Prof. Javier Esparza
Philipp Czerner, Martin Helfrich

Technische Universität München
Lehrstuhl für Theoretische Informatik

# Einführung in die Theoretische Informatik

Sommersemester 2021 – Übungsblatt 2

- Das Übungsblatt ist in zwei Teile gegliedert: den Vorbereitungsteil, den Sie vor der Übung selbstständig bearbeiten sollen, und den Übungs-/Nachbereitungsteil, der Aufgaben enthält, die in der Übung besprochen werden und von Ihnen anschließend zur Nachbereitung verwendet werden können.
- Das ist nicht das Hausaufgabenblatt! Die Hausaufgaben finden Sie auf einem separaten Blatt.

## Vorbereitung (→ vor der Übung selbständig zu bearbeiten)

**Individualaufgabe Ü2.1.** (*Wichtige Begriffe*)

Überprüfen Sie, dass Sie die folgenden Begriffe oder Notationen korrekt definieren können.

- Regulärer Ausdruck
- $\epsilon$-NFA
- Rechtslineare Grammatik
- Potenzmengenkonstruktion
- Produktkonstruktion
- Strukturelle Induktion

**Individualaufgabe Ü2.2.** (*Automata Tutor: Reguläre Ausdrücke*)

Lösen Sie die Aufgaben Ü2.2 (a–g) auf Automata Tutor.[1] Beachten Sie, dass wir für einen regulären Ausdruck $r$ das folgende Makro definieren: $r^+ = rr^*$.
**Achtung:** Für die Übungsaufgaben haben Sie beliebig viele Versuche. Für die Aufgaben in Hausaufgabe H2.1 nicht!

**Individualaufgabe Ü2.3.** (*Konstruieren von NFAs mit Einschränkungen*)

Entscheiden Sie, ob die folgenden Aussagen korrekt sind, und begründen Sie Ihre Behauptung, indem Sie entweder ein Gegenbeispiel oder eine passende Konstruktion angeben.

Für jeden NFA $N = (Q, \Sigma, \delta, q_0, F)$ gibt es einen NFA $N' = (Q', \Sigma, \delta', q_0', F')$ mit $L(N) = L(N')$ und ...

(a) der Startzustand hat keine eingehenden Kanten.

(b) kein Endzustand hat eine ausgehende Kante.

(c) für jeden Zustand $q$ gilt: alle eingehenden Kanten von $q$ sind mit demselben Zeichen beschriftet.

(d) für jeden Zustand $q$ gilt: alle ausgehenden Kanten von $q$ sind mit demselben Zeichen beschriftet.

Zu dieser Aufgabe gibt es Video-Lösungen: (a) (b) (c) (d)

[1] Wenn Sie Automata Tutor noch nicht verwendet haben, folgen Sie erst den Schritten in Ü1.2, um sich richtig zu registrieren.

*Lösungsskizze.*

(a) Ja: Man erstellt vom Startzustand $q_0$ eine Kopie $q_0'$, die nur die ausgehenden Kanten von $q_0$ erbt. Danach wird $q_0$ zu einem normalen Zustand, während $q_0'$ zu einem Startzustand wird. Formal: $N' = (Q', \Sigma, \delta', q_0', F')$, wobei

- $Q' = Q \cup \{q_0'\}$,
- $\delta' = \delta \cup \{(q_0', x, q) : (q_0, x, q) \in \delta\}$
- $F' = F \cup \{q_0' : q_0 \in F\}$

(b) Nein: Betrachte einen NFA $N$ für $L = \{\varepsilon, a\}$.

(c) Ja: man spaltet jeden Zustand $q \in Q$ nach dem Buchstaben der eingehenden Kanten auf, d.h. aus $q$ macht man die Zustände $(q, x)$ für jedes $x \in \Sigma$.

Danach setzt man $\delta'((q, x), y, (q', y)) := \delta(q, y, q')$.

Wird dabei der Startzustand aufgespalten, wählt man einen beliebigen daraus hervorgehenden Zustand als neuen Startzustand.

(d) Nein: Betrachte einen NFA $N$ mit $L(N) = \{a, b\} = \Sigma$. Dann gilt $\delta(q_0, a) \cap F \neq \emptyset$ und $\delta(q_0, b) \cap F \neq \emptyset$.

**Individualaufgabe Ü2.4.** (*Potenzmengenkonstruktion*)

Mit $|w|_x$ bezeichnen wir die Anzahl der Vorkommen des Buchstabens $x \in \Sigma$ in $w \in \Sigma^*$. Sei $\Sigma = \{a, b, c\}$ und $L = \{w \in \Sigma^* \mid \exists\, x \in \Sigma : |w|_x = 0\}$.

