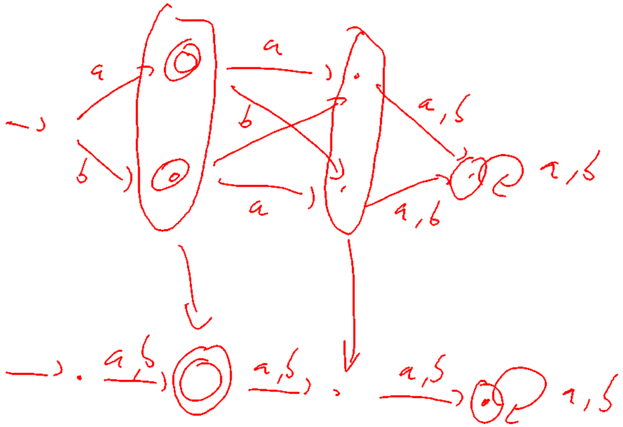
Script **generated by TTT**

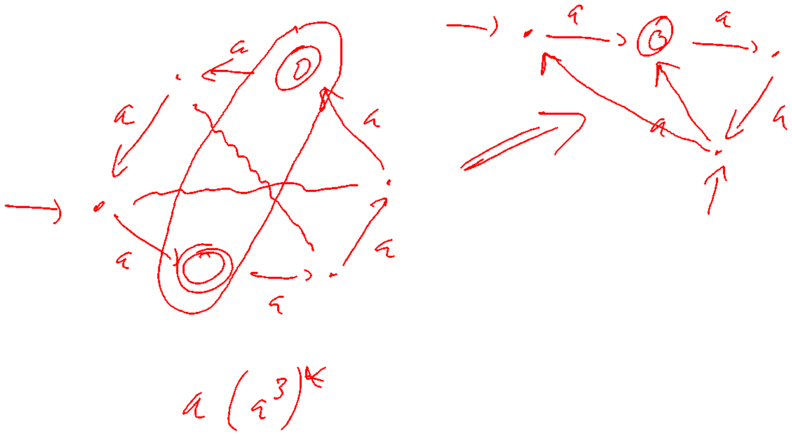
Title: Nipkow: Theo (13.05.2019)

Date: Mon May 13 14:22:41 CEST 2019

Duration: 82:42 min

Pages: 89





3.12 Minimierung endlicher Automaten

Wir zeigen, dass jede reguläre Sprache einen einzigen minimalen DFA hat und geben Algorithmen an, die diesen DFA konstruieren.

1. Beispiele
2. Algorithmen
3. Minimalitätsbeweis

Algorithmus zur Minimierung eines DFA

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

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.

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,

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:

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

- Gilt $p \in F$ und $q \notin F$, dann sind p und q unterscheidbar.

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:

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

- Gilt $p \in F$ und $q \notin F$, dann sind p und q unterscheidbar.

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:

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

- Gilt $p \in F$ und $q \notin F$, dann sind p und q unterscheidbar.
- Sind $\delta(p, a)$ und $\delta(q, a)$ unterscheidbar, dann auch p und q.

$$p \xrightarrow{a} \cdot \xrightarrow{w} \in F$$
$$q \xrightarrow{a} \cdot \xrightarrow{w} \notin F$$

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:

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

- Gilt $p \in F$ und $q \notin F$, dann sind p und q unterscheidbar.
- 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

Berechnung äquivalenter Zustände eines DFA

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

Berechnung äquivalenter Zustände eines DFA

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:

Berechnung äquivalenter Zustände eines DFA

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

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

Eingabe: DFA $A = (Q, \Sigma, \delta, q_0, F)$ $\bar{U} = \emptyset$

Ausgabe: Äquivalenzrelation auf Q.

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

Algorithmus U:

1. $U := \{\{p, q\} \mid p \in F \wedge q \notin F\}$
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 3.48
Am Ende gilt: U ist Menge aller unterscheidbaren Zustandspaare.

Berechnung äquivalenter Zustände eines DFA

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. $U := \{\{p, q\} \mid p \in F \wedge q \notin F\}$
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 3.48
Am Ende gilt: U ist Menge aller unterscheidbaren Zustandspaare.

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

Berechnung äquivalenter Zustände eines DFA

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. $U := \{\{p, q\} \mid p \in F \wedge q \notin F\}$ —
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 3.48
Am Ende gilt: U ist Menge aller unterscheidbaren Zustandspaare.

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

$w = a_1 \dots a_n \quad \{\hat{\delta}(p, w), \hat{\delta}(q, w)\} \in U$

Berechnung äquivalenter Zustände eines DFA

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

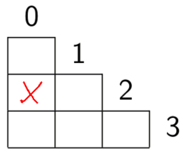
Invariante: $\{p, q\} \in U \implies p$ und q unterscheidbar

Lemma 3.48
Am Ende gilt: U ist Menge aller unterscheidbaren Zustandspaare.

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. $\square$

Implementierung von U:

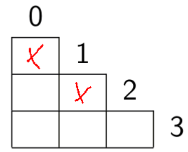
Tabelle von anfangs unmarkierten Paaren $\{p, q\}$, $p \neq q$.



Implementierung von U:

Tabelle von anfangs unmarkierten Paaren $\{p, q\}$, $p \neq q$.

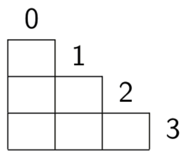
U



for all $p \in F$, $q \in Q \setminus F$ **do** markiere $\{p, q\}$

Implementierung von U:

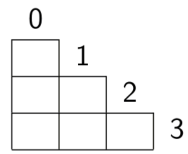
Tabelle von anfangs unmarkierten Paaren $\{p, q\}$, $p \neq q$.



for all $p \in F$, $q \in Q \setminus F$ **do** markiere $\{p, q\}$
while $\exists$ unmarkiertes $\{p, q\}$

Implementierung von U:

Tabelle von anfangs unmarkierten Paaren $\{p, q\}$, $p \neq q$.



for all $p \in F$, $q \in Q \setminus F$ **do** markiere $\{p, q\}$
while $\exists$ unmarkiertes $\{p, q\}$ $\exists a \in \Sigma$. $\{\delta(p, a), \delta(q, a)\}$ ist markiert

Implementierung von U:

Tabelle von anfangs unmarkierten Paaren $\{p,q\}$, $p \neq q$.

