Theoretische Informatik 1

WiSe 23/24

Prof. Dr. Sebastian Siebertz
AG Theoretische Informatik
MZH, Raum 3160
siebertz@uni-bremen.de

Universität
Bremen

Wiederholung Pumping Lemma

- Pumping-Lemma für reguläre Sprachen:

  Wenn L eine reguläre Sprache ist, dann gilt:

  es existiert $n_0 \geq 1$, so dass

  für alle $w \in L$ mit $|w| \geq n_0$ gilt:

  es existiert eine Zerlegung $w = xyz$ mit $y \neq \varepsilon$ und $|xy| \leq n_0$, so dass

  für alle $k \geq 0$ gilt $xy^k z \in L$.

- Pumping-Lemma als Kontraposition:

  Sei L eine Sprache. Wenn

  für alle $n_0 \geq 1$

  existiert ein $w \in L$ mit $|w| \geq n_0$, so dass

  für alle Zerlegungen $w = xyz$ mit $y \neq \varepsilon$ und $|xy| \leq n_0$ gilt:

  es existiert $k \geq 0$ mit $xy^k z \notin L$.

  Dann ist L **nicht** regulär.

Reguläre Ausdrücke

- Wir kennen bereits 4 äquivalente Charakterisierungen der Klasse der regulären Sprachen.

> Eine Sprache $L \subseteq \Sigma^*$ ist regulär gdw.
>
> (1) $L = L(\mathcal{A})$ für einen DEA $\mathcal{A}$.
> (2) $L = L(\mathcal{A})$ für einen NEA $\mathcal{A}$.
> (3) $L = L(\mathcal{A})$ für einen ε-NEA $\mathcal{A}$.
> (4) $L = L(\mathcal{A})$ für einen NEA mit Wortübergängen $\mathcal{A}$.

- Im Folgenden betrachten wir eine weitere nützliche Charakterisierung mittels regulärer Ausdrücke.

Reguläre Ausdrücke

- Sei Σ ein endliches Alphabet.
- Die Menge Reg_Σ der regulären Ausdrücke über Σ ist induktiv definiert:
  - $\varnothing$, ε, a (für $a \in \Sigma$) sind Elemente von Reg_Σ.
  - Sind $r, s \in \mathrm{Reg}_\Sigma$, so auch
    - $(r + s) \in \mathrm{Reg}_\Sigma$,
    - $(r \cdot s) \in \mathrm{Reg}_\Sigma$,
    - $r^* \in \mathrm{Reg}_\Sigma$.
- Beispiele
  - $((a \cdot b^*) + (a + b)^*)^* \in \mathrm{Reg}_\Sigma$ für $\Sigma = \{a, b\}$.
  - $(((a \cdot b)^* + ((c \cdot b) \cdot a)^*) + a^*) \in \mathrm{Reg}_\Sigma$ für $\Sigma = \{a, b, c\}$.

Notation reguläre Ausdrücke

- Wir vereinbaren:
  - * bindet stärker als $\cdot$,
  - $\cdot$ bindet stärker als $+$,
  - $\cdot$ wird ganz weggelassen.
- Statt $((a \cdot b^*) + (a + b)^*)^*$ schreiben wir also $(ab^* + (a + b))^*)^*$.
- Wir werden gleich sehen: $+$ und $\cdot$ sind assoziativ, d. h.

$$((\alpha + \beta) + \gamma) \text{ und } (\alpha + (\beta + \gamma))$$

beschreiben dieselbe Sprache und

$$((\alpha' \cdot \beta') \cdot \gamma') \text{ und } (\alpha' \cdot (\beta' \cdot \gamma'))$$

ebenfalls.

- Also lassen wir auch hier Klammern weg und schreiben $\alpha + \beta + \gamma$, bzw. $\alpha'\beta'\gamma'$.
- Statt $(((a \cdot b)^* + ((c \cdot b) \cdot a)^*) + a^*)$ also $(ab)^* + (cba)^* + a^*$.

