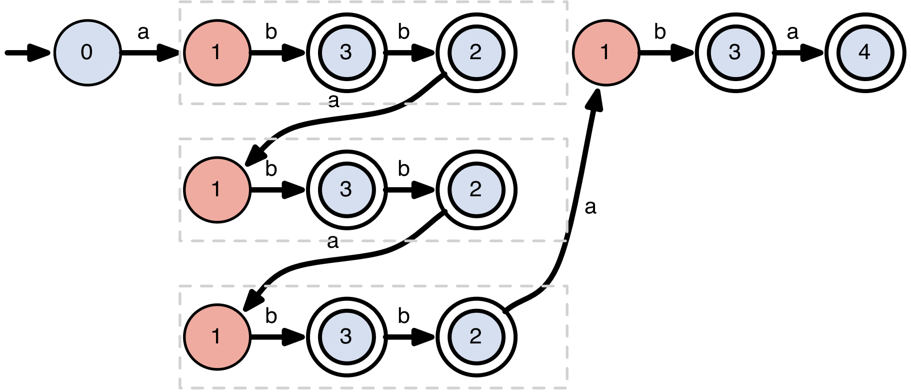
Berechenbarkeitstheorie

5. Vorlesung



Dr. Franziska Jahnke
Institut für Mathematische Logik und Grundlagenforschung
WWU Münster

WS 15/16

Äquivalenz von Zuständen

Äquivalenz von Zuständen

Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, dann heißen zwei Zustände $p, q \in Q$ **äquivalent** (Schreibweise $p \approx q$) gdw.

$$\forall z \in \Sigma^* : \quad \delta^*(p, z) \in F \Leftrightarrow \delta^*(q, z) \in F$$

Äquivalenz von Zuständen

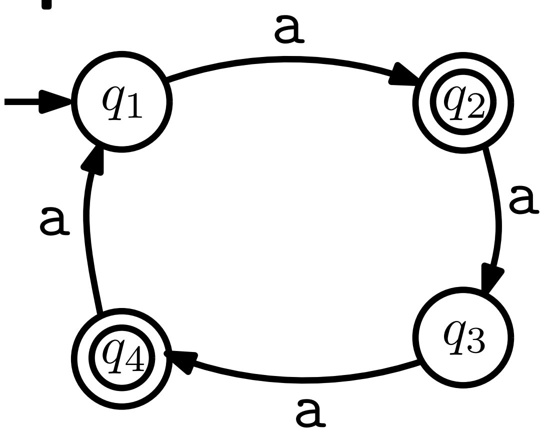
Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, dann heißen zwei Zustände $p, q \in Q$ **äquivalent** (Schreibweise $p \approx q$) gdw.

$$\forall z \in \Sigma^* : \qquad \delta^*(p, z) \in F \Leftrightarrow \delta^*(q, z) \in F$$

Zwei nicht äquivalente Zustände nennen wir auch trennbar.

Bsp.



Äquivalenz von Zuständen

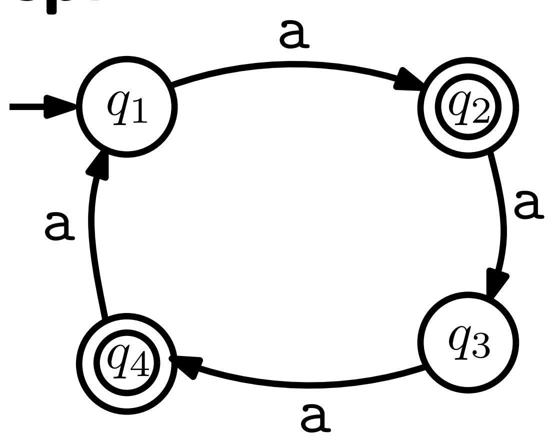
Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, dann heißen zwei Zustände $p, q \in Q$ **äquivalent** (Schreibweise $p \approx q$) gdw.

$$\forall z \in \Sigma^* : \qquad \delta^*(p, z) \in F \Leftrightarrow \delta^*(q, z) \in F$$

Zwei nicht äquivalente Zustände nennen wir auch trennbar.

Bsp.



- q_1 und q_2 sind trennbar (z.B. durch das Trennwort a)

Äquivalenz von Zuständen

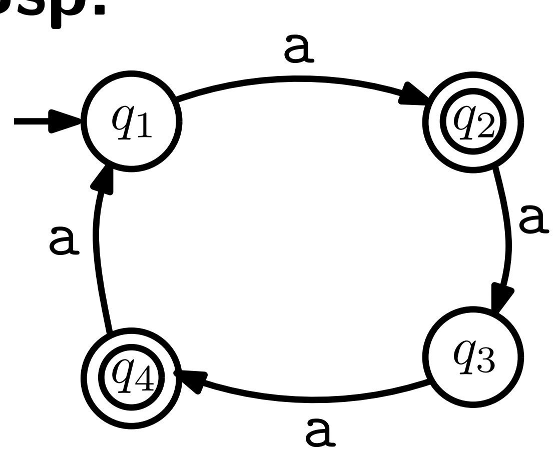
Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, dann heißen zwei Zustände $p, q \in Q$ **äquivalent** (Schreibweise $p \approx q$) gdw.

$$\forall z \in \Sigma^* : \qquad \delta^*(p, z) \in F \Leftrightarrow \delta^*(q, z) \in F$$

Zwei nicht äquivalente Zustände nennen wir auch trennbar.

Bsp.



- q_1 und q_2 sind trennbar (z.B. durch das Trennwort a)
- q_1 und q_3 sind äquivalent

Äquivalenz von Zuständen

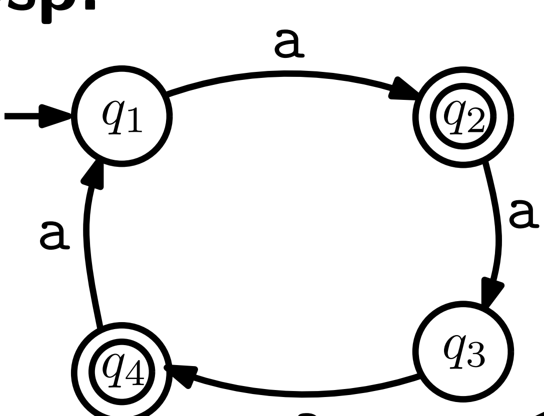
Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA, dann heißen zwei Zustände $p, q \in Q$ **äquivalent** (Schreibweise $p \approx q$) gdw.

$$\forall z \in \Sigma^* : \qquad \delta^*(p, z) \in F \Leftrightarrow \delta^*(q, z) \in F$$

Zwei nicht äquivalente Zustände nennen wir auch trennbar.

