Berechnungen und Logik
Hausaufgabenserie

Henri Heyden, Nike Pulow
stu240825, stu239549

A1

Vor.: $A = 0, 1, \overline{\cdot} : A^* \to A^*, ab \mapsto \begin{cases} 0, & ab = 1 \\ 1, & ab = 0 \\ \overline{b} \cdot \overline{a}, & \text{sonst} \end{cases}$

$f : A^* \to A^*, w \mapsto w\overline{w}$
Beh.: f ist nicht sequenziell.
Bew.: Wir werden zeigen, dass f nicht das notwendige Kriterium des linearen Abstandswachstums erfüllt.
Es ist also zu zeigen: $\forall c \in \mathbb{N} : \exists x, y \in A^* : d(f(x), f(y)) > c \cdot d(x, y)$.
Sei $c \in \mathbb{N},\ x := 1^c,\ y := 1^{c-1}0$.
Dann gilt: $c \cdot d(x, y) = c \cdot 2$
Um ein bisschen besser die andere Seite zu vereinfachen, werden wir zwei Hilfsfunktion definieren:

$$\overline{\cdot}' : A^* \to A^*, ab \mapsto \begin{cases} 0, & ab = 1 \\ 1, & ab = 0 \\ \overline{a} \cdot \overline{b}, & sonst \end{cases}$$

$$f' : A^* \to A^*, w \mapsto w \cdot \text{rev}(\overline{x}')$$

Nach den Definitionen von $\overline{\cdot}$ und $\overline{\cdot}'$ sieht man leicht, dass $f = f'$ gilt.
Somit berechnen wir $d(f(x), f(y))$ wie folgt:

$$\begin{aligned}
& d(f(x), f(y)) && \mid \text{Def. } f' \\
=& d\left(1^c \cdot 0^c,\ 1^{c-1} \cdot 0 \cdot \text{rev}\left(\overline{1^{c-1} \cdot 0}'\right)\right) && \mid \text{Def. } \overline{\cdot}' \\
=& d(1^c \cdot 0^c,\ 1^{c-1} \cdot 0 \cdot \text{rev}\left(0^{c-1} \cdot 1\right)) && \mid \text{Def. rev} \\
=& d(1^c \cdot 0^c,\ 1^{c-1} \cdot 0 \cdot 1 \cdot 0^{c-1}) && \mid \text{Rausfiltern von größtem Prefix, Def. } d \\
=& |1 \cdot 0^c| + |0 \cdot 1 \cdot 0^{c-1}| && \mid \text{Def. } |\cdot| \\
=& 2 + |0^c| + |1 \cdot 0^{c-1}| && \mid \text{Def. } |\cdot| \\
=& 2 + 2c
\end{aligned}$$

Dann gilt somit: $c \cdot d(x, y) = 2c < 2c + 2 = d(f(x), f(y))$ – was zu zeigen war. $\square$

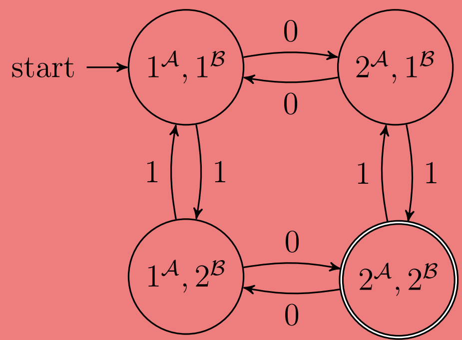
A2

a)

$\mathcal{A}$ akzeptiert Wörter, die eine ungerade Anzahl an 0 haben.
$\mathcal{B}$ akzeptiert Wörter, die eine ungerade Anzahl an 1 haben.
$\mathcal{C}$ akzeptiert Wörter, die eine ungerade Anzahl an 0 haben und eine gerade Anzahl an 1 haben.
Letzteres ergibt sich, da $\mathcal{C}$ aus den Transitionsfunktionen von $\mathcal{A}$ und $\mathcal{B}$ besteht, nur, dass diese auf die Zustände von $\mathcal{C}$ abbilden, welche, wie ihre Namen anspielen, aus $\{1^{\mathcal{A}}, 2^{\mathcal{A}}\} \times \{1^{\mathcal{B}}, 2^{\mathcal{B}}\}$ sind.
Jegliche originellen Transitionsfunktionen haben zwei isomorphe Komplemente in $\mathcal{C}$.