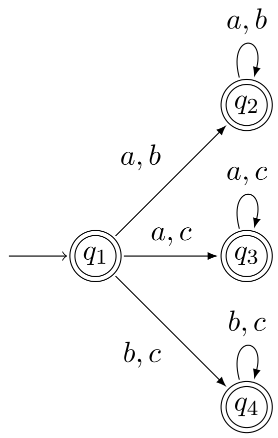
(a) Konstruieren Sie einen NFA $N$ mit genau 4 Zuständen und $L(N) = L$.

(b) Determinisieren Sie den NFA $N$ aus (a) mittels der Potenzmengenkonstruktion, um einen DFA $M$ mit $L(M) = L(N)$ zu erhalten.

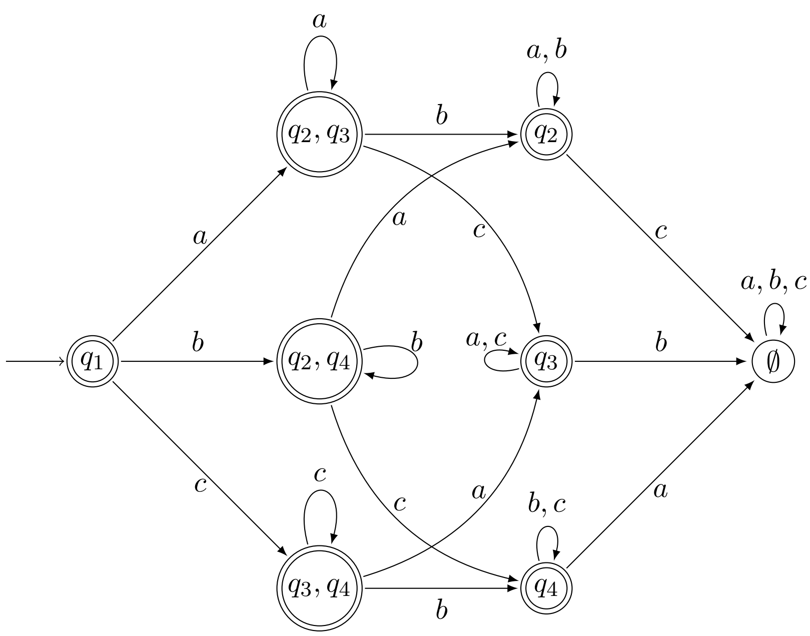
Zu dieser Aufgabe gibt es Video-Lösungen: (a) (b)

*Lösungsskizze.*

(a) Der folgende NFA rät im initialen Zustand welche beiden Buchstaben im Wort auftauchen:




(b) Anwendung der Potenzmengenkonstruktion:




## Übung und Nachbereitung

**Fokusaufgabe Ü2.5.** (*RA* $\to$ $\epsilon$-*NFA* $\to$ *NFA* $\to$ *DFA*)

Wir betrachten den regulären Ausdruck $r = 1(0\,|\,1)^*\,|\,0$.

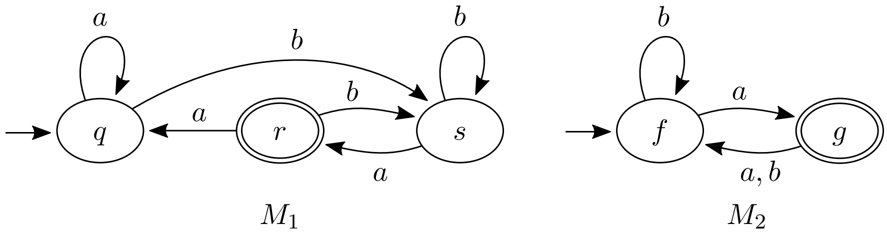
(a) Konstruieren Sie für $r$ mit dem Standardverfahren aus der Vorlesung einen $\epsilon$-NFA $A$, so dass $L(r) = L(A)$ gilt.

(b) Wandeln Sie diesen Automaten in einen äquivalenten NFA ohne $\epsilon$-Übergänge um.

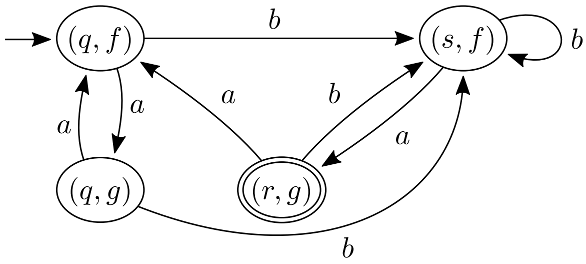
(c) Konstruieren Sie durch Anwendung des Potenzmengenverfahrens einen DFA, der die Sprache des Ausdrucks $r$ akzeptiert.

