TECHNISCHE UNIVERSITÄT DRESDEN

FORMALE SYSTEME

9. Vorlesung: Minimale Automaten (2)

Markus Krötzsch

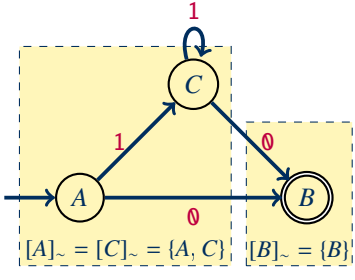
Professur für Wissensbasierte Systeme

TU Dresden, 23. November 2020

Rückblick

Automaten verkleinern mit Quotientenbildung

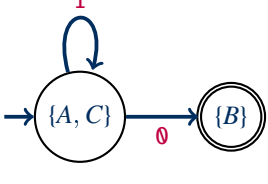
Wir betrachten DFAs mit totaler Übergangsfunktion.



Ermittlung der Zustandsäquivalenz $\sim$:

	A	B
C		ϵ
B	ϵ	

Quotientenautomat:



Sei $\mathcal{M}$ ein DFA mit totaler Übergangsfunktion. Der reduzierte Automat $\mathcal{M}_r$ ergibt sich durch folgende Schritte:

(1) Entferne alle unerreichbaren Zustände aus $\mathcal{M}$

(2) Berechne den Quotientenautomaten

Verschiedene Äquivalenzrelationen

Für einen DFA $\mathcal{M}$ und eine Sprache **L** haben wir verschiedene Äquivalenzrelationen verwendet:

DFA-Zustandsäquivalenz $\sim_{\mathcal{M}} \subseteq Q \times Q$:
$q \sim_{\mathcal{M}} p$ wenn für alle $w \in \Sigma^*$ gilt: $\delta(q, w) \in F$ gdw. $\delta(p, w) \in F$.

Anwendung: Vereinfachung von DFAs durch Quotientenbildung

Nerode-Rechtskongruenz $\simeq_{\mathbf{L}} \subseteq \Sigma^* \times \Sigma^*$:
$u \simeq_{\mathbf{L}} v$ wenn für alle $w \in \Sigma^*$ gilt: $uw \in \mathbf{L}$ gdw. $vw \in \mathbf{L}$.

Anwendung: automatenunabhängige Analyse von Sprachen; alternative Konstruktion Minimalautomat

Der Satz von Myhill und Nerode

$\simeq$ und reguläre Sprachen

Satz: Wenn **L** regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

(Die Zahl der Äquivalenzklassen wird auch als Index bezeichnet)

$\simeq$ und reguläre Sprachen

Satz: Wenn **L** regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

(Die Zahl der Äquivalenzklassen wird auch als Index bezeichnet)

Beweis: Wir erhalten diese Eigenschaft aus der Darstellung von **L** mit einem DFA.

Für einen DFA $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F \rangle$ definieren wir eine Relation $\simeq_{\mathcal{M}}$ wie folgt. Für Wörter $u, v \in \Sigma^*$ sei $u \simeq_{\mathcal{M}} v$ wenn gilt:

$$\delta(q_0, u) = \delta(q_0, v).$$

Offensichtlich ist $\simeq_{\mathcal{M}}$ eine Äquivalenzrelation (Eigenschaften „geerbt“ von $=$) mit endlichem Index (es gibt nur endlich viele Zustände).

$\simeq$ und reguläre Sprachen (2)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Behauptung: $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$ das heißt

$$\text{wenn } u \simeq_{\mathcal{M}} v \text{ dann } u \simeq_{\mathbf{L}} v \qquad .$$

Wir betrachten einen DFA $\mathcal{M}$ mit $\mathbf{L} = \mathbf{L}(\mathcal{M})$.

$\simeq$ und reguläre Sprachen (2)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Behauptung: $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$ das heißt

$$\text{wenn } u \simeq_{\mathcal{M}} v \text{ dann } u \simeq_{\mathbf{L}} v \qquad .$$

Wir betrachten einen DFA $\mathcal{M}$ mit $\mathbf{L} = \mathbf{L}(\mathcal{M})$.

- Seien $u, v \in \Sigma^*$ Wörter mit $u \simeq_{\mathcal{M}} v$

$\simeq$ und reguläre Sprachen (2)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Behauptung: $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$ das heißt

$$\text{wenn } u \simeq_{\mathcal{M}} v \text{ dann } u \simeq_{\mathbf{L}} v \qquad .$$

Wir betrachten einen DFA $\mathcal{M}$ mit $\mathbf{L} = \mathbf{L}(\mathcal{M})$.

- Seien $u, v \in \Sigma^*$ Wörter mit $u \simeq_{\mathcal{M}} v$
- Dann ist $\delta(q_0, u) = \delta(q_0, v)$

$\simeq$ und reguläre Sprachen (2)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Behauptung: $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$ das heißt

$$\text{wenn } u \simeq_{\mathcal{M}} v \text{ dann } u \simeq_{\mathbf{L}} v \qquad .$$

Wir betrachten einen DFA $\mathcal{M}$ mit $\mathbf{L} = \mathbf{L}(\mathcal{M})$.

