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

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

Sommersemester 2023 – Hausaufgabenblatt 1

- Beachten Sie die Abgaberichtlinien für die Hausaufgaben auf der Vorlesungswebsite!
- 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.

AT-Aufgabe H1.1. (*Mein Lieblingstutor*) 0 Punkte

Diese Hausaufgabe wird mit Automata Tutor bearbeitet. Falls Sie es noch nicht bereits gemacht haben, folgen Sie den Schritten in Ü1.2, um ein Konto zu erstellen. Achten Sie darauf, dass Sie sich, wie dort beschrieben, mit Ihrer TUM-Kennung anmelden.

Bearbeiten Sie die Hausaufgaben H1.1 (a–c).

Aufgabe H1.2. (*Verstehen Sie Sprachen?*) 3 Punkte

Sei $\Sigma := \{a, b\}$, $A := \{\varepsilon, a, ab\}$, $B := \{b, ba\}$. Bestimmen Sie folgende Mengen:

(a) $AB \setminus (A \cup B)$ (b) $A^3\emptyset$ (c) $\emptyset^* A^0$ (d) $(A \times \emptyset)^* \times B$

Lösungsskizze.

(a) $\{aba, abb, abba\}$

(b) $\emptyset$

(c) $\{\varepsilon\}$

(d) $\{(\varepsilon, b), (\varepsilon, ba)\}$

Aufgabe H1.3. (*Fragwürdige Wahrheiten*) 4 Punkte

Beim Chat mit einem textbasierten Onlinedialogsystem™ erhält Dora die folgenden, angeblich gültigen, Aussagen. Sie ist sich jedoch nicht sicher, ob Sie dem System vertrauen kann...Sie bittet euch daher um Hilfe, die Behauptungen zu überprüfen.

Sei $\Sigma \neq \emptyset$ ein Alphabet, und $A, B, C \subseteq \Sigma^*$ Sprachen über Σ. Beweisen oder widerlegen Sie folgende Aussagen.

(a) $AA^* = A^* \implies \varepsilon \in A$

(b) $A \subseteq B \Leftrightarrow A^+ \subseteq B^+$

(c) $|A \times A| = |AA|$

(d) $A(BC) = (AB)C$

(e) $AB \setminus AC = A(B \setminus C)$

(f) $|AB| \geq |A|$

Hinweis: Für alle ungültigen Aussagen gibt es ein Gegenbeispiel, in den jede Sprache höchstens zwei Wörter mit höchsten zwei Buchstaben hat.

Lösungsskizze. In allen Gegenbeispielen verwenden wir $\Sigma := \{a, b\}$.

(a) Wahr. Sei $A^* \subseteq AA^*$. Da $\varepsilon \in A^*$ gilt, folgt auch $\varepsilon \in AA^*$. Deshalb folgt $\varepsilon = uv$ mit $u \in A, v \in A^*$. Dann muss $u = \varepsilon$ sein und daher auch $\varepsilon \in A$.

(b) Falsch. $A := \{aa\}, B := \{a\}$, womit $A^+ \subseteq B^+$ gilt, $A \subseteq B$ aber nicht.

(c) Falsch. Sei $A := \{\varepsilon, a\}$. Dann ist $|A \times A| = |\{(\varepsilon, \varepsilon), (\varepsilon, a), (a, \varepsilon), (a, a)\}| = 4$ aber $|AA| = |\{\varepsilon, a, aa\}| = 3$.

(d) Wahr. Nutze die Assoziativität von Konkatenation auf Wörtern:

$$\begin{aligned} u \in A(BC) &\iff u = v(xy) \text{ für } v \in A, x \in B, y \in C \\ &\iff u = (vx)y \text{ für } v \in A, x \in B, y \in C \\ &\iff u \in (AB)C \end{aligned}$$

(e) Falsch. $A := \{a, aa\}, B := \{ab\}, C := \{b\}$, also gelten $AB \setminus AC = \{aaab\}$ und $A(B \setminus C) = \{aab, aaab\}$.

