Script generated by TTT

Title: Seidl: Theoretische_Informatik (03.05.2012)

Date: Thu May 03 16:01:45 CEST 2012

Duration: 87:49 min

Pages: 109

$\alpha\alpha^* = \alpha^*\alpha$

Lemma 2.48

Seien M, M_1, M_2 Mengen von REs.

- $L[M_1 M_2] = L[M_1]L[M_2]$
- $L[M_1 \cup M_2] = L[M_1] \cup L[M_2]$
- $L[M^*] = (L[M])^*$

Beweis:

$$\begin{aligned}
L[M_1M_2] &= \bigcup_{\alpha \in M_1M_2} L(\alpha) \\
&= \bigcup_{\alpha_1 \in M_1, \alpha_2 \in M_2} L(\alpha_1\alpha_2) \\
&= \bigcup_{\alpha_1 \in M_1, \alpha_2 \in M_2} L(\alpha_1)L(\alpha_2) \\
&= \bigcup_{\alpha_1 \in M_1, \alpha_2 \in M_2} \{w_1w_2 \mid w_1 \in L(\alpha_1) \wedge w_2 \in L(\alpha_2)\} \\
&= \{w_1w_2 \mid w_1 \in \bigcup_{\alpha_1 \in M_1} L(\alpha_1) \wedge w_2 \in \bigcup_{\alpha_2 \in M_2} L(\alpha_2)\} \\
&= \{w_1w_2 \mid w_1 \in L[M_1] \wedge w_2 \in L[M_2]\} \\
&= L[M_1]L[M_2]
\end{aligned}$$

□

Gilt auch $L[M_1 \cap M_2] = L[M_1] \cap L[M_2]$?

Lemma 2.48

Seien M, M_1, M_2 Mengen von REs.

- $L[M_1 M_2] = L[M_1]L[M_2]$
- $L[M_1 \cup M_2] = L[M_1] \cup L[M_2]$
- $L[M^*] = (L[M])^*$

Beweis:

$$\begin{aligned}
L[M_1M_2] &= \bigcup_{\alpha \in M_1M_2} L(\alpha) \\
&= \bigcup_{\alpha_1 \in M_1, \alpha_2 \in M_2} L(\alpha_1\alpha_2) \\
&= \bigcup_{\alpha_1 \in M_1, \alpha_2 \in M_2} L(\alpha_1)L(\alpha_2) \\
&= \bigcup_{\alpha_1 \in M_1, \alpha_2 \in M_2} \{w_1w_2 \mid w_1 \in L(\alpha_1) \wedge w_2 \in L(\alpha_2)\} \\
&= \{w_1w_2 \mid w_1 \in \bigcup_{\alpha_1 \in M_1} L(\alpha_1) \wedge w_2 \in \bigcup_{\alpha_2 \in M_2} L(\alpha_2)\} \\
&= \{w_1w_2 \mid w_1 \in L[M_1] \wedge w_2 \in L[M_2]\} \\
&= L[M_1]L[M_2]
\end{aligned}$$

□

Gilt auch $L[M_1 \cap M_2] = L[M_1] \cap L[M_2]$?

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:
Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))]$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:
Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}]$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:
Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}] = L(\sigma(X))$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:
Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}] = L(\sigma(X))$$

- Fall $E = E_1E_2$:

$$L[\sigma(L(E_1E_2))]$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:
Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}] = L(\sigma(X))$$

- Fall $E = E_1E_2$:

$$L[\sigma(L(E_1E_2))] = L[\sigma(L(E_1)\,L(E_2))]$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:
Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}] = L(\sigma(X))$$

- Fall $E = E_1E_2$:

$$\begin{aligned} L[\sigma(L(E_1E_2))] &= L[\sigma(L(E_1)\,L(E_2))] \\ &= L[\sigma(L(E_1))\,\sigma(L(E_2))] \end{aligned}$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:
Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}] = L(\sigma(X))$$

- Fall $E = E_1E_2$:

$$\begin{aligned} L[\sigma(L(E_1E_2))] &= L[\sigma(L(E_1)\,L(E_2))] \\ &= L[\sigma(L(E_1))\,\sigma(L(E_2))] \\ &= L[\sigma(L(E_1))]\,L[\sigma(L(E_2))] \end{aligned}$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:

Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}] = L(\sigma(X))$$

- Fall $E = E_1E_2$:

$$\begin{aligned}
L[\sigma(L(E_1E_2))] &= L[\sigma(L(E_1)\,L(E_2))] \\
&= L[\sigma(L(E_1))\,\sigma(L(E_2))] \\
&= L[\sigma(L(E_1))]\,L[\sigma(L(E_2))] \\
&= L(\sigma(E_1))\,L(\sigma(E_2))
\end{aligned}$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:

Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}] = L(\sigma(X))$$

- Fall $E = E_1E_2$:

$$\begin{aligned}
L[\sigma(L(E_1E_2))] &= L[\sigma(L(E_1)\,L(E_2))] \\
&= L[\sigma(L(E_1))\,\sigma(L(E_2))] \\
&= L[\sigma(L(E_1))]\,L[\sigma(L(E_2))] \\
&= L(\sigma(E_1))\,L(\sigma(E_2)) \\
&= L(\sigma(E_1)\,\sigma(E_2))
\end{aligned}$$

Satz 2.49 (Substitutionslemma)

$$L[\sigma(L(E))] = L(\sigma(E))$$

Beweis:

Mit struktureller Induktion über E.

- Fall $E = X$:

$$L[\sigma(L(X))] = L[\sigma(\{X\})] = L[\{\sigma(X)\}] = L(\sigma(X))$$

- Fall $E = E_1E_2$:

$$\begin{aligned}
L[\sigma(L(E_1E_2))] &= L[\sigma(L(E_1)\,L(E_2))] \\
&= L[\sigma(L(E_1))\,\sigma(L(E_2))] \\
&= L[\sigma(L(E_1))]\,L[\sigma(L(E_2))] \\
&= L(\sigma(E_1))\,L(\sigma(E_2)) \\
&= L(\sigma(E_1)\,\sigma(E_2)) \\
&= L(\sigma(E_1E_2))
\end{aligned}$$

$\square$

Korollar 2.50

Falls $E_1 \equiv E_2$ (wobei Variablen als neue Konstanten betrachtet werden), dann $\sigma(E_1) \equiv \sigma(E_2)$ für alle Substitutionen σ.