Semantik regulärer Ausdrücke

- Die durch einen regulären Ausdruck r definierte Sprache $L(r)$ ist induktiv definiert:
  - $L(\varnothing) := \varnothing$
  - $L(\varepsilon) := \{\varepsilon\}$
  - $L(a) := \{a\}$
  - $L(r + s) := L(r) \cup L(s)$
  - $L(r \cdot s) := L(r) \cdot L(s)$
  - $L(r^*) := L(r)^*$
- Beispiele:
  - $L((a+b)^* ab (a+b)^*) = \{w \in \{a,b\}^* : w \text{ hat Infix } ab\}$.
  - $L(ab^* + b) = \{ab^i \mid i \geq 0\} \cup \{b\}$.

Notation

- Statt $L(r)$ schreiben wir im Folgenden häufig einfach r.
- Beispiel
  - $(ab)^*a = a(ba)^*$ statt $L((ab)^*a) = L(a(ba)^*)$.
  - $L(\mathcal{A}) = ab^* + b$ statt $L(\mathcal{A}) = L(ab^* + b)$.

Der Satz von Kleene

Satz von Kleene

Für eine Sprache $L \subseteq \Sigma^*$ sind äquivalent:

1) Es gibt einen regulären Ausdruck r mit $L = L(r)$.
2) L ist regulär.

- Was müssen wir zeigen?

  1) $\Rightarrow$ 2): Angenommen $L = L(r)$ für einen regulären Ausdruck r.

  Zeige: es existiert ein DEA $\mathcal{A}$, NEA $\mathcal{A}$, ε-NEA $\mathcal{A}$, oder Wort-NEA $\mathcal{A}$ mit $L = L(\mathcal{A})$.

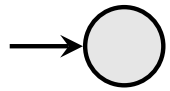
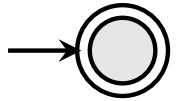
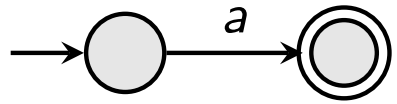
  2) $\Rightarrow$ 1): Angenommen $L = L(\mathcal{A})$ für einen NEA $\mathcal{A}$.

  Idee: Erzeuge aus $\mathcal{A}$ einen regulären Ausdruck r mit $L(r) = L$.

Beweis Satz von Kleene

1) $\Rightarrow$ 2): Beweis per Induktion über den Aufbau regulärer Ausdrücke.

Induktionsanfang:

- $L(\varnothing) = \varnothing$ regulär:  ist NEA für $\varnothing$.
- $L(\varepsilon) = \{\varepsilon\}$ regulär:  ist NEA für $\{\varepsilon\}$.
- $L(a) = \{a\}$ regulär: 

ist NEA für $\{a\}$.

Induktionsschritt:

- $L(r)$ und $L(s)$ regulär $\Rightarrow$ $L(r+s)$, $L(r \cdot s)$ und $L(r^*)$ regulär:
  - $L(r+s) = L(r) \cup L(s)$ (Abschluss unter Vereinigung)
  - $L(r \cdot s) = L(r) \cdot L(s)$ (Abschluss unter Konkatenation)
  - $L(r^*) = L(r)^*$ (Abschluss unter Kleene Stern)

Beweis Satz von Kleene

2) $\Rightarrow$ 1):

- Sei $\mathcal{A} = (Q, \Sigma, q_s, \Delta, F)$ ein NEA mit $L = L(\mathcal{A})$.
- Für $p, q \in Q$ und $X \subseteq Q$ sei $L^X_{p,q}$ die Sprache aller Wörter $w = a_1 \cdots a_n$, für die es einen Pfad

$$p_0 \xrightarrow{a_1}_{\mathcal{A}} p_1 \xrightarrow{a_2}_{\mathcal{A}} \cdots \xrightarrow{a_{n-1}}_{\mathcal{A}} p_{n-1} \xrightarrow{a_n}_{\mathcal{A}} p_n$$

  gibt mit $p_0 = p$, $p_n = q$ und $\{p_1, \ldots, p_{n-1}\} \subseteq X$.
- Es gilt:

$$L(\mathcal{A}) = \bigcup_{q_f \in F} L^Q_{q_s, q_f}.$$

- Wir zeigen, dass für alle Sprachen $L^X_{p,q}$ ein regulärer Ausdruck existiert.