(f) Falsch. $A := \{a\}, B := \emptyset$

Aufgabe H1.4. (*Obacht! Ein n-DFA!*) 2 + 2 + 3 Punkte

Der Superschurke Dr. Evilsparza ist zurück! Seine lange Winterpause hat er sinnvoll genutzt, um an neuen Automaten in seiner Werkstube zu schrauben. Seine erste Erfindung: der *n-DFA*. Teuflisch lässt er die theoretische Konstruktion auf die arme Studierendenschaft los. Gelingt es euch, Dr. Evilsparzas Erfindung durch die Kraft des Beweises zu zähmen?

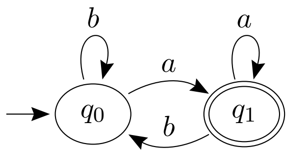
Sei $n \in \mathbb{N}_+$. Ein n-DFA M ist ähnlich zu einem DFA und wird beschrieben durch ein Tupel $M = (Q, \Sigma, \delta, q_0, F)$. Allerdings akzeptiert ein n-DFA ein Wort nicht, indem er beim Lesen des Wortes in einem Zustand in F endet, sondern indem er während des Lesens des Wortes *mindestens* n-mal Zustände in F besucht. Insbesondere kann ein n-DFA sich am Ende eines akzeptierten Laufes also auch in einem Zustand in $Q \setminus F$ befinden.

Genauer definieren wir die *Multimenge*[1] der von M besuchten Endzuständen bei Eingabe w aus Zustand q rekursiv durch

$$V(\epsilon, q) := \begin{cases} \{q\}_m, & \text{falls } q \in F \\ \emptyset_m, & \text{sonst} \end{cases} \qquad \text{und} \qquad V(aw, q) := \begin{cases} \{q\}_m \cup_m V(w, \delta(q, a)), & \text{falls } q \in F \\ V(w, \delta(q, a)), & \text{sonst} \end{cases}$$

und die von M akzeptierte Sprache als $L(M) := \{w \in \Sigma^* \mid |V(w, q_0)|_m \geq n\}$.

Beispiel: der folgende n-DFA akzeptiert die Sprache $\{w \in \{a, b\}^* \mid |w|_a \geq n\}$.



Es gilt beispielsweise

$$\begin{aligned} V(aba, q_0) &= V(ba, q_1) = \{q_1\}_m \cup_m V(a, q_0) \\ &= \{q_1\}_m \cup_m V(\epsilon, q_1) = \{q_1\}_m \cup_m \{q_1\}_m = \{q_1, q_1\}_m. \end{aligned}$$

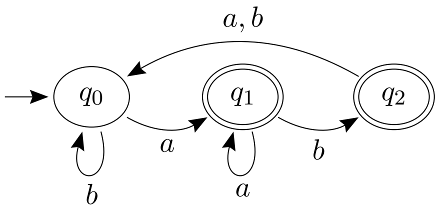
(a) Zeigen Sie: Es gibt einen DFA dessen Sprache von keinem 1-DFA erkannt werden kann.

(b) Zeigen oder widerlegen Sie: für jeden DFA M existiert ein n, sodass es einen n-DFA gibt, der dieselbe Sprache wie M akzeptiert.

[1] d.h. Mengen, die möglicherweise Duplikate enthalten; zur Klarstellung indizieren wir Operationen auf Multimengen mit einem m, z.B. $\emptyset_m, \cup_m, \{\cdot\}_m, |\cdot|_m$

Falls Sie die Aussage widerlegen möchten, geben Sie ein passendes Gegenbeispiel an und zeigen Sie die Korrektheit ihres Gegenbeispiels. Falls Sie die Aussage zeigen möchten, geben Sie eine formale Übersetzung von DFAs zu n-DFAs an. Die Korrektheit der Übersetzung müssen Sie dabei nicht beweisen, allerdings informell erläutern.

(c) Zeichnen Sie einen DFA, der dieselbe Sprache wie der folgende 3-DFA erkennt:



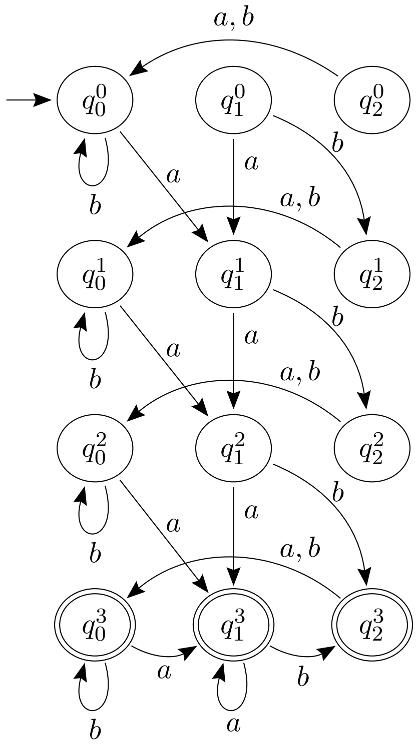
Lösungsskizze.