Korollar 2.50

Falls $E_1 \equiv E_2$ (wobei Variablen als neue Konstanten betrachtet werden), dann $\sigma(E_1) \equiv \sigma(E_2)$ für alle Substitutionen σ.

Denn $E_1 \equiv E_2$, also $L(E_1) = L(E_2)$, impliziert

$$L(\sigma(E_1))$$

Korollar 2.50

Falls $E_1 \equiv E_2$ (wobei Variablen als neue Konstanten betrachtet werden), dann $\sigma(E_1) \equiv \sigma(E_2)$ für alle Substitutionen σ.

Denn $E_1 \equiv E_2$, also $L(E_1) = L(E_2)$, impliziert

$$L(\sigma(E_1)) = L[\sigma(L(E_1))] = L[\sigma(L(E_2))] = L(\sigma(E_2)),$$

2.11 Automaten und Gleichungssysteme

Nicht mehr *Beweisen* von Äquivalenzen

Gilt $XX^* \equiv X^*X$ für alle X?

2.11 Automaten und Gleichungssysteme

Nicht mehr *Beweisen* von Äquivalenzen

Gilt $XX^* \equiv X^*X$ für alle X?

sondern *Lösen*:

Für welches X gilt $X \equiv aX \mid b$?

2.11 Automaten und Gleichungssysteme

Nicht mehr *Beweisen* von Äquivalenzen

Gilt $XX^* \equiv X^*X$ für alle X?

sondern *Lösen*:

Für welches X gilt $X \equiv aX \mid b$?

Anwendung:

Automat $\rightsquigarrow$ Gleichungssystem $\rightsquigarrow$ RE

2.11 Automaten und Gleichungssysteme

Nicht mehr *Beweisen* von Äquivalenzen

Gilt $XX^* \equiv X^*X$ für alle X?

sondern *Lösen*:

Für welches X gilt $X \equiv aX \mid b$?

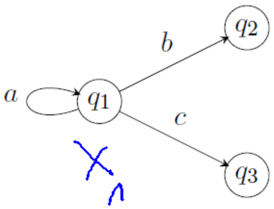
Anwendung:

Automat $\rightsquigarrow$ Gleichungssystem $\rightsquigarrow$ RE

Technische Vereinfachung: die X_i stehen für (unbekannte) REs.

Beispiel 2.51

Ein Automatenfragment:

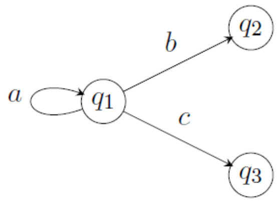


X_1 X_2 X_3

$X_2 = X$

Beispiel 2.51

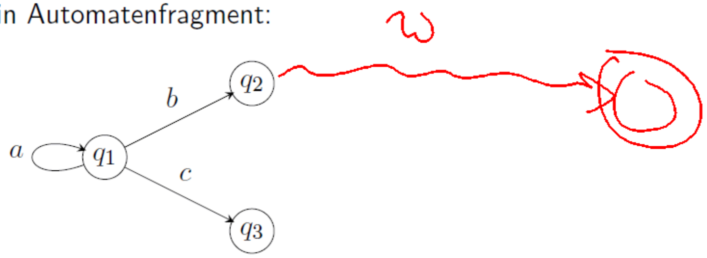
Ein Automatenfragment:



$$L_i := \{w \mid \hat{\delta}(q_i, w) \in F\}$$

Beispiel 2.51

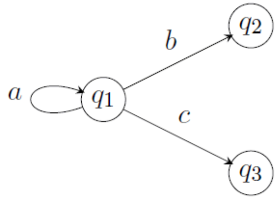
Ein Automatenfragment:



$$X_1 = bX_2 \mid cX_3 \mid aX_1$$

Beispiel 2.51

Ein Automatenfragment:

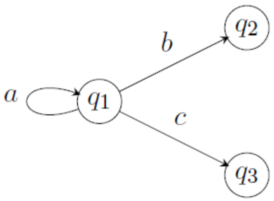


$$L_i := \{w \mid \hat{\delta}(q_i, w) \in F\}$$
$$\Longrightarrow$$
$$L_1 = \{a\}L_1 \cup \{b\}L_2 \cup \{c\}L_3$$

Da die L_i regulär sein müssen(?),

Beispiel 2.51

Ein Automatenfragment:



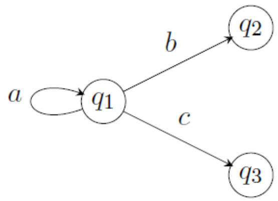
$$L_i := \{w \mid \hat{\delta}(q_i, w) \in F\}$$
$$\Longrightarrow$$
$$L_1 = \{a\}L_1 \cup \{b\}L_2 \cup \{c\}L_3$$

Da die L_i regulär sein müssen(?), arbeiten wir direkt mit REs:

$$X_1 \equiv aX_1 \mid bX_2 \mid cX_3$$

Beispiel 2.51

Ein Automatenfragment:



$$L_i := \{w \mid \hat{\delta}(q_i, w) \in F\}$$
$$\Longrightarrow$$
$$L_1 = \{a\}L_1 \cup \{b\}L_2 \cup \{c\}L_3$$

Da die L_i regulär sein müssen(?), arbeiten wir direkt mit REs:

$$X_1 \equiv aX_1 \mid bX_2 \mid cX_3$$

Lösung X_i ist RE für die von q_i aus akzeptierte Sprache.