	0	1	2
1			
2			
3			

for all $p \in F$, $q \in Q \setminus F$ **do** markiere $\{p,q\}$
while $\exists$ unmarkiertes $\{p,q\}$ $\exists a \in \Sigma.$ $\{\delta(p,a), \delta(q,a)\}$ ist markiert
do markiere $\{p,q\}$

Implementierung von U:

Tabelle von anfangs unmarkierten Paaren $\{p,q\}$, $p \neq q$.

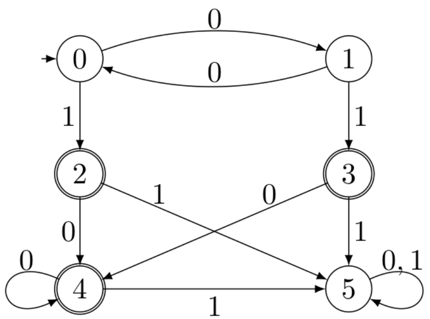
	0	1	2
1			
2			
3			

for all $p \in F$, $q \in Q \setminus F$ **do** markiere $\{p,q\}$
while $\exists$ unmarkiertes $\{p,q\}$ $\exists a \in \Sigma.$ $\{\delta(p,a), \delta(q,a)\}$ ist markiert
do markiere $\{p,q\}$

Komplexität:

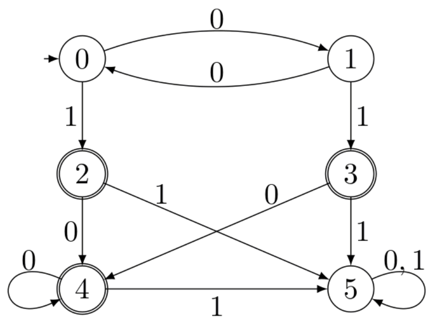
$$O\left(\binom{n}{2} + \binom{n}{2}\binom{n}{2}|\Sigma|\right) = O\left(\frac{n(n-1)}{2} + \frac{n^2(n-1)^2}{4}|\Sigma|\right)$$

Beispiel 3.49



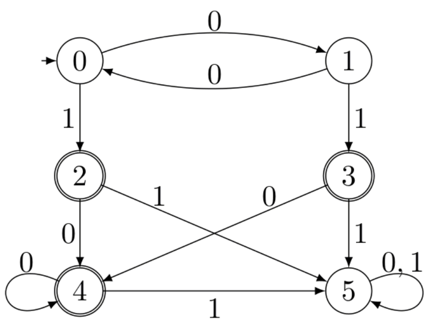
	0	1	2	3	4
1					
2	×	×			
3	×	×			
4	×	×			
5			×	×	×

Beispiel 3.49

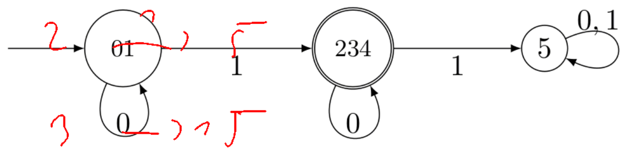


	0	1	2	3	4
1					
2	×	×			
3	×	×			
4	×	×			
5	×	×	×	×	×

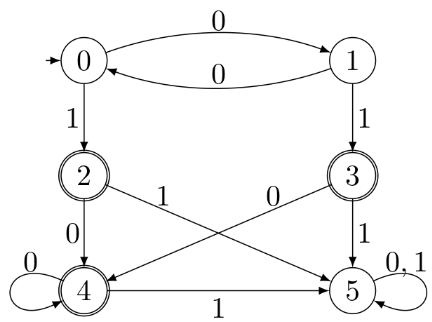
Beispiel 3.49



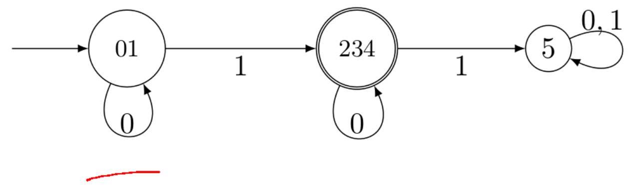
	0	1	2	3	4
1					
2	×	×			
3	×	×	*		
4	×	×			
5	×	×	×	×	×