Beweis Satz von Kleene

- Wir zeigen, dass für alle Sprachen $L_{p,q}^{X}$ ein regulärer Ausdruck existiert.
- Beweis per Induktion über $|X|$.
- Induktionsanfang: $X = \varnothing$.
  - 1. Fall: $p \neq q$

    $L_{p,q}^{\varnothing} = \{a \in \Sigma \mid (p, a, q) \in \Delta\} = \{a_1, \ldots, a_k\}$ für gewisse $a_i \in \Sigma$ und der entsprechende reguläre Ausdruck ist $a_1 + \cdots + a_k$.
  - 2. Fall: $p = q$

    $L_{p,q}^{\varnothing} = \{a \in \Sigma \mid (p, a, q) \in \Delta\} \cup \{\varepsilon\} = \{a_1, \ldots, a_k\} \cup \{\varepsilon\}$ für gewisse $a_i \in \Sigma$ und der entsprechende reguläre Ausdruck ist $a_1 + \cdots + a_k + \varepsilon$.

Beweis Satz von Kleene

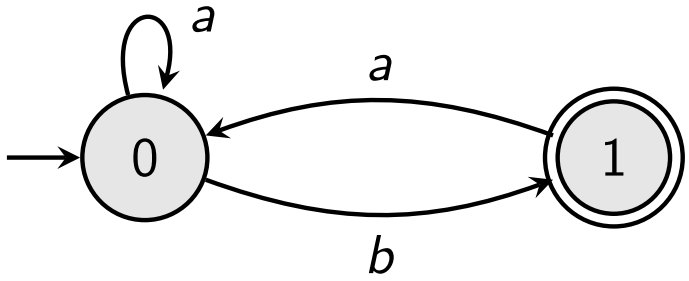
- Induktionsschritt: $X \neq \emptyset$.
- Wähle ein beliebiges $\hat{q} \in X$.
- Dann gilt:

$$L_{p,q}^{X} = L_{p,q}^{X \setminus \{\hat{q}\}} \cup \left(L_{p,\hat{q}}^{X \setminus \{\hat{q}\}} \cdot \left(L_{\hat{q},\hat{q}}^{X \setminus \{\hat{q}\}} \right)^{*} \cdot L_{\hat{q},q}^{X \setminus \{\hat{q}\}} \right).$$

- Für die Sprachen $L_{p,q}^{X \setminus \{\hat{q}\}}$, $L_{p,\hat{q}}^{X \setminus \{\hat{q}\}}$, $L_{\hat{q},\hat{q}}^{X \setminus \{\hat{q}\}}$ und $L_{\hat{q},q}^{X \setminus \{\hat{q}\}}$ gibt es nach Induktionsvoraussetzung reguläre Ausdrücke und die regulären Ausdrücke sind unter Vereinigung, Konkatenation und Kleene Stern abgeschlossen.

□

Beispiel



- Da 1 der einzige akzeptierende Zustand ist, gilt $L(\mathcal{A}) = L_{0,1}^{Q}$.
- Induktionsschritte:

$$\begin{aligned} L_{0,1}^{Q} &= L_{0,1}^{\{0\}} \cup L_{0,1}^{\{0\}} \cdot (L_{1,1}^{\{0\}})^* \cdot L_{1,1}^{\{0\}} \\ &= a^*b + a^*b \cdot (\varepsilon + aa^*b)^* \cdot (\varepsilon + aa^*b) = a^*b(aa^*b)^* \end{aligned}$$

$$L_{0,1}^{\{0\}} = L_{0,1}^{\varnothing} \cup L_{0,0}^{\varnothing} \cdot (L_{0,0}^{\varnothing})^* \cdot L_{0,1}^{\varnothing} = b + (a+\varepsilon) \cdot (a+\varepsilon)^* \cdot b = a^*b$$

$$L_{1,1}^{\{0\}} = L_{1,1}^{\varnothing} \cup L_{1,0}^{\varnothing} \cdot (L_{0,0}^{\varnothing})^* \cdot L_{0,1}^{\varnothing} = \varepsilon + a \cdot (a+\varepsilon)^* \cdot b = \varepsilon + aa^*b$$

- Induktionsanfang:

$$L_{0,1}^{\varnothing} = b \quad L_{0,0}^{\varnothing} = a + \varepsilon \quad L_{1,1}^{\varnothing} = \varepsilon \quad L_{1,0}^{\varnothing} = a$$

Bemerkungen zu regulären Ausdrücken

- Zur Größe der konstruierten Ausdrücke/Automaten:
  - Die Konstruktion „Ausdruck → NEA“ ist polynomiell.
  - Die Konstruktion „NEA → Ausdruck“ ist im Allgemeinen exponentiell.
  - Es kann gezeigt werden, dass dies nicht vermeidbar ist.
- Für alle regulären Ausdrücke r und s
  - gibt es einen Ausdruck t mit $L(t) = L(r) \cap L(s)$;
  - gibt es einen Ausdruck t' mit $L(t') = \overline{L(r)}$.
  - Es ist schwierig, diese Ausdrücke direkt aus r und s (also ohne den Umweg über Automaten) zu konstruieren.