Satz 2.52 (Ardens Lemma)

Sind A, B und X Spachen mit $\epsilon \notin A$, so gilt

$$X = AX \cup B \quad \Longrightarrow \quad X = A^*B$$

Satz 2.52 (Ardens Lemma)

Sind A, B und X Spachen mit $\epsilon \notin A$, so gilt

$$X = AX \cup B \quad \Longrightarrow \quad X = A^*B$$

Korollar 2.53

Sind α, β und X reguläre Ausdrücke mit $\epsilon \notin L(\alpha)$, so gilt

$$X \equiv \alpha X \mid \beta \quad \Longrightarrow \quad X \equiv \alpha^* \beta$$

Satz 2.52 (Ardens Lemma)

Sind A, B und X Spachen mit $\epsilon \notin A$, so gilt

$$X = AX \cup B \quad \Longrightarrow \quad X = A^*B$$

Korollar 2.53

Sind α, β und X reguläre Ausdrücke mit $\epsilon \notin L(\alpha)$, so gilt

$$X \equiv \alpha X \mid \beta \quad \Longrightarrow \quad X \equiv \alpha^* \beta$$

Bemerkungen

- $X = \{\epsilon\}X \cup B$ hat keine eindeutige Lösung:

Satz 2.52 (Ardens Lemma)

Sind A, B und X Spachen mit $\epsilon \notin A$, so gilt

$$X = AX \cup B \quad \Longrightarrow \quad X = A^*B$$

Korollar 2.53

Sind α, β und X reguläre Ausdrücke mit $\epsilon \notin L(\alpha)$, so gilt

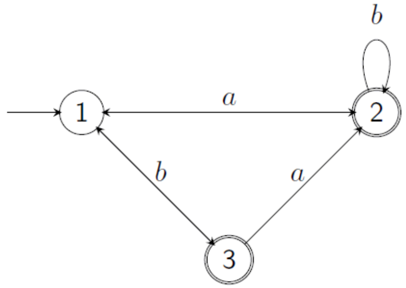
$$X \equiv \alpha X \mid \beta \quad \Longrightarrow \quad X \equiv \alpha^* \beta$$

Bemerkungen

- $X = \{\epsilon\}X \cup B$ hat keine eindeutige Lösung:
  jede Sprache $X \supseteq B$ ist Lösung.
- $X \equiv aXb \mid \epsilon$ hat keine *reguläre* Lösung.

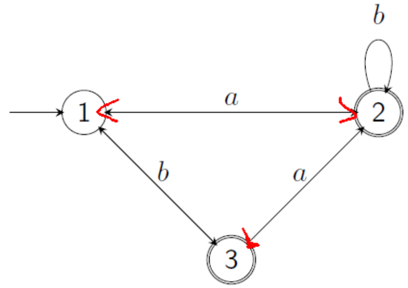
Umwandlung eines DFAs in einen äquivalenten regulären Ausdruck:

Beispiel 2.54



Umwandlung eines DFAs in einen äquivalenten regulären Ausdruck:

Beispiel 2.54



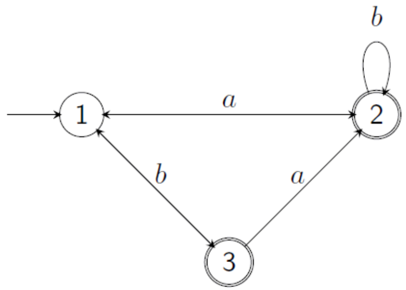
Äquivalentes Gleichungssystem:

X_i ist ein RE für die von q_i aus akzeptierte Sprache.

$$X_1 \equiv$$

Umwandlung eines DFAs in einen äquivalenten regulären Ausdruck:

Beispiel 2.54



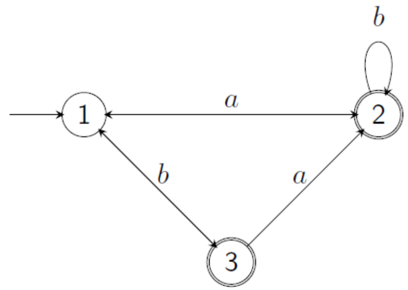
Äquivalentes Gleichungssystem:

X_i ist ein RE für die von q_i aus akzeptierte Sprache.

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \end{aligned}$$

Umwandlung eines DFAs in einen äquivalenten regulären Ausdruck:

Beispiel 2.54



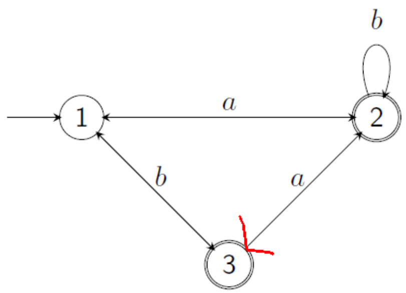
Äquivalentes Gleichungssystem:

X_i ist ein RE für die von q_i aus akzeptierte Sprache.

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \end{aligned}$$

Umwandlung eines DFAs in einen äquivalenten regulären Ausdruck:

Beispiel 2.54



Äquivalentes Gleichungssystem:

X_i ist ein RE für die von q_i aus akzeptierte Sprache.

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Löse $X_2 \equiv bX_2 \mid (aX_1 \mid \epsilon)$ nach X_2 auf:

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Löse $X_2 \equiv bX_2 \mid (aX_1 \mid \epsilon)$ nach X_2 auf:

$$X_2 \equiv b^*(aX_1 \mid \epsilon)$$

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Löse $X_2 \equiv bX_2 \mid (aX_1 \mid \epsilon)$ nach X_2 auf:

$$X_2 \equiv b^*(aX_1 \mid \epsilon)$$

Zurück einsetzen:

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Löse $X_2 \equiv bX_2 \mid (aX_1 \mid \epsilon)$ nach X_2 auf:

$$X_2 \equiv b^*(aX_1 \mid \epsilon)$$

Zurück einsetzen:

$$X_1 \equiv a(b^*(aX_1 \mid \epsilon)) \mid bX_3$$

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Löse $X_2 \equiv bX_2 \mid (aX_1 \mid \epsilon)$ nach X_2 auf:

$$X_2 \equiv b^*(aX_1 \mid \epsilon)$$

Zurück einsetzen:

$$\begin{aligned} X_1 &\equiv a(\boxed{b^*(aX_1 \mid \epsilon)}) \mid bX_3 \\ X_3 &\equiv bX_1 \mid a(\boxed{b^*(aX_1 \mid \epsilon)}) \mid \epsilon \end{aligned}$$

