TECHNISCHE UNIVERSITÄT DRESDEN

FORMALE SYSTEME

ÜBUNG 4

Eric Kunze

eric.kunze@tu-dresden.de

TU Dresden, 13. November 2021

Aufgabe 1 & 2:
Reguläre Ausdrücke

REGULÄRE AUSDRÜCKE

Die Menge der **regulärer Ausdrücke** über einem Alphabet Σ ist induktiv wie folgt definiert:

- $\emptyset$ ist ein regulärer Ausdruck
- ε ist ein regulärer Ausdruck
- $\texttt{a}$ ist ein regulärer Ausdruck für jedes $\texttt{a} \in \Sigma$
- Wenn α und β reguläre Ausdrücke sind, dann sind auch $(\alpha\beta)$, $(\alpha \mid \beta)$ und $(\alpha)^*$ reguläre Ausdrücke

Die **Sprache eines regulären Ausdrucks** α wird mit $\mathbf{L}(\alpha)$ bezeichnet und rekursiv definiert:

- $\mathbf{L}(\emptyset) = \emptyset$
- $\mathbf{L}(\varepsilon) = \{\varepsilon\}$
- $\mathbf{L}(\texttt{a}) = \{\texttt{a}\} \quad \forall \texttt{a} \in \Sigma$
- $\mathbf{L}((\alpha\beta)) = \mathbf{L}(\alpha) \circ \mathbf{L}(\beta)$
- $\mathbf{L}((\alpha \mid \beta)) = \mathbf{L}(\alpha) \cup \mathbf{L}(\beta)$
- $\mathbf{L}((\alpha)^*) = \mathbf{L}(\alpha)^*$

AUFGABE 1

Gegeben sind das Alphabet $\Sigma = \{a, b, c\}$ und die Sprache

$$L = \{w \in \Sigma^* \mid \begin{array}{l} \text{es gibt } u, v \in \Sigma^* \text{ mit } w = ubabcv \text{ und} \\ \text{es gibt } u, v \in \Sigma^* \text{ mit } w = ucccv \text{ und} \\ \text{es gibt kein } u \in \Sigma^* \text{ mit } w = au\}. \end{array}$$

Geben Sie für L einen regulären Ausdruck r mit $L = L(r)$ an.

Idee:

- die ersten beiden Bedingungen beschreiben zusammen die „Wortmitte“
- die Teile *ccc* und *babc* können in drei verschiedenen Kombinationen auftreten
- die dritte Bedingung schließt ein *a* am Anfang aus

Lösung:

$$
\begin{aligned}
r \;=\; & \Big((b \mid c)(a \mid b \mid c)^*\Big)^* \\
& \Big((babc\,(a \mid b \mid c)^*\, ccc) \;\Big|\; ccc\,(a \mid b \mid c)^*\, babc \;\Big|\; babccc\Big) \\
& \Big(a \mid b \mid c\Big)^*
\end{aligned}
$$

AUFGABE 2

Beweisen Sie die folgenden Gleichungen für reguläre Ausdrücke r, s und t.

Erinnerung: $r \equiv s$ bedeutet $L(r) = L(s)$

a) $r \mid s \equiv s \mid r$

b) $(r \mid s) \mid t \equiv r \mid (s \mid t)$

c) $(rs)t \equiv r(st)$

d) $r(s \mid t) \equiv rs \mid rt$

e) $\emptyset^* \equiv \varepsilon$

f) $(r^*)^* \equiv r^*$

g) $r^* \equiv rr^* \mid \varepsilon$

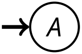
h) $(\varepsilon \mid r)^* \equiv r^*$

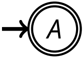
Aufgabe 3:
RegExp $\rightarrow$ *NFA*

REGEXP $\to$ NFA: KOMPOSITIONELLE METHODE

Für einen Ausdruck α definieren wir rekursiv den ε-NFA $\mathcal{M}(\alpha)$:

Grundfälle:

- Wenn $\alpha = \emptyset$ dann $\mathcal{M}(\alpha) =$ 

- Wenn $\alpha = \varepsilon$ dann $\mathcal{M}(\alpha) =$ 

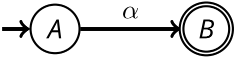
- Wenn $\alpha = \mathtt{a}$ dann $\mathcal{M}(\alpha) =$ 

Rekursive Fälle: Wir bezeichnen mit $\mathrm{elim}_\varepsilon(\mathcal{M})$ den NFA, der aus einem ε-NFA $\mathcal{M}$ durch Eliminierung der ε-Übergänge entsteht.