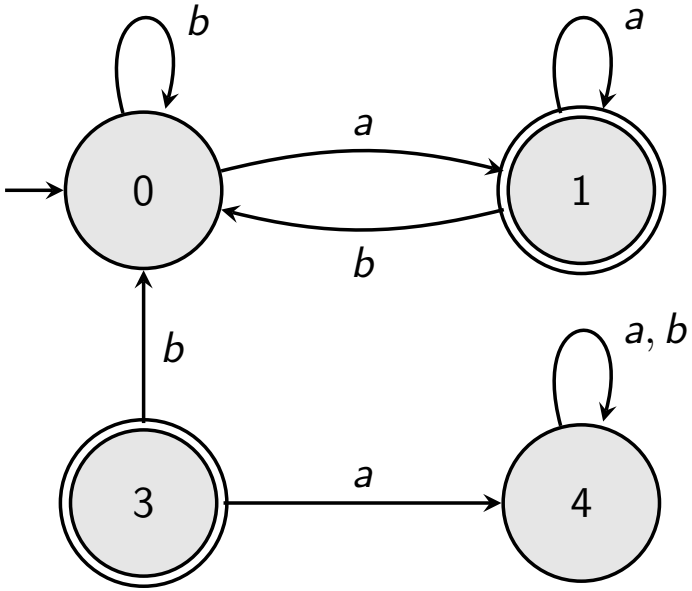
Zweites Thema für heute: minimale DEAs

- Ziel: zu gegebenem DEA einen äquivalenten DEA mit minimaler Zustandszahl konstruieren.
- Zwei Schritte:
  - Eliminieren von Zuständen die nicht erreichbar sind.
  - Zusammenfassen äquivalenter Zustände.

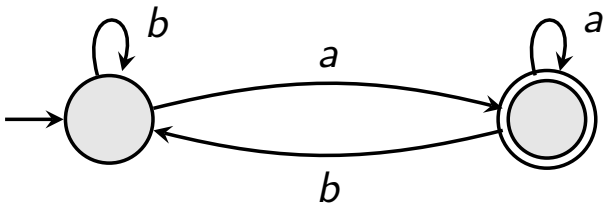
Minimale DEAs

- Ein Zustand p eines DEA $\mathcal{A} = (Q, \Sigma, q_s, \delta, F)$ heißt erreichbar, falls es ein Wort $w \in \Sigma^*$ gibt mit $\hat{\delta}(q_s, w) = p$.
- Für die erkannte Sprache sind nur Zustände wichtig, die von q_s erreichbar sind.
- Durch Weglassen aller anderen Zustände erhalten wir einen äquivalenten Automaten: $\mathcal{A}_0 = (Q_0, \Sigma, Q_0, \delta_0, F_0)$ mit
  - $Q_0 = \{q \in Q \mid q \text{ ist erreichbar}\}$
  - $\delta_0 : Q_0 \times \Sigma \to Q_0 : (q, a) \mapsto \delta(q, a)$ (δ_0 wie δ, aber eingeschränkt auf Q_0)
  - $F_0 = F \cap Q_0$

Beispiel

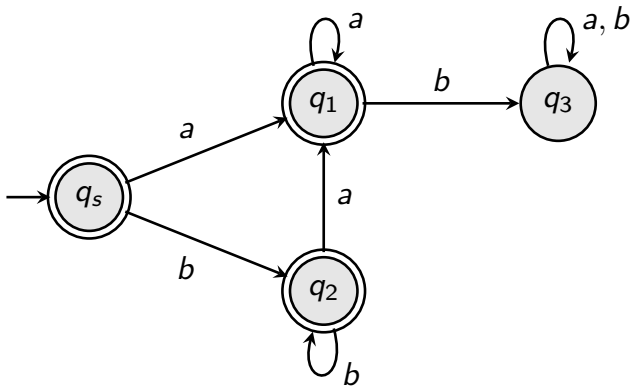


- Die Zustände 3 und 4 sind nicht erreichbar.
- Durch Weglassen dieser Zustände erhalten wir den äquivalenten DEA $\mathcal{A}_0$:

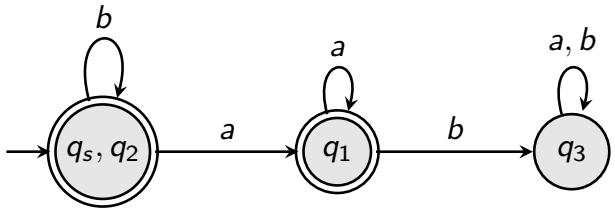


Minimale DEAs

- Ein DEA ohne unerreichbare Zustände muss noch nicht minimal sein, da er noch verschiedene Zustände enthalten kann, die sich „gleich“ verhalten.



- q_s und q_2 sind äquivalent.



Äquivalenzrelationen

- Eine Relation $R \subseteq M \times M$ heißt Äquivalenzrelation, wenn R
  - reflexiv ist: $(x, x) \in R$ für all $x \in M$,
  - symmetrisch ist: $(x, y) \in R$ impliziert $(y, x) \in R$,
  - transitiv ist: $(x, y) \in R$ und $(y, z) \in R$ impliziert $(x, z) \in R$.

  Die Äquivalenzklasse von $x \in M$ bezüglich R ist $[x] = \{y \in M : (x, y) \in R\}$.

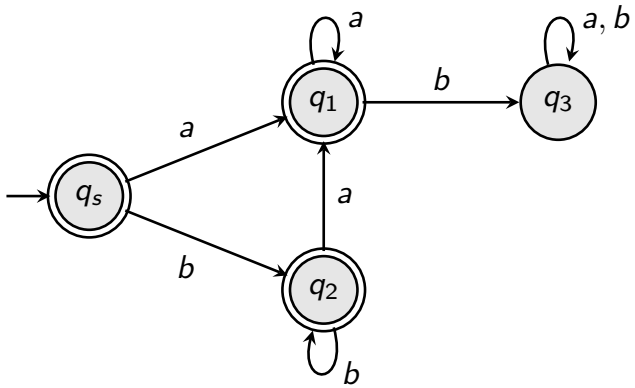
- Äquivalenzrelationen beschreiben die Gleichwertigkeit von Objekten.
- Die Äquivalenzklassen partitionieren die Grundmenge M in disjunkte Teilmengen.
- Jede Äquivalenzklasse ist eine Menge von Objekten, die sich gleich verhalten.

Äquivalenz von Zuständen

- Es sei $\mathcal{A} = (Q, \Sigma, q_s, \delta, F)$ ein DEA.
- Für $q \in Q$ sei $\mathcal{A}_q = (Q, \Sigma, q, \delta, F)$ der Automat mit Startzustand q
- Zwei Zustände $q, q' \in Q$ heißen $\mathcal{A}$-äquivalent ($q \sim_{\mathcal{A}} q'$) wenn

$$L(\mathcal{A}_q) = L(\mathcal{A}_{q'}).$$

- Beispiel:



- $q_s \sim_{\mathcal{A}} q_2$, da $L(\mathcal{A}_{q_s}) = b^* a^* = L(\mathcal{A}_{q_2})$.
- $q_s \not\sim_{\mathcal{A}} q_1$, da $b \in L(\mathcal{A}_{q_s}) \setminus L(\mathcal{A}_{q_1})$.

Äquivalenz von Zuständen

- Lemma
  1) $\sim_{\mathcal{A}}$ ist eine Äquivalenzrelation auf Q.
  2) $\sim_{\mathcal{A}}$ ist verträglich mit der Übergangsfunktion, d. h.

$$q \sim_{\mathcal{A}} q' \Rightarrow \forall a \in \Sigma : \delta(q, a) \sim_{\mathcal{A}} \delta(q', a)$$

Beweis.

1) ist klar, da die Relation „=“ reflexiv, transitiv und symmetrisch ist.