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Löse $X_2 \equiv bX_2 \mid (aX_1 \mid \epsilon)$ nach X_2 auf:

$$X_2 \equiv b^*(aX_1 \mid \epsilon)$$

Zurück einsetzen:

$$\begin{aligned} X_1 &\equiv a(b^*(aX_1 \mid \epsilon)) \mid bX_3 \\ X_3 &\equiv bX_1 \mid a(b^*(aX_1 \mid \epsilon)) \mid \epsilon \end{aligned}$$

Ausmultiplizieren und X_i ausklammern:

$$X_1 \equiv$$

Lösen des Gleichungssystems:

$$\begin{aligned} X_1 &\equiv aX_2 \mid bX_3 \\ X_2 &\equiv aX_1 \mid bX_2 \mid \epsilon \\ X_3 &\equiv bX_1 \mid aX_2 \mid \epsilon \end{aligned}$$

Löse $X_2 \equiv bX_2 \mid (aX_1 \mid \epsilon)$ nach X_2 auf:

$$X_2 \equiv b^*(aX_1 \mid \epsilon)$$

Zurück einsetzen:

$$\begin{aligned} X_1 &\equiv a(b^*(aX_1 \mid \epsilon)) \mid bX_3 \\ X_3 &\equiv bX_1 \mid a(b^*(aX_1 \mid \epsilon)) \mid \epsilon \end{aligned}$$

Ausmultiplizieren und X_i ausklammern:

$$\begin{aligned} X_1 &\equiv ab^*aX_1 \mid ab^* \mid bX_3 \\ X_3 &\equiv (b \mid ab^*a)X_1 \mid ab^* \mid \epsilon \end{aligned}$$

$$\begin{aligned} X_1 &\equiv ab^*aX_1 \mid bX_3 \mid ab^* \\ X_3 &\equiv (b \mid ab^*a)X_1 \mid ab^* \mid \epsilon \end{aligned}$$

X_3 ist gelöst, in 1. Gleichung einsetzen:

$$X_1 \equiv ab^*aX_1 \mid b((b \mid ab^*a)X_1 \mid ab^* \mid \epsilon) \mid ab^*$$

$$\begin{aligned} X_1 &\equiv ab^*aX_1 \mid bX_3 \mid ab^* \\ X_3 &\equiv (b \mid ab^*a)X_1 \mid ab^* \mid \epsilon \end{aligned}$$

X_3 ist gelöst, in 1. Gleichung einsetzen:

$$X_1 \equiv ab^*aX_1 \mid b((b \mid ab^*a)X_1 \mid ab^* \mid \epsilon) \mid ab^*$$

Ausmultiplizieren und X_1 ausklammern:

$$X_1 \equiv (ab^*a \mid bb \mid bab^*a)X_1$$

$$\begin{aligned} X_1 &\equiv ab^*aX_1 \mid bX_3 \mid ab^* \\ X_3 &\equiv (b \mid ab^*a)X_1 \mid ab^* \mid \epsilon \end{aligned}$$

X_3 ist gelöst, in 1. Gleichung einsetzen:

$$X_1 \equiv ab^*aX_1 \mid b((b \mid ab^*a)X_1 \mid ab^* \mid \epsilon) \mid ab^*$$

Ausmultiplizieren und X_1 ausklammern:

$$X_1 \equiv (ab^*a \mid bb \mid bab^*a)X_1 \mid bab^* \mid b \mid ab^*$$

$$\begin{aligned} X_1 &\equiv ab^*aX_1 \mid bX_3 \mid ab^* \\ X_3 &\equiv (b \mid ab^*a)X_1 \mid ab^* \mid \epsilon \end{aligned}$$

X_3 ist gelöst, in 1. Gleichung einsetzen:

$$X_1 \equiv ab^*aX_1 \mid b((b \mid ab^*a)X_1 \mid ab^* \mid \epsilon) \mid ab^*$$

Ausmultiplizieren und X_1 ausklammern:

$$X_1 \equiv (ab^*a \mid bb \mid bab^*a)X_1 \mid bab^* \mid b \mid ab^*$$

Nach X_1 auflösen:

$$X_1 \equiv (ab^*a \mid bb \mid bab^*a)^*(bab^* \mid b \mid ab^*)$$

Umwandlung eines FAs in einen äquivalenten regulären Ausdruck:

- Wandle FA mit n Zuständen in ein System von n Gleichungen mit n Variablen um:

$$X_i \equiv a_{i1}X_1 \mid \cdots \mid a_{in}X_n \mid b_i$$

Umwandlung eines FAs in einen äquivalenten regulären Ausdruck:

- Wandle FA mit n Zuständen in ein System von n Gleichungen mit n Variablen um:

$$X_i \equiv a_{i1}X_1 \mid \cdots \mid a_{in}X_n \mid b_i$$

$$\begin{aligned} a_{ij} \quad := \quad & c_1 \mid \cdots \mid c_k \quad \text{falls } \{c_1, \ldots, c_k\} = \{c \in \Sigma \mid q_i \xrightarrow{c} q_j\} \\ & \text{wobei } a_{ij} := \emptyset \text{ falls } q_i \xrightarrow{c} q_j \text{für kein } c \in \Sigma \end{aligned}$$

Umwandlung eines FAs in einen äquivalenten regulären Ausdruck:

- Wandle FA mit n Zuständen in ein System von n Gleichungen mit n Variablen um:

$$X_i \equiv a_{i1}X_1 \mid \cdots \mid a_{in}X_n \mid b_i$$

$$a_{ij} \quad := \quad c_1 \mid \cdots \mid c_k \quad \text{falls } \{c_1, \ldots, c_k\} = \{c \in \Sigma \mid q_i \xrightarrow{c} q_j\}$$
$$\text{wobei } a_{ij} := \emptyset \text{ falls } q_i \xrightarrow{c} q_j \text{für kein } c \in \Sigma$$

$$b_i \quad := \quad \begin{cases} \epsilon & \text{falls } q_i \in F \\ \emptyset & \text{sonst} \end{cases}$$

Umwandlung eines FAs in einen äquivalenten regulären Ausdruck:

- Wandle FA mit n Zuständen in ein System von n Gleichungen mit n Variablen um:

$$X_i \equiv a_{i1}X_1 \mid \cdots \mid a_{in}X_n \mid b_i$$

$$a_{ij} \quad := \quad c_1 \mid \cdots \mid c_k \quad \text{falls } \{c_1, \ldots, c_k\} = \{c \in \Sigma \mid q_i \xrightarrow{c} q_j\}$$
$$\text{wobei } a_{ij} := \emptyset \text{ falls } q_i \xrightarrow{c} q_j \text{für kein } c \in \Sigma$$

$$b_i \quad := \quad \begin{cases} \epsilon & \text{falls } q_i \in F \\ \emptyset & \text{sonst} \end{cases}$$

- Löse das System durch schrittweise Elimination von Variablen mit Hilfe von Ardens Lemma für REs (Korollar 2.53).