Bsp.



- q_1 und q_2 sind trennbar (z.B. durch das Trennwort a)
- q_1 und q_3 sind äquivalent

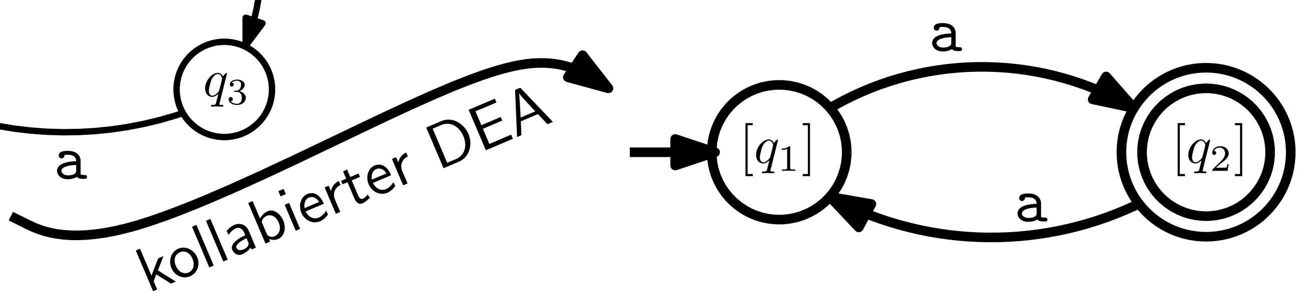


Table-Filling Algorithmus

Table-Filling Algorithmus

- Algorithmus zum effizienten Finden aller äquivalenten Zustände

Table-Filling Algorithmus

- Algorithmus zum effizienten Finden aller äquivalenten Zustände
- Datenstruktur: Tabelle T, mit $|Q|$ Zeilen und Spalten, $Q = \{1, 2, 3, \ldots, n\}$.

Table-Filling Algorithmus

- Algorithmus zum effizienten Finden aller äquivalenten Zustände
- Datenstruktur: Tabelle T, mit $|Q|$ Zeilen und Spalten, $Q = \{1, 2, 3, \ldots, n\}$.
- **Invariante:** Enthält $T[p, q]$ die Markierung **1** dann sind p und q trennbar ($p \not\approx q$)

Table-Filling Algorithmus

- Algorithmus zum effizienten Finden aller äquivalenten Zustände
- Datenstruktur: Tabelle T, mit $|Q|$ Zeilen und Spalten, $Q = \{1, 2, 3, \ldots, n\}$.
- **Invariante:** Enthält $T[p, q]$ die Markierung **1** dann sind p und q trennbar ($p \not\approx q$)
- Tabelle T wird nach und nach mit 1en gefüllt bis eine Abruchbedingung eintritt

Table-Filling Algorithmus

- Algorithmus zum effizienten Finden aller äquivalenten Zustände
- Datenstruktur: Tabelle T, mit $|Q|$ Zeilen und Spalten, $Q = \{1, 2, 3, \ldots, n\}$.
- **Invariante:** Enthält $T[p, q]$ die Markierung **1** dann sind p und q trennbar ($p \not\approx q$)
- Tabelle T wird nach und nach mit 1en gefüllt bis eine Abruchbedingung eintritt
- am Ende notieren alle unmarkierten Einträge $T[p, q]$ äquivalente Zustandspaare p, q

Table-Filling Algorithmus

Table-Filling Algorithmus

- Wir füllen nur die Hälfte der Tabelle T aus ($T[p, q]$ mit $p < q$)

Table-Filling Algorithmus

- Wir füllen nur die Hälfte der Tabelle T aus ($T[p,q]$ mit $p < q$)

Algorithm 1: TableFilling Algorithmus

Initialisiere $T \equiv 0$;
for all $(p, q) \in Q \times Q$ **do**
    if *($p \in F$ und $q \notin F$) oder ($p \notin F$ und $q \in F$)* **then**
    $T[p, q] = 1$;
end
repeat
    for all $(p, q) \in Q \times Q$ *mit* $T[p, q] \neq 1$ **do**
        if $\exists a \in \Sigma\colon T[\delta(p, a), \delta(q, a)] == 1$ **then** $T[p, q] = 1$;
    end
until *keine neue Markierung gesetzt*;

Table-Filling Algorithmus

- Wir füllen nur die Hälfte der Tabelle T aus ($T[p,q]$ mit $p < q$)

Algorithm 1: TableFilling Algorithmus

Initialisierung

Initialisiere $T \equiv 0$;
for all $(p,q) \in Q \times Q$ **do**
  if *($p \in F$ und $q \notin F$) oder ($p \notin F$ und $q \in F$)* **then**
    $T[p,q] = 1$;
end
repeat
  for all $(p,q) \in Q \times Q$ *mit* $T[p,q] \neq 1$ **do**
    if $\exists a \in \Sigma \colon T[\delta(p,a),\delta(q,a)] == 1$ **then** $T[p,q] = 1$;
  end
until *keine neue Markierung gesetzt*;

Table-Filling Algorithmus

- Wir füllen nur die Hälfte der Tabelle T aus ($T[p,q]$ mit $p < q$)

Algorithm 1: TableFilling Algorithmus

Initialisiere $T \equiv 0$; Initialisierung
for all $(p,q) \in Q \times Q$ **do**
    if *($p \in F$ und $q \notin F$) oder ($p \notin F$ und $q \in F$)* **then**
    $T[p,q] = 1$;
end
repeat
    for all $(p,q) \in Q \times Q$ *mit* $T[p,q] \neq 1$ **do**
        if $\underline{\exists a \in \Sigma \colon T[\delta(p,a), \delta(q,a)] == 1}$ **then** $T[p,q] = 1$;
    end Bedingung zum Setzen neuer Marken
until *keine neue Markierung gesetzt*;

Table-Filling Algorithmus

- Wir füllen nur die Hälfte der Tabelle T aus ($T[p,q]$ mit $p < q$)

Algorithm 1: TableFilling Algorithmus

Initialisiere $T \equiv 0$; Initialisierung
for all $(p,q) \in Q \times Q$ **do**
  if *(*$p \in F$ *und* $q \notin F$*) oder (*$p \notin F$ *und* $q \in F$*)* **then**
  $T[p,q] = 1$;
end
repeat
  for all $(p,q) \in Q \times Q$ *mit* $T[p,q] \neq 1$ **do**
    if $\exists a \in \Sigma \colon T[\delta(p,a), \delta(q,a)] == 1$ **then** $T[p,q] = 1$;
  end Bedingung zum Setzen neuer Marken
until *keine neue Markierung gesetzt*;

