Prof. Javier Esparza
Philipp Czerner, Kevin Kappelmann, Valentin Krasotin
Technische Universität München
Lehrstuhl für Theoretische Informatik

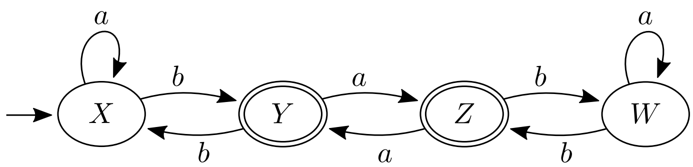
Einführung in die Theoretische Informatik

Sommersemester 2024 – Hausaufgabenblatt 3

- Die Hausaufgaben werden in folgender Reihenfolge korrigiert: H3.4, H3.2, H3.3.
- Die Knobelaufgabe bitte separat auf Moodle abgeben. Sie wird korrigiert.

AT-Aufgabe H3.1. (*DFA zu RE*) unkorrigiert (3 Punkte)

Bearbeiten Sie diese Aufgabe in Automata Tutor. Konvertieren Sie den folgenden DFA M zu einem regulären Ausdruck, z.B. indem Sie ein geeignetes Gleichungssystem aufstellen und dieses lösen.



Lösungsskizze. Wir stellen folgende Gleichungen auf:

$$\begin{aligned} X &\equiv aX \mid bY \\ Y &\equiv aZ \mid bX \mid \boldsymbol{\epsilon} \\ Z &\equiv aY \mid bW \mid \boldsymbol{\epsilon} \\ W &\equiv aW \mid bZ \end{aligned}$$

Um dieses System zu lösen, setzen wir $r := (ba^*b)^*$, $s := (ara)^*$ und folgern:

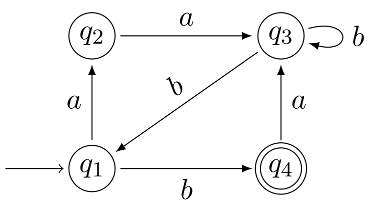
$$\begin{aligned} W &\overset{*}{\equiv} a^*bZ \\ Z &\equiv aY \mid ba^*bZ \mid \boldsymbol{\epsilon} \overset{*}{\equiv} (ba^*b)^*aY \mid (ba^*b)^* \equiv raY \mid r \\ Y &\equiv araY \mid ar \mid bX \mid \boldsymbol{\epsilon} \overset{*}{\equiv} s(ar \mid \boldsymbol{\epsilon}) \mid sbX \\ X &\equiv aX \mid bs(ar \mid \boldsymbol{\epsilon}) \mid bsbX \overset{*}{\equiv} (a \mid bsb)^*bs(ar \mid \boldsymbol{\epsilon}) \end{aligned}$$

Äquivalenzen, die aus Ardens Lemma folgen, sind mit $*$ markiert. Indem wir r und s wieder einsetzen, erhalten wir

$$X \equiv (a \mid b(a(ba^*b)^*a)^*b)^*b(a(ba^*b)^*a)^*(a(ba^*b)^* \mid \boldsymbol{\epsilon})$$

Aufgabe H3.2. (*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 H3.3. (*ein paar as, dann ein b...*) 1+1+1 Punkte

Geben sie an, zu welchen regulären Ausdrücken die gleiche Sprache gehört. Geben Sie außerdem für jede Sprache, die Sie so identifizieren, ein Wort an, das nur in dieser Sprache liegt. Die sechs regulären Ausdrücke erzeugen drei unterschiedliche Sprachen.

Beispiel: Sie stellen fest, dass (1,2), (3,4) und (5,6) die gleiche Sprache erzeugen. Dann müssen Sie ein Wort in $L_{1,2} \setminus (L_{3,4} \cup L_{5,6})$ finden, eines in $L_{3,4} \setminus (L_{1,2} \cup L_{5,6})$, und eines in $L_{5,6} \setminus (L_{1,2} \cup L_{3,4})$.

Wie üblich schreiben wir r^+ anstelle von rr^* für einen regulären Ausdruck r.