b)

Der komposierte Automat $\mathcal{C}$ wird durch folgendes Bild dargestellt:



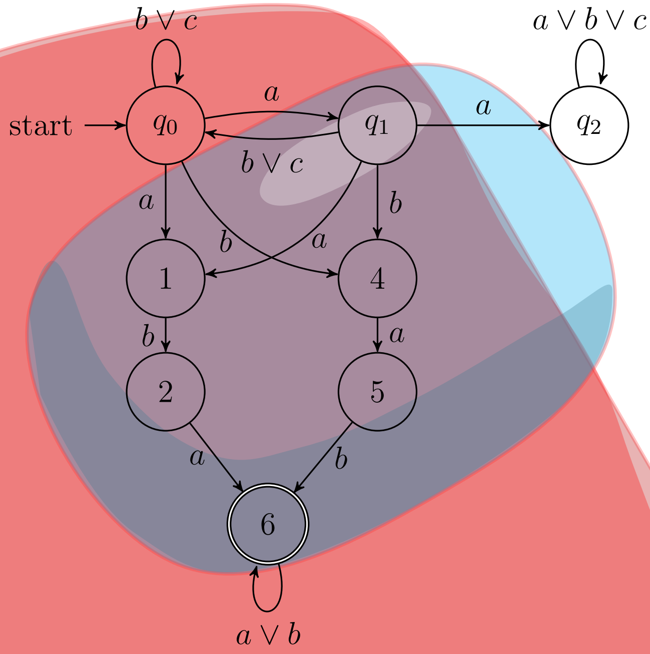
Der Gedanke ist der gleiche, wie in a), nur, dass wir den Endzustand verschieben auf den Zustand, der nur erreicht werden kann, wenn die Anzahl an Nullen und Einsen ungerade ist.

c)

Da $\mathcal{A}$ DEA ist, können wir einfach $Q^{\mathcal{A}} \setminus F^{\mathcal{A}}$ als Menge der Endzustände verwenden, um $\overline{\mathrm{L}(\mathcal{A})}$ zu erhalten, da ein Wort akzeptiert wird nach Definition, wenn der Automat in einem Endzustand endet. Jedes Wort, das keinen Endzustand im Automaten hervorruft, wird somit nicht akzeptiert.
Um genau diese Wörter zu finden, muss man nur die derzeitigen Endzustände falsifizieren und die ehemaligen nicht Endzustände als neue Endzustände setzen – genau dann erreichen wir $\overline{\mathrm{L}(\mathcal{A})}$.

A3

a)



Da wir nur mit 6 enden wollen, sind q_0 und q_1 keine Endzustände mehr, jedoch erfüllen sie beide ihre ehemaligen Aufgaben, sowohl als auch existiert ein Teilisomorphismus zu den Zuständen und ihren Transitionsfunktionen über 0, 3.

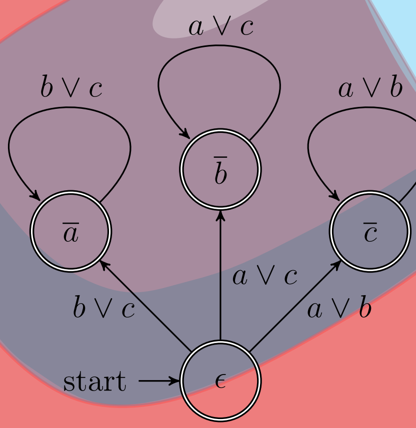
b)