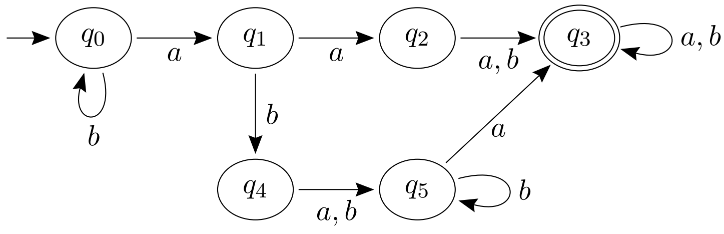
(a) Beispiel: der DFA, der nur das leere Wort akzeptiert. Gäbe es einen passenden 1-DFA, müsste dessen Initialzustand in F liegen. Damit wäre die Akzeptanzbedingung des 1-DFA jedoch bereits bei Start für jedes Wort erfüllt und somit erkennt der 1-DFA die Sprache Σ^*.

Bemerkung: allgemein gilt für einen n-DFA M, dass für jedes $w \in L(M)$ auch alle Erweiterungen $w' \in \{w\}\Sigma^*$ von M akzeptiert werden.

(b) Gegenbeispiel: Es gibt keinen n-DFA für die Sprache $\{\epsilon\}$. Angenommen es gäbe einen solchen n-DFA M. Da $\epsilon \in L(M)$, gilt $|V(\epsilon, q_0)|_m \geq n \geq 1$. Außerdem gilt $1 \geq |V(\epsilon, q_0)|_m$ Somit $n = 1$. Damit folgt die Aussage mit Aufgabe (a).

(c) Idee: Erstelle Kopien des 3-DFAs und zähle die Anzahl der Besuche von F im Zustand (linkes Bild). Rechts im Bild: der dazugehörige minimale Automat.





Aufgabe H1.5. (*Buchstabenfhler*) 2 + 2 Punkte

Nicht nur in seiner Werkstube war Dr. Evilsparza aktiv. Er hat sich auch noch ins Campusnetzwerk gehackt und dabei ist es ihm gelungen, wichtige Daten zu löschen! Sie sind Teil des Cybersecurityteams, das versucht das Ausmaß des Schadens zu bestimmen.

Sei $\Sigma \coloneqq \{a, b\}$ ein Alphabet.

(a) Sei G eine beliebige Grammatik, und G' eine Grammatik, die durch das Entfernen von einer beliebigen Produktion aus G entsteht. Beweisen oder widerlegen Sie $L(G') \subseteq L(G)$.

(b) Sei G wieder eine beliebige Grammatik, und G' eine Grammatik, die man erhält, wenn man von G ein beliebiges Nichtterminalzeichen aus der rechten Seite einer beliebigen Produktion löscht. Beweisen oder widerlegen Sie: $L(G') \subseteq L(G)$

Lösungsskizze.

(a) Die Aussage ist wahr: Sei $w \in L(G')$ beliebig, und S das Startsymbol von G. Nach Definition von $L(G')$ gilt also $S \rightarrow^*_{G'} w$. Da wir für G' aber nur eine Produktion entfernt haben, gilt $S \rightarrow^*_G w$ ebenfalls, und somit $w \in L(G)$.

(b) Falsch! Ein Gegenbeispiel wäre $G = (\{S\}, \Sigma, P, S)$ mit $P = \{S \rightarrow aS\}$. Es ist nicht möglich, in G ein Wort zu produzieren, das nur aus Terminalzeichen besteht, also $L(G) = \emptyset$. Wenn wir allerdings die Produktionsregel zu $S \rightarrow a$ verändern, erhalten wir $L(G') = \{a\}$.