Äquivalenz von Zuständen

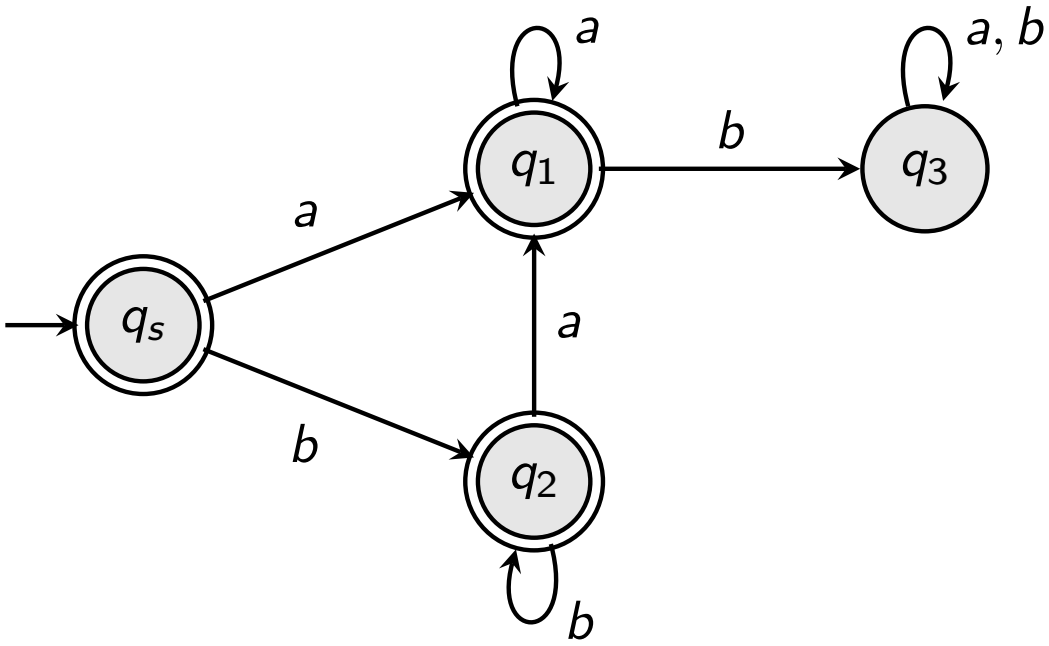
2) $(q \sim_{\mathcal{A}} q' \Rightarrow \forall a \in \Sigma : \delta(q,a) \sim_{\mathcal{A}} \delta(q',a))$:

$$
\begin{aligned}
q \sim_{\mathcal{A}} q' &\Leftrightarrow L(\mathcal{A}_q) = L(\mathcal{A}_{q'}) \\
&\Leftrightarrow \forall w \in \Sigma^* : \hat{\delta}(q,w) \in F \Leftrightarrow \hat{\delta}(q',w) \in F \\
&\Rightarrow \forall a \in \Sigma\ \forall v \in \Sigma^* : \hat{\delta}(q,av) \in F \Leftrightarrow \hat{\delta}(q',av) \in F \\
&\Leftrightarrow \forall a \in \Sigma\ \forall v \in \Sigma^* : \hat{\delta}(\delta(q,a),v) \in F \Leftrightarrow \hat{\delta}(\delta(q',a),v) \in F \\
&\Leftrightarrow \forall a \in \Sigma : L(\mathcal{A}_{\delta(q,a)}) = L(\mathcal{A}_{\delta(q',a)}) \\
&\Leftrightarrow \forall a \in \Sigma : \delta(q,a) \sim_{\mathcal{A}} \delta(q',a).
\end{aligned}
$$

Berechnung der Äquivalenzrelation

- Die Relation $\sim_{\mathcal{A}}$ kann wie folgt berechnet werden.
- Definiere eine Folge von Relationen $\sim_0, \sim_1, \sim_2, \ldots$
  - $q \sim_0 q' \quad \Leftrightarrow \quad (q \in F \Leftrightarrow q' \in F)$
  - $q \sim_{k+1} q' \quad \Leftrightarrow \quad q \sim_k q'$ und $\forall a \in \Sigma : \delta(q, a) \sim_k \delta(q', a)$
- Es gilt $Q \times Q \supseteq \sim_0 \supseteq \sim_1 \supseteq \sim_2 \supseteq \ldots$.
- Da Q endlich ist, gibt es ein $k \geq 0$ mit $\sim_k = \sim_{k+1}$.
- Wir zeigen, dass $\sim_k$ die gewünschte Relation $\sim_{\mathcal{A}}$ ist.