Umwandlung eines FAs in einen äquivalenten regulären Ausdruck:

- Wandle FA mit n Zuständen in ein System von n Gleichungen mit n Variablen um:

$$X_i \equiv a_{i1}X_1 \mid \cdots \mid a_{in}X_n \mid b_i$$

$$a_{ij} \quad := \quad c_1 \mid \cdots \mid c_k \quad \text{falls } \{c_1, \ldots, c_k\} = \{c \in \Sigma \mid q_i \xrightarrow{c} q_j\}$$
$$\text{wobei } a_{ij} := \emptyset \text{ falls } q_i \xrightarrow{c} q_j \text{für kein } c \in \Sigma$$

$$b_i \quad := \quad \begin{cases} \epsilon & \text{falls } q_i \in F \\ \emptyset & \text{sonst} \end{cases}$$

- Löse das System durch schrittweise Elimination von Variablen mit Hilfe von Ardens Lemma für REs (Korollar 2.53).

- Ist k der Startzustand, so beschreibt X_k die vom Automaten akzeptierte Sprache.

Das System in Matrix-Schreibweise, mit $+$ statt $|$:

$$x = Ax + b$$

Das System in Matrix-Schreibweise, mit $+$ statt $|$:

$$x = Ax + b$$

wobei

$$x = \begin{pmatrix} X_1 \\ \vdots \\ X_n \end{pmatrix} \quad A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

Matrix-Addition und Multiplikation sind wie üblich definiert,

---

Das System in Matrix-Schreibweise, mit $+$ statt $|$:

$$x = Ax + b$$

wobei

$$x = \begin{pmatrix} X_1 \\ \vdots \\ X_n \end{pmatrix} \quad A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

Matrix-Addition und Multiplikation sind wie üblich definiert, aber auf der Element-Ebene mit

- REs statt Zahlen,
- Konkatenation statt Multiplikation,
- Alternative statt Addition.

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} \boxed{aX_1} + \boxed{bX_2} \\ cX_1 + dX_2 \end{pmatrix}$$

Das System in Matrix-Schreibweise, mit $+$ statt $|$:

$$x = Ax + b \qquad x = A^* b$$

wobei

$$x = \begin{pmatrix} X_1 \\ \vdots \\ X_n \end{pmatrix} \quad A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

Matrix-Addition und Multiplikation sind wie üblich definiert, aber auf der Element-Ebene mit

- REs statt Zahlen,
- Konkatenation statt Multiplikation,
- Alternative statt Addition.

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} aX_1 + bX_2 \\ cX_1 + dX_2 \end{pmatrix}$$

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} aX_1 + bX_2 \\ cX_1 + dX_2 \end{pmatrix}$$

Achtung: Die Einträge in den Matrizen sind beliebige REs!
Wenn wir $+$ und $\cdot$ auf Matrizen haben, was könnte * sein?

$$A^* := A^0 + A^1 + A^2 + \cdots$$

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} aX_1 + bX_2 \\ cX_1 + dX_2 \end{pmatrix}$$

Achtung: Die Einträge in den Matrizen sind beliebige REs!
Wenn wir $+$ und $\cdot$ auf Matrizen haben, was könnte * sein?

$$A^* := A^0 + A^1 + A^2 + \cdots$$

Existiert das? Was ist das?

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} aX_1 + bX_2 \\ cX_1 + dX_2 \end{pmatrix}$$

Achtung: Die Einträge in den Matrizen sind beliebige REs!
Wenn wir $+$ und $\cdot$ auf Matrizen haben, was könnte * sein?

$$A^* := A^0 + A^1 + A^2 + \cdots$$

Existiert das? Was ist das?
Es gilt: $A(i,j)$ beschreibt alle Wege von i nach j der Länge 1.

$$A^0 = \begin{pmatrix} \varepsilon & & & \emptyset \\ & \varepsilon & & \\ & & \varepsilon & \\ \emptyset & & & \varepsilon \end{pmatrix}$$

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} aX_1 + bX_2 \\ cX_1 + dX_2 \end{pmatrix}$$

Achtung: Die Einträge in den Matrizen sind beliebige REs!
Wenn wir $+$ und $\cdot$ auf Matrizen haben, was könnte * sein?

$$A^* := A^0 + A^1 + A^2 + \cdots$$

Existiert das? Was ist das?
Es gilt: $A(i,j)$ beschreibt alle Wege von i nach j der Länge 1.
$A^n(i,j)$ beschreibt alle Wege von i nach j der Länge n.
Daher sollte gelten:
$A^*(i,j)$ beschreibt alle Wege von i nach j endlicher Länge.

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} aX_1 + bX_2 \\ cX_1 + dX_2 \end{pmatrix}$$

Achtung: Die Einträge in den Matrizen sind beliebige REs!
Wenn wir $+$ und $\cdot$ auf Matrizen haben, was könnte * sein?

$$A^* := A^0 + A^1 + A^2 + \cdots$$

Existiert das? Was ist das?
Es gilt: $A(i,j)$ beschreibt alle Wege von i nach j der Länge 1.
$A^n(i,j)$ beschreibt alle Wege von i nach j der Länge n.
Daher sollte gelten:
$A^*(i,j)$ beschreibt alle Wege von i nach j endlicher Länge.

Aus dem Beweis des Satzes von Kleene wissen wir:
$A^*(i,j)$ existiert und ist der reguläre Ausdruck α_{ij}^n.

