Aufgabe 2

Übungsblatt 3

Eric Kunze — 10. November 2021

Keine Garantie auf Vollständigkeit und/oder Korrektheit!

Aufgabe. Sei L eine reguläre Sprache über einem mindestens zweielementigen Alphabet Σ. Zeigen Sie, dass die folgenden Sprachen regulär sind.

(a) $L_1 = \{x \in L : \text{es gibt kein } y \in \Sigma^+, \text{ so dass } xy \in L\}$

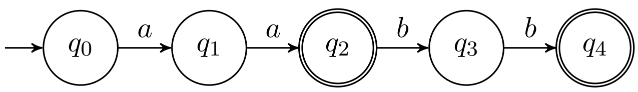
(b) $L_2 = \{x \in L : \text{kein echtes Präfix von } x \text{ liegt in } L\}$

Ausgangspunkt ist also in beiden Fällen eine reguläre Sprache L, für die wir einen DFA $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F\rangle$ finden können, sodass $L = L(\mathcal{M})$ (d.h. $\mathcal{M}$ erkennt genau die Sprache L). Diesen Automaten haben wir gegeben und können ihn nun so verändern, dass er die Sprachen L_1 bzw. L_2 erkennt.

Teil (a): es gibt keine Verlängerung

In L_1 sollen nur noch die Wörter x aus L enthalten sein, die sich nicht in L verlängern lassen. Es soll also kein echtes Wort $y \in \Sigma^+$ geben, sodass das verlängerte Wort xy in L läge. Wann kann eine solche Verlängerung auftreten? Wir müssten in L sowohl das kurze Wort als auch das verlängerte Wort erkennen. Das können wir uns beispielhaft wie folgt vorstellen.

Beispiel. *Der Automat $\mathcal{M}$ für L sei wie folgt gegeben:*



Damit ist $L = \{aa, aabb\}$. Für das Wort $x = aa$ existiert folglich eine Verlängerung $y = bb \in \Sigma^+$, sodass $xy = aabb \in L$ gilt. Für $x = aabb$ existiert eine solche Verlängerung jedoch nicht. Somit gilt für dieses Beispiel $L_1 = \{aabb\}$.

Anhand des Beispiels erkennen wir relativ gut, dass wir die Akzeptanz von Wörtern „auf dem Weg“ unterbinden müssen. Im Beispiel muss also die Akzeptanz in q_2 verboten werden, d.h. q_2 darf kein Finalzustand mehr sein, weil wir ausgehend von q_2 noch einmal in einen Finalzustand gelangen können.

Deswegen definieren wir uns die Menge

$$F' := \{q \in F : \text{ex. kein } y \in \Sigma^* : \delta(q, y) \in F\}\,.$$

Diese Menge besteht aus allen Finalzuständen $q \in F$ des Automaten $\mathcal{M}$ für L, für die wir kein Wort $y \in \Sigma^+$ finden können, sodass der Wortübergang $\delta(q, y)$ wieder in einem Finalzustand landet.

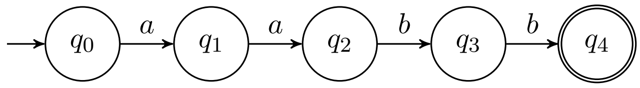
Beispiel. *Wir wollen diese Definition anhand des obigen Beispiels nachvollziehen. Es gilt auf jeden Fall $F' \subseteq F$, d.h. wir müssen nur Finalzustände des Originalautomaten betrachten. Für q_2 existiert $y = bb \in \Sigma^*$, sodass $\delta(q_2, y) = \delta(q_2, bb) = q_4 \in F$. Damit ist also $q_2 \notin F'$. Für q_4 finden wir aber offensichtlich keinen Wortübergang, der noch einmal in einem Finalzustand landet (da es überhaupt gar keinen Übergang mehr gibt), d.h. $q_4 \in F'$. Zusammengefasst ist also $F' = \{q_4\}$.*

Ersetzen wir nun die Menge der Finalzustände F im Originalautomaten durch F', dann fallen alle unterwegs akzeptierenden Finalzustände weg und wir erreichen genau unser Ziel. Setze also

$$M' := \langle Q, \Sigma, \delta, q_0, F' \rangle$$

und übernehmen dabei alle anderen Komponenten (Zustandsmenge, Alphabet, Übergangsfunktion, Startzustand) des Originalautomaten.