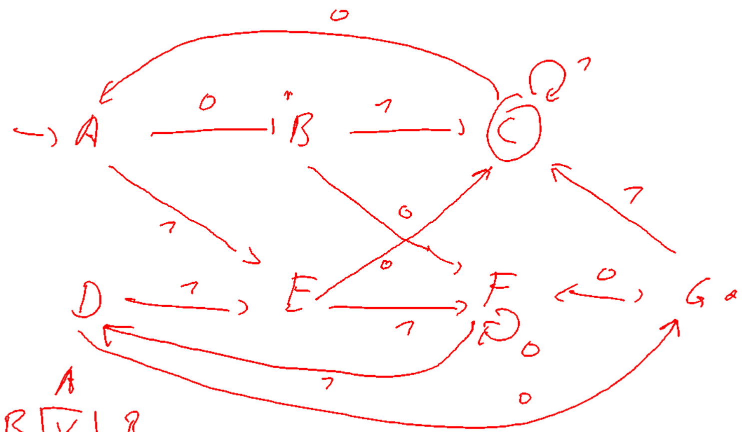


Beispiel 3.49



	0	1	2	3	4
1					
2	×	×			
3	×	×			
4	×	×			
5	×	×	×	×	×



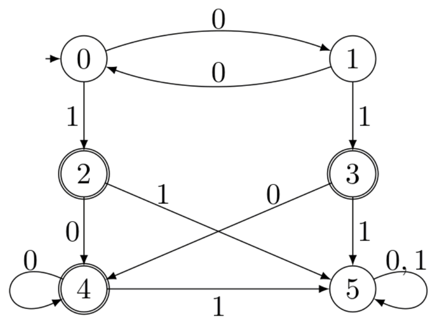


	A	B	C	D	E	F
B	X					
C	X	X				
D	=	X	X			
E	X	X	X	X		
F	X	X	X	X	X	
G	X	=	X	X	X	X

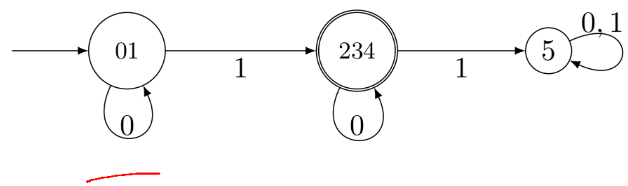
$A \approx D$

$B \approx G$

Beispiel 3.49



	0	1	2	3	4
1					
2	×	×			
3	×	×			
4	×	×			
5	×	×	×	×	×



Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

$\{p', q'\}$ unterscheidbar $\Longrightarrow$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

$\{p', q'\}$ unterscheidbar $\Longrightarrow$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

$D[\{p', q'\}]$: Menge der Paare $\{p, q\}$ wie oben, anfangs leer

Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

$\{p', q'\}$ unterscheidbar $\Longrightarrow$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

$D[\{p', q'\}]$: Menge der Paare $\{p, q\}$ wie oben, anfangs leer

for all $\{p, q\} \subseteq Q$ mit $p \neq q$,

Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

$\{p', q'\}$ unterscheidbar $\Longrightarrow$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

$D[\{p', q'\}]$: Menge der Paare $\{p, q\}$ wie oben, anfangs leer

for all $\{p, q\} \subseteq Q$ mit $p \neq q$, $a \in \Sigma$ **do**
  $p' := \delta(p, a)$; $q' := \delta(q, a)$
  if $p' \neq q'$ **then** $D[\{p', q'\}] := D[\{p', q'\}] \cup \{\{p, q\}\}$

$p \xrightarrow{a} p'$
$\neq$
$q \xrightarrow{a} q'$

Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

$\{p', q'\}$ unterscheidbar $\Longrightarrow$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

$D[\{p', q'\}]$: Menge der Paare $\{p, q\}$ wie oben, anfangs leer

for all $\{p, q\} \subseteq Q$ mit $p \neq q$, $a \in \Sigma$ **do**
  $p' := \delta(p, a)$; $q' := \delta(q, a)$
  if $p' \neq q'$ **then** $D[\{p', q'\}] := D[\{p', q'\}] \cup \{\{p, q\}\}$
for all $p' \in F$, $q' \in Q \setminus F$ **do** $mark(\{p', q'\})$

Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

$\{p', q'\}$ unterscheidbar $\Longrightarrow$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

$D[\{p', q'\}]$: Menge der Paare $\{p, q\}$ wie oben, anfangs leer

for all $\{p, q\} \subseteq Q$ mit $p \neq q$, $a \in \Sigma$ **do**
  $p' := \delta(p, a)$; $q' := \delta(q, a)$
  if $p' \neq q'$ **then** $D[\{p', q'\}] := D[\{p', q'\}] \cup \{\{p, q\}\}$
for all $p' \in F$, $q' \in Q \setminus F$ **do** $mark(\{p', q'\})$

$mark(\{p', q'\}) =$
  if $\{p', q'\}$ unmarkiert **then**

Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

$\{p', q'\}$ unterscheidbar $\Longrightarrow$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

$D[\{p', q'\}]$: Menge der Paare $\{p, q\}$ wie oben, anfangs leer

for all $\{p, q\} \subseteq Q$ mit $p \neq q$, $a \in \Sigma$ **do**
  $p' := \delta(p, a)$; $q' := \delta(q, a)$
  if $p' \neq q'$ **then** $D[\{p', q'\}] := D[\{p', q'\}] \cup \{\{p, q\}\}$
for all $p' \in F$, $q' \in Q \setminus F$ **do** $mark(\{p', q'\})$

$mark(\{p', q'\}) =$
  if $\{p', q'\}$ unmarkiert **then**
    markiere $\{p', q'\}$

Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

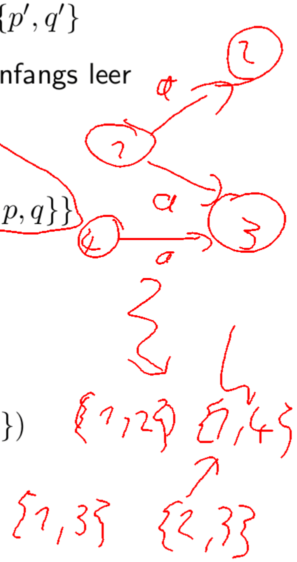
$\{p', q'\}$ unterscheidbar $\implies$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

$D[\{p', q'\}]$: Menge der Paare $\{p, q\}$ wie oben, anfangs leer

for all $\{p, q\} \subseteq Q$ mit $p \neq q,\ a \in \Sigma$ **do**
  $\cancel{p' := \delta(p, a);\ q' := \delta(q, a)}$
  if $p' \neq q'$ **then** $D[\{p', q'\}] := D[\{p', q'\}] \cup \{\{p, q\}\}$
for all $p' \in F,\ q' \in Q \setminus F$ **do** $mark(\{p', q'\})$

$mark(\{p', q'\}) =$
  if $\{p', q'\}$ unmarkiert **then**
    markiere $\{p', q'\}$
    for all $\{p, q\} \in D[\{p', q'\}]$ **do** $mark(\{p, q\})$

Komplexität: $O(n^2 + n^2) = O(n^2)$.