Abbruchbedingung

Table-Filling Algorithmus Beispiel

Table-Filling Algorithmus Beispiel

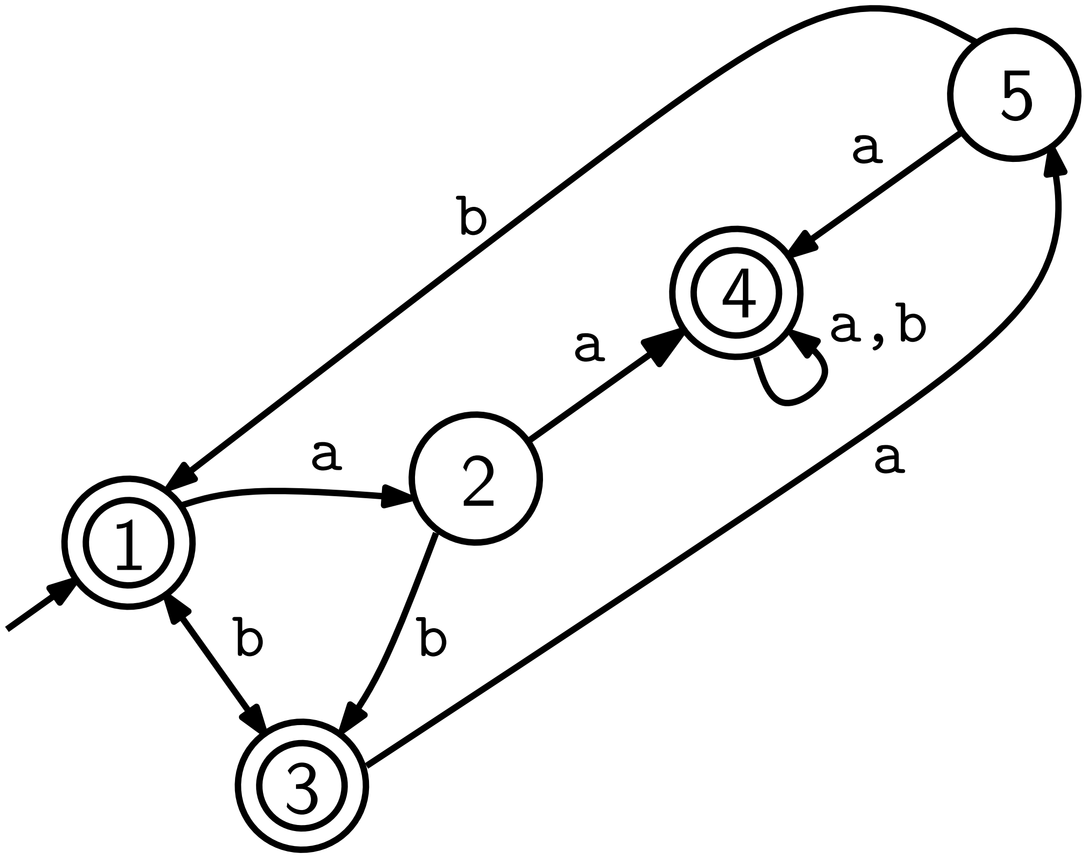
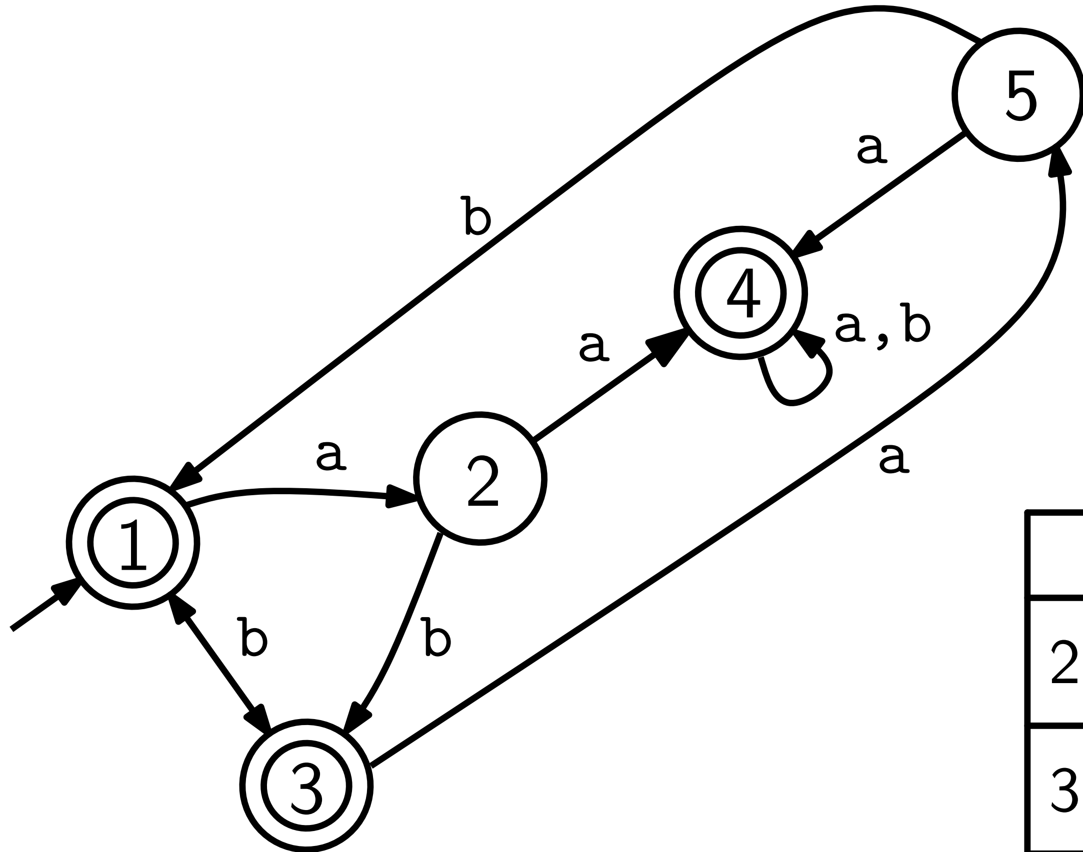
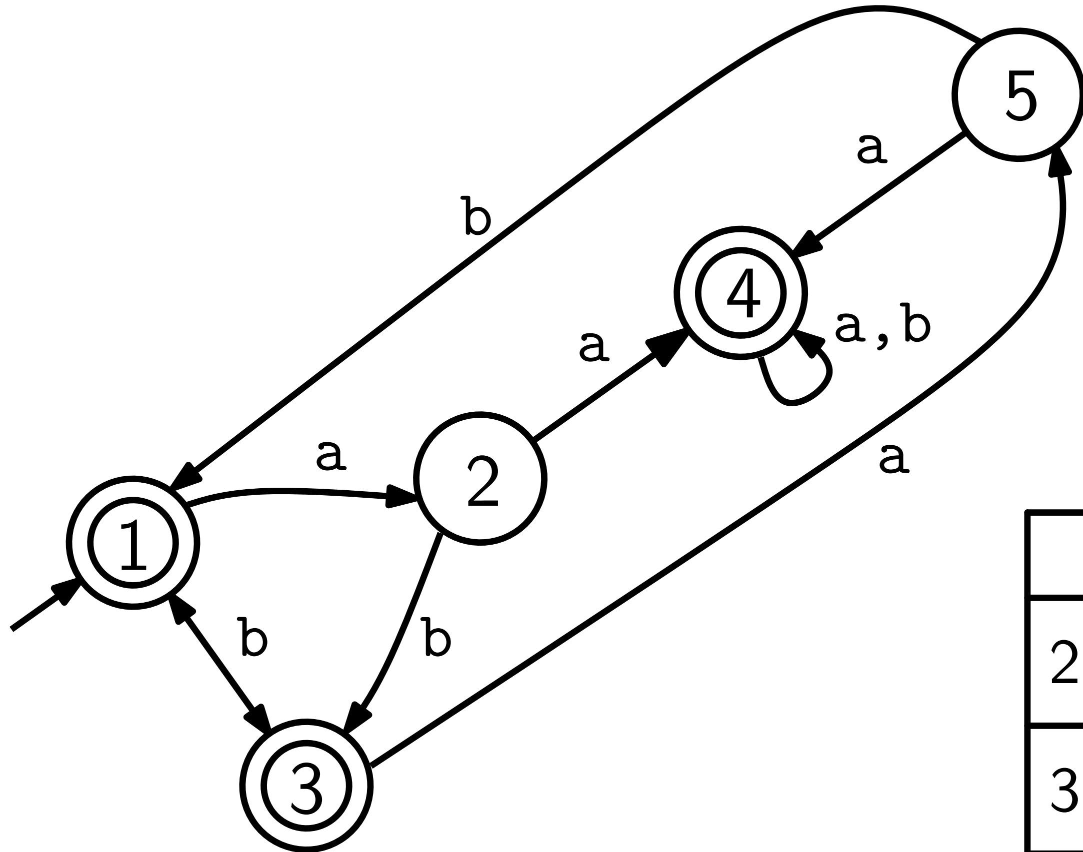


