Prof. Javier Esparza
Kevin Kappelmann, Simon Roßkopf, Lukas Stevens

Technische Universität München
Lehrstuhl für Theoretische Informatik

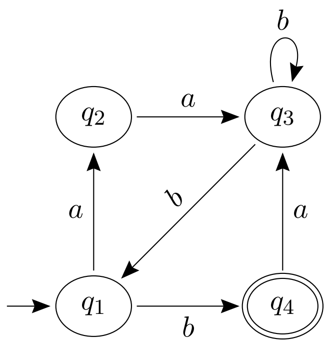
Einführung in die Theoretische Informatik

Sommersemester 2023 – Hausaufgabenblatt 3

- Bitte beachten Sie, dass in dieser Vorlesung generell Antworten mit Begründung gefordert werden, solange die Aufgabe nicht explizit das Gegenteil sagt.
- Zum Bestehen diese Blattes müssen Sie 50% der Punkte erreichen.

Aufgabe H3.1. (*Arden*) 3 + 1 Punkte

Gegeben sei folgender Automat $M = (\{q_1, q_2, q_3, q_4\}, \{a, b\}, \delta, q_1, \{q_4\})$:



(a) Berechnen Sie einen regulären Ausdruck α mit $L(\alpha) = L(M)$. Verwenden Sie dafür das aus der Vorlesung bekannte Verfahren zum Lösen eines Gleichungssystems mit Hilfe von Ardens Lemma.

(b) Folgt aus der Gleichung $X \equiv (a \mid \epsilon)X \mid a$, dass $X \equiv (a \mid \epsilon)^* a$? Beweisen Sie die Implikation oder geben Sie ein Gegenbeispiel an (mit kurzer Begründung).

Lösungsskizze.

(a) Gleichungssystem:

$$X_1 \equiv aX_2 \mid bX_4 \tag{1}$$
$$X_2 \equiv aX_3 \tag{2}$$
$$X_3 \equiv bX_3 \mid bX_1 \tag{3}$$
$$X_4 \equiv aX_3 \mid \boldsymbol{\epsilon} \tag{4}$$

Gleichung (2) in (1) einsetzen:

$$X_1 \equiv aaX_3 \mid bX_4 \tag{5}$$

Gleichung (4) in (5) einsetzen:

$$X_1 \equiv aaX_3 \mid b(aX_3 \mid \boldsymbol{\epsilon}) \equiv (aa \mid ba)X_3 \mid b \tag{6}$$

Gleichung (6) in (3) einsetzen und auflösen:

$$X_3 \equiv bX_3 \mid b((aa \mid ba)X_3 \mid b) \tag{7}$$
$$\equiv (b \mid baa \mid bba)X_3 \mid bb \ \equiv \ (b \mid baa \mid bba)^* bb \tag{8}$$

Einsetzen in (6):

$$X_1 \equiv \alpha \equiv (aa \mid ba)(b \mid baa \mid bba)^* bb \mid b \tag{9}$$

(b) Gegenbeispiel: $X \equiv a^*$ ist eine Lösung von $X \equiv (a \mid \epsilon)X \mid a$ aber $a^* \not\equiv (a \mid \epsilon)^* a \equiv a^+$

Aufgabe H3.2. (*THEO-TV*) 3 + 3 + 3 Punkte

Dr. Evilsparza ist nach seinem letzten Gefecht ziemlich ausgelaugt und binge-watched auf seinem Fernsehautomaten. Beim Zappen auf THEO-TV läuft gerade die Wiederholung der Tutorübung 3.6, welche seine Aufmerksamkeit erregt. Diese Funktionen $h : \Sigma \to \Delta^*$ sind ja ganz nett, denkt er sich. Mit ein paar Kniffen könnte man doch damit möglichst viel Unheil anrichten...

Seien Σ und Δ zwei Alphabete, L eine Sprache über Δ und $h : \Sigma \to \Delta^*$ eine Funktion. Dr. Evilsparza definiert nun die schauderbare Sprache aller Wörter, deren Bild unter h^* in L liegen. Formell definiert er $h^{-1}(L) := \{w \mid h^*(w) \in L\}$. Beachte, dass $h^{-1}(L) \subseteq \Sigma^*$.

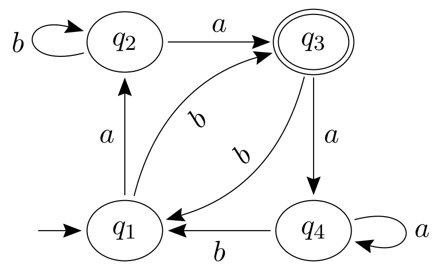
Dr. Evilsparza ist sich sicher, dass er nun endlich die Grenzen der regulären Sprachen verlassen hat und sein Unheil von dannen ziehen kann. Doch nicht so schnell – rettet erneut die Welt vor der böswilligen Konstruktion!