Da wir die kombinierten Wörter aus $L(A_1)$ und $L(A_2)$ suchen, müssen wir alle Endzustände von A_1 eliminieren aber als „Startzustände" für A_2 markieren. Damit ist gemeint, dass alle Transitionsfunktionen und Zustände von A_2 ein isomorphes Komplement in C haben müssen wobei die Endzustände von A_1 auch alle Eigenschaften von den Startzuständen von A_2 besitzen müssen. Wir übertragen also alle Zustände und die Transitionsfunktion von A_2 so, dass die ehemaligen Endzustände von A_1 in die ehemaligen Startzustände von A_2 übergehen und nach der Transition sich genau so verhalten.
Bei vielen Endzuständen von A_1 und Startzuständen von A_2 kann die Darstellung unübersichtlich werden, deswegen empfiehlt sich eine andere Notation, bei der man die Startpfeile von den Startzuständen von A_2 unbeschriftet von den Endzuständen von A_1 schreiben kann verwendet, sodass die Darstellung ähnlicher zu den originellen Automaten aussieht.

A4

a)

Wir definieren $\mathcal{A} := (Q, A, \delta, q_0, F)$ mit $Q := \{\epsilon, \overline{a}, \overline{b}, \overline{c}\}$, $q_0 := \epsilon$, $F := Q$ und der Transitionsfunktion δ dargestellt in folgender Illustration:



Dann gilt $L = L(A)$, da wir Zustände und Übergänge, die die Sprache brechen würde, nicht zulassen. Damit bleiben falsche Eingaben undefiniert, wodurch ein solches Wort nicht akzeptiert werden kann.

b)

Wir nennen $\mathcal{A}$ einen DEA, der die gleiche Sprache wie A (der Automat) erkennt. Es gilt für diesen DEA:

$$\mathcal{A} := \{Q', A, \delta_{\mathcal{A}}, \{\epsilon\}, Q' \setminus \{\emptyset, \{\overline{a}, \overline{b}, \overline{c}\}\}$$

wobei $Q' = \mathcal{P}(Q)$ gilt.

Um die Potenzmengenkonstruktion von A (dem Automaten) zu erreichen, fangen wir mit dem Startzustand $\{\epsilon\}$ an.
Dann akzeptieren wir a, b, c, was jeweils zu den Folgezuständen $\{\overline{c}, \overline{b}\}$, $\{\overline{c}, \overline{a}\}$, $\{\overline{a}, \overline{b}\}$ führt. Für diese Zustände akzeptieren wir jeweils $c \vee b, c \vee a, a \vee b$, was zu den jeweiligen Zuständen $\{\overline{b}\} \vee \{\overline{c}\}, \{\overline{a}\} \vee \{\overline{c}\}, \{\overline{b}\} \vee \{\overline{a}\}$ führt. Die anderen Eingaben, also jeweils a, b, c, führen alle zu dem jeweiligen Zustand selbst, da dies nicht unsere Regeln verletzt oder irgendwas für uns interessantes an dem Wort verändert.
Nun betrachten wir die Zustände $\{\overline{c}\}, \{\overline{a}\}, \{\overline{b}\}$. Für diese müssen die Eingaben c, a und b auf $\emptyset$ abbilden, da nun das Wort unser Kriterium verletzt, die pro Zustand zwei anderen Eingaben bilden, wie im NEA auf sich selbst ab. Den Zustand $\{\overline{a}, \overline{b}, \overline{c}\}$ und die Zustände $\{m \in \mathcal{P}(Q) | \epsilon \in m \wedge |m| > 1\}$ ignorieren wir, da es keine Möglichkeit gibt, sie zu erreichen. $\emptyset$ bildet alle Eingaben auf sich selbst, da wir alle Wörter ignorieren, die diesen Zustand erreichen. Hieraus folgt, dass wir alle Zustände außer $\emptyset$ und $\{\overline{a}, \overline{b}, \overline{c}\}$ als Endzustände zulassen.
Es ergibt sich folgende Darstellung (wir ignorieren außer bei $\emptyset$ die äußeren Klammern):