Table-Filling Algorithmus Beispiel



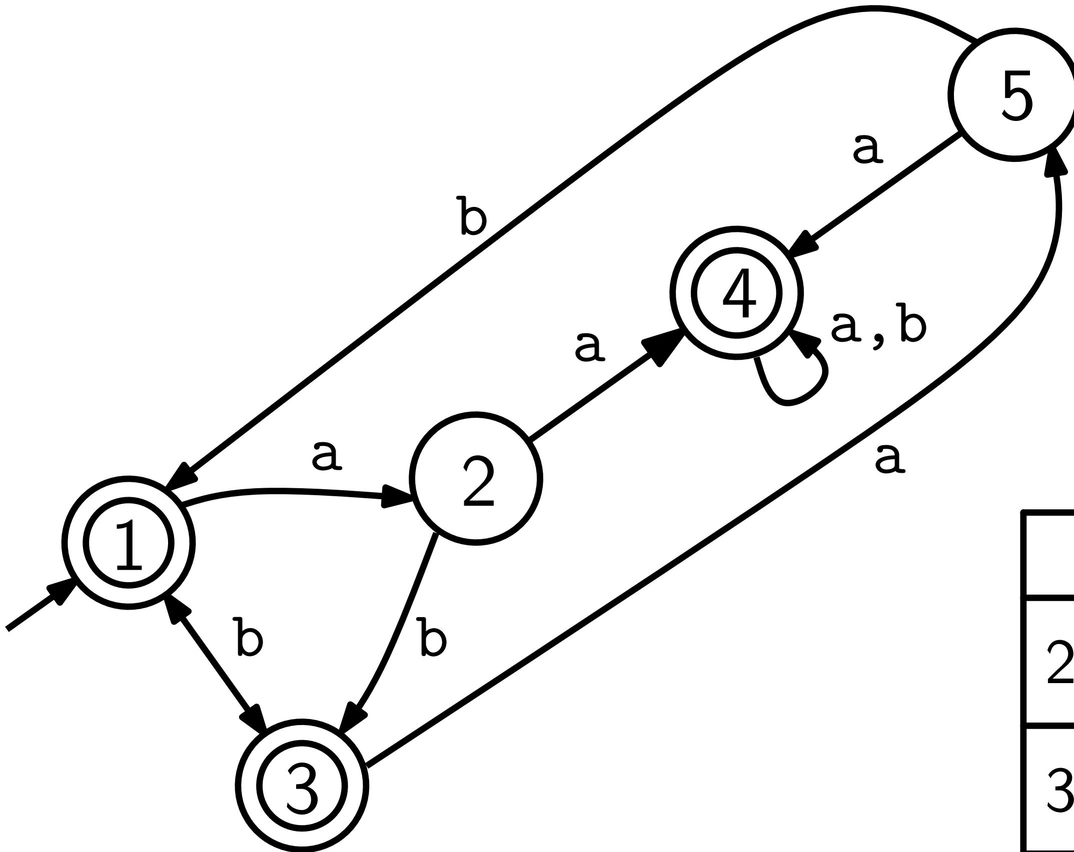
	1	2	3	4
2				
3				
4				
5				

Table-Filling Algorithmus Beispiel



	1	2	3	4
2	1			
3		1		
4		1		
5	1		1	1

Table-Filling Algorithmus Beispiel

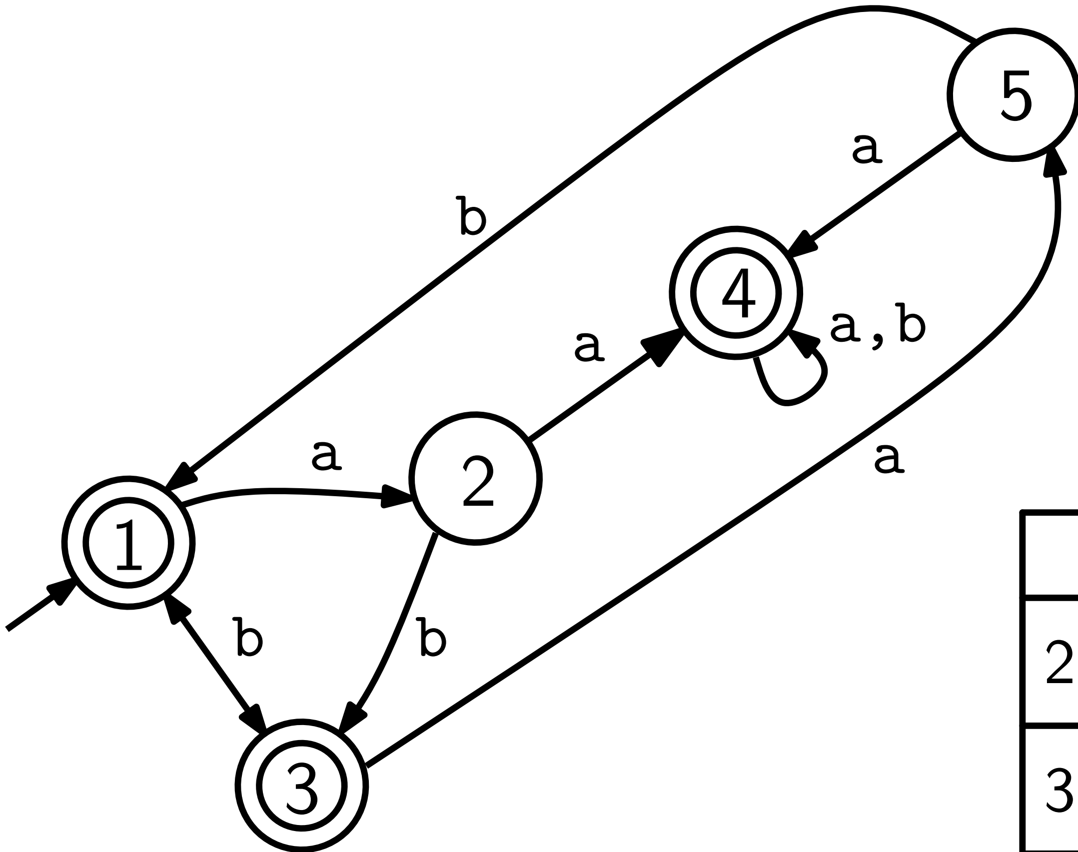


$$(1,3) \xrightarrow{\mathtt{a}} (2,5)$$

$$(1,3) \xrightarrow{\mathtt{b}} (1,3)$$

	1	2	3	4
2	1			
3		1		
4		1		
5	1		1	1

Table-Filling Algorithmus Beispiel

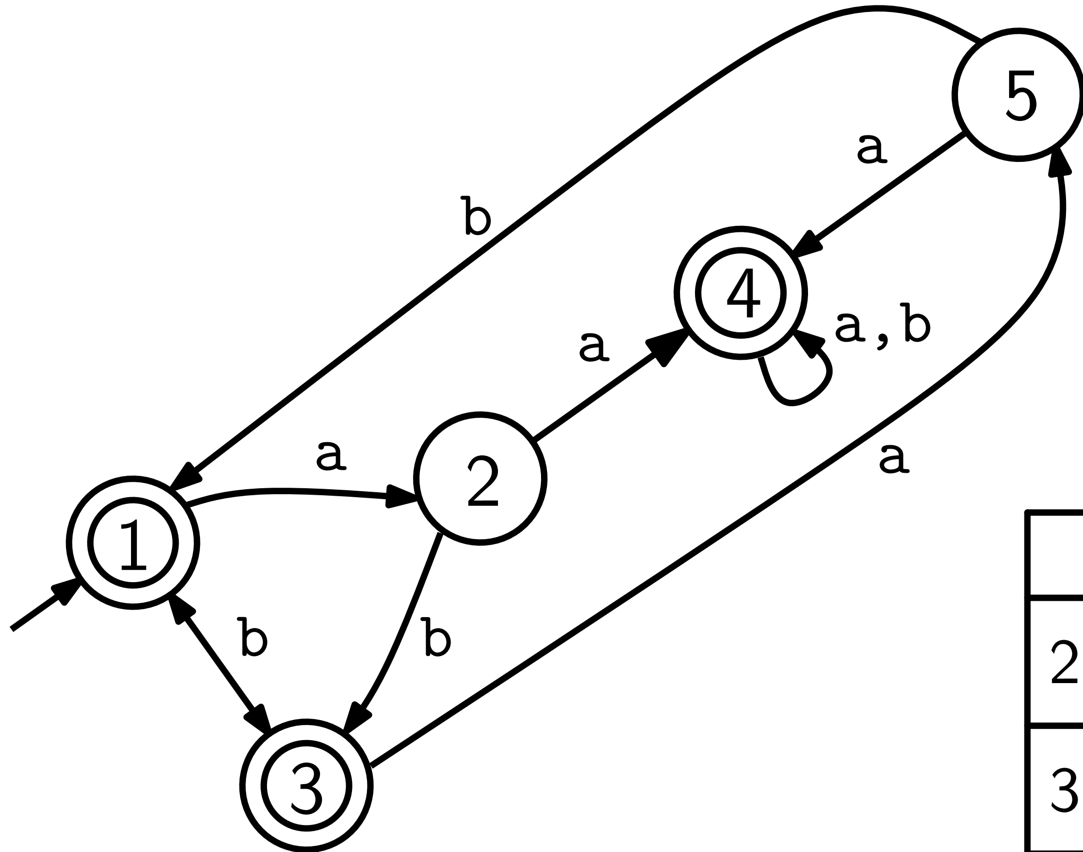


$$(1,4) \xrightarrow{\mathtt{a}} (2,4)$$

