

Theoretische Informatik - Übungsblatt 8
Übungsgruppe 7 - Mi, 16-18 Uhr
Elena Kvassova

Jan Hendrik Dithmar (Matrikelnummer 2031259)
Dirk Heine (Matrikelnummer 2030941)

Aufgabe 1

- (a) Wahr. Eine endliche Sprache ist eine endlich große Menge von Worten, die die Sprache akzeptiert. Jede solche Sprache läßt sich trivial mit einem regulären Ausdruck beschreiben, indem wir die Vereinigung $+$ über alle Wörter der Sprache bilden. Nach Satz 5.3.3 ist jede Sprache, die mit einem regulären Ausdruck beschrieben werden kann, regulär. $\square$
- (b) Falsch. Als Gegenbeispiel nehmen wir die Sprache $\mathbb{B}^*$. Die Sprache $\{w \in \mathbb{B}^* \mid |w|_0 = |w|_1\}$ auf Aufgabenteil (d) ist (wie dort gezeigt) nicht regulär, ist jedoch offensichtlich eine Teilmenge der regulären Sprache $\mathbb{B}^*$. $\nmid$
- (c) Wahr. Sei L also eine reguläre Sprache und somit existiert ein endlicher Automat A , der die Sprache L akzeptiert. Dieser Automat A kann problemlos in einen Automaten A^{-1} überführt werden, der alle Worte akzeptiert, die *nicht* in der Sprache L enthalten sind (indem alle Endzustände zu normalen Zuständen und alle normalen Zustände zu Endzuständen werden).

Wir bauen somit einen (nicht-deterministischen) endlichen Automaten A' , aus A und A^{-1} , der die geforderte Sprache akzeptiert. Das Eingabewort uv wird als Eingabe in A gegeben (der Startzustand des Automaten A ist der Startzustand des Automaten A'). Jedesmal, wenn A einen Endzustand erreicht (auch wenn noch nicht das Ende der Eingabe erreicht wurde), gibt es zwei mögliche Rechenwege:

- die Abarbeitung des Automaten A wird normal fortgesetzt oder
- der Rest der Eingabe wird als Eingabe für den Automaten A^{-1} verwendet.

Erreicht in einem Rechenweg der Automat A^{-1} einen seiner Endzustände, dann ist das Wort in der Sprache enthalten (die Endzustände des Automaten A^{-1} sind die Endzustände des Automaten A'). Beweis durch Konstruktion erbracht. $\square$

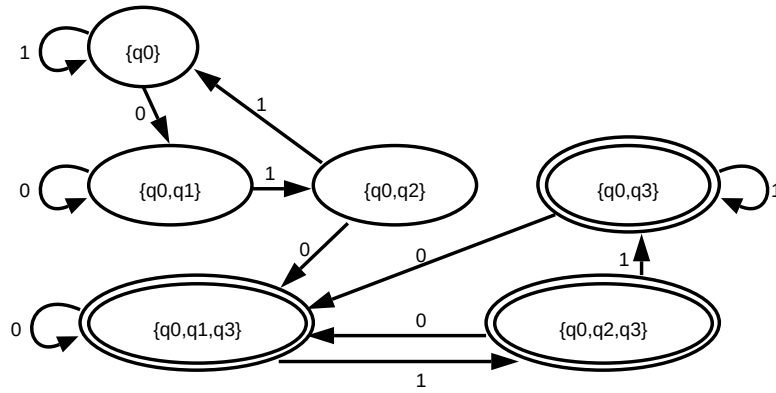
- (d) Falsch. Angenommen die Sprache sei regulär und es existiere somit ein endlicher Automat, der die Sprache akzeptiert. Habe dieser Automat ohne Beschränkung der Allgemeinheit k Zustände. Für eine Eingabe, die z.B. mit $k + 1$ -mal dem Zeichen „0“ beginnt, kann nicht geprüft werden, ob das Wort mindestens noch $k + 1$ -mal das Zeichen „1“ enthält, da der Automat außer der Zuständen kein „Gedächtnis“ hat. $\nmid$