Von $O(n^4)$ zu $O(n^2)$ mit Abhängigkeitsanalyse:

$\{p', q'\}$ unterscheidbar $\implies$
$\{p, q\}$ unterscheidbar falls $\{p, q\} \xrightarrow{a} \{p', q'\}$

$D[\{p', q'\}]$: Menge der Paare $\{p, q\}$ wie oben, anfangs leer

for all $\{p, q\} \subseteq Q$ mit $p \neq q,\ a \in \Sigma$ **do**
  $p' := \delta(p, a);\ q' := \delta(q, a)$
  if $p' \neq q'$ **then** $D[\{p', q'\}] := D[\{p', q'\}] \cup \{\{p, q\}\}$
for all $p' \in F,\ q' \in Q \setminus F$ **do** $mark(\{p', q'\})$

$mark(\{p', q'\}) =$
  if $\{p', q'\}$ unmarkiert **then**
    markiere $\{p', q'\}$
    for all $\{p, q\} \in D[\{p', q'\}]$ **do** $mark(\{p, q\})$

Komplexität: $O(n^2 + n^2) = O(n^2)$.
Korrektheit?

Eine weitere Anwendung: Äquivalenztest von DFAs.

Eine weitere Anwendung: Äquivalenztest von DFAs.

1. Gegeben DFAs A und B, bilde disjunkte Vereiningung. („Male A und B nebeneinander.“)

Eine weitere Anwendung: Äquivalenztest von DFAs.

1. Gegeben DFAs A und B, bilde disjunkte Vereiningung. („Male A und B nebeneinander.“)
2. Berechne Menge der äquivalenten Zustände.

Der Minimierungsalgorithmus (zur Erinnerung).

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.

Formale Definition des “kollabierten Automaten”

Formale Definition des “kollabierten Automaten”

Eine Relation $\approx \subseteq A \times A$ ist eine Äquivalenzrelation falls

- Reflexivität: $\forall a \in A.\ a \approx a$
- Symmetrie: $\forall a, b \in A.\ a \approx b \implies b \approx a$
- Transitivität: $\forall a, b, c \in A.\ a \approx b \wedge b \approx c \implies a \approx c$

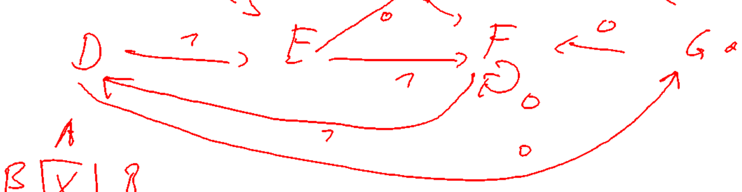
Äquivalenzklasse:

$$[a]_\approx \ := \ \{b \mid a \approx b\}$$

Formale Definition des "kollabierten Automaten"

Eine Relation $\approx\ \subseteq A \times A$ ist eine Äquivalenzrelation falls

- Reflexivität: $\forall a \in A.\ a \approx a$
- Symmetrie: $\forall a, b \in A.\ a \approx b \implies b \approx a$
- Transitivität: $\forall a, b, c \in A.\ a \approx b \wedge b \approx c \implies a \approx c$



	A	B	C	D	E	F
B	X					
C	X	X				
D	=	X	X			
E	X	X	X	X		
F	X	X	X	X	X	
G	X	=	X	X	X	X

$A \approx D$

$B \approx G$

$m \approx n \iff m \mid n \wedge n \mid m$

Formale Definition des "kollabierten Automaten"

$m \approx n \iff m \mid n \wedge n \mid m$

Eine Relation $\approx\ \subseteq A \times A$ ist eine Äquivalenzrelation falls

- Reflexivität: $\forall a \in A.\ a \approx a$
- Symmetrie: $\forall a, b \in A.\ a \approx b \implies b \approx a$
- Transitivität: $\forall a, b, c \in A.\ a \approx b \wedge b \approx c \implies a \approx c$

Äquivalenzklasse:

$$[a]_\approx \ := \ \{b \mid a \approx b\}$$

Formale Definition des "kollabierten Automaten"

$m \approx n \iff m \mid n \wedge n \mid m$

Eine Relation $\approx\ \subseteq A \times A$ ist eine Äquivalenzrelation falls

- Reflexivität: $\forall a \in A.\ a \approx a$ $\{0\}$ $\{1\}$ $\{1\}$ $\{2\} = \{2\}$ ...
- Symmetrie: $\forall a, b \in A.\ a \approx b \implies b \approx a$
- Transitivität: $\forall a, b, c \in A.\ a \approx b \wedge b \approx c \implies a \approx c$

Äquivalenzklasse:

$[0]_\approx$ $[1]_\approx = \{1\}$

$$[a]_\approx \ := \ \{b \mid a \approx b\}$$

Es gilt:

$$[a]_\approx = [b]_\approx \ \Leftrightarrow \ a \approx b$$

$\{(1,1), (2,2), \ldots, (6,6), (2,3), (3,2), (4,5), (5,4), (4,6), (6,4), (5,6), (6,5)\}$

$1 \mid 2 \approx 3 \mid 4 \approx 5 \approx 6$

Quotientenmenge:

$$A/\approx \ := \ \{[a]_\approx \mid a \in A\}$$

$m \approx n \iff m \mid n \wedge n \mid m$

$\{0\} \quad \{1,-1\} \quad \{2,-2\} \quad \ldots$

$[0]_\approx \quad [1]_\approx = [-1]_\approx$

$1 \quad 2 \approx 3 \quad 4 \approx 5 \approx 6$

$\{(1,1),(2,2),\ldots,(6,6),$
$(2,3),(3,2),$
$(4,5),(5,4),(4,6),(6,4),(5,6),(6,5)\}$

Formale Definition des "kollabierten Automaten"