$$(1,4) \xrightarrow{\mathtt{b}} (3,4)$$

	1	2	3	4
2	1			
3		1		
4	1	1		
5	1		1	1

Table-Filling Algorithmus Beispiel

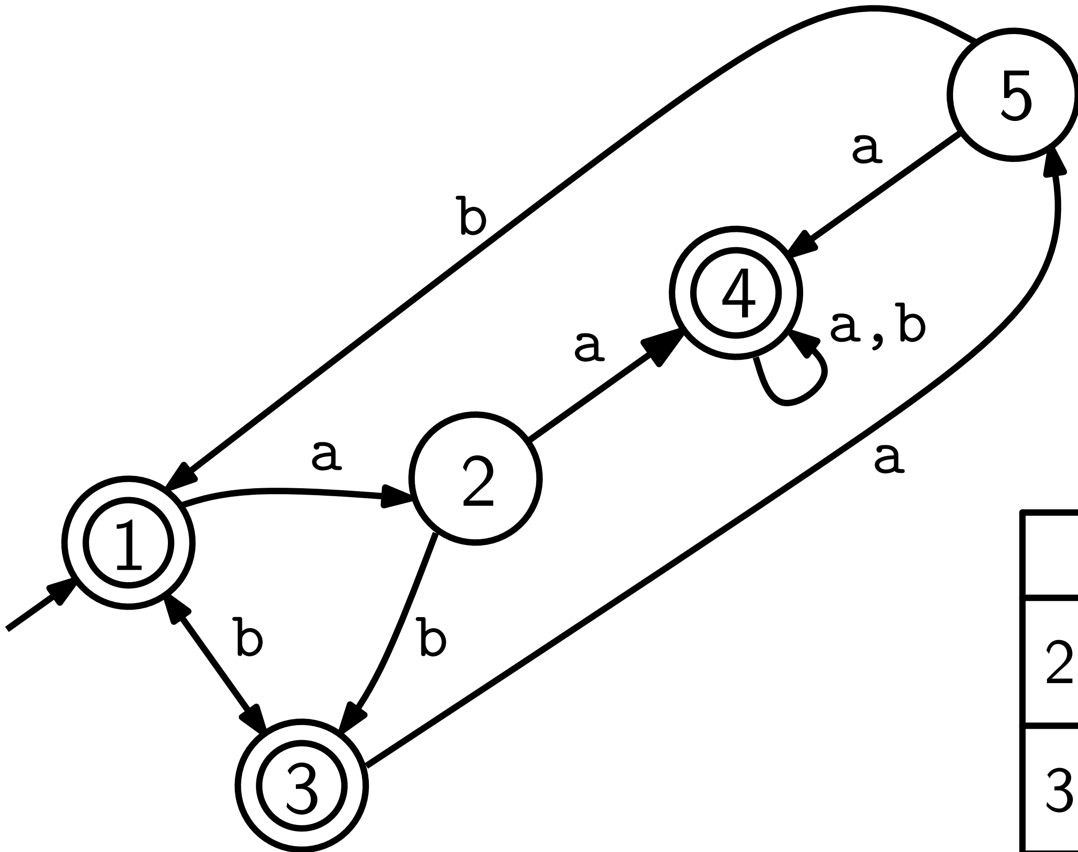


$$(3,4) \xrightarrow{\texttt{a}} (5,4)$$

$$(3,4) \xrightarrow{\texttt{b}} (1,4)$$

	1	2	3	4
2	1			
3		1		
4	1	1	1	
5	1		1	1

Table-Filling Algorithmus Beispiel

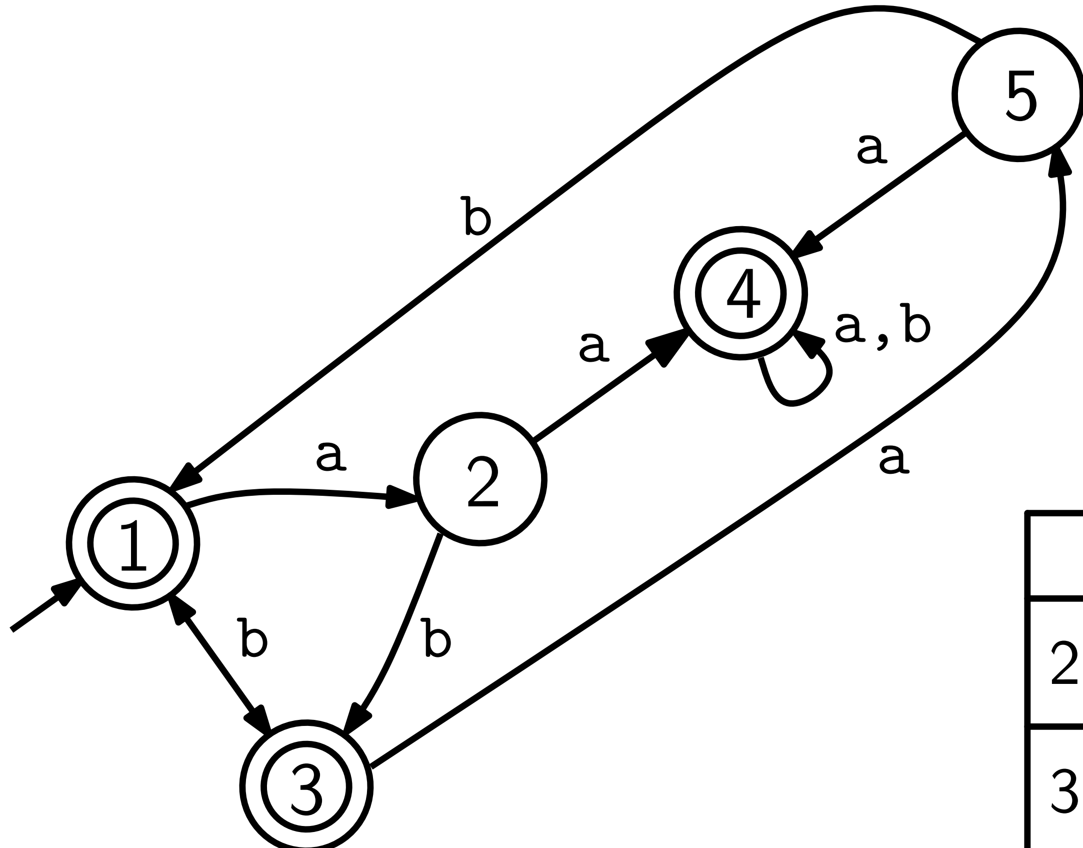


$$(2,5) \xrightarrow{\mathtt{a}} (4,4)$$

$$(2,5) \xrightarrow{\mathtt{b}} (1,3)$$

	1	2	3	4
2	1			
3		1		
4	1	1	1	
5	1		1	1

Table-Filling Algorithmus Beispiel



	1	2	3	4
2	1			
3		1		
4	1	1	1	
5	1		1	1

Keine weiteren Veränderungen

$\rightarrow 1 \approx 3$ und $2 \approx 5$

Korrektheit TF-Algorithmus

Korrektheit TF-Algorithmus

Zu zeigen:
1. nur trennbare Paare wurden markiert (Invariante bleibt erhalten)
2. alle trennbaren Paare wurden markiert

Korrektheit TF-Algorithmus

Zu zeigen:
1. nur trennbare Paare wurden markiert (Invariante bleibt erhalten)
2. alle trennbaren Paare wurden markiert

1.