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} aX_1 + bX_2 \\ cX_1 + dX_2 \end{pmatrix}$$

Achtung: Die Einträge in den Matrizen sind beliebige REs!
Wenn wir $+$ und $\cdot$ auf Matrizen haben, was könnte * sein?

$$A^* := A^0 + A^1 + A^2 + \cdots$$

Existiert das? Was ist das?
Es gilt: $A(i,j)$ beschreibt alle Wege von i nach j der Länge 1.
$A^n(i,j)$ beschreibt alle Wege von i nach j der Länge n.
Daher sollte gelten:
$A^*(i,j)$ beschreibt alle Wege von i nach j endlicher Länge.

Beispiel 2.55

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} aX_1 + bX_2 \\ cX_1 + dX_2 \end{pmatrix}$$

Satz 2.56

Sei A eine $n \times n$ Matrix regulärer Ausdrücke.
Seien x und b Vektoren der Größe n.
Falls $\epsilon \notin L(A(i,j))$ für alle i, j, dann gilt

$$x = Ax + b \implies x = A^*b$$

Satz 2.56

Sei A eine $n \times n$ Matrix regulärer Ausdrücke.
Seien x und b Vektoren der Größe n.
Falls $\epsilon \notin L(A(i,j))$ für alle i, j, dann gilt

$$x = Ax + b \implies x = A^*b$$

Beweis: zB Kozen.
Berechnung von A^*: zB Kozen oder Satz von Kleene.

Satz 2.56

Sei A eine $n \times n$ Matrix regulärer Ausdrücke.
Seien x und b Vektoren der Größe n.
Falls $\epsilon \notin L(A(i,j))$ für alle i, j, dann gilt

$$x = Ax + b \implies x = A^*b$$

Beweis: zB Kozen.
Berechnung von A^*: zB Kozen oder Satz von Kleene.

Bemerkung:
Die Nebenbedingung $\epsilon \notin L(A(i,j))$ ist automatisch gegeben, wenn A von einem Automaten ohne ϵ-Übergänge stammt,

Wann wendet man welches Verfahren zum Lösen von Gleichungssystemen an:

Wann wendet man welches Verfahren zum Lösen von Gleichungssystemen an:

- Schrittweises Lösen des Gleichungssystems mit Ardens Lemma: für Berechnungen per Hand.

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

$$X = A(AX \cup B) \cup B$$

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

$$X = A(AX \cup B) \cup B = A^2X \cup AB \cup B$$

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

$$\begin{aligned} X &= A(AX \cup B) \cup B = A^2X \cup AB \cup B \\ &= A^2(AX \cup B) \cup AB \cup B \end{aligned}$$

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

$$\begin{aligned} X &= A(AX \cup B) \cup B = A^2X \cup AB \cup B \\ &= A^2(AX \cup B) \cup AB \cup B = A^3X \cup A^2B \cup AB \cup B = \dots \end{aligned}$$

Mit Induktion zeigt man für alle $n \in \mathbb{N}$:

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^iB$$

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

$$\begin{aligned} X &= A(AX \cup B) \cup B = A^2X \cup AB \cup B \\ &= A^2(AX \cup B) \cup AB \cup B = A^3X \cup A^2B \cup AB \cup B = \dots \end{aligned}$$

Mit Induktion zeigt man für alle $n \in \mathbb{N}$:

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^iB$$

0: Behauptung wird zu $X = AX \cup B$, der Annahme.

$n+1$:

$$\begin{aligned} X &= AX \cup B \\ &= A(A^{n+1}X \cup \textstyle\bigcup_{i \leq n} A^iB) \cup B \end{aligned}$$

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

$$\begin{aligned} X &= A(AX \cup B) \cup B = A^2X \cup AB \cup B \\ &= A^2(AX \cup B) \cup AB \cup B = A^3X \cup A^2B \cup AB \cup B = \dots \end{aligned}$$

Mit Induktion zeigt man für alle $n \in \mathbb{N}$:

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^iB$$

0: Behauptung wird zu $X = AX \cup B$, der Annahme.

$n+1$:

$$\begin{aligned} X &= AX \cup B \\ &= A(A^{n+1}X \cup \textstyle\bigcup_{i \leq n} A^iB) \cup B \\ &= A^{n+2}X \cup (\textstyle\bigcup_{i \leq n} A^{i+1}B) \cup B \end{aligned}$$

$0 \leq$

Beweis

von Ardens Lemma:
Wir nehmen an $X = AX \cup B$.

$$\begin{aligned} X &= A(AX \cup B) \cup B = A^2X \cup AB \cup B \\ &= A^2(AX \cup B) \cup AB \cup B = A^3X \cup A^2B \cup AB \cup B = \dots \end{aligned}$$

Mit Induktion zeigt man für alle $n \in \mathbb{N}$:

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^iB$$

0: Behauptung wird zu $X = AX \cup B$, der Annahme.

$n+1$:

$$\begin{aligned} X &= AX \cup B \\ &= A(A^{n+1}X \cup \textstyle\bigcup_{i \leq n} A^iB) \cup B \\ &= A^{n+2}X \cup (\textstyle\bigcup_{i \leq n} A^{i+1}B) \cup B \\ &= A^{n+2}X \cup (\textstyle\bigcup_{1 \leq i \leq n+1} A^{i}B) \cup B \\ &= A^{n+2}X \cup \textstyle\bigcup_{i \leq n+1} A^{i}B \end{aligned}$$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^iB$$

Wir zeigen nun $X = A^*B$.

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^iB$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X$: $\quad w \in A^*B = \bigcup_{i \in \mathbb{N}} A^iB$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$\boxed{A^*B \subseteq X:}$ $w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$\implies \exists n.\ w \in A^n B$

$\implies w \in X$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X:$ $w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$\implies \exists n.\ w \in A^n B$

$\implies w \in X$

$X \subseteq A^*B:$ Sei $w \in X$ und $n := |w|$.

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X:$ $w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$\implies \exists n.\ w \in A^n B$

$\implies w \in X$

$X \subseteq A^*B:$ Sei $w \in X$ und $n := |w|$.