Eine Relation $\approx \subseteq A \times A$ ist eine Äquivalenzrelation falls

- Reflexivität: $\forall a \in A.\ a \approx a$
- Symmetrie: $\forall a, b \in A.\ a \approx b \implies b \approx a$
- Transitivität: $\forall a, b, c \in A.\ a \approx b \wedge b \approx c \implies a \approx c$

Äquivalenzklasse:

$$[a]_\approx := \{b \mid a \approx b\}$$

$\mathbb{Z}/\approx\ = \{[0],[1],[2], \ldots\}$

Es gilt:

$$[a]_\approx = [b]_\approx \quad \Leftrightarrow \quad a \approx b$$

Quotientenmenge:

$$A/\approx\ :=\ \{[a]_\approx \mid a \in A\}$$

$A/\approx\ = \{[1],[2],[4]\}$

Im Folgenden sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DFA ohne unerreichbare Zustände.

Definition 3.50 (Äquivalenz von Zuständen)

$$p \equiv_M q \quad \Leftrightarrow \quad (\forall w \in \Sigma^*.\ \hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F)$$

Intuition: Zwei Zustände sind äquivalent gwd sie die selbe Sprache erkennen.

Im Folgenden sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DFA ohne unerreichbare Zustände.

Definition 3.50 (Äquivalenz von Zuständen)

$$p \equiv_M q \quad \Leftrightarrow \quad (\forall w \in \Sigma^*.\ \hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F)$$

Intuition: Zwei Zustände sind äquivalent gwd sie die selbe Sprache erkennen.

Wir schreiben $\equiv$ statt $\equiv_M$ wenn M klar ist.

Im Folgenden sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DFA ohne unerreichbare Zustände.

Definition 3.50 (Äquivalenz von Zuständen)

$$p \equiv_M q \quad \Leftrightarrow \quad (\forall w \in \Sigma^*.\ \hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F)$$

Intuition: Zwei Zustände sind äquivalent gwd sie die selbe Sprache erkennen.

Wir schreiben $\equiv$ statt $\equiv_M$ wenn M klar ist.

Einfache Fakten:

- $\equiv_M$ ist eine Äquivalenzrelation.
- $p \equiv_M q \implies \delta(p, a) \equiv_M \delta(q, a)$
- Algorithmus U liefert die unterscheidbaren Zustände, also $\not\equiv$.

Im Folgenden sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DFA ohne unerreichbare Zustände.

Definition 3.50 (Äquivalenz von Zuständen)

$$p \equiv_M q \quad \Leftrightarrow \quad (\forall w \in \Sigma^*.\ \hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F)$$

Intuition: Zwei Zustände sind äquivalent gwd sie die selbe Sprache erkennen.

Wir schreiben $\equiv$ statt $\equiv_M$ wenn M klar ist.

Einfache Fakten:

- $\equiv_M$ ist eine Äquivalenzrelation.
- $p \equiv_M q \implies \delta(p, a) \equiv_M \delta(q, a)$
- Algorithmus U liefert die unterscheidbaren Zustände, also $\not\equiv$.

In der weiteren Analyse beziehen wir uns direkt auf $\equiv$, nicht mehr auf den Algorithmus.

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

Definition 3.51 (Quotientenautomat)

$$M/\equiv \quad :=$$

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

Definition 3.51 (Quotientenautomat)

$$M/\equiv \quad := \quad (Q/\equiv,$$

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

Definition 3.51 (Quotientenautomat)

$$M/\!\equiv \; := \; (Q/\!\equiv,\ \Sigma,$$

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

Definition 3.51 (Quotientenautomat)

$$\begin{aligned} M/\!\equiv \; &:= \; (Q/\!\equiv,\ \Sigma, \delta',\ [q_0]_\equiv,\ F/\!\equiv) \\ \delta'([p]_\equiv, a) \; &:= \; \underline{[\delta(p,a)]_\equiv} \end{aligned}$$

$f : \mathbb{Z}/\!\approx \;\to \{-1, 0, 1\}$

$f([n]_\approx) = \mathrm{sgn}(n)$

$f([1]_\approx) = 1$

$f([-1]_\approx) = -1$

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

Definition 3.51 (Quotientenautomat)

$$\begin{aligned} M/\!\equiv \; &:= \; (Q/\!\equiv,\ \Sigma, \delta',\ [q_0]_\equiv,\ F/\!\equiv) \\ \delta'([p]_\equiv, a) \; &:= \; [\delta(p,a)]_\equiv \end{aligned}$$

Die Definition von δ' ist wohlgeformt da unabhängig von der Wahl des Repräsentanten p:

$$[p]_\equiv = [p']_\equiv$$

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

Definition 3.51 (Quotientenautomat)

$$\begin{aligned} M/\!\equiv \; &:= \; (Q/\!\equiv,\ \Sigma, \delta',\ [q_0]_\equiv,\ F/\!\equiv) \\ \delta'([p]_\equiv, a) \; &:= \; [\delta(p,a)]_\equiv \end{aligned}$$

Die Definition von δ' ist wohlgeformt da unabhängig von der Wahl des Repräsentanten p:

$$[p]_\equiv = [p']_\equiv \quad \Longrightarrow \quad p \equiv p'$$

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

minimal ⟺ minimal

Definition 3.51 (Quotientenautomat)

$\{0\}$ $\{1\}$ $\{2\}$

$Z/\approx = \{[0], [1], [2], \ldots\}$

$$\begin{array}{rcl} M/\equiv & := & (Q/\equiv,\ \Sigma, \delta',\ [q_0]_\equiv,\ F/\equiv) \\ \delta'([p]_\equiv, a) & := & [\delta(p,a)]_\equiv \end{array}$$

Die Definition von δ' ist wohlgeformt da unabhängig von der Wahl des Repräsentanten p:

$[0]$ $[1] = \{2, 1\}$

$$\cancel{[p]_\equiv = [p']_\equiv \quad \Longrightarrow \quad p \equiv p' \quad \Longrightarrow \quad \delta(p,a) \equiv \delta(p',a)}$$

$\{(1,1), (2,2), \ldots, (6,6),$
$(2,3), (3,2),$
$(4,5), (5,4), (4,6), (6,4), (5,6), (6,5)\}$

$A/\approx = \{[1], [2], [4]\}$

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

Definition 3.51 (Quotientenautomat)

$$\begin{array}{rcl} M/\equiv & := & (Q/\equiv,\ \Sigma, \delta',\ [q_0]_\equiv,\ F/\equiv) \\ \delta'([p]_\equiv, a) & := & [\delta(p,a)]_\equiv \end{array}$$

Die Definition von δ' ist wohlgeformt da unabhängig von der Wahl des Repräsentanten p:

$$\begin{array}{rcl} [p]_\equiv = [p']_\equiv & \Longrightarrow & p \equiv p' \quad \Longrightarrow \quad \delta(p,a) \equiv \delta(p',a) \\ & \Longrightarrow & [\delta(p,a)]_\equiv = [\delta(p',a)]_\equiv \end{array}$$

Die „Kollabierung“ von M bzgl. $\equiv$ ist der *Quotientenautomat*:

Definition 3.51 (Quotientenautomat)

$$\begin{array}{rcl} M/\equiv & := & (Q/\equiv,\ \Sigma, \delta',\ [q_0]_\equiv,\ F/\equiv) \\ \delta'([p]_\equiv, a) & := & [\delta(p,a)]_\equiv \end{array}$$

Die Definition von δ' ist wohlgeformt da unabhängig von der Wahl des Repräsentanten p:

$$\begin{array}{rcl} [p]_\equiv = [p']_\equiv & \Longrightarrow & p \equiv p' \quad \Longrightarrow \quad \delta(p,a) \equiv \delta(p',a) \\ & \Longrightarrow & [\delta(p,a)]_\equiv = [\delta(p',a)]_\equiv \end{array}$$

Lemma 3.52

$$L(M/\equiv) = L(M)$$

Beweis: Übung.

Minimalität des Quotientenautomaten

Eine Beobachtung: Für $p := \hat{\delta}(q_0, u)$ und $q := \hat{\delta}(q_0, v)$ gilt

$$p \equiv_M q \quad \Leftrightarrow \quad \forall w \in \Sigma^*.\ \underline{\hat{\delta}(p,w) \in F} \Leftrightarrow \underline{\hat{\delta}(q,w) \in F}$$

Minimalität des Quotientenautomaten

Eine Beobachtung: Für $p := \hat{\delta}(q_0, u)$ und $q := \hat{\delta}(q_0, v)$ gilt

$$\begin{aligned} p \equiv_M q \quad &\Leftrightarrow \quad \forall w \in \Sigma^*.\ \hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F \\ &\Leftrightarrow \quad \forall w \in \Sigma^*.\ \hat{\delta}(q_0, uw) \in F \Leftrightarrow \hat{\delta}(q_0, vw) \in F \end{aligned}$$

Minimalität des Quotientenautomaten

Eine Beobachtung: Für $p := \hat{\delta}(q_0, u)$ und $q := \hat{\delta}(q_0, v)$ gilt

$$\begin{aligned} p \equiv_M q \quad &\Leftrightarrow \quad \forall w \in \Sigma^*.\ \hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F \\ &\Leftrightarrow \quad \forall w \in \Sigma^*.\ \hat{\delta}(q_0, uw) \in F \Leftrightarrow \hat{\delta}(q_0, vw) \in F \\ &\Leftrightarrow \quad \forall w \in \Sigma^*.\ uw \in L(M) \Leftrightarrow vw \in L(M) \end{aligned}$$

Minimalität des Quotientenautomaten

Eine Beobachtung: Für $p := \hat{\delta}(q_0, u)$ und $q := \hat{\delta}(q_0, v)$ gilt

$$\begin{aligned} p \equiv_M q \quad &\Leftrightarrow \quad \forall w \in \Sigma^*.\ \hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F \\ &\Leftrightarrow \quad \forall w \in \Sigma^*.\ \hat{\delta}(q_0, uw) \in F \Leftrightarrow \hat{\delta}(q_0, vw) \in F \\ &\Leftrightarrow \quad \forall w \in \Sigma^*.\ uw \in L(M) \Leftrightarrow vw \in L(M) \end{aligned}$$

Definition 3.53
Jede Sprache $L \subseteq \Sigma^*$ induziert eine Äquivalenzrelation $\equiv_L \subseteq \Sigma^* \times \Sigma^*$:

Minimalität des Quotientenautomaten

Eine Beobachtung: Für $p := \hat{\delta}(q_0, u)$ und $q := \hat{\delta}(q_0, v)$ gilt

$$\begin{aligned} p \equiv_M q \quad &\Leftrightarrow \quad \forall w \in \Sigma^*.\ \hat{\delta}(p, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F \\ &\Leftrightarrow \quad \forall w \in \Sigma^*.\ \hat{\delta}(q_0, uw) \in F \Leftrightarrow \hat{\delta}(q_0, vw) \in F \\ &\Leftrightarrow \quad \forall w \in \Sigma^*.\ uw \in L(M) \Leftrightarrow vw \in L(M) \end{aligned}$$

Definition 3.53
Jede Sprache $L \subseteq \Sigma^*$ induziert eine Äquivalenzrelation $\equiv_L \subseteq \Sigma^* \times \Sigma^*$:

$$u \equiv_L v \quad \Leftrightarrow \quad \forall w \in \Sigma^*.\ uw \in L \Leftrightarrow vw \in L$$