Beispiel. *Der Automat $\mathcal{M}'$ für obigen Beispiel sieht dann wie folgt aus:*



Man beachte, dass sich wirklich nur die Menge der Finalzustände geändert hat. Die erkannte Sprache ist nun $L(\mathcal{M}') = \{aabb\}$, was genau L_1 für das Beispiel entspricht.

Nun haben wir also einen DFA $\mathcal{M}'$ konstruiert, von dem wir hoffen, dass er genau L_1 erkennt. Um das zu verifizieren, müssen wir noch $L(\mathcal{M}') = L_1$ zeigen.

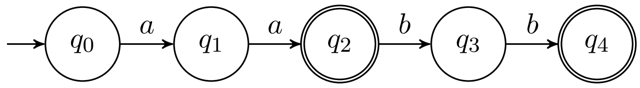
$$\begin{aligned} x \in L(\mathcal{M}') &\Leftrightarrow \delta(q_0, x) \in F' && \text{(Def. Akzeptanz)} \\ &\Leftrightarrow \delta(q_0, x) \in F \text{ und ex. kein } y \in \Sigma^+ : \delta(q_0, xy) \in F && \text{(Def. } F'\text{)} \\ &\Leftrightarrow x \in L_1 && \text{(Def. von } L_1\text{)} \end{aligned}$$

Unsere Konstruktion ist folglich korrekt.

Teil (b): keinen Präfix

Die Sprache L_2 soll genau die Wörter von L enthalten, die keinen echten[1] Präfix in L haben. Es soll also das lange Wort xy in L liegen, nicht jedoch der Präfix x.

Beispiel. *Wir arbeiten wieder mit unserem Beispielautomaten $\mathcal{M}$.*



Nach wie vor gilt $L = L(\mathcal{M}) = \{aa, aabb\}$. Für $xy = aa$ finden wir keinen Präfix, der auch in L liegt. Es gäbe nur die Möglichkeiten $x = \epsilon \notin L$ und $x = a \notin L$. Das ergibt $aa \in L_2$. Für $xy = aabb$ finden wir jedoch einen solchen Präfix, nämlich mit $x = aa \in L$. Somit ist $aabb \notin L_2$

Am Beispiel machen wir uns wieder schnell klar, was wir verbieten müssen: erreichen wir einmal einen Finalzustand, dürfen wir nicht mehr weitergehen, da sonst ein längeres Wort enstehen kann, dessen Präfix wir mit Erreichen des ersten Finalzustands auch akzeptieren. Im Beispielautomaten darf vom Finalzustand q_2 kein Übergang mehr erfolgen, weil wir sonst noch in den Finalzustand q_4 gelangen können. Dort wird das Wort *aabb* akzeptiert, bis q_2 haben wir aber schon dessen Präfix *aa* akzeptiert und das soll in L_2 nicht möglich sein. Wir definieren also für alle Symbole $a \in \Sigma$ die Übergänge

$$\delta''(q, a) := \begin{cases} q_\perp & \text{falls } q \in F \\ \delta(q, a) & \text{sonst} \end{cases}$$

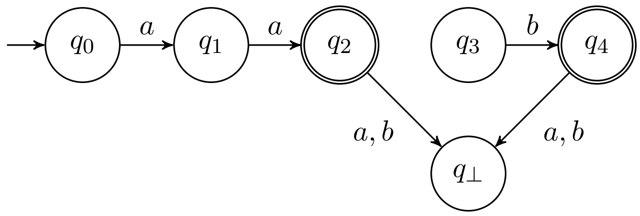
Dabei soll $q_\perp \notin Q$ ein neu eingeführter Zustand sein, in dem wir anhalten ohne zu akzeptieren (es gibt keinen ausgehenden Übergang, aber $q_\perp$ ist auch kein Finalzustand).