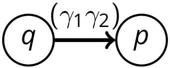
- Wenn $\alpha = (\beta\gamma)$ dann $\mathcal{M}(\alpha) = \mathrm{elim}_\varepsilon(\mathcal{M}(\beta) \odot \mathcal{M}(\gamma))$
- Wenn $\alpha = (\beta \mid \gamma)$ dann $\mathcal{M}(\alpha) = \mathcal{M}(\beta) \oplus \mathcal{M}(\gamma)$
- Wenn $\alpha = (\beta)^*$ dann $\mathcal{M}(\alpha) = \mathrm{elim}_\varepsilon(\mathcal{M}(\beta)^*)$

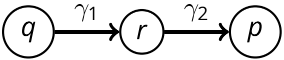
REGEXP $\rightarrow$ NFA: EXPLIZITE KONSTRUKTION

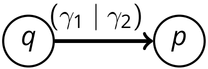
Gegeben: regulärer Ausdruck α ohne $\emptyset$

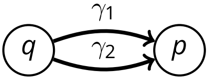
Initialisierung: $\mathcal{M}_\alpha =$ 

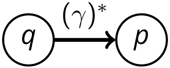
Solange es in $\mathcal{M}_\alpha$ Übergänge $q \xrightarrow{\beta} p$ gibt, die mit einem Ausdruck $\beta \notin \{\varepsilon\} \cup \Sigma$ beschriftet sind, wende eine der folgenden Regeln an:

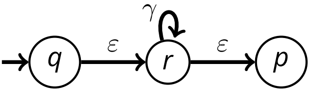
- Ersetze 

durch 

- Ersetze 

durch 

- Ersetze 

durch 

AUFGABE 3

Geben Sie zu jedem der regulären Ausdrücke r_i einen NFA $\mathcal{M}_i$ mit $L(\mathcal{M}_i) = L(r_i)$ an.

a) $r_1 = (ab)^*$

b) $r_2 = a(b \mid c)a^* \mid a^*$

Wenden Sie dabei jeweils den *kompositionellen Ansatz* sowie den *expliziten Ansatz* zur Konstruktion von NFAs aus der Vorlesung an.

Aufgabe 4:
NFA $\rightarrow$ *RegExp*

NFA $\rightarrow$ REGEXP: ERSETZUNGSMETHODE

Gegeben: NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$

Gesucht: regulärer Ausdruck α mit $\mathbf{L}(\alpha) = \mathbf{L}(\mathcal{M})$

Idee:

Für jeden Zustand $q \in Q$, berechne einen regulären Ausdruck α_q für die Sprache $\mathbf{L}(\alpha_q) = \mathbf{L}(\mathcal{M}_q)$ mit $\mathcal{M}_q = \langle Q, \Sigma, \delta, \{q\}, F\rangle$

Für $Q_0 = \{q_1, q_2, \ldots, q_n\}$ gilt dann

$$\mathbf{L}(\mathcal{M}) = \bigcup_{q \in Q_0} \mathbf{L}(\alpha_q) = \mathbf{L}(\alpha_{q_1} \mid \alpha_{q_2} \mid \cdots \mid \alpha_{q_n})$$

(1) **Vereinfache den Automaten** (entferne offensichtlich unnötige Zustände)

(2) **Bestimme das Gleichungssystem**

Intuition: Beschreibe α_q in Abhängigkeit von Folgezuständen

- ▷ Für jeden Zustand $q \in Q \setminus F$: $\alpha_q \equiv \sum_{\mathtt{a} \in \Sigma} \sum_{p \in \delta(q,\mathtt{a})} \mathtt{a}\alpha_p$
- ▷ Für jeden Zustand $q \in F$: $\alpha_q \equiv \varepsilon \mid \sum_{\mathtt{a} \in \Sigma} \sum_{p \in \delta(q,\mathtt{a})} \mathtt{a}\alpha_p$

(3) **Löse das Gleichungssystem** durch Einsetzen und

Regel von Arden: Aus $\alpha \equiv \beta\alpha \mid \gamma$ mit $\varepsilon \notin \mathbf{L}(\beta)$ folgt $\alpha \equiv \beta^*\gamma$.