Achtung

$p \equiv_M q$ ist eine Relation auf Zuständen von M

$u \equiv_L v$ ist eine Relation auf Wörtern

$$u \equiv_{L(M)} v \quad \Leftrightarrow \quad \hat{\delta}(q_0, u) \equiv_M \hat{\delta}(q_0, v)$$

wohldef

$$u \equiv_{L(M)} v \quad \Leftrightarrow \quad \hat{\delta}(q_0, u) \equiv_M \hat{\delta}(q_0, v)$$

inj

Da alle Zustände von q_0 erreichbar sind, gilt sogar: Die Abbildung

$$[u]_{\equiv_{L(M)}} \quad \mapsto \quad [\hat{\delta}(q_0, u)]_{\equiv_M}$$

ist eine Bijektion zwischen den $\equiv_{L(M)}$ und $\equiv_M$ Äquivalenzklassen.

$$u \equiv_{L(M)} v \quad \Leftrightarrow \quad \hat{\delta}(q_0, u) \equiv_M \hat{\delta}(q_0, v)$$

Da alle Zustände von q_0 erreichbar sind, gilt sogar: Die Abbildung

$$[u]_{\equiv_{L(M)}} \quad \mapsto \quad [\hat{\delta}(q_0, u)]_{\equiv_M}$$

ist eine Bijektion zwischen den $\equiv_{L(M)}$ und $\equiv_M$ Äquivalenzklassen.

Satz 3.54

Ist M ein DFA ohne unerreichbare Zustände, so ist der von Algorithmus U *berechnete Quotientenautomat $M/\equiv$ ein minimaler DFA für $L(M)$.*

$$u \equiv_{L(M)} v \quad \Leftrightarrow \quad \hat\delta(q_0,u) \equiv_M \hat\delta(q_0,v)$$

Da alle Zustände von q_0 erreichbar sind, gilt sogar: Die Abbildung

$$[u]_{\equiv_{L(M)}} \quad \mapsto \quad [\hat\delta(q_0,u)]_{\equiv_M}$$

ist eine Bijektion zwischen den $\equiv_{L(M)}$ und $\equiv_M$ Äquivalenzklassen.

Satz 3.54

Ist M ein DFA ohne unerreichbare Zustände, so ist der von Algorithmus U berechnete Quotientenautomat $M/\equiv$ ein minimaler DFA für $L(M)$.

Beweis:

Sei $L := L(M)$ und M' ein DFA mit $L(M') = L$. Dann gilt:

$$|Q'| \geq |Q'/\equiv_{M'}|$$

$m \simeq n \Leftrightarrow m \mid n \wedge n \mid m$

$$u \equiv_{L(M)} v \quad \Leftrightarrow \quad \hat\delta(q_0,u) \equiv_M \hat\delta(q_0,v)$$

Da alle Zustände von q_0 erreichbar sind, gilt sogar: Die Abbildung

$\{0\}$ $\{1,-1\}$ $\{2,-2\}$ ...

$\mathbb{Z}/\simeq = \{[0],[1],[2],$

$$[u]_{\equiv_{L(M)}} \quad \mapsto \quad [\hat\delta(q_0,u)]_{\equiv_M}$$

ist eine Bijektion zwischen den $\equiv_{L(M)}$ und $\equiv_M$ Äquivalenzklassen.

$[0]$ $[1] = \{-1\}$

Satz 3.54

Ist M ein DFA ohne unerreichbare Zustände, so ist der von Algorithmus U berechnete Quotientenautomat $M/\equiv$ ein minimaler DFA für $L(M)$.

$1 \simeq 2$, $4 \simeq 5$, 3, 6

$\{(1,1),(2,2),\dots,(6,6),$ $(1,2),(2,1),$ $(4,5),(5,4),(4,6),(6,4),(5,6),(6,5)\}$

Beweis:

Sei $L := L(M)$ und M' ein DFA mit $L(M') = L$. Dann gilt:

$$|Q'| \geq |Q'/\equiv_{M'}|$$

$A/\simeq = \{[1],[2],[4]\}$

$$u \equiv_{L(M)} v \quad \Leftrightarrow \quad \hat\delta(q_0,u) \equiv_M \hat\delta(q_0,v)$$

Da alle Zustände von q_0 erreichbar sind, gilt sogar: Die Abbildung

$$[u]_{\equiv_{L(M)}} \quad \mapsto \quad [\hat\delta(q_0,u)]_{\equiv_M}$$

ist eine Bijektion zwischen den $\equiv_{L(M)}$ und $\equiv_M$ Äquivalenzklassen.

Satz 3.54

Ist M ein DFA ohne unerreichbare Zustände, so ist der von Algorithmus U berechnete Quotientenautomat $M/\equiv$ ein minimaler DFA für $L(M)$.

Beweis:

Sei $L := L(M)$ und M' ein DFA mit $L(M') = L$. Dann gilt:

$$|Q'| \geq |Q'/\equiv_{M'}| = |\Sigma^*/\equiv_L|$$

$$u \equiv_{L(M)} v \quad \Leftrightarrow \quad \hat\delta(q_0,u) \equiv_M \hat\delta(q_0,v)$$

Da alle Zustände von q_0 erreichbar sind, gilt sogar: Die Abbildung

$$[u]_{\equiv_{L(M)}} \quad \mapsto \quad [\hat\delta(q_0,u)]_{\equiv_M}$$

ist eine Bijektion zwischen den $\equiv_{L(M)}$ und $\equiv_M$ Äquivalenzklassen.

Satz 3.54

Ist M ein DFA ohne unerreichbare Zustände, so ist der von Algorithmus U berechnete Quotientenautomat $M/\equiv$ ein minimaler DFA für $L(M)$.

Beweis:

Sei $L := L(M)$ und M' ein DFA mit $L(M') = L$. Dann gilt:

$$|Q'| \geq |Q'/\equiv_{M'}| = |\Sigma^*/\equiv_L| = |Q/\equiv_M| \qquad \square$$

Es gilt sogar:

Fakt 3.55
Alle Quotientenautomaten $M/{\equiv_M}$ für die gleiche Sprache $L(M)$ haben die gleiche Struktur, d.h. sie unterscheiden sich nur durch eine Umbenennung der Zustände.