Damit können wir nun einen DFA

$$\mathcal{M}'' := \langle Q \cup \{q_\perp\}, \Sigma, \delta'', q_0, F \rangle$$

definieren, der obige Übergangsfunktion δ'' verwendet sowie den neu eingeführten Fangzustand $q_\perp$.

Beispiel. *Für unser Beispiel fügen wir den Zustand $q_\perp$ und Übergänge von den Finalzuständen q_2 und q_4 mit den Symbolen a und b aus Σ zum Fangzustand. Der Übergang $q_2 \xrightarrow{b} q_3$ wird dabei überschrieben durch den Übergang $q_2 \xrightarrow{a,b} q_\perp$.*



Nun bleibt auch hier zu zeigen, dass unsere Konstruktion korrekt ist, d.h. dass $L(\mathcal{M}'') = L_2$

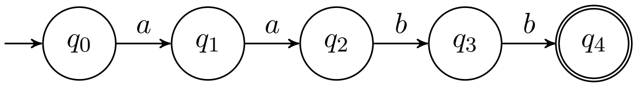
[1] im folgenden steht Präfix immer für ein *echtes* Präfix

gilt.

$$\begin{aligned}
x \in L(\mathcal{M}'') \quad &\Leftrightarrow \quad \delta''(q_0, x) \in F && \text{(Def. Akzeptanz)} \\
&\Leftrightarrow \quad \delta(q_0, x) \in F \text{ und ex. kein echtes Präfix } \overline{x} \text{ von } x \text{ mit } \delta(q_0, \overline{x}) \in F && \text{(Def. } \delta''\text{)} \\
&\Leftrightarrow \quad x \in L_2 && \text{(Def. von } L_2\text{)}
\end{aligned}$$

Erkennt $\mathcal{M}'$ auch L_2?

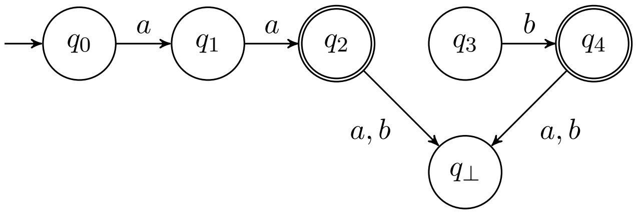
Wir verwenden wieder unser Beispiel und wollen $L_2 = \{aa\}$ mit dem Automat $\mathcal{M}'$ erkennen.



Wir können zwar die Übergänge $q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_2$ durchführen und das Wort aa zwischenzeitlich lesen, jedoch ist nach unserer Konstruktion q_2 kein Finalzustand mehr, sodass wir an dieser Stelle nicht akzeptieren und weitere Übergänge durchführen müssten. Dadurch verlängert sich das Wort aa zwangsweise zu $aabb$. Es gilt also $aa \notin L(\mathcal{M}')$ und somit schon $L_2 \neq L(\mathcal{M}')$. L_2 wird folglich nicht von $\mathcal{M}'$ erkannt.

Erkennt $\mathcal{M}''$ auch L_1?

Wieder versuchen wir anhand des Beispiels die Sprache $L_1 = \{aabb\}$ mit dem Automaten $\mathcal{M}''$ zu erkennen.



Wir starten und können zumindest den Teil aa lesen und würden diesen auch schon akzeptieren. Es ist also $aa \in L(\mathcal{M}'')$ aber offensichtlich $aa \notin L_1$. Somit muss schon $L_1 \neq L(\mathcal{M}'')$ gelten, d.h. L_1 wird nicht von $\mathcal{M}''$ erkannt.