**Übungsaufgabe Ü2.6.** (*Produktkonstruktion*)

Konstruieren Sie einen DFA $M$ mit $L(M) = L(M_1) \cap L(M_2)$, indem Sie die Produktkonstruktion verwenden. Ist $M$ minimal?




*Lösungsskizze.*




Der DFA $M$ ist nicht minimal. Er akzeptiert die gleiche Sprache wie $M_1$.

**Übungsaufgabe Ü2.7.** (*Teilwörter und reguläre Sprachen*)

Ein Teilwort eines Wortes $w$ ist ein zusammenhängendes Wort in $w$. Wir definieren die Menge aller Teilwörter von $w$ als $\downarrow[w] := \{w' \in \Sigma^* : w \text{ enthält } w'\}$. Also gilt beispielsweise $\mathit{the}, \mathit{he}, \mathit{theo}, \varepsilon \in \downarrow[\mathit{theo}]$, aber $\mathit{to}, \mathit{theoo} \notin \downarrow[\mathit{theo}]$. Die Menge aller Wörter, von denen $w$ ein Teilwort ist, ist dann entsprechend definiert als $\uparrow[w] := \{w' \in \Sigma^* : w' \text{ enthält } w\}$; z.B. $\mathit{thetheotee} \in \uparrow[\mathit{theo}]$. Wir erweitern dies auf Sprachen: Für eine beliebige Sprache $L \subseteq \Sigma^*$, definieren wir also $\downarrow[L] := \bigcup_{w\in L} \downarrow[w]$, und $\uparrow[L] := \bigcup_{w\in L} \uparrow[w]$.

Sei nun $\Sigma := \{t, h, e, o\}$ und $L$ eine beliebige reguläre Sprache.

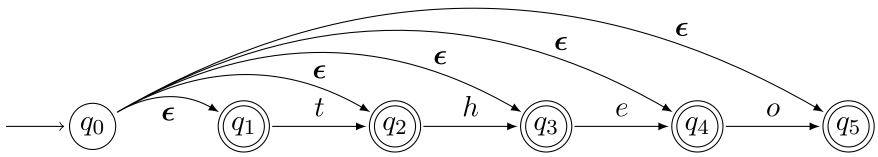
(a) Geben Sie für $\downarrow[\mathit{theo}]$ und $\uparrow[\mathit{theo}]$ einen $\epsilon$-NFA an.

(b) Zeigen Sie, dass $\downarrow[L]$ regulär ist.

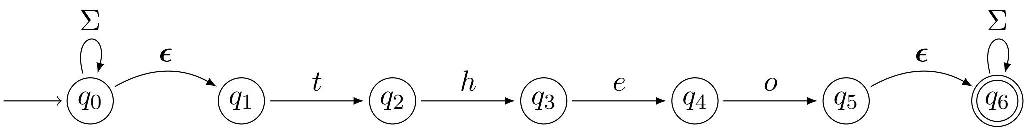
(c) Zeigen Sie, dass $\uparrow[L]$ regulär ist.

*Lösungsskizze.*

(a) $\epsilon$-NFA für $\downarrow[\mathit{theo}]$:




$\epsilon$-NFA für $\uparrow[\mathit{theo}]$:




(b) Wenn $L$ regulär ist, gibt es einen NFA $N = (Q, \Sigma, \delta, q_0, F)$ der L akzeptiert und bei dem jeder Zustand von $q_0$ erreichbar ist. Wir zeigen, dass es einen $\epsilon$-NFA $N'$ gibt, der $\downarrow[L]$ akzeptiert, indem wir $N$ modifizieren.

**Idee:** Da wir alle Teilwörter akzeptieren wollen, muss es in $N'$ möglich sein, ein Präfix zu überspringen. Dafür führen wir einen neuen Startzustand ein, der jeden anderen Zustand mit einem $\epsilon$-Übergang erreichen kann. Da auch ein beliebiges Suffix gelöscht werden darf, wird jeder Zustand in $N$, der einen Endzustand erreichen kann, akzeptierend.

Wir definieren also $\epsilon$-NFA $N' = (Q \cup \{q_0'\}, \Sigma, \delta', q_0', F')$. Dabei ist

$$\delta' = \delta \uplus \{(q_0', \boldsymbol{\epsilon}, q) : q \in Q\}$$
$$F' = \{q \in Q : \exists w \in \Sigma^* : \hat{\hat{\delta}}(q, w) \cap F \neq \emptyset\}$$

Wir müssen zeigen: $L(N') = \downarrow[L]$.

