(z_4, z_5)$	—	—	—
z_5	$*$	$*$		$*$	$*$	—	—
z_6	$2(z_1, z_6)$	$1(z_2, z_1)$	$*$		$1(z_2, z_4)$	$*$	—

Für durch $*$ inital markierte Zustandspaare ist jeweils ein Zustand Endzustand und der andere nicht.

Zustandspaare mit Markierung $i(p, q)$ (wobei $i \in \mathbb{N}_{>0}$; $p, q \in Z$; Z ist die Zustandsmenge von A) werden im i-ten Durchlauf ausgefüllt. Dass die Zustände p und q nicht vereinigt werden dürfen, ist der Grund dafür, dass das so markierte Zustandspaar auch nicht vereinigt werden darf.

Aus der Tabelle ergibt sich der Minimalautomat B:



TIMI10-2 *Turingmaschinen*

(0 Punkte)

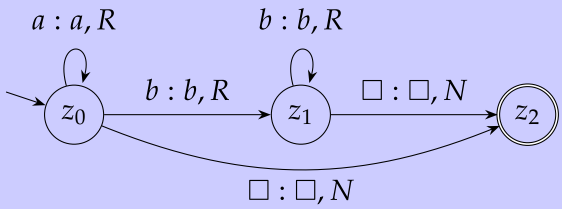
a) Seien $M = \{w \mid \#_a(w) = \#_b(w)\}$ und $N = L(a^*b^*)$ Sprachen über $\Sigma = \{a, b\}$. Zeigen Sie, dass es eine Funktion f gibt mit $\forall x \in \Sigma^* . x \in M \iff f(x) \in N$.

LÖSUNGSVORSCHLAG:

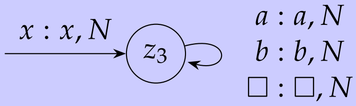
$$f(x) = \begin{cases} ab & \text{wenn} \quad \#_a(x) = \#_b(x) \\ ba & \text{wenn} \quad \#_a(x) \neq \#_b(x) \end{cases}$$

b) Geben Sie eine deterministische Turingmaschine an, welche N erkennt.

LÖSUNGSVORSCHLAG:



Fehlende Übergänge mit $x \in \{a, b, \square\}$:



c) Was muss für f gelten, damit man daraus folgern kann, dass M von einer Turingmaschine erkannt werden kann (ohne die Turingberechenbarkeit von M direkt zu zeigen)?

LÖSUNGSVORSCHLAG: f muss turingberechenbar sein. Sei F die passende Turingmaschine. Dann kann man M erkennen, indem man zunächst F ausführt und danach die Turingmaschine für N ausführt. Wie wir im Laufe der Vorlesung sehen, nennt man f in diesem Fall eine Reduktionsfunktion, für eine Reduktion von M auf N.