Aufgabe 2

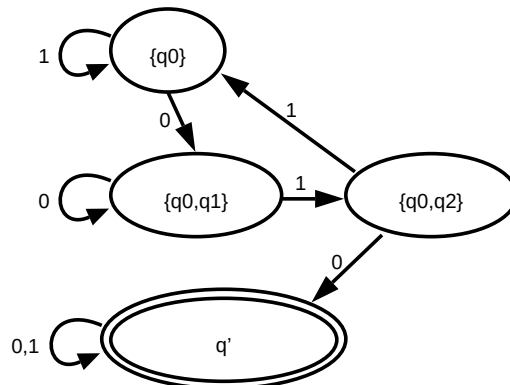
- (a) Dieser Automat akzeptiert die Sprache, die durch den regulären Ausdruck $(0+1)^* 010 (0+1)^*$ beschrieben wird. Dies ist die Sprache aller Wörter aus Σ^* , die an beliebiger Stelle die Zeichenkette 010 enthält.
- (b) Konstruieren DFA für die Sprache L mittels Potenzautomaten-Konstruktion:

	0	1
$\{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_1\}$	$\emptyset$	$\{q_2\}$
$\{q_2\}$	$\{q_3\}$	$\emptyset$
$\{q_3\}$	$\{q_3\}$	$\{q_3\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$
$\{q_0, q_2\}$	$\{q_0, q_1, q_3\}$	$\{q_0\}$
$\{q_0, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_0, q_3\}$
$\{q_0, q_1, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_0, q_2, q_3\}$
$\{q_0, q_2, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_0, q_3\}$

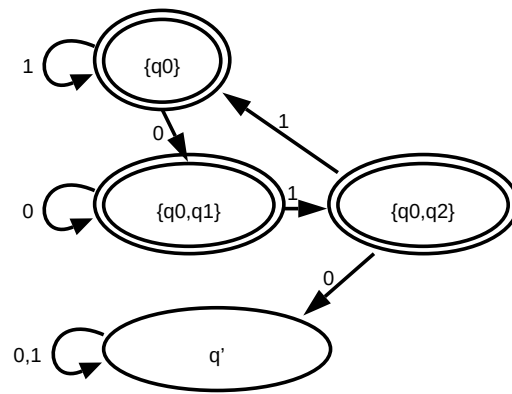
Die Zustandsmengen $\{q_1\}$, $\{q_2\}$ und $\{q_3\}$ werden nie erreicht und können somit weggelassen werden. Alle Zustandsmengen, die den alten Endzustand q_3 enthalten, werden zu akzeptierenden Endzuständen im konstruierten DFA. $\{q_0\}$ wird zum Anfangszustand.



- (c) Offensichtlich sind die Zustandsmengen $\{q_0, q_3\}$, $\{q_0, q_1, q_3\}$ und $\{q_0, q_2, q_3\}$ äquivalent. Der Äquivalenzklassenautomat für den obigen DFA ist somit:



- (d) Der DFA für $\bar{L}$ ist offensichtlich:



Aufgabe 3

(a) Konstruktionsvorschrift für den NFA A^R durch Modifikation des NFA A :

- Wir invertieren die Richtungen aller Kanten aus A .
- Sei q_0 der Startzustand des NFA A . Die Menge $\{q_0\}$ ist die Menge der akzeptierenden Endzustände von A^R .
- Wir führen einen neuen Startzustand q'_0 für A^R ein, von dem aus zu jedem Zustand aus F eine neue ε -Kante führt, wobei F die Menge der Endzustände von A ist.

(b) Beweis durch Gegenbeispiel. Alle Wörter der Sprache $a^n b b a^n$ sind offensichtlich Palindrome. Es gibt jedoch keinen regulären Ausdruck, der diese Sprache darstellen kann. $\square$

(c) Habe ohne Beschränkung der Allgemeinheit der NFA A genau einen akzeptierenden Endzustand. Anderenfalls kann dies trivialerweise durch Hinzufügung eines neuen Zustandes (welcher dann der einzige akzeptierende Endzustand wird) und eine Reihe von ε -Kanten erreicht werden.

Der zu diesem NFA A gehörende NFA A^R , der die selben Worte jedoch gespiegelt akzeptiert, ergibt sich dann trivialerweise durch Spiegelung aller Kanten und dem Austausch von Start- und Endzustand.