- $\subseteq$: Sei $w \in L(N')$. Dann gibt es einen akzeptierenden Lauf $q_0'q_1q_2...q_k$ in $N'$. Da jeder Zustand in $N$ und somit auch in $N'$ erreichbar ist, gibt es ein Wort $u \in \Sigma^*$, so dass $q_1 \in \hat{\delta}(q_0, u)$. Per Definition von $F'$ gibt es auch ein Wort $v \in \Sigma^*$, so dass $\hat{\delta}(q_k, v) \cap F \neq \emptyset$. Somit gilt, $uwv \in L(N)$. Aus $w \in \downarrow[uwv]$ folgt dann $w \in \downarrow[L]$.
- $\supseteq$: Sei $w \in \downarrow[L]$. Per Definition wissen wir, dass es $u, v \in \Sigma^*$ gibt, sodass $uwv \in L$. Betrachte einen akzeptierenden Lauf für $uwv$ in $N$. Sei $q_1...q_k$ der Teil des Laufes, bei dem $w$ eingelesen wird. Da in $N$ von $q_k$ mit $v$ akzeptiert werden kann, ist $q_k$ akzeptierend in $N'$. Dann ist $q_0'q_1...q_k$ ein akzeptierender Lauf in $N'$ und somit $w \in L(N')$.

(c) Wenn $L$ regulär ist, gibt es einen NFA $N = (Q, \Sigma, \delta, q_0, F)$, der L akzeptiert. Wir zeigen, dass es einen $\epsilon$-NFA $N'$ gibt, der $\uparrow[L]$ akzeptiert, indem wir $N$ modifizieren.

**Idee:** Da wir alle Wörter akzeptieren, die ein Wort aus $L$ als Teilwort enthalten, muss es in $N'$ möglich sein ein beliebiges Präfix, dann ein beliebiges Wort aus $L$, und dann einen beliebiges Suffix zu lesen. Also fügen wir einen neuen Startzustand ein, bei dem beliebige Zeichen gelesen werden kann. Dieser erreicht den alten Startzustand mit einem $\epsilon$-Übergang. Statt zu akzeptieren, haben alle in $N$ akzeptierenden Zustand einen $\epsilon$-Übergang zum neuen Endzustand. In diesem wird jedes Wort akzeptiert.

Wir definieren also $\epsilon$-NFA $N' = (Q \cup \{q_0', q_f\}, \Sigma, \delta', q_0', \{q_f\})$. Dabei ist

$$\begin{aligned}\delta' = \delta &\uplus \{(q_0', x, q_0') \mid x \in \Sigma\} \uplus \{(q_0', \epsilon, q_0)\} \\ &\uplus \{(q, \epsilon, q_f) \mid q \in F\} \quad \uplus \{(q_f, x, q_f) \mid x \in \Sigma\}\end{aligned}$$

Wir müssen zeigen: $L(N') = \uparrow[L]$.

- $\subseteq$: Sei $w \in L(N')$. Dann gibt es einen akzeptierenden Lauf für $w$ in $N'$. Nach Konstruktion lässt sich der Lauf in drei Teile unterteilen: Im ersten Teil befindet sich $N$ nur in Zustand $q_0'$, bis er mit einem $\epsilon$-Übergang nach $q_0$ übergeht. Im zweiten Teil befindet sich $N'$ nur in Zuständen von $Q$, bis er mit einem $\epsilon$-Übergang von einem Zustand $q \in F$ nach $q_f$ übergeht. Im letzten Teil befindet sich $N'$ nur in $q_f$. Wir nennen die drei Teilworte von $w$, die in den einzelnen Abschnitten eingelesen wurden, $u, w'$ und $v$. Da es möglich ist von $q_0$ beim Einlesen von $w'$ einen akzeptierenden Zustand $q$ zu erreichen, gilt $w' \in L(N) = L$. Da $w = uw'v \in \uparrow[w']$, gilt somit $w \in \uparrow[L]$.
- $\supseteq$: Sei $w \in \uparrow[N]$, es gibt also $u, w', v \in \Sigma^*$ mit $w = uw'v$ und $w' \in L$. Wegen $L(N) = L$ gibt es einen akzeptierenden Lauf für $w'$ in $N$. Somit gibt es einen akzeptierenden Lauf für $uw'v$ in $N'$, bei dem erst $u$ in $q_0'$ eingelesen wird, dann folgt der Lauf für $w'$ aus $N$ und danach wird $v$ in $q_f$ eingelesen. Somit gilt $w \in L(N')$.