(4) **Gib den Ausdruck für die Sprache des NFA an**

Für $Q_0 = \{q_1, q_2, \dots, q_n\}$ gilt dann

$$\mathbf{L}(\mathcal{M}) = \bigcup_{q \in Q_0} \mathbf{L}(\alpha_q) = \mathbf{L}(\alpha_{q_1} \mid \alpha_{q_2} \mid \cdots \mid \alpha_{q_n})$$

NFA $\rightarrow$ REGEXP: DYNAMISCHE ERMITTLUNG

Gegeben: NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$

Gesucht: regulärer Ausdruck α mit $\mathbf{L}(\alpha) = \mathbf{L}(\mathcal{M})$

Ansatz:

Für jedes Paar von Zuständen $q, p \in Q$, berechne einen regulären Ausdruck $\alpha_{q,p}$ für die Sprache $\mathbf{L}(\alpha_{q,p}) = \mathbf{L}(\mathcal{M}_{q,p})$ mit $\mathcal{M}_{q,p} = \langle Q, \Sigma, \delta, \{q\}, \{p\} \rangle$

Dann gilt:

$$\mathbf{L}(\mathcal{M}) = \bigcup_{q \in Q_0} \bigcup_{p \in F} \mathbf{L}(\alpha_{q,p}) = \mathbf{L}\left(\sum_{q \in Q_0} \sum_{p \in F} \alpha_{q,p} \right)$$

- $\mathbf{L}^k[i,j] \ldots$ Sprache mit Start in q_i, Ende in q_j und nutzt nur Zwischenzustände $q_1, \ldots, q_k$
- $\alpha^k[i,j]$ zugehöriger regulärer Ausdruck

Idee: lasse immer mehr Zwischenzustände von q nach p zu

- $k = n$: nutze alle Zustände — Ergebnis ablesbar
- $k = 0$: nutze keine Zwischenzustände — $\alpha^0[i,j]$ direkt ablesbar:

Sei $\{\mathtt{a}_1, \ldots, \mathtt{a}_m\} = \{\mathtt{a} \in \Sigma \mid q_i \xrightarrow{\mathtt{a}} q_j\}$ die Menge der Beschriftungen von direkten Übergängen von q_i zu q_j.

- Falls $i \neq j$, dann ist $\alpha^0[i,j] = \mathtt{a}_1 \mid \ldots \mid \mathtt{a}_m$
- Falls $i = j$, dann ist $\alpha^0[i,j] = \mathtt{a}_1 \mid \ldots \mid \mathtt{a}_m \mid \varepsilon$

Update-Formel:

$$\alpha^{k+1}[i,j] = \alpha^k[i,j] \mid \left(\alpha^k[i,k+1](\alpha^k[k+1,k+1])^*\alpha^k[k+1,j]\right)$$

vgl. VL „Algorithmen & Datenstrukturen", Prozess-Problem im Aho-Hopcroft-Ullman-Algorithmus

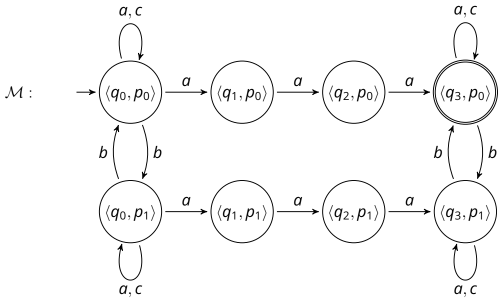
AUFGABE 4

Entwickeln Sie für die Sprache L über dem Alphabet $\Sigma = \{a, b, c\}$ einen regulären Ausdruck r mit $L = L(r)$. Für alle Wörter $w \in L$ gilt:

- w enthält aaa.
- w endet mit c.
- Die Anzahl der b in w ist gerade.

Idee: $L = \underbrace{\left(\{w \in \Sigma^* : w \text{ enthält } aaa\} \cap \left\{w \in \Sigma^* : |w|_b \text{ gerade}\right\}\right)}_{:=L(\mathcal{M})} \cdot \{c\}$

- Konstruiere Automaten für die erste und dritte Bedingung
- Produktautomat $\mathcal{M}$ erkennt die Sprache ohne c am Ende



- Umwandlung in regulären Ausdruck $\alpha_{\mathcal{M}}$ (z.B. via Lemma von Arden)

$$\alpha_{\mathcal{M}} = \Big(a \mid c \mid b\,(a|c)^*\,b\Big)^* \\ \Big(aaa(a \mid c \mid b\,(a|c)^*\,b)^* \,\Big|\, b(a|c)^* aaa(a|c)^* b\,\big(a \mid c \mid b(a|c)^* b\big)^*\Big)$$

- Mit $r = \alpha_{\mathcal{M}} c$ gilt dann $L(r) = L$.