- Seien $u, v \in \Sigma^*$ Wörter mit $u \simeq_{\mathcal{M}} v$
- Dann ist $\delta(q_0, u) = \delta(q_0, v)$
- Dann gilt, für alle $w \in \Sigma^*$, $\delta(q_0, uw) = \delta(q_0, vw)$

$\simeq$ und reguläre Sprachen (2)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Behauptung: $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$ das heißt

$$\text{wenn } u \simeq_{\mathcal{M}} v \text{ dann } u \simeq_{\mathbf{L}} v \qquad .$$

Wir betrachten einen DFA $\mathcal{M}$ mit $\mathbf{L} = \mathbf{L}(\mathcal{M})$.

- Seien $u, v \in \Sigma^*$ Wörter mit $u \simeq_{\mathcal{M}} v$
- Dann ist $\delta(q_0, u) = \delta(q_0, v)$
- Dann gilt, für alle $w \in \Sigma^*$, $\delta(q_0, uw) = \delta(q_0, vw)$
- Dann gilt, für alle $w \in \Sigma^*$, $\delta(q_0, uw) \in F$ gdw. $\delta(q_0, vw) \in F$

$\simeq$ und reguläre Sprachen (2)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Behauptung: $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$ das heißt

$$\text{wenn } u \simeq_{\mathcal{M}} v \text{ dann } u \simeq_{\mathbf{L}} v \quad .$$

Wir betrachten einen DFA $\mathcal{M}$ mit $\mathbf{L} = \mathbf{L}(\mathcal{M})$.

- Seien $u, v \in \Sigma^*$ Wörter mit $u \simeq_{\mathcal{M}} v$
- Dann ist $\delta(q_0, u) = \delta(q_0, v)$
- Dann gilt, für alle $w \in \Sigma^*$, $\delta(q_0, uw) = \delta(q_0, vw)$
- Dann gilt, für alle $w \in \Sigma^*$, $\delta(q_0, uw) \in F$ gdw. $\delta(q_0, vw) \in F$
- Dann gilt, für alle $w \in \Sigma^*$, $uw \in \mathbf{L}(\mathcal{M})$ gdw. $vw \in \mathbf{L}(\mathcal{M})$

$\simeq$ und reguläre Sprachen (2)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Behauptung: $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$ das heißt

$$\text{wenn } u \simeq_{\mathcal{M}} v \text{ dann } u \simeq_{\mathbf{L}} v \quad .$$

Wir betrachten einen DFA $\mathcal{M}$ mit $\mathbf{L} = \mathbf{L}(\mathcal{M})$.

- Seien $u, v \in \Sigma^*$ Wörter mit $u \simeq_{\mathcal{M}} v$
- Dann ist $\delta(q_0, u) = \delta(q_0, v)$
- Dann gilt, für alle $w \in \Sigma^*$, $\delta(q_0, uw) = \delta(q_0, vw)$
- Dann gilt, für alle $w \in \Sigma^*$, $\delta(q_0, uw) \in F$ gdw. $\delta(q_0, vw) \in F$
- Dann gilt, für alle $w \in \Sigma^*$, $uw \in \mathbf{L}(\mathcal{M})$ gdw. $vw \in \mathbf{L}(\mathcal{M})$
- Dann ist $u \simeq_{\mathbf{L}(\mathcal{M})} v$ und also $u \simeq_{\mathbf{L}} v$

Damit ist die Behauptung gezeigt.

$\simeq$ und reguläre Sprachen (3)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Wir haben gezeigt, dass $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$.

$\simeq$ und reguläre Sprachen (3)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Wir haben gezeigt, dass $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$.

$\rightsquigarrow$ Jede $\simeq_{\mathbf{L}}$-Äquivalenzklasse besteht aus einer oder mehr $\simeq_{\mathcal{M}}$-Äquivalenzklassen

$\simeq$ und reguläre Sprachen (3)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Wir haben gezeigt, dass $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$.

- $\rightsquigarrow$ Jede $\simeq_{\mathbf{L}}$-Äquivalenzklasse besteht aus einer oder mehr $\simeq_{\mathcal{M}}$-Äquivalenzklassen
- $\rightsquigarrow$ Der Index von $\simeq_{\mathbf{L}}$ ist kleiner oder gleich dem Index von $\simeq_{\mathcal{M}}$

$\simeq$ und reguläre Sprachen (3)

Satz: Wenn $\mathbf{L}$ regulär ist, dann hat $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen.

Beweis (Fortsetzung): Wir haben gezeigt, dass $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$.

- $\rightsquigarrow$ Jede $\simeq_{\mathbf{L}}$-Äquivalenzklasse besteht aus einer oder mehr $\simeq_{\mathcal{M}}$-Äquivalenzklassen
- $\rightsquigarrow$ Der Index von $\simeq_{\mathbf{L}}$ ist kleiner oder gleich dem Index von $\simeq_{\mathcal{M}}$
- $\rightsquigarrow$ Der Index von $\simeq_{\mathbf{L}}$ ist endlich $\square$

Anmerkung: Der letzte Teil ist eine allgemeine Eigenschaft aller Äquivalenzrelationen: wenn $R \subseteq S$, dann ist der Index von R größer oder gleich dem Index von S.

Verschiedene Äquivalenzrelationen

Für einen DFA $\mathcal{M}$ und eine Sprache **L** haben wir verschiedene Äquivalenzrelationen verwendet:

DFA-Zustandsäquivalenz $\sim_{\mathcal{M}} \subseteq Q \times Q$:
$q \sim_{\mathcal{M}} p$ wenn für alle $w \in \Sigma^*$ gilt: $\delta(q, w) \in F$ gdw. $\delta(p, w) \in F$.