(a) Zeigen Sie: wenn L regulär ist, dann ist auch $h^{-1}(L)$ regulär. Da Dr. Evilsparza großer Automatenfan ist und ihr ihn beeindrucken möchtet, entwerft hierfür für einen gegebenen DFA M einen neuen DFA M' mit $L(M') = h^{-1}(L(M))$. Erläutern Sie ihre Konstruktionsidee auch kurz informell.

(b) Sei $\Sigma = \{T, H, \heartsuit, E, O\}$, $\Delta = \{a, b\}$ und $h : \Sigma \to \Delta^*$ definiert durch

$$h(T) := ab \qquad h(H) := \epsilon \qquad h(\heartsuit) := abba \qquad h(E) := ba \qquad h(O) := ab$$

Wenden Sie ihre Konstruktion aus Aufgabe (a) auf $h(\cdot)$ und den folgenden DFA an:



Hinweis: Ihr Automat sollte beispielsweise die Wörter

$$\heartsuit, \qquad THEO\heartsuit, \qquad THEOTHEO\heartsuit, \qquad ET\heartsuit H$$

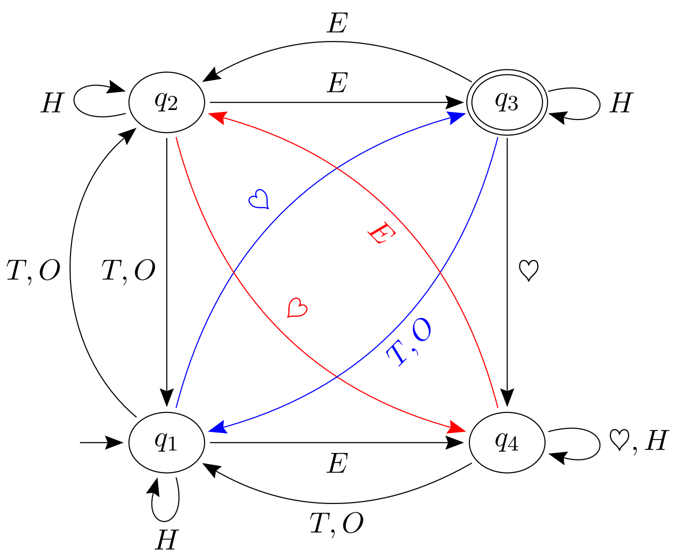
akzeptieren und die Wörter *HOT* und *OTTO* ablehnen.

(c) Beweisen Sie, dass Ihre Konstruktion aus Aufgabe (a) korrekt ist.

Lösungsskizze.

(a) Sei $M = (Q, \Delta, \delta, q_0, F)$ ein DFA. Wir konstruieren einen DFA M' mit $L(M') = h^{-1}(L(M))$. Die Konstruktionsidee liegt darin, bei Eingabe w den Automaten M auf $h^*(w)$ laufen zu lassen. Formal definieren wir $M' := (Q, \Sigma, \delta', q_0, F)$ mit $\delta'(q, a) := \hat{\delta}(q, h(a))$.

(b)



(c) Wir schreiben $q \rightarrow_M^w q'$ für $\hat{\delta}(q, w) = q'$ und $q \rightarrow_{M'}^w q'$ für $\hat{\delta}'(q, w) = q'$. Wir müssen zeigen, dass $L(M') = h^{-1}(L(M))$. Sei nun $w = a_1 \cdots a_n \in \Sigma^*$. Dann gilt

$$\begin{aligned}
w \in L(M') &\iff q_0 \rightarrow_{M'}^{w} q_n \in F \\
&\iff q_0 \rightarrow_{M'}^{a_1} q_1 \cdots \rightarrow_{M'}^{a_n} q_n \in F \\
&\iff q_0 \rightarrow_{M}^{h(a_1)} q_1 \cdots \rightarrow_{M}^{h(a_n)} q_n \in F \\
&\iff q_0 \rightarrow_{M}^{h^*(w)} q_n \in F \\
&\iff h^*(w) \in L(M).
\end{aligned}$$

Aufgabe H3.3. (*Prima, große Produkte*) 1+2+1+1+1 Punkte

Die Produktkonstruktion, wie in der Vorlesung definiert, erzeugt für zwei DFAs mit n und m Zuständen einen DFA mit nm Zuständen. Sie haben (zum Beispiel in der Übung) Fälle gesehen, in denen es möglich ist, nur eine Teilmenge dieser Zustäde zu erzeugen. Im folgenden soll gezeigt werden, dass es jedoch notwendig sein kann, die komplette Zustandsmenge zu erzeugen. Hierfür betrachten wir DFAs für die Sprachen $L_n := \{a^{nm} | m \in \mathbb{N}\}$ für $n \in \mathbb{N}_+$ über dem Alphabet $\Sigma = \{a\}$, d.h. ein Wort ist in L_n, wenn seine Länge ein Vielfaches von n ist.