$\epsilon \notin A$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X:$ $w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$\implies \exists n.\ w \in A^n B$

$\implies w \in X$

$X \subseteq A^*B:$ Sei $w \in X$ und $n := |w|$.

$\epsilon \notin A$

$\implies \forall u \in A^{n+1}.\ |u| \geq n+1$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X$: $\quad w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$$\implies \exists n.\ w \in A^n B$$

$$\implies w \in X$$

$X \subseteq A^*B$: $\quad$ Sei $w \in X$ und $n := |w|$.

$$\epsilon \notin A$$

$$\implies \forall u \in A^{n+1}.\ |u| \geq n+1$$

$$\implies w \notin \boxed{A^{n+1}X}$$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X$: $\quad w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$$\implies \exists n.\ w \in A^n B$$

$$\implies w \in X$$

$X \subseteq A^*B$: $\quad$ Sei $w \in X$ und $n := |w|$.

$$\epsilon \notin A$$

$$\implies \forall u \in A^{n+1}.\ |u| \geq n+1$$

$$\implies w \notin A^{n+1}X$$

$$\implies w \in \bigcup_{i \leq n} A^i B$$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X$: $\quad w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$$\implies \exists n.\ w \in A^n B$$

$$\implies w \in X$$

$X \subseteq A^*B$: $\quad$ Sei $w \in X$ und $n := |w|$.

$$\epsilon \notin A$$

$$\implies \forall u \in A^{n+1}.\ |u| \geq n+1$$

$$\implies w \notin A^{n+1}X$$

$$\implies w \in \bigcup_{i \leq n} A^i B \subseteq \bigcup_{i \in \mathbb{N}} A^i B$$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X$: $\quad w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$$\implies \exists n.\ w \in A^n B$$

$$\implies w \in X$$

$X \subseteq A^*B$: $\quad$ Sei $w \in X$ und $n := |w|$.

$$\epsilon \notin A$$

$$\implies \forall u \in A^{n+1}.\ |u| \geq n+1$$

$$\implies w \notin A^{n+1}X$$

$$\implies w \in \bigcup_{i \leq n} A^i B \subseteq \bigcup_{i \in \mathbb{N}} A^i B = A^*B$$

$\square$

$$X = A^{n+1}X \cup \bigcup_{i \leq n} A^i B$$

Wir zeigen nun $X = A^*B$.

$A^*B \subseteq X$: $w \in A^*B = \bigcup_{i \in \mathbb{N}} A^i B$

$\implies \exists n.\ w \in A^n B$

$\implies w \in X$

$X \subseteq A^*B$: Sei $w \in X$ und $n := |w|$.

$\epsilon \notin A$

$\implies \forall u \in A^{n+1}.\ |u| \geq n+1$

$\implies w \notin A^{n+1}X$

$\implies w \in \bigcup_{i \leq n} A^i B \subseteq \bigcup_{i \in \mathbb{N}} A^i B = A^*B$

$\square$

2.12 Minimierung endlicher Automaten

1. Beispiele
2. Algorithmen
3. Minimalitätsbeweis

Der Algorithmus zur Minimierung eines DFA:

1. Entferne alle von q_0 aus nicht erreichbaren Zustände.

Der Algorithmus zur Minimierung eines DFA:

1. Entferne alle von q_0 aus nicht erreichbaren Zustände.
2. Berechne die *äquivalenten* Zustände des Automaten.

Der Algorithmus zur Minimierung eines DFA:

1. Entferne alle von q_0 aus nicht erreichbaren Zustände.
2. Berechne die *äquivalenten* Zustände des Automaten.
3. Kollabiere den Automaten durch Zusammenfassung aller äquivalenten Zustände.

Zustände p und q sind unterscheidbar wenn es $w \in \Sigma^*$ gibt mit $\hat{\delta}(p, w) \in F$ und $\hat{\delta}(q, w) \notin F$ oder umgekehrt.

Der Algorithmus zur Minimierung eines DFA:

1. Entferne alle von q_0 aus nicht erreichbaren Zustände.
2. Berechne die *äquivalenten* Zustände des Automaten.
3. Kollabiere den Automaten durch Zusammenfassung aller äquivalenten Zustände.

Zustände p und q sind unterscheidbar wenn es $w \in \Sigma^*$ gibt mit $\hat{\delta}(p, w) \in F$ und $\hat{\delta}(q, w) \notin F$ oder umgekehrt.
Zustände sind äquivalent wenn sie nicht unterscheidbar sind, d.h. wenn für alle $w \in \Sigma^*$ gilt:

$$\hat{\delta}(p, w) \in F \quad \Leftrightarrow \quad \hat{\delta}(q, w) \in F$$

Der Algorithmus zur Minimierung eines DFA:

1. Entferne alle von q_0 aus nicht erreichbaren Zustände.
2. Berechne die *äquivalenten* Zustände des Automaten.
3. Kollabiere den Automaten durch Zusammenfassung aller äquivalenten Zustände.

Zustände p und q sind unterscheidbar wenn es $w \in \Sigma^*$ gibt mit $\hat{\delta}(p, w) \in F$ und $\hat{\delta}(q, w) \notin F$ oder umgekehrt.
Zustände sind äquivalent wenn sie nicht unterscheidbar sind, d.h. wenn für alle $w \in \Sigma^*$ gilt:

$$\hat{\delta}(p, w) \in F \quad \Leftrightarrow \quad \hat{\delta}(q, w) \in F$$

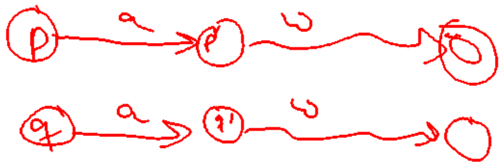
Der Algorithmus zur Minimierung eines DFA:

1. Entferne alle von q_0 aus nicht erreichbaren Zustände.
2. Berechne die *äquivalenten* Zustände des Automaten.
3. Kollabiere den Automaten durch Zusammenfassung aller äquivalenten Zustände.