Anwendung: Vereinfachung von DFAs durch Quotientenbildung

Nerode-Rechtskongruenz $\simeq_{\mathbf{L}} \subseteq \Sigma^* \times \Sigma^*$:
$u \simeq_{\mathbf{L}} v$ wenn für alle $w \in \Sigma^*$ gilt: $uw \in \mathbf{L}$ gdw. $vw \in \mathbf{L}$.

Anwendung: automatenunabhängige Analyse von Sprachen; alternative Konstruktion Minimalautomat

DFA-Präfixäquivalenz $\simeq_{\mathcal{M}} \subseteq \Sigma^* \times \Sigma^*$:
$u \simeq_{\mathcal{M}} v$ wenn gilt: $\delta(q_0, u) = \delta(q_0, v)$.

Anwendung: Beweis der Endlichkeit des $\simeq_{\mathbf{L}}$-Indexes regulärer Sprachen

Der Satz von Myhill und Nerode

Das vorige Resultat kann noch verstärkt werden:

Satz (Myhill & Nerode): Eine Sprache **L** ist genau dann regulär, wenn $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen hat.

Der Satz von Myhill und Nerode

Das vorige Resultat kann noch verstärkt werden:

Satz (Myhill & Nerode): Eine Sprache **L** ist genau dann regulär, wenn $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen hat.

Beweis: Die Richtung „⇒“ haben wir soeben gezeigt.

Der Satz von Myhill und Nerode

Das vorige Resultat kann noch verstärkt werden:

Satz (Myhill & Nerode): Eine Sprache **L** ist genau dann regulär, wenn $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen hat.

Beweis: Die Richtung „$\Rightarrow$“ haben wir soeben gezeigt.

Für die Richtung „$\Leftarrow$“ zeigen wir, wie man einen DFA $\mathcal{M}_{\mathbf{L}}$ für **L** erhalten kann, wenn $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen hat.

Der Satz von Myhill und Nerode

Das vorige Resultat kann noch verstärkt werden:

Satz (Myhill & Nerode): Eine Sprache **L** ist genau dann regulär, wenn $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen hat.

Beweis: Die Richtung „$\Rightarrow$“ haben wir soeben gezeigt.

Für die Richtung „$\Leftarrow$“ zeigen wir, wie man einen DFA $\mathcal{M}_{\mathbf{L}}$ für **L** erhalten kann, wenn $\simeq_{\mathbf{L}}$ endlich viele Äquivalenzklassen hat.

Der DFA $\mathcal{M}_{\mathbf{L}} = \langle Q, \Sigma, \delta, q_0, F \rangle$ ist wie folgt definiert:

- $Q = \{[w]_\simeq \mid w \in \Sigma^*\}$ ist die (endliche) Menge der $\simeq$-Äquivalenzklassen
- $q_0 = [\epsilon]_\simeq$
- $F = \{[w]_\simeq \mid w \in \mathbf{L}\}$
- $\delta([w]_\simeq, \mathtt{a}) = [w\mathtt{a}]_\simeq$

Anmerkung: Diese Definition von F und δ ist zulässig, weil sie nicht vom gewählten Repräsentanten w abhängt (da $\simeq$ eine Rechtskongruenz ist)

Satz von Myhill und Nerode: Beweis

Beweis (Fortsetzung): Wir müssen noch zeigen, dass $\mathbf{L} = \mathbf{L}(\mathcal{M}_{\mathbf{L}})$.

$$w \in \mathbf{L}(\mathcal{M}_{\mathbf{L}}) \text{ gdw. } \delta(q_0, w) \in F$$

Satz von Myhill und Nerode: Beweis

Beweis (Fortsetzung): Wir müssen noch zeigen, dass $\mathbf{L} = \mathbf{L}(\mathcal{M}_{\mathbf{L}})$.

$$
\begin{aligned}
w \in \mathbf{L}(\mathcal{M}_{\mathbf{L}}) \quad &\text{gdw.} \quad \delta(q_0, w) \in F \\
&\text{gdw.} \quad \delta([\epsilon]_{\simeq}, w) \in F
\end{aligned}
$$

Satz von Myhill und Nerode: Beweis

Beweis (Fortsetzung): Wir müssen noch zeigen, dass $\mathbf{L} = \mathbf{L}(\mathcal{M}_{\mathbf{L}})$.

$$
\begin{aligned}
w \in \mathbf{L}(\mathcal{M}_{\mathbf{L}}) \quad &\text{gdw.} \quad \delta(q_0, w) \in F \\
&\text{gdw.} \quad \delta([\epsilon]_{\simeq}, w) \in F \\
&\text{gdw.} \quad [w]_{\simeq} \in F \qquad \text{wegen } \delta([u], v) = [uv] \\
& \qquad\qquad\qquad\quad \text{(Induktion über } |v|)
\end{aligned}
$$

Satz von Myhill und Nerode: Beweis

Beweis (Fortsetzung): Wir müssen noch zeigen, dass $\mathbf{L} = \mathbf{L}(\mathcal{M}_{\mathbf{L}})$.

$$
\begin{aligned}
w \in \mathbf{L}(\mathcal{M}_{\mathbf{L}}) \quad &\text{gdw.} \quad \delta(q_0, w) \in F \\
&\text{gdw.} \quad \delta([\epsilon]_{\simeq}, w) \in F \\
&\text{gdw.} \quad [w]_{\simeq} \in F && \text{wegen } \delta([u], v) = [uv] \\
& && \text{(Induktion über } |v|) \\
&\text{gdw.} \quad w \in \mathbf{L}
\end{aligned}
$$