Beispiel



- $Q \times Q = \{(q_s, q_s), (q_s, q_1), \ldots, (q_3, q_3)\}$.
- $\sim_0 = \{(q_s, q_s), (q_s, q_1), (q_1, q_s), (q_s, q_2), (q_2, q_s), (q_1, q_1),$
  $(q_1, q_2), (q_2, q_1), (q_2, q_2)\} \cup \{(q_3, q_3)\}$.
- $\sim_1 = \{(q_s, q_s), (q_s, q_2), (q_2, q_s), (q_2, q_2)\} \cup \{(q_1, q_1)\} \cup \{(q_3, q_3)\}$.
- $\sim_2 = \sim_1 = \sim_{\mathcal{A}}$.

Berechnung der Äquivalenzrelation

Hilfsaussage

Für alle $k \geq 0$ gilt: $q \sim_k q'$ genau dann, wenn für alle $w \in \Sigma^*$ mit $|w| \leq k$:

$$w \in L(\mathcal{A}_q) \Leftrightarrow w \in L(\mathcal{A}_{q'}).$$

Beweis per Induktion über $|w|$.

- Induktionsanfang: Nach Def. von $\sim_0$ gilt $q \sim_0 q'$ genau dann, wenn $(q \in F \Leftrightarrow q' \in F) \Leftrightarrow (\varepsilon \in L(\mathcal{A}_q) \Leftrightarrow \varepsilon \in L(\mathcal{A}_{q'}))$.
- Induktionsschritt:

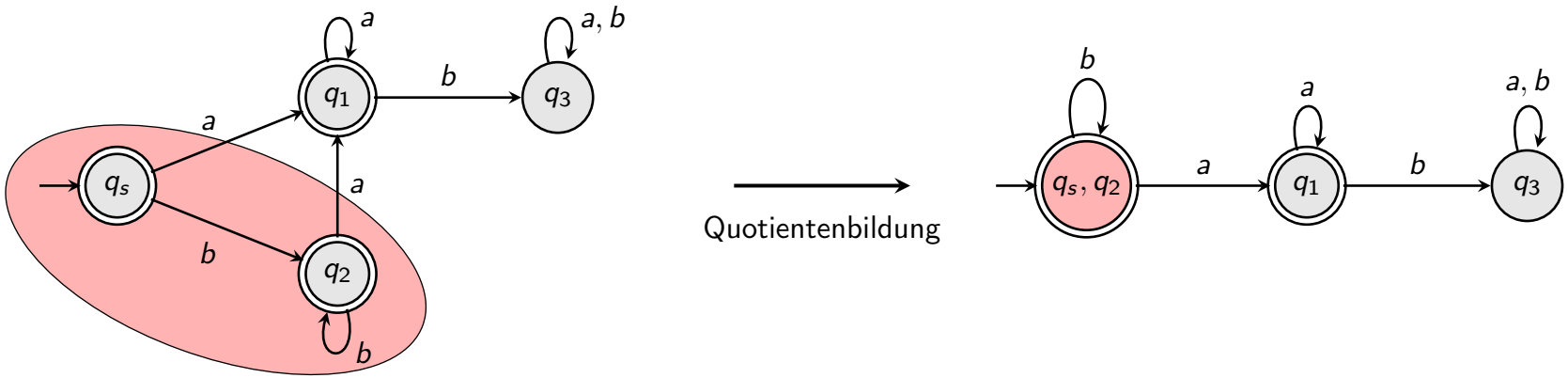
$$\begin{aligned} q \sim_{k+1} q' &\Leftrightarrow q \sim_k q' \text{ und } \forall a \in \Sigma : \delta(q,a) \sim_k \delta(q',a) \\ &\Leftrightarrow \forall w \in \Sigma^* \text{ mit } |w| \leq k : w \in L(\mathcal{A}_q) \Leftrightarrow w \in L(\mathcal{A}_{q'}) \text{ und} \\ &\quad \forall a \in \Sigma : \forall w \in \Sigma^* \text{ mit } |w| \leq k : w \in L(\mathcal{A}_{\delta(q,a)}) \Leftrightarrow w \in L(\mathcal{A}_{\delta(q',a)}) \\ &\Leftrightarrow \forall w \in \Sigma^* \text{ mit } |w| \leq k+1 : w \in L(\mathcal{A}_q) \Leftrightarrow w \in L(\mathcal{A}_{q'}). \end{aligned}$$