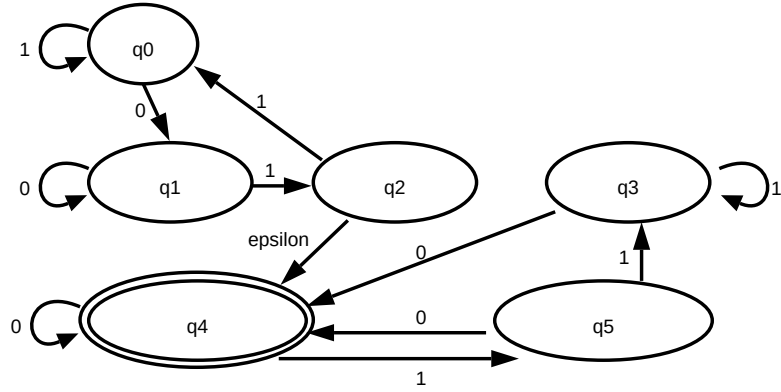
Seien ohne Beschränkung der Allgemeinheit q_i die Zustände des NFA A (und somit auch des NFA A^R). Sei ferner q_0 der Startzustand von A (und somit der Endzustand von A^R) und q_1 der Startzustand von A^R (und somit der Endzustand von A).

Der Automat $A \times A^R$ führt praktisch beide Automaten gleichzeitig aus. Alle Zustände haben die Form (q_i, q_j) . Befindet sich der Automat $A \times A^R$ am Ende des Wortes in einem Zustand (q_i, q_j) mit $i = j$, dann ist das Wort in der Sprache $L_{\text{PAL}/2}$. Der Startzustand des Automaten ist (q_0, q_1) und die Menge der Endzustände ist $\cup_i \{(q_i, q_i)\}$.

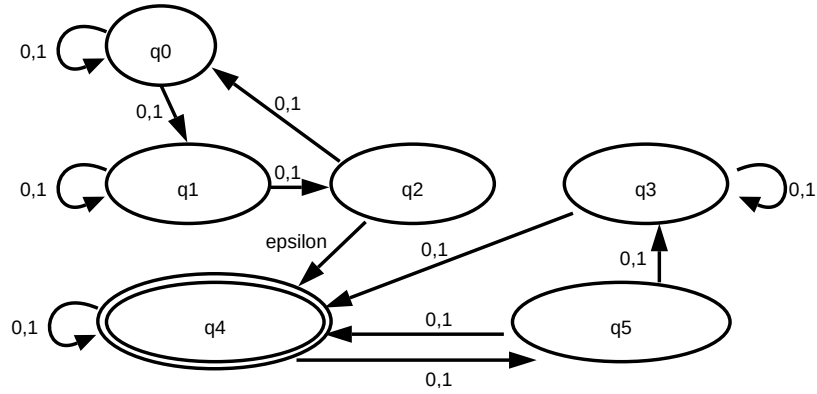
Die Worte der Form ww^R sind naturgemäß immer Palindrome. Indem wir A^R von vorne beginnend auf w ausführen, führen wir praktisch A von hinten beginnend auf w aus. Mit dem Automaten $A \times A^R$ führen wir somit gleichzeitig A von vorne *und* von hinten auf w aus. Wenn wir uns bei der Ausführung „in der Mitte“, also im „selben“ (das heißt entsprechenden oder äquivalenten) Zustand, treffen, dann werden beide Richtungen auf *jeden* Fall ihren jeweiligen Endzustand erreichen. Somit ist $ww^R \in L$ und damit $w \in L$.

(d) **Konstruktion von A_1**

Für die Konstruktion von A_1 werden in A alle Kanten, die *keine* ε -Kanten sind, mit *allen* Zeichen des Alphabets Σ beschriftet. Es ist also an der jeweiligen Stelle egal, welches Zeichen gelesen wird. Dies ist auch korrekt, da uns nur die Übereinstimmung der Länge des Wortes interessiert. Der Anfangszustand und die Endzustände bleiben unverändert. Zum Beispiel wird aus dem Automaten



über dem Alphabet $\Sigma = \mathbb{B}$ der Automat A_1



Konstruktion von A_k

Für die Konstruktion von A_k hängen wir einfach k -mal den Automaten A_1 hintereinander. Die Automaten werden in einer Reihe verbunden, indem von allen Endzuständen einer A_1 -Instanz ε -Kanten auf den Startzustand der nachfolgenden A_1 -Instanz hinzugefügt werden. Der Startzustand der ersten A_1 -Instanz ist der Startzustand von A_k und die Endzustände der letzten A_1 -Instanz sind die Endzustände von A_k (weitere Endzustände gibt es nicht). Die Verkettung hat also für zum Beispiel A_3 die Form