□

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$		

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$		

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_{\simeq}$		

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_{\simeq}$	keine	

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_{\simeq}$	keine	$\{v \mid v \notin \mathbf{L}\}$

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_{\simeq}$	keine	$\{v \mid v \notin \mathbf{L}\}$

Beispiel

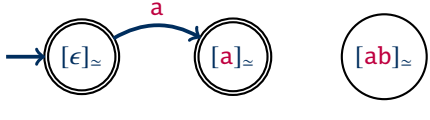
Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_\simeq$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_\simeq$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_\simeq$	keine	$\{v \mid v \notin \mathbf{L}\}$

Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

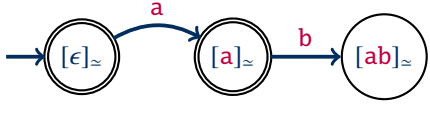
Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_\simeq$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_\simeq$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_\simeq$	keine	$\{v \mid v \notin \mathbf{L}\}$



Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

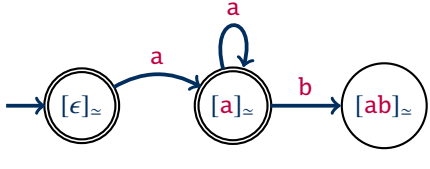
Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_{\simeq}$	keine	$\{v \mid v \notin \mathbf{L}\}$



Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

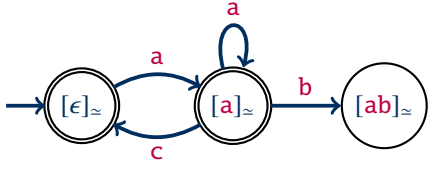
Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_{\simeq}$	keine	$\{v \mid v \notin \mathbf{L}\}$



Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

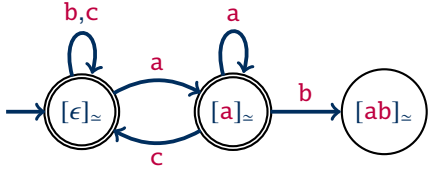
Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_\simeq$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_\simeq$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_\simeq$	keine	$\{v \mid v \notin \mathbf{L}\}$



Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

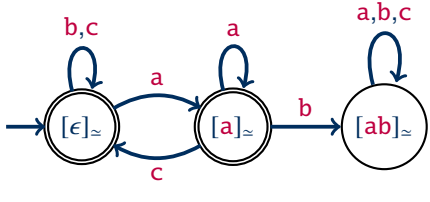
Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_\simeq$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_\simeq$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_\simeq$	keine	$\{v \mid v \notin \mathbf{L}\}$



Beispiel

Die Konstruktion von Myhill & Nerode für die Sprache $\mathbf{L} = \{w \mid w \text{ enthält kein Infix } \mathtt{ab}\}$ über dem Alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Klasse C	erlaubte Suffixe w mit $C \cdot \{w\} \subseteq \mathbf{L}$	C als Menge
$[\epsilon]_{\simeq}$	$w \in \mathbf{L}$	$\{v \in \mathbf{L} \mid v \text{ endet nicht auf } \mathtt{a}\}$
$[\mathtt{a}]_{\simeq}$	$w \in \mathbf{L}$, w beginnt nicht mit $\mathtt{b}$	$\{v \in \mathbf{L} \mid v \text{ endet auf } \mathtt{a}\}$
$[\mathtt{ab}]_{\simeq}$	keine	$\{v \mid v \notin \mathbf{L}\}$



$\mathcal{M}_{\mathbf{L}}$ ist minimal

Für reguläre **L** haben wir bereits gezeigt:

- $\mathcal{M}_{\mathbf{L}}$ ist ein DFA, da er insbesondere endlich viele Zustände hat (Index von $\simeq$)
- $\mathbf{L}(\mathcal{M}_{\mathbf{L}}) = \mathbf{L}$

Als nächstes interessiert uns der Bezug zum Minimalautomat

$\mathcal{M}_{\mathbf{L}}$ ist minimal

Für reguläre **L** haben wir bereits gezeigt:

- $\mathcal{M}_{\mathbf{L}}$ ist ein DFA, da er insbesondere endlich viele Zustände hat (Index von $\simeq$)
- $\mathbf{L}(\mathcal{M}_{\mathbf{L}}) = \mathbf{L}$

Als nächstes interessiert uns der Bezug zum Minimalautomat:

Satz: $\mathcal{M}_{\mathbf{L}}$ hat unter allen totalen DFAs, die **L** erkennen, eine minimale Anzahl an Zuständen.

$\mathcal{M}_{\mathbf{L}}$ ist minimal

Für reguläre **L** haben wir bereits gezeigt:

- $\mathcal{M}_{\mathbf{L}}$ ist ein DFA, da er insbesondere endlich viele Zustände hat (Index von $\simeq$)
- $\mathbf{L}(\mathcal{M}_{\mathbf{L}}) = \mathbf{L}$

Als nächstes interessiert uns der Bezug zum Minimalautomat:

Satz: $\mathcal{M}_{\mathbf{L}}$ hat unter allen totalen DFAs, die **L** erkennen, eine minimale Anzahl an Zuständen.