(a) Geben Sie DFAs für L_2 und L_3 an

(b) Geben Sie eine Konstruktion für einen DFA $A_n = (Q_n, \Sigma, \delta_n, q_{0n}, F_n)$ mit $L(A_n) = L_n$ und $|Q_n| = n$ an. Warum kann es keinen DFA mit weniger Zuständen für L_n geben?

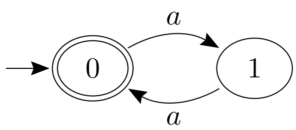
(c) Berechnen Sie einen Automaten für $L_2 \cap L_3$, indem Sie eine Produktkonstruktion durchführen.

(d) Geben Sie eine unendliche Menge von Paaren (n, m) an, sodass der Produktautomat von A_n und A_m genau mn Zustände hat.

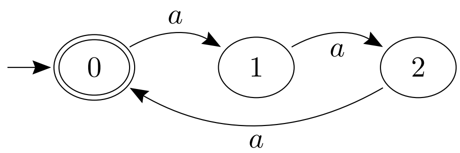
(e) Begründen Sie ihre Antwort für (d), indem Sie beweisen, das der resultierende Produktautomat genau nm Zustände hat.

Lösungsskizze.

(a) L_2:



L_3:



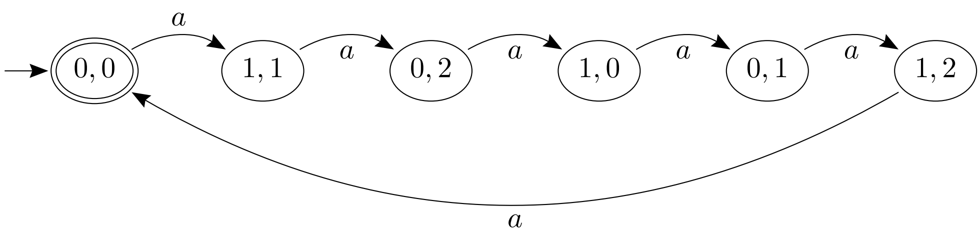
(b) Definiere $A_n := ([n-1], \Sigma, \delta_n, 0, \{0\})$ mit $\delta_n(q, a) := (q + 1) \bmod n$

Korrektheit: Wir schreiben $q \rightarrow^{w}_{A_n} q'$ für $\hat{\delta}_n(q, w) = q'$
Wir müssen zeigen: $w \in L_n \iff w \in L(A_n)$.

$$
\begin{aligned}
w \in L(A_n) &\iff 0 \rightarrow^{w}_{A_n} 0 \\
&\iff \underbrace{0 \rightarrow^{a^n}_{A_n} 0 \rightarrow^{a^n}_{A_n} \dots \rightarrow^{a^n}_{A_n} 0}_{m \text{ mal}} \\
&\iff \exists m. w = a^{mn} \\
&\iff w \in L_n
\end{aligned}
$$

Minimalität: Beweis durch Wiederspruch. Nehme an es gäbe einen DFA $A' = (Q', \Sigma, \delta', q'_0, F')$ der L_n erkennt, mit weniger Zuständen. Da $\varepsilon \in L_n$ gilt $q'_0 \in F'$. Da $a^n \in L_n$ auch $q'_0 \rightarrow^{a^n}_{A'} q'_f \in F'$, d.h. es existiert ein akzeptierender Lauf der Länge n. Da aber A' weniger als n Zustände hat und $|\Sigma| = 1$ muss der Automat während dieses Laufes einen Zustand zweimal besuchen, d.h. $q'_0 \rightarrow^{a^x}_{A'} q' \rightarrow^{a^y}_{A'} q' \rightarrow^{a^z}_{A'} q'_f \in F$, für $y > 0, n = x + y + z$. Da A' weniger als n Zustände hat, muss insbesondere auch $y < n$ sein. Dann ist aber auch $q'_0 \rightarrow^{a^x}_{A'} q' \rightarrow^{a^z}_{A'} q'_f \in F'$ ein akzeptierender Lauf und daher wäre $a^{x+z} \in L_n$. Wiederspruch!

(c) Produktautomat:



(d) Zum Beispiel $\{(n, m) | n$ und m sind teilerfremd$\}$. (Auch möglich: n, m Primzahlen, $n, n+1$ für $n > 2$, beide implizieren Teilerfremdheit).

(e) Da n, m teilerfremd sind ist ihr kleinstes gemeinsames Vielfaches nm. Dann ist $L_n \cap L_m = \{a^{knm} | k \in \mathbb{N}\} = L_{nm}$. Die Produkkonstruktion produziert einen Automaten für $L_n \cap L_m$ mit (maximal) nm Zuständen. Nach (b) hat ein Automat für L_{nm} mindestens nm Zustände.