(1) $b \mid a^+(baa^+)^*(b|bab)$

(2) $(a \mid ab)^+ \mid a(b^*a\boldsymbol{\emptyset})^+$

(3) $(a^+b)^*$

(4) $(aba \mid a)^* b$

(5) $a(a^*ba)^*a^*(b \mid \boldsymbol{\epsilon})$

(6) $\boldsymbol{\emptyset}^* \mid a(a \mid ba)^* b$

Update: Sie können diese Aufgabe nun auch in apotheosis lösen.

Lösungsskizze.

- (1) $\equiv$ (4): b
- (2) $\equiv$ (5): a (alle Wörter in a^+ sind möglich)
- (3) $\equiv$ (6): ε

Aufgabe H3.4. (*Pump für den Schurken*) 3+1+4 Punkte

In dieser Aufgabe nehmt Ihr die Rolle des trickreichen Superschurken Dr. Evilsparza ein. Seid bereit, möglichst großes Unheil anzurichten.

Mit eurem Kristallomaten beobachtet ihr – Dr. Evilsparzas – wie die tapferen Theo-Studierenden in Ü3.6 die Nicht-Regularität verschiedenster Sprachen mithilfe des Pumping-Lemmas bewiesen haben. Den vielen Erfolgen möchtet ihr natürlich schnell ein Ende bereiten, weswegen ihr euch in eure Werkstube begebt, um an trickreichen Sprachen zu tüfteln. Und nach kurzer Zeit ist es euch dann auch gelungen: eine teuflische Sprache, um die Studierenden in die Irre zu führen!

Um die Studierenden zu überlisten, fixiert ihr das Alphabet $\Sigma = \{a, b, c\}$ und die folgenden Sprachen:

$$L_1 \coloneqq \{ab^nc^n \mid n \in \mathbb{N}_+\}, \quad L_2 \coloneqq \{a^lb^mc^n \mid l \in (\mathbb{N} \setminus \{1\}), m, n \in \mathbb{N}\}, \quad L_3 \coloneqq L_1 \cup L_2.$$

Mit der Sprache L_3, so eure Vermutung, werdet ihr die Studierenden so richtig übers Ohr hauen. Denn es ist eine nicht-reguläre Sprache, die dennoch die Bedingungen des Pumping-Lemmas erfüllt! Ihr freut euch schon darauf, wie die Studierenden aussichtslos und tagelang dennoch versuchen, Nicht-Regularität der Sprache mit dem Pumping-Lemma zu beweisen.

Bevor ihr nun aber die Falle einsetzt und euch an den Misserfolgen ergötzt, möchtet ihr natürlich sicherstellen, dass sie auch wirklich funktioniert...

(a) Zeigen Sie mithilfe des Pumping-Lemmas, dass L_1 nicht regulär ist.

(b) Zeigen Sie mithilfe des Ergebnisses aus (a) und der Abschlusseigenschaft von regulären Sprachen unter Schnitt ($\cap$), dass L_3 nicht regulär ist.

Hinweis: Nehmen Sie hierfür an, dass L_3 regulär ist und führen sie dies mit Hilfe der genannten Abschlusseigenschaft zu einem Widerspruch.

(c) Zeigen Sie, dass L_3 die Bedingungen des Pumping-Lemmas erfüllt.

Tipp: Machen Sie eine Fallunterscheidung, ob das zu zerlegende Wort in L_1 oder L_2 ist.

Lösungsskizze.

(a) Angenommen, L_1 wäre regulär.

Dann können wir das Pumping-Lemma anwenden.

Sei $n \in \mathbb{N}_+$ eine Pumping-Lemma-Zahl.

Dann gilt $z = ab^nc^n \in L_1$ und $|z| \geq n$.

Es gibt also für z eine Zerlegung $z = uvw$ mit $v \neq \varepsilon$ und $|uv| \leq n$, sodass $uv^iw \in L_1$ für alle $i \in \mathbb{N}$ $(*)$.