Beweis: Sei $\mathcal{M}$ ein beliebiger totaler DFA mit $\mathbf{L}(\mathcal{M}) = \mathbf{L}$

- Wir haben bereits gezeigt: $\simeq_{\mathcal{M}} \subseteq \simeq_{\mathbf{L}}$
- Daraus folgerten wir:
  „Zahl der $\simeq_{\mathbf{L}}$-Äquivalenzklassen $\leq$ Anzahl der erreichbaren Zustände von $\mathcal{M}$“
- Die Zahl der $\simeq_{\mathbf{L}}$-Äquivalenzklassen ist aber die Anzahl der Zustände von $\mathcal{M}_{\mathbf{L}}$ □

Weitere Eigenschaften von $\mathcal{M}_{\mathbf{L}}$

Für reguläre **L** haben wir bereits gezeigt:

- $\mathcal{M}_{\mathbf{L}}$ ist ein DFA, da er insbesondere endlich viele Zustände hat (Index von $\simeq$)
- $\mathbf{L}(\mathcal{M}_{\mathbf{L}}) = \mathbf{L}$
- Jeder Zustand von $\mathcal{M}_{\mathbf{L}}$ ist vom Startzustand aus erreichbar, da $\delta([\epsilon]_{\simeq}, w) = [w]_{\simeq}$
- $\simeq_{\mathcal{M}_{\mathbf{L}}} = \simeq_{\mathbf{L}}$

Weitere Eigenschaften von $\mathcal{M}_{\mathbf{L}}$

Für reguläre **L** haben wir bereits gezeigt:

- $\mathcal{M}_{\mathbf{L}}$ ist ein DFA, da er insbesondere endlich viele Zustände hat (Index von $\simeq$)
- $\mathbf{L}(\mathcal{M}_{\mathbf{L}}) = \mathbf{L}$
- Jeder Zustand von $\mathcal{M}_{\mathbf{L}}$ ist vom Startzustand aus erreichbar, da $\delta([\epsilon]_{\simeq}, w) = [w]_{\simeq}$
- $\simeq_{\mathcal{M}_{\mathbf{L}}} = \simeq_{\mathbf{L}}$ (♠)

Beweis (♠): $\simeq_{\mathcal{M}_{\mathbf{L}}} \subseteq \simeq_{\mathbf{L}}$ haben wir bereits für beliebige DFA gezeigt

$\simeq_{\mathcal{M}_{\mathbf{L}}} \supseteq \simeq_{\mathbf{L}}$ kann wie folgt gezeigt werden:

- Seien $u, v \in \Sigma^*$ beliebige Wörter mit $u \simeq_{\mathbf{L}} v$
- Dann gilt für $\mathcal{M}_{\mathbf{L}}$: $\delta([\epsilon]_{\simeq}, u) = [u]_{\simeq} = [v]_{\simeq} = \delta([\epsilon]_{\simeq}, v)$
- Also ist $u \simeq_{\mathcal{M}_{\mathbf{L}}} v$ □

Isomorphismen von Automaten

„Isomorph“ bedeutet „gleich bis auf Umbenennung von Zuständen“:

Ein Isomorphismus zwischen zwei DFAs $\mathcal{M}_1 = \langle Q_1, \Sigma, \delta_1, q_1, F_1 \rangle$ und $\mathcal{M}_2 = \langle Q_2, \Sigma, \delta_2, q_2, F_2 \rangle$ ist eine bijekive Funktion $f : Q_1 \to Q_2$, so dass gilt:

- $f(q_1) = q_2$
- $f(\delta_1(q, \mathtt{a})) = \delta_2(f(q), \mathtt{a})$ für alle $\mathtt{a} \in \Sigma$
- $\{f(q) \mid q \in F_1\} = F_2$

Zwei Automaten sind isomorph, wenn es einen Isomorphismus zwischen ihnen gibt.

Man kann leicht zeigen, dass isomorphe Automaten die selbe Sprache akzeptieren

Eindeutigkeit des Minimalautomaten

Satz: Ist $\mathbf{L}$ eine reguläre Sprache, $\mathcal{M}$ ein DFA mit totaler Übergangsfunktion und $\mathbf{L}(\mathcal{M}) = \mathbf{L}$, so sind der reduzierte Automat $\mathcal{M}_r$ und der Myhill-Nerode-Minimalautomat $\mathcal{M}_\mathbf{L}$ isomorph.

Beweis: Sei $\mathcal{M}_\mathbf{L} = \langle Q, \Sigma, \delta, q, F\rangle$ und $\mathcal{M}_r = \langle Q_r, \Sigma, \delta_r, q_r, F_r\rangle$.

$\mathcal{M}_\mathbf{L} = \langle Q, \Sigma, \delta, q_0, F\rangle$ mit:

- $Q = \{[w]_\simeq \mid w \in \Sigma^*\}$
- $q_0 = [\epsilon]_\simeq$
- $F = \{[w]_\simeq \mid w \in \mathbf{L}\}$
- $\delta([w]_\simeq, \mathrm{a}) = [w\mathrm{a}]_\simeq$

$\mathcal{M}_r = \langle Q_r, \Sigma, \delta_r, q_r, F_r\rangle$:

(1) Entferne unerreichbare Zustände

(2) Berechne den Quotienten-DFA bzgl. $\sim$

Eindeutigkeit des Minimalautomaten

Satz: Ist **L** eine reguläre Sprache, $\mathcal{M}$ ein DFA mit totaler Übergangsfunktion und $\mathbf{L}(\mathcal{M}) = \mathbf{L}$, so sind der reduzierte Automat $\mathcal{M}_r$ und der Myhill-Nerode-Minimalautomat $\mathcal{M}_\mathbf{L}$ isomorph.