Berechnung der Äquivalenzrelation

- Wir zeigen: wenn $\sim_k = \sim_{k+1}$, dann $\sim_k = \sim_{\mathcal{A}}$.
- $\sim_{\mathcal{A}} \subseteq \sim_k$:
  - $q \sim_{\mathcal{A}} q' \Leftrightarrow$ für alle $w \in \Sigma^*$: $w \in L(\mathcal{A}_q) \Leftrightarrow w \in L(\mathcal{A}_{q'})$.
  - $q \sim_k q' \Leftrightarrow$ für alle $w \in \Sigma^*$ mit $|w| \leq k$: $w \in L(\mathcal{A}_q) \Leftrightarrow w \in L(\mathcal{A}_{q'})$.
  - Also $q \sim_{\mathcal{A}} q' \Rightarrow q \sim_k q'$, d.h. $\sim_{\mathcal{A}} \subseteq \sim_k$.
- $\sim_k \subseteq \sim_{\mathcal{A}}$:
  - Angenommen $\sim_k \not\subseteq \sim_{\mathcal{A}}$.
  - Wähle q, q' mit $q \sim_k q'$ und $q \not\sim_{\mathcal{A}} q'$.
  - Es gibt also ein $w \in \Sigma^*$ mit $w \in L(\mathcal{A}_q)$ und $w \notin L(\mathcal{A}_{q'})$.
  - Mit Hilfsaussage folgt $q \not\sim_n q'$ für $n = |w|$.
  - Da $\sim_k \subseteq \sim_i$ für alle $i \geq 0$ folgt $q \not\sim_k q'$, ein Widerspruch.

Der Quotientenautomat

- Der Quotientenautomat $\widetilde{\mathcal{A}} = (\widetilde{Q}, \Sigma, [q_s]_{\mathcal{A}}, \widetilde{\delta}, \widetilde{F})$ zu $\mathcal{A} = (Q, \Sigma, q_s, \delta, F)$ ist definiert durch:
  - $\widetilde{Q} := \{[q]_{\mathcal{A}} \mid q \in Q\}$
  - $\widetilde{\delta}([q]_{\mathcal{A}}, a) := [\delta(q, a)]_{\mathcal{A}}$ (repräsentantenunabhängig!)
  - $\widetilde{F} := \{[q]_{\mathcal{A}} \mid q \in F\}$



Der Quotientenautomat

Satz

$\widetilde{\mathcal{A}}$ ist äquivalent zu $\mathcal{A}$.

Beweis.

- Es folgt leicht per Induktion über $|w|$:

$$\hat{\widetilde{\delta}}([q_s]_{\mathcal{A}}, w) = [\hat{\delta}(q_s, w)]_{\mathcal{A}} \text{ für alle } w \in \Sigma^*. \qquad (*)$$

- Nun gilt:

$$\begin{aligned} w \in L(\mathcal{A}) &\Leftrightarrow \hat{\delta}(q_s, w) \in F \\ &\Leftrightarrow [\hat{\delta}(q_s, w)]_{\mathcal{A}} \in \widetilde{F} \quad (\text{Def. } \widetilde{F}) \\ &\Leftrightarrow \hat{\widetilde{\delta}}([q_0]_{\mathcal{A}}, w) \in \widetilde{F} \quad (*) \\ &\Leftrightarrow w \in L(\widetilde{\mathcal{A}}). \end{aligned}$$

Der Quotientenautomat

- Für einen DEA $\mathcal{A}$ bezeichnet $\mathcal{A}_{\text{red}}$ den reduzierten Automaten, den wir aus $\mathcal{A}$ durch Eliminieren unerreichbarer Zustände und nachfolgendes Bilden des Quotientenautomaten erhalten.
- $\mathcal{A}_{\text{red}}$ kann nicht weiter vereinfacht werden.
- Wir werden zeigen, dass $\mathcal{A}_{\text{red}}$ der kleinste DEA ist, der $L(\mathcal{A})$ akzeptiert.
- Um dies zu zeigen werden wir die Nerode-Rechtskongruenz verwenden, die auch von unabhängigem Interesse ist.
  - Charakterisierung der regulären Sprachen ohne Automaten.