- 2 Möglichkeiten wie markiert wurde

Korrektheit TF-Algorithmus

Zu zeigen:
1. nur trennbare Paare wurden markiert (Invariante bleibt erhalten)
2. alle trennbaren Paare wurden markiert

1.

- 2 Möglichkeiten wie markiert wurde
- Bei der Initialisierung:
  if $(p \in F$ und $q \notin F)$ oder $(p \notin F$ und $q \in F)$
  then $T[p, q] = 1$

Korrektheit TF-Algorithmus

Zu zeigen:
1. nur trennbare Paare wurden markiert (Invariante bleibt erhalten)
2. alle trennbaren Paare wurden markiert

1.

- 2 Möglichkeiten wie markiert wurde
- Bei der Initialisierung:
  if $(p \in F$ und $q \notin F)$ oder $(p \notin F$ und $q \in F)$
  then $T[p, q] = 1$

  $\rightarrow p, q$ trennbar mit ε

Korrektheit TF-Algorithmus

Zu zeigen:
1. nur trennbare Paare wurden markiert
   (Invariante bleibt erhalten)
2. alle trennbaren Paare wurden markiert

1.

- 2 Möglichkeiten wie markiert wurde
- Bei der Initialisierung:
  if $(p \in F$ und $q \notin F)$ oder $(p \notin F$ und $q \in F)$
  then $T[p,q] = 1$

  $\rightarrow p, q$ trennbar mit ε
- In der **repeat** Schleife:
  if $\exists a \in \Sigma\colon T[\delta(p,a), \delta(q,a)] == 1$ **then** $T[p,q] = 1$

Korrektheit TF-Algorithmus

Zu zeigen:
1. nur trennbare Paare wurden markiert (Invariante bleibt erhalten)
2. alle trennbaren Paare wurden markiert

1.

- 2 Möglichkeiten wie markiert wurde
- Bei der Initialisierung:
  if $(p \in F$ und $q \notin F)$ oder $(p \notin F$ und $q \in F)$
  then $T[p, q] = 1$

  $\rightarrow p, q$ trennbar mit ε
- In der **repeat** Schleife:
  if $\exists a \in \Sigma \colon T[\delta(p, a), \delta(q, a)] == 1$ **then** $T[p, q] = 1$

  $\rightarrow \delta(p, a), \delta(q, a)$ trennbar mit $w \in \Sigma^*$ (Invariante)

Korrektheit TF-Algorithmus

Zu zeigen:
1. nur trennbare Paare wurden markiert (Invariante bleibt erhalten)
2. alle trennbaren Paare wurden markiert

1.

- 2 Möglichkeiten wie markiert wurde
- Bei der Initialisierung:
  if ($p \in F$ und $q \notin F$) oder ($p \notin F$ und $q \in F$)
  then $T[p, q] = 1$

  $\to p, q$ trennbar mit ε

- In der **repeat** Schleife:
  if $\exists a \in \Sigma \colon T[\delta(p, a), \delta(q, a)] == 1$ **then** $T[p, q] = 1$

  $\to \delta(p, a), \delta(q, a)$ trennbar mit $w \in \Sigma^*$ (Invariante)

  $\to p, q$ trennbar mit aw

2. Alle trennbaren Paare wurden markiert

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

- Annahme: sei (p, q) schlechtes Paar mit kürzestem Trennwort w und kein schlechtes Paar besitzt ein Trennwort kürzer als w

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

- Annahme: sei (p, q) schlechtes Paar mit kürzestem Trennwort w und kein schlechtes Paar besitzt ein Trennwort kürzer als w
- $w \neq \varepsilon$, denn sonst wäre (p, q) während der Initialisierung markiert worden

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

- Annahme: sei (p, q) schlechtes Paar mit kürzestem Trennwort w und kein schlechtes Paar besitzt ein Trennwort kürzer als w
- $w \neq \varepsilon$, denn sonst wäre (p, q) während der Initialisierung markiert worden
- $w = au$ mit $a \in \Sigma$
- $p' = \delta(p, a)$ und $q' = \delta(q, a)$

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

- Annahme: sei (p, q) schlechtes Paar mit kürzestem Trennwort w und kein schlechtes Paar besitzt ein Trennwort kürzer als w
- $w \neq \varepsilon$, denn sonst wäre (p, q) während der Initialisierung markiert worden
- $w = au$ mit $a \in \Sigma$
- $p' = \delta(p, a)$ und $q' = \delta(q, a)$

$w = \square\square\square\square\square\square\square\square\square\square$

OBdA

$\delta^*(p, w) \in F$

$\delta^*(q, w) \notin F$

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

- Annahme: sei (p, q) schlechtes Paar mit kürzestem Trennwort w und kein schlechtes Paar besitzt ein Trennwort kürzer als w
- $w \neq \varepsilon$, denn sonst wäre (p, q) während der Initialisierung markiert worden
- $w = au$ mit $a \in \Sigma$
- $p' = \delta(p, a)$ und $q' = \delta(q, a)$

$$w = \boxed{a}\underbrace{\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}}_{u}$$

OBdA

$$\delta^*(p', u) = \delta^*(p, w) \in F$$
$$\delta^*(q', u) = \delta^*(q, w) \notin F$$

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

- Annahme: sei (p, q) schlechtes Paar mit kürzestem Trennwort w und kein schlechtes Paar besitzt ein Trennwort kürzer als w
- $w \neq \varepsilon$, denn sonst wäre (p, q) während der Initialisierung markiert worden
- $w = au$ mit $a \in \Sigma$
- $p' = \delta(p, a)$ und $q' = \delta(q, a)$

$$w = \boxed{a}\underbrace{\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}}_{u}$$

OBdA

$$\delta^*(p', u) = \delta^*(p, w) \in F$$
$$\delta^*(q', u) = \delta^*(q, w) \notin F$$

- $T[p', q'] \neq 1$, denn sonst wäre während der Ausführung $T[p, q] = 1$ gesetzt

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

- Annahme: sei (p, q) schlechtes Paar mit kürzestem Trennwort w und kein schlechtes Paar besitzt ein Trennwort kürzer als w
- $w \neq \varepsilon$, denn sonst wäre (p, q) während der Initialisierung markiert worden
- $w = au$ mit $a \in \Sigma$
- $p' = \delta(p, a)$ und $q' = \delta(q, a)$

$$w = \boxed{a}\underbrace{\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}}_{u}$$