Beweis: Sei $\mathcal{M}_\mathbf{L} = \langle Q, \Sigma, \delta, q, F\rangle$ und $\mathcal{M}_r = \langle Q_r, \Sigma, \delta_r, q_r, F_r\rangle$.

$\mathcal{M}_\mathbf{L} = \langle Q, \Sigma, \delta, q_0, F\rangle$ mit:

- $Q = \{[w]_\simeq \mid w \in \Sigma^*\}$
- $q_0 = [\epsilon]_\simeq$
- $F = \{[w]_\simeq \mid w \in \mathbf{L}\}$
- $\delta([w]_\simeq, \mathrm{a}) = [w\mathrm{a}]_\simeq$

$\mathcal{M}_r = \langle Q_r, \Sigma, \delta_r, q_r, F_r\rangle$:

(1) Entferne unerreichbare Zustände

(2) Berechne den Quotienten-DFA bzgl. $\sim$

Wir definieren eine Funktion $f : Q_r \to Q$ wie folgt:

für $q \in Q_r$ wähle ein Wort w mit $\delta_r(q_r, w) = q$ und setze $f(q) = [w]_\simeq$

Eindeutigkeit des Minimalautomaten (2)

für $q \in Q_r$ wähle ein Wort w mit $\delta_r(q_r, w) = q$ und setze $f(q) = [w]_{\simeq}$

Ist f wohldefiniert?

Eindeutigkeit des Minimalautomaten (2)

für $q \in Q_r$ wähle ein Wort w mit $\delta_r(q_r, w) = q$ und setze $f(q) = [w]_{\simeq}$

Ist f wohldefiniert?

- für jeden Zustand $q \in Q_r$ gibt es ein geeignetes Wort w, da alle Zustände in $\mathcal{M}_r$ vom Startzustand q_r aus erreichbar sind.

Eindeutigkeit des Minimalautomaten (2)

für $q \in Q_r$ wähle ein Wort w mit $\delta_r(q_r, w) = q$ und setze $f(q) = [w]_{\simeq}$

Ist f wohldefiniert?

- für jeden Zustand $q \in Q_r$ gibt es ein geeignetes Wort w, da alle Zustände in $\mathcal{M}_r$ vom Startzustand q_r aus erreichbar sind.
- die Wahl von w ist dabei unerheblich:
  - wenn $\delta_r(q_r, w) = q = \delta_r(q_r, w')$
  - dann $w \simeq_{\mathcal{M}_r} w'$
  - dann $w \simeq_{\mathsf{L}} w'$ (wie zuvor gezeigt)
  - dann $[w]_{\simeq} = [w']_{\simeq}$

$\rightsquigarrow f$ ist wohldefiniert

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_{\mathbf{L}}$

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_{\mathbf{L}}$

(1) „$f(q_r) = q_0$“:

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_{\mathbf{L}}$

(1) „$f(q_r) = q_0$“: es gilt $\delta_r(q_r, \epsilon) = q_r$ und daher $f(q_r) = [\epsilon]_{\simeq} = q_0$

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_\mathbf{L}$

(1) „$f(q_r) = q_0$“: es gilt $\delta_r(q_r, \epsilon) = q_r$ und daher $f(q_r) = [\epsilon]_\simeq = q_0$

(2) „$f(\delta_r(q, \mathsf{a})) = \delta(f(q), \mathsf{a})$“:

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_\mathbf{L}$

(1) „$f(q_r) = q_0$“: es gilt $\delta_r(q_r, \epsilon) = q_r$ und daher $f(q_r) = [\epsilon]_\simeq = q_0$

(2) „$f(\delta_r(q, \mathsf{a})) = \delta(f(q), \mathsf{a})$“:

- Sei w ein Wort mit $\delta_r(q_r, w) = q$

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_\mathbf{L}$

(1) „$f(q_r) = q_0$“: es gilt $\delta_r(q_r, \epsilon) = q_r$ und daher $f(q_r) = [\epsilon]_\simeq = q_0$

(2) „$f(\delta_r(q, \mathsf{a})) = \delta(f(q), \mathsf{a})$“:

- Sei w ein Wort mit $\delta_r(q_r, w) = q$
- Dann ist $\delta_r(q_r, w\mathsf{a}) = \delta_r(q, \mathsf{a})$ und daher $f(\delta_r(q, \mathsf{a})) = [w\mathsf{a}]_\simeq$

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_\mathbf{L}$

(1) „$f(q_r) = q_0$“: es gilt $\delta_r(q_r, \epsilon) = q_r$ und daher $f(q_r) = [\epsilon]_\simeq = q_0$

(2) „$f(\delta_r(q, \mathsf{a})) = \delta(f(q), \mathsf{a})$“:

- Sei w ein Wort mit $\delta_r(q_r, w) = q$
- Dann ist $\delta_r(q_r, w\mathsf{a}) = \delta_r(q, \mathsf{a})$ und daher $f(\delta_r(q, \mathsf{a})) = [w\mathsf{a}]_\simeq$
- Andererseits gilt $f(q) = [w]_\simeq$ und daher $\delta(f(q), \mathsf{a}) = [w\mathsf{a}]_\simeq$

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_\mathbf{L}$

(1) „$f(q_r) = q_0$“: es gilt $\delta_r(q_r, \epsilon) = q_r$ und daher $f(q_r) = [\epsilon]_\simeq = q_0$

(2) „$f(\delta_r(q, \mathsf{a})) = \delta(f(q), \mathsf{a})$“:

- Sei w ein Wort mit $\delta_r(q_r, w) = q$
- Dann ist $\delta_r(q_r, w\mathsf{a}) = \delta_r(q, \mathsf{a})$ und daher $f(\delta_r(q, \mathsf{a})) = [w\mathsf{a}]_\simeq$
- Andererseits gilt $f(q) = [w]_\simeq$ und daher $\delta(f(q), \mathsf{a}) = [w\mathsf{a}]_\simeq$

(3) „$\{f(q) \mid q \in F_r\} = F$“:

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_\mathbf{L}$

(1) „$f(q_r) = q_0$“: es gilt $\delta_r(q_r, \epsilon) = q_r$ und daher $f(q_r) = [\epsilon]_\simeq = q_0$

(2) „$f(\delta_r(q, \mathrm{a})) = \delta(f(q), \mathrm{a})$“:

- Sei w ein Wort mit $\delta_r(q_r, w) = q$
- Dann ist $\delta_r(q_r, w\mathrm{a}) = \delta_r(q, \mathrm{a})$ und daher $f(\delta_r(q, \mathrm{a})) = [w\mathrm{a}]_\simeq$
- Andererseits gilt $f(q) = [w]_\simeq$ und daher $\delta(f(q), \mathrm{a}) = [w\mathrm{a}]_\simeq$

(3) „$\{f(q) \mid q \in F_r\} = F$“: Übung

Eindeutigkeit des Minimalautomaten (3)

Es bleibt zu zeigen: f ist ein Isomorphismus von $\mathcal{M}_r$ zu $\mathcal{M}_{\mathbf{L}}$

(1) „$f(q_r) = q_0$“: es gilt $\delta_r(q_r, \epsilon) = q_r$ und daher $f(q_r) = [\epsilon]_\simeq = q_0$

(2) „$f(\delta_r(q, \mathsf{a})) = \delta(f(q), \mathsf{a})$“:
- Sei w ein Wort mit $\delta_r(q_r, w) = q$
- Dann ist $\delta_r(q_r, w\mathsf{a}) = \delta_r(q, \mathsf{a})$ und daher $f(\delta_r(q, \mathsf{a})) = [w\mathsf{a}]_\simeq$
- Andererseits gilt $f(q) = [w]_\simeq$ und daher $\delta(f(q), \mathsf{a}) = [w\mathsf{a}]_\simeq$

(3) „$\{f(q) \mid q \in F_r\} = F$“: Übung

□

Damit ist der Satz bewiesen:

Satz: Ist **L** eine reguläre Sprache, $\mathcal{M}$ ein DFA mit totaler Übergangsfunktion und $\mathbf{L}(\mathcal{M}) = \mathbf{L}$, so sind der reduzierte Automat $\mathcal{M}_r$ und der Myhill-Nerode-Minimalautomat $\mathcal{M}_{\mathbf{L}}$ isomorph.

Zusammenfassung: Eigenschaften von $\mathcal{M}_{\mathbf{L}}$

Für reguläre **L** haben wir bereits gezeigt:

- $\mathcal{M}_{\mathbf{L}}$ ist ein DFA, da er insbesondere endlich viele Zustände hat (Index von $\simeq$)
- $\mathbf{L}(\mathcal{M}_{\mathbf{L}}) = \mathbf{L}$
- Jeder Zustand von $\mathcal{M}_{\mathbf{L}}$ ist vom Startzustand aus erreichbar, da $\delta([\epsilon]_{\simeq}, w) = [w]_{\simeq}$
- $\simeq_{\mathcal{M}_{\mathbf{L}}} = \simeq_{\mathbf{L}}$
- $\mathcal{M}_{\mathbf{L}}$ ist isomorph zu jedem reduzierten Automaten für **L**

Daraus folgt auch:

Satz: Alle minimalen DFA mit totaler Übergangsfunktion, die $\mathbf{L}(\mathcal{M})$ erkennen, sind bis auf Umbenennung von Zuständen gleich (sie sind isomorph). Daher hängt $\mathcal{M}_r$ nur von $\mathbf{L}(\mathcal{M})$ ab, nicht von $\mathcal{M}$.

Automaten vergleichen

Wir wissen, dass reduzierte Automaten isomorph sind, wenn sie die selbe Sprache akzeptieren.

Damit können wir Äquivalenz zweier Automaten testen:

Eingabe: Zwei endliche Automaten $\mathcal{M}_1$ und $\mathcal{M}_2$
Ausgabe: Ist $\mathbf{L}(\mathcal{M}_1) = \mathbf{L}(\mathcal{M}_2)$?

- Transformiere $\mathcal{M}_1$ und $\mathcal{M}_2$ falls nötig in DFAs mit totaler Übergangsfunktion
- Bestimme die reduzierten Automaten
- Teste, ob die reduzierten Automaten isomorph sind (z.B. naiv durch systematisches Durchprobieren aller Bijektionen)

Auf ähnlichem Weg kann man auch reguläre Ausdrücke und reguläre Grammatiken vergleichen.
Es gibt allerdings zum Teil effizientere Verfahren . . .