Zustände p und q sind unterscheidbar wenn es $w \in \Sigma^*$ gibt mit $\hat{\delta}(p, w) \in F$ und $\hat{\delta}(q, w) \notin F$ oder umgekehrt.
Zustände sind äquivalent wenn sie nicht unterscheidbar sind, d.h. wenn für alle $w \in \Sigma^*$ gilt:

$$\hat{\delta}(p, w) \in F \quad \Leftrightarrow \quad \hat{\delta}(q, w) \in F$$

Sind $\delta(p, a)$ und $\delta(q, a)$ unterscheidbar, dann auch p und q.



Der Algorithmus zur Minimierung eines DFA:

1. Entferne alle von q_0 aus nicht erreichbaren Zustände.
2. Berechne die *äquivalenten* Zustände des Automaten.
3. Kollabiere den Automaten durch Zusammenfassung aller äquivalenten Zustände.

Zustände p und q sind unterscheidbar wenn es $w \in \Sigma^*$ gibt mit $\hat{\delta}(p, w) \in F$ und $\hat{\delta}(q, w) \notin F$ oder umgekehrt.
Zustände sind äquivalent wenn sie nicht unterscheidbar sind, d.h. wenn für alle $w \in \Sigma^*$ gilt:

$$\hat{\delta}(p, w) \in F \quad \Leftrightarrow \quad \hat{\delta}(q, w) \in F$$

Sind $\delta(p, a)$ und $\delta(q, a)$ unterscheidbar, dann auch p und q.

$\Rightarrow$ Unterscheidbarkeit pflanzt sich rückwärts fort.

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$
Ausgabe: Äquivalenzrelation auf Q.
Datenstruktur: Eine Menge U *ungeordneter* Paare $\{p, q\} \subseteq Q$.

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$
Ausgabe: Äquivalenzrelation auf Q.
Datenstruktur: Eine Menge U *ungeordneter* Paare $\{p, q\} \subseteq Q$.
Algorithmus U:

1. $U := \{\{p, q\} \mid p \in F \land q \notin F\}$

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$

Ausgabe: Äquivalenzrelation auf Q.

Datenstruktur: Eine Menge U *ungeordneter* Paare $\{p, q\} \subseteq Q$.

Algorithmus U:

1. $U := \{\{p, q\} \mid p \in F \land q \notin F\}$
2. **while** $\exists \{p, q\} \notin U.\ \exists a \in \Sigma.\ \{\delta(p, a), \delta(q, a)\} \in U$
   do $U := U \cup \{\{p, q\}\}$

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$

Ausgabe: Äquivalenzrelation auf Q.

Datenstruktur: Eine Menge U *ungeordneter* Paare $\{p, q\} \subseteq Q$.

Algorithmus U:

1. $U := \{\{p, q\} \mid p \in F \land q \notin F\}$
2. **while** $\exists \{p, q\} \notin U.\ \exists a \in \Sigma.\ \{\delta(p, a), \delta(q, a)\} \in U$
   do $U := U \cup \{\{p, q\}\}$

Invariante: $\{p, q\} \in U \implies p$ und q unterscheidbar

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$

Ausgabe: Äquivalenzrelation auf Q.

Datenstruktur: Eine Menge U *ungeordneter* Paare $\{p, q\} \subseteq Q$.

Algorithmus U:

1. $U := \{\{p, q\} \mid p \in F \land q \notin F\}$
2. **while** $\exists \{p, q\} \notin U.\ \exists a \in \Sigma.\ \{\delta(p, a), \delta(q, a)\} \in U$
   do $U := U \cup \{\{p, q\}\}$

Invariante: $\{p, q\} \in U \implies p$ und q unterscheidbar

Lemma 2.57

Am Ende gilt: U ist Menge aller unterscheidbaren Zustände.

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$

Ausgabe: Äquivalenzrelation auf Q.

Datenstruktur: Eine Menge U *ungeordneter* Paare $\{p, q\} \subseteq Q$.

Algorithmus U:

1. $U := \{\{p, q\} \mid p \in F \land q \notin F\}$
2. **while** $\exists \{p, q\} \notin U.\ \exists a \in \Sigma.\ \{\delta(p, a), \delta(q, a)\} \in U$
   do $U := U \cup \{\{p, q\}\}$

Invariante: $\{p, q\} \in U \implies p$ und q unterscheidbar

Lemma 2.57

Am Ende gilt: U ist Menge aller unterscheidbaren Zustände.

Beweis:

$\{p, q\} \in U \implies p$ und q unterscheidbar: Invariante

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$

Ausgabe: Äquivalenzrelation auf Q.

Datenstruktur: Eine Menge U *ungeordneter* Paare $\{p, q\} \subseteq Q$.

Algorithmus U:

1. $U := \{\{p, q\} \mid p \in F \land q \notin F\}$
2. **while** $\exists \{p, q\} \notin U.\ \exists a \in \Sigma.\ \{\delta(p, a), \delta(q, a)\} \in U$
   do $U := U \cup \{\{p, q\}\}$

Invariante: $\{p, q\} \in U \implies p$ und q unterscheidbar

Lemma 2.57

Am Ende gilt: U ist Menge aller unterscheidbaren Zustände.

Beweis:

$\{p, q\} \in U \implies p$ und q unterscheidbar: Invariante

p und q unterscheidbar $\implies \{p, q\} \in U$:

Induktion über die Länge eines unterscheidenden Worts. □

Berechnung äquivalenter Zustände eines DFA

durch Berechnung der unterscheidbaren Zustände

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$

Ausgabe: Äquivalenzrelation auf Q.

Datenstruktur: Eine Menge U *ungeordneter* Paare $\{p, q\} \subseteq Q$.

Algorithmus U:

1. $U := \{\{p, q\} \mid p \in F \land q \notin F\}$
2. **while** $\exists \{p, q\} \notin U.\ \exists a \in \Sigma.\ \{\delta(p, a), \delta(q, a)\} \in U$
   do $U := U \cup \{\{p, q\}\}$

Invariante: $\{p, q\} \in U \implies p$ und q unterscheidbar

Lemma 2.57

Am Ende gilt: U ist Menge aller unterscheidbaren Zustände.

Beweis:

$\{p, q\} \in U \implies p$ und q unterscheidbar: Invariante

p und q unterscheidbar $\implies \{p, q\} \in U$:

Induktion über die Länge eines unterscheidenden Worts. □