OBdA

$$\delta^*(p', u) = \delta^*(p, w) \in F$$
$$\delta^*(q', u) = \delta^*(q, w) \notin F$$

- $T[p', q'] \neq 1$, denn sonst wäre während der Ausführung $T[p, q] = 1$ gesetzt
- **aber:** (p', q') trennbar mit u, und $|u| < |w|$

2. Alle trennbaren Paare wurden markiert

Schlechtes Paar: trennbar aber nicht markiert

- Annahme: sei (p, q) schlechtes Paar mit kürzestem Trennwort w und kein schlechtes Paar besitzt ein Trennwort kürzer als w
- $w \neq \varepsilon$, denn sonst wäre (p, q) während der Initialisierung markiert worden
- $w = au$ mit $a \in \Sigma$
- $p' = \delta(p, a)$ und $q' = \delta(q, a)$

$$w = \boxed{a}\underbrace{\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}\boxed{}}_{u}$$

OBdA

$$\delta^*(p', u) = \delta^*(p, w) \in F$$
$$\delta^*(q', u) = \delta^*(q, w) \notin F$$

- $T[p', q'] \neq 1$, denn sonst wäre während der Ausführung $T[p, q] = 1$ gesetzt
- **aber:** (p', q') trennbar mit u, und $|u| < |w|$

  $\rightarrow$ Widerspruch zur Annahme, d.h. es gibt kein schlechtes Paar!

Die Myhill-Nerode Relation

Die Myhill-Nerode Relation

Definition

Sei L eine Sprache über Σ. Die **Myhill-Nerode Relation (MN-Relation)** $\equiv_L$ zu L ist definiert als

$$\forall u, v \in \Sigma^* : \quad u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L.$$

Die Myhill-Nerode Relation

Definition

Sei L eine Sprache über Σ. Die **Myhill-Nerode Relation (MN-Relation)** $\equiv_L$ zu L ist definiert als

$$\forall u, v \in \Sigma^* : \quad u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L.$$

- hängt nur von der Sprache, nicht vom Modell (z.B. DEA) ab

Die Myhill-Nerode Relation

Definition

Sei L eine Sprache über Σ. Die **Myhill-Nerode Relation (MN-Relation)** $\equiv_L$ zu L ist definiert als

$$\forall u, v \in \Sigma^*: \quad u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^*: uz \in L \Leftrightarrow vz \in L.$$

- hängt nur von der Sprache, nicht vom Modell (z.B. DEA) ab
- L muss nicht unbedingt regulär sein

Die Myhill-Nerode Relation

Definition

Sei L eine Sprache über Σ. Die **Myhill-Nerode Relation (MN-Relation)** $\equiv_L$ zu L ist definiert als

$$\forall u, v \in \Sigma^* : \quad u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L.$$

- hängt nur von der Sprache, nicht vom Modell (z.B. DEA) ab
- L muss nicht unbedingt regulär sein

Beispiel $\quad L = \{\mathrm{a}^n \mid n \text{ gerade } \}$

Die Myhill-Nerode Relation

Definition

Sei L eine Sprache über Σ. Die **Myhill-Nerode Relation (MN-Relation)** $\equiv_L$ zu L ist definiert als

$$\forall u, v \in \Sigma^*: \quad u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^*: uz \in L \Leftrightarrow vz \in L.$$

- hängt nur von der Sprache, nicht vom Modell (z.B. DEA) ab
- L muss nicht unbedingt regulär sein

Beispiel $\quad L = \{\mathtt{a}^n \mid n \text{ gerade }\}$

$\rightarrow$ $\mathtt{aa} \equiv_L \mathtt{aaaa}$

Die Myhill-Nerode Relation

Definition

Sei L eine Sprache über Σ. Die **Myhill-Nerode Relation (MN-Relation)** $\equiv_L$ zu L ist definiert als

$$\forall u, v \in \Sigma^*: \quad u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^*: uz \in L \Leftrightarrow vz \in L.$$

- hängt nur von der Sprache, nicht vom Modell (z.B. DEA) ab
- L muss nicht unbedingt regulär sein

Beispiel $L = \{\mathtt{a}^n \mid n \text{ gerade }\}$

$\rightarrow \mathtt{aa} \equiv_L \mathtt{aaaa}$
$\rightarrow \mathtt{a} \not\equiv_L \mathtt{aa}$

Die Myhill-Nerode Relation

Definition

Sei L eine Sprache über Σ. Die **Myhill-Nerode Relation (MN-Relation)** $\equiv_L$ zu L ist definiert als

$$\forall u, v \in \Sigma^*: \quad u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^*: uz \in L \Leftrightarrow vz \in L.$$

- hängt nur von der Sprache, nicht vom Modell (z.B. DEA) ab
- L muss nicht unbedingt regulär sein

Beispiel $L = \{\mathrm{a}^n \mid n \text{ gerade } \}$

$\rightarrow$ aa $\equiv_L$ aaaa

$\rightarrow$ a $\not\equiv_L$ aa Trennwort zum Beispiel $z = \varepsilon$

Eigenschaften der MN Relation

$$(u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$$

Eigenschaften der MN Relation

$$(u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$$

(1) $\equiv_L$ ist **symmetrisch**

Eigenschaften der MN Relation

$$(u \equiv_L v :\iff \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

Eigenschaften der MN Relation

$(u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

Eigenschaften der MN Relation

$(u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

$\boxed{\text{Äquivalenzrelation}}$

Eigenschaften der MN Relation

$(u \equiv_L v :\iff \forall z \in \Sigma^*: uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

Äquivalenzrelation

(4) $\forall a \in \Sigma\colon u \equiv_L v \implies ua \equiv_L va$

Eigenschaften der MN Relation

$(u \equiv_L v :\iff \forall z \in \Sigma^*: uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

Äquivalenzrelation

(4) $\forall a \in \Sigma\colon u \equiv_L v \implies ua \equiv_L va$

sonst trennt $a\circ$(Trennwort uz, vz) auch u, v

Eigenschaften der MN Relation

$(u \equiv_L v :\iff \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

$\boxed{\text{Äquivalenzrelation}}$

(4) $\forall a \in \Sigma\colon u \equiv_L v \implies ua \equiv_L va$

sonst trennt $a\circ$(Trennwort uz, vz) auch u, v

(4') $\forall z \in \Sigma^* : u \equiv_L v \implies uz \equiv_L vz$

Eigenschaften der MN Relation

$(u \equiv_L v :\Longleftrightarrow \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

Äquivalenzrelation

(4) $\forall a \in \Sigma : u \equiv_L v \implies ua \equiv_L va$

sonst trennt $a \circ$