Wir betrachten zwei Fälle:

- **Fall** $|v|_a = 1$**:** Dann folgt $|uv^0w|_a = 0$ und daher $uv^0w \notin L_1$. Widerspruch zu $(*)$.
- **Fall** $|v|_a = 0$**:** Da $|uv| \leq n$ und $v \neq \epsilon$, folgt aufgrund der Wahl von z dass $|v|_b > 0$ und $|uv|_c = 0$. Dann folgt $|uv^0w|_b < |uvw|_b = |uvw|_c = |uv^0w|_c$. Somit $uv^0w \notin L_1$. Widerspruch zu $(*)$.

Das heißt unsere ursprüngliche Annahme ist falsch und L_1 nicht regulär.

(b) Wenn L_3 regulär wäre, dann wäre auch $L_3 \cap L(ab^*c^*) = L_1$ regulär (reguläre Sprachen sind abgeschlossen unter Durchschnitt). Widerspruch zu (a).

(c) Wir setzen $n := 3$. Sei nun $z \in L_3$ mit $|z| \geq n$. Wir müssen z geeignet zerlegen, sodass die Bedingungen des Pumping-Lemmas erfüllt sind.

- **Fall** $z \in L_1$**:** Dann gilt $z = ab^mc^m$ mit $m \in \mathbb{N}_+$. Wir setzen $u := \epsilon$, $v := a$, $w := b^mc^m$. Dann gillt $v = a \neq \epsilon$, $|uv| = 1 \leq 3 = n$ und $uv^iw = a^ib^mc^m$. Für $i = 1$ gilt dann $a^ib^mc^m \in L_1 \subseteq L_3$ und für $i \neq 1$ gilt $a^ib^mc^m \in L_2 \subseteq L_3$.
- **Fall** $z \in L_2$**:** Dann gilt $z = a^lb^mc^k$ mit $l \in (\mathbb{N} \setminus \{1\})$ und $m, k \in \mathbb{N}$.
  - **Fall** $l > 2$**:** Wir setzen $u := \epsilon$, $v := a$, $w := a^{l-1}b^mc^k$. Dann gillt $v = a \neq \epsilon$, $|uv| = 1 \leq 3 = n$ und $uv^iw = a^{i+l-1}b^mc^k$. Da $l > 2$, folgt $i + l - 1 \geq 2$ für alle $i \in \mathbb{N}$. Somit $a^{i+l-1}b^mc^k \in L_2 \subseteq L_3$.
  - **Fall** $l \leq 2$ **und** $m > 0$**:** Wir setzen $u := a^l$, $v := b$, $w := b^{m-1}c^k$. Dann gillt $v = b \neq \epsilon$, $|uv| = l + 1 \leq 2 + 1 = 3 = n$ und $uv^iw = a^lb^{i+m-1}c^k$. Da $m > 0$, folgt $i + m - 1 \in \mathbb{N}$ für alle $i \in \mathbb{N}$ und somit $a^lb^{i+m-1}c^k \in L_2 \subseteq L_3$ (bemerke: $l \in \{0, 2\}$).
  - **Fall** $l \leq 2$ **und** $m = 0$**:** Dann folgt $z = a^lc^k$ und da $l \leq 2$ und $|z| \geq n = 3$ folgt $k > 0$. Wir setzen $u := a^l$, $v := c$, $w := c^{k-1}$. Dann gillt $v = c \neq \epsilon$, $|uv| = l + 1 \leq 2 + 1 = 3 = n$ und $uv^iw = a^lc^{i+k-1}$. Da $k > 0$, folgt $i + k - 1 \in \mathbb{N}$ für alle $i \in \mathbb{N}$ und somit $a^lc^{i+k-1} \in L_2 \subseteq L_3$ (bemerke: $l \in \{0, 2\}$).

Quizaufgabe H3.5. (*Spieglein, Spieglein an der Wand...*) unkorrigiert (2+1+1 Punkte)

Sei $\Sigma := \{a, b\}$. Wie üblich definieren wir w^R als die Spiegelung des Wortes w. Für eine beliebige Sprache L definieren wir nun $L^R := \{w^R : w \in L\}$.

(a) Sei M ein NFA mit n Zuständen. Konstruieren Sie einen NFA für $L(M)^R$ mit höchstens $n + 1$ Zuständen.

(b) Sei r ein regulärer Ausdruck. Konstruieren Sie einen regulären Ausdruck r' für $L(r)^R$, dessen Länge mit r übereinstimmt.