Es gilt sogar:

Fakt 3.55
Alle Quotientenautomaten $M/{\equiv_M}$ für die gleiche Sprache $L(M)$ haben die gleiche Struktur, d.h. sie unterscheiden sich nur durch eine Umbenennung der Zustände.

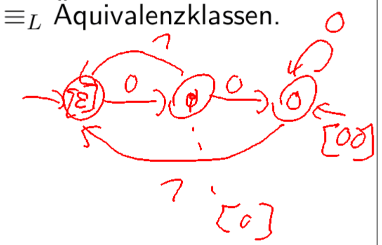
Daher beschriften wir die Zustände des kanonischen Minimalautomaten für eine Sprache L mit $\equiv_L$ Äquivalenzklassen.

Es gilt sogar:

Fakt 3.55
Alle Quotientenautomaten $M/{\equiv_M}$ für die gleiche Sprache $L(M)$ haben die gleiche Struktur, d.h. sie unterscheiden sich nur durch eine Umbenennung der Zustände.

Daher beschriften wir die Zustände des kanonischen Minimalautomaten für eine Sprache L mit $\equiv_L$ Äquivalenzklassen.

Beispiel 3.56
Sei $L := \{w \in \{0,1\}^* \mid w \text{ endet mit } 00\}$.



Definition 3.57 (Kanonischer Minimalautomat)

$$M_L \; :=$$

Definition 3.57 (Kanonischer Minimalautomat)

$$M_L := (\Sigma^*/{\equiv_L},\ \Sigma,\ \delta_L,\ [\epsilon]_{\equiv_L},\ F_L)$$

mit $\delta_L([w]_{\equiv_L}, a) := [wa]_{\equiv_L}$ und $F_L := \{[w]_{\equiv_L} \mid w \in L\}$.

Definition 3.57 (Kanonischer Minimalautomat)

$$M_L := (\Sigma^*/{\equiv_L},\ \Sigma,\ \delta_L,\ [\epsilon]_{\equiv_L},\ F_L)$$

mit $\delta_L([w]_{\equiv_L}, a) := [wa]_{\equiv_L}$ und $F_L := \{[w]_{\equiv_L} \mid w \in L\}$.

Man sieht: δ_L ist wohldefiniert und $\hat{\delta}_L([\epsilon]_{\equiv_L}, w) = [w]_{\equiv_L}$.
Dann gilt offensichtlich $L(M_L) = L$.

Definition 3.57 (Kanonischer Minimalautomat)

$$M_L := (\Sigma^*/{\equiv_L},\ \Sigma,\ \delta_L,\ [\epsilon]_{\equiv_L},\ F_L)$$

mit $\delta_L([w]_{\equiv_L}, a) := [wa]_{\equiv_L}$ und $F_L := \{[w]_{\equiv_L} \mid w \in L\}$.

Man sieht: δ_L ist wohldefiniert und $\hat{\delta}_L([\epsilon]_{\equiv_L}, w) = [w]_{\equiv_L}$.
Dann gilt offensichtlich $L(M_L) = L$.

Satz 3.58 (Myhill-Nerode)

Eine Sprache $L \subseteq \Sigma^*$ ist genau dann regulär, wenn $\equiv_L$ endlich viele Äquivalenzklassen hat.

Beweis:

„$\Longrightarrow$“: Ist L regulär, so wird L von einem DFA M akzeptiert.

Definition 3.57 (Kanonischer Minimalautomat)

$$M_L := (\Sigma^*/{\equiv_L},\ \Sigma,\ \delta_L,\ [\epsilon]_{\equiv_L},\ F_L)$$

mit $\delta_L([w]_{\equiv_L}, a) := [wa]_{\equiv_L}$ und $F_L := \{[w]_{\equiv_L} \mid w \in L\}$.

Man sieht: δ_L ist wohldefiniert und $\hat{\delta}_L([\epsilon]_{\equiv_L}, w) = [w]_{\equiv_L}$.
Dann gilt offensichtlich $L(M_L) = L$.

Satz 3.58 (Myhill-Nerode)

Eine Sprache $L \subseteq \Sigma^*$ ist genau dann regulär, wenn $\equiv_L$ endlich viele Äquivalenzklassen hat.

Beweis:

„$\Longrightarrow$“: Ist L regulär, so wird L von einem DFA M akzeptiert.
Daher hat $\equiv_L$ so viele Äquivalenzklassen wie $M/{\equiv_M}$ Zustände.

Definition 3.57 (Kanonischer Minimalautomat)

$$M_L \;:=\; (\Sigma^*/{\equiv_L},\ \Sigma,\ \delta_L,\ [\epsilon]_{\equiv_L},\ F_L)$$

mit $\delta_L([w]_{\equiv_L}, a) := [wa]_{\equiv_L}$ und $F_L := \{[w]_{\equiv_L} \mid w \in L\}$.

Man sieht: δ_L ist wohldefiniert und $\hat{\delta}_L([\epsilon]_{\equiv_L}, w) = [w]_{\equiv_L}$.
Dann gilt offensichtlich $L(M_L) = L$.

Satz 3.58 (Myhill-Nerode)

Eine Sprache $L \subseteq \Sigma^*$ ist genau dann regulär, wenn $\equiv_L$ endlich viele Äquivalenzklassen hat.

Beweis:
„$\Longrightarrow$“: Ist L regulär, so wird L von einem DFA M akzeptiert.
Daher hat $\equiv_L$ so viele Äquivalenzklassen wie $M/{\equiv_M}$ Zustände.
„$\Longleftarrow$“: Hat $\equiv_L$ endlich viele Äquivalenzklasse,
so ist M_L ein DFA mit $L(M_L) = L$. □

Wie sieht M_L aus wenn L *nicht* regulär ist?