Aufgabe 5:
Minimierung von Automaten

ÄQUIVALENZ VON ZUSTÄNDEN & QUOTIENTENAUTOMAT

DFA $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F\rangle \rightsquigarrow$ DFA $\mathcal{M}_q = \langle Q, \Sigma, \delta, q, F\rangle$

Äquivalenz von Zuständen:	$p \sim_{\mathcal{M}} q \quad \Leftrightarrow \quad \mathbf{L}(\mathcal{M}_p) = \mathbf{L}(\mathcal{M}_q)$
Äquivalenzklasse:	$[q]_\sim = \{p \in Q \mid q \sim p\}$
Quotient von $P \subseteq Q$:	$P/_\sim = \{[p]_\sim \mid p \in P\}$

Quotientenautomat: Verschmelzen von äquivalenten Zuständen

Für einen DFA $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F\rangle$ mit totaler Übergangsfunktion ist der Quotientenautomat $\mathcal{M}/_\sim = \langle Q/_\sim, \Sigma, \delta_\sim, [q_0]_{\sim_\mathcal{M}}, F/_\sim\rangle$ gegeben durch

- $Q/_\sim = \{[q]_\sim \mid q \in Q\}$
- $\delta_\sim([q]_\sim, \mathtt{a}) = [\delta(q, \mathtt{a})]_\sim$
- $F/_\sim = \{[q]_\sim \mid q \in F\}$

Bestimmung von $\sim$:

- Initialisiere $\not\sim \; := \emptyset$
- **Regel 1**: Für jedes Paar von Zuständen $\langle q, p \rangle \in Q \times Q$:
  falls $q \in F$ und $p \notin F$, dann „speichere $q \not\sim p$"
- **Regel 2**: Für jedes Paar $\langle q, p \rangle \in Q \times Q \setminus \not\sim$ und jedes $\mathtt{a} \in \Sigma$:
  falls $\delta(q, \mathtt{a}) \not\sim \delta(p, \mathtt{a})$ dann „speichere $q \not\sim p$"
- Wiederhole Regel 2 bis keine Änderungen mehr auftreten
- Das Ergebnis ist $(Q \times Q) \setminus \not\sim$

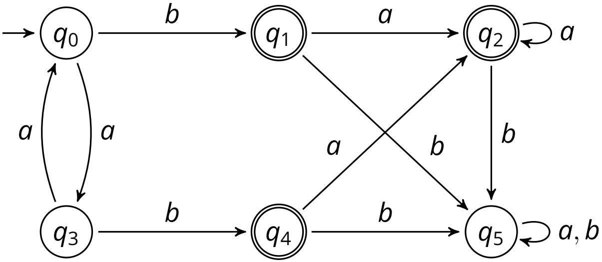
Beispiel: Für einen DFA mit Zuständen $Q = \{A, B, C, D, E\}$ genügt eine Tabelle mit zehn Feldern (statt $5^2 = 25$).

(dazu reihen wir Zustände vertikal in umgekehrter Reihenfolge)

	A	B	C	D
E				
D				
C				
B				

AUFGABE 5

Berechnen Sie für folgenden DFA
$\mathcal{M} = (\{q_0, q_1, q_2, q_3, q_4, q_5\}, \{a, b\}, \delta, q_0, \{q_1, q_2, q_4\})$ mit δ:



die Äquivalenzrelation $\sim_{\mathcal{M}}$, und geben Sie den Quotientenautomaten $\mathcal{M}/_{\sim}$ an.

$$
\begin{aligned}
\sim\, = & \{\langle q_1, q_2\rangle, \langle q_2, q_4\rangle, \langle q_1, q_4\rangle, \langle q_0, q_3\rangle\} \\
\cup & \ \{\langle q_2, q_1\rangle, \langle q_4, q_2\rangle, \langle q_4, q_1\rangle, \langle q_3, q_0\rangle\} \\
\cup & \ \{\langle q, q\rangle : q \in Q\}
\end{aligned}
$$

$$
Q/_\sim = \Big\{ \underbrace{\{q_0, q_3\}}_{=[q_0]_\sim}, \underbrace{\{q_1, q_2, q_4\}}_{=[q_1]_\sim}, \underbrace{\{q_5\}}_{=[q_5]_\sim} \Big\}
$$