Hinweise: Es bietet sich an, ähnlich zur strukturellen Induktion, ein rekursives Verfahren anzugeben. Die Länge eines regulären Ausdrucks ist die Länge der üblichen Darstellung im Alphabet $\{(,), *, |, \epsilon, \emptyset\} \cup \Sigma$, z.B. hat $(ab^* \mid \epsilon)$ Länge 7.

Lösungsskizze. **(a)** Idee: Wenn M ein Wort akzeptiert, gibt es einen Lauf von einem Start- in einen Endzustand. Der neue NFA soll diesen Lauf rückwärts ausführen; das bedeutet, wir wollen die letzte Transition zuerst rückwärts ausführen, dann die vorletzte, und so weiter. Wir müssen in einem Finalzustand starten, wir wissen jedoch nicht, in welchem. Dies wird der neue NFA am Anfang erraten. Wenn wir von irgendeinem dieser

Finalzustände aus beim Rückwärtsausführen im Initialzustand landen, dann wollen wir das Wort akzeptieren.

Sei $M = (Q, \delta, \Sigma, q_0, F)$. Wir definieren uns einen NFA $M' := (Q', \delta', \Sigma, q_0', F')$ mit $Q' := Q \uplus \{q_0'\}$, wobei $q_0' \notin Q$ ein neuer Zustand ist und unser Startzustand wird. Wir übernehmen wir die Transitionen aus M, drehen sie aber um:

$$\delta'(q', a) := \{q \in Q : q' \in \delta(q, a)\} \quad \text{für alle } q' \in Q, a \in \Sigma$$

Es gibt also eine Transition von q' zu q in M' genau dann, wenn es eine von q zu q' in M gibt. Wenn wir von q_0' ein Zeichen a lesen, gehen wir in einen beliebigen Zustand, den man von F mit einem a erreichen kann, wir setzen also

$$\delta'(q_0', a) := \delta'(F, a) \quad \text{für alle } a \in \Sigma.$$

Die akzeptierenden Zustände sind $F' := \{q_0\}$, falls $\varepsilon \notin L(M)$, sonst $F' := \{q_0, q_0'\}$.

Dieser NFA hat wie gefordert $n + 1$ Zustände.

(b) Wir geben die Konstruktion rekursiv an. Es gibt folgende Möglichkeiten:

- $r \in \{\boldsymbol{\emptyset}, \boldsymbol{\epsilon}\} \cup \Sigma$: Wir setzen $r' := r$, da hier $L(r) = L(r)^R$ gilt.
- $r = s^*$: Wir konstruieren rekursiv einen regulären Ausdruck t mit $L(t) = L(s)^R$ und setzen $r' := t^*$. Dann gilt

  $$L(r)^R = (L(s)^*)^R = (L(s)^R)^* = L(t)^* = L(t^*) = L(r')$$

- $r = st$: Rekursiv gibt es reguläre Ausdrücke s', t' mit $L(s') = L(s)^R$ und $L(t') = L(t)^R$. Wir definieren $r' := t's'$ und erhalten

  $$L(r)^R = L(st)^R = L(t)^R L(s)^R = L(t')L(s') = L(t's') = L(r')$$

- $r = s \,|\, t$: Wieder gibt es s', t' mit $L(s') = L(s)^R$ und $L(t') = L(t)^R$. Wir definieren $r' := s' \,|\, t'$ und erhalten

  $$L(r)^R = L(s \,|\, t)^R = L(s)^R \cup L(t)^R = L(s') \cup L(t') = L(s' \,|\, t') = L(r')$$

In allen Fällen bleibt die Länge des regulären Ausdrucks erhalten.

Knobelaufgabe H3.6. (*Nach den Sternen greifen*)

Sei L eine reguläre Sprache. Zeigen Sie, dass $L' := \{w : \{w\}^* \subseteq L\}$ regulär ist.