Minimieren ohne totale Übergänge?

Alle Ergebnisse zu Minimalautomaten gehen von DFAs mit totaler Übergangsfunktion aus

Diskussion:

- Jeder DFA kann leicht in einen totalen umgewandelt werden (ein zusätzlicher Fangzustand)
- Der Fangzustand entspricht der Nerode-Kongruenzklasse $\{w \mid w$ ist kein Präfix eines Wortes in $\mathbf{L}\}$
- In minimalen Automaten gibt es daher maximal einen entsprechenden Fangzustand, den man weglassen könnte, ohne die akzeptierte Sprache zu ändern*
- Man kann DFAs ohne totale Übergangsfunktion auch direkt reduzieren, wobei der Algorithmus zur Bestimmung von $\sim$ ausgeführt wird, als ob man einen Fangzustand definiert hätte

* Ausnahme: der DFA der leeren Sprache benötigt diese (einzige!) Klasse

Minimale NFAs?

Alle Ergebnisse zu Minimalautomaten gehen von DFAs aus
$\rightsquigarrow$ NFAs könnten viel kleiner sein als minimale DFAs!

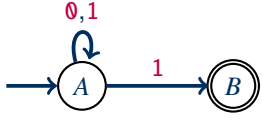
Für NFAs gibt es keine vergleichbar elegante Art der Minimierung

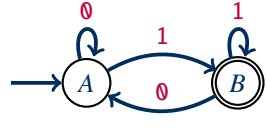
Minimale NFAs?

Alle Ergebnisse zu Minimalautomaten gehen von DFAs aus
$\rightsquigarrow$ NFAs könnten viel kleiner sein als minimale DFAs!

Für NFAs gibt es keine vergleichbar elegante Art der Minimierung

Beispiel: Die folgenden beiden NFAs akzeptieren die selbe Sprache („Wörter die auf 1 enden“) und sind von minimaler Größe, aber sie sind nicht isomorph:





- NFA-Minimierung ist algorithmisch aufwändiger
- … und liefert kein eindeutiges Ergebnis,
- … ist aber durchaus praktisch machbar

Sind NFAs wirklich kompakter?

Beispiel aus Vorlesung 4: $\mathbf{L}_n = \{0, 1\}^* 1 \{0, 1\}^{n-1}$ („Wörter mit 1 an n-letzter Stelle“)

Wir haben gesehen, dass es dafür NFAs der Größe $n + 1$ gibt

DFAs benötigen dagegen mindestens 2^n Zustände

Sind NFAs wirklich kompakter?

Beispiel aus Vorlesung 4: $\mathbf{L}_n = \{0, 1\}^* 1 \{0, 1\}^{n-1}$ („Wörter mit 1 an n-letzter Stelle“)

Wir haben gesehen, dass es dafür NFAs der Größe $n + 1$ gibt

DFAs benötigen dagegen mindestens 2^n Zustände

Beweis: mit Hilfe von Myhill/Nerode:

- Die Kongruenzklassen haben die Form $[w]_{\simeq}$ mit $w \in \{0, 1\}^n$
- Jede dieser Klassen ist unterschiedlich, z.B. für $n = 3$:
  wenn $v \in [011]$ dann $v\epsilon \notin \mathbf{L}_3$, $v0 \in \mathbf{L}_3$ und $v00 \in \mathbf{L}_3$
  wenn $v \in [101]$ dann $v\epsilon \in \mathbf{L}_3$, $v0 \notin \mathbf{L}_3$ und $v00 \in \mathbf{L}_3$
  ...
- Es gibt also mindestens 2^n Kongruenzklassen
  (=Zustände im Minimal-DFA)

Ausblick: Nicht-reguläre Sprachen

Nicht-reguläre Sprachen

Sind alle Sprachen regulär?

Nicht-reguläre Sprachen

Sind alle Sprachen regulär?

Sicher nicht (dazu gibt es zu viele Sprachen)

Nicht-reguläre Sprachen

Sind alle Sprachen regulär?
Sicher nicht (dazu gibt es zu viele Sprachen)

Sind alle Typ-0-Sprachen regulär?

Nicht-reguläre Sprachen

Sind alle Sprachen regulär?
Sicher nicht (dazu gibt es zu viele Sprachen)

Sind alle Typ-0-Sprachen regulär?
Nein, auch das gilt nicht

Wie aber zeigt man das?

- Behauptung „Sprache **L** ist regulär!“ $\rightsquigarrow$ es genügt, einen Automaten, regulären Ausdruck oder eine Typ-3-Grammatik für **L** anzugeben
- Behauptung „Sprache **L** ist nicht regulär!“ $\rightsquigarrow$ man müsste zeigen, dass es keinen Automaten, keinen regulären Ausdruck bzw. keine Typ-3-Grammatik für **L** gibt

Zusammenfassung und Ausblick

Der Satz von Myhill und Nerode charakterisiert reguläre Sprachen und liefert einen direkt konstruierten DFA für eine Sprache

Der reduzierte DFA ist minimal und eindeutig

NFAs können viel kleiner sein als minimale DFAs, aber ihre Minimierung ist viel komplizierter

Offene Fragen:

- Wie findet man nicht-reguläre Sprachen?
- Welche weiteren Berechnungsaufgaben gibt es im Zusammenhang mit regulären Sprachen?
- Und was ist mit kontextfreien Sprachen?