Lösungsskizze. Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DFA, der L akzeptiert. Die Idee ist, dass wir einen DFA M' für L' konstruieren, der sich nach Einlesen eines Wortes $w \in \Sigma^*$ merkt, wie M sich auf dem Wort w verhält. Dafür benötigen wir lediglich die Funktion $f_w : Q \to Q$, die wir definieren als $f_w(q) := \hat{\delta}(q, w)$ für alle Zustände $q \in Q$. Es gilt also beispielsweise $f_{ww}(q) = f_w(f_w(q))$; oder auch $f_{wv} = f_v \circ f_w$ für alle Wörter w, v. Beachten Sie, dass es zwar unendliche viele Wörter w gibt, aber nur endlich viele Funktionen f_w.

Schreiben wir nun f_w^k für die k-fache Anwendung von f_w (also $f_w^0(q) := q$ und $f_w^{k+1}(q) := f_w^k(f_w(q))$), dann ist gilt allgemein $f_{w^k}(q) = f_w^k(q)$ für alle $k \in \mathbb{N}$, und $w \in L'$ ist äquivalent zu $q_0, f_w(q_0), f_w^2(q_0), ... \in F$.

Für den DFA $M' = (Q', \Sigma, \delta', q_0', F')$ verwenden wir als Zustandsmenge die Menge aller Funktionen von Q nach Q (üblicherweise als Q^Q geschrieben), also $Q' := Q^Q$. Der

Anfangszustand ist $q_0' := f_\varepsilon = \mathrm{Id}$, also die Identitätsfunktion. Wenn wir ein Zeichen $c \in \Sigma$ lesen, wollen wir von f_w zu f_{wc} übergehen, es soll also gelten

$$\delta'(f_w, c) = f_{wc} = f_c \circ f_w \qquad \text{für alle } w \in \Sigma^*$$

Der Term $f_c \circ f_w$ hängt nur von f_w (dem aktuellen Zustand) und c ab – insbesondere hängt es nicht unmittelbar von w ab. Es ist für einen Zustandsübergang also (glücklicherweise) nicht wichtig, mit welchem Wort wir den Zustand betreten haben.

Für einen Zustand $f \in Q'$ und Zeichen c definieren wir also $\delta'(f, c) := f_c \circ f$. Die Finalzustände sind nun $F' := \{f \in Q' \mid \forall i \in \mathbb{N} : f^i(q_0) \in F\}$; wie oben erläutert ist $f_w^i(q_0)$ der Zustand, den M erreicht, nachdem er das Wort w genau i-mal eingelesen hat. Wir merken an, dass es uns zwar genügen würde, die Menge F' mathematisch zu definieren, ohne dass wir angeben, wie man feststellt, ob ein bestimmter Zustand $q \in Q'$ nun tatsächlich dort enthalten wäre. Allerdings lässt sich die Eigenschaft $\forall i : f^i(q_0) \in F$ auch unschwer entscheiden: man berechnet sukzessiv die Folge $q_0, f(q_0), f^2(q_0), \ldots$ und überprüft, ob alle Zustände in F enthalten sind. Sobald sich ein Zustand wiederholt, kann man aufhören.

Abschließend beweisen wir noch formal, dass M' tatsächlich L' akzeptiert. Es gilt

$$w \in L' \;\Leftrightarrow\; \{w\}^* \subseteq L \;\Leftrightarrow\; \forall i : w^i \in L \;\Leftrightarrow\; \forall i : \hat{\delta}(q_0, w^i) \in F \;\Leftrightarrow\; \forall i : f_{w^i}(q_0) \in F$$

Da $f_{w^i} = f_w^i$ nach Definition von $\hat{\delta}$, erhalten wir also $w \in L'$ genau dann, wenn $f_w \in F'$. Es genügt also, $\hat{\delta}'(q_0', w) = f_w$ für alle $w \in \Sigma^*$ zu zeigen. Dies folgt über Induktion; als Basis erhalten wir $\hat{\delta}'(q_0', \varepsilon) = q_0' = \mathrm{Id} = f_\varepsilon$. Für den Induktionsschritt sei nun $w \in \Sigma^*$ und $c \in \Sigma$ beliebig.

$$\hat{\delta}'(q_0', wc) = \delta'(\hat{\delta}'(q_0', w), c) \overset{\text{IA}}{=} \delta'(f_w, c) = f_c \circ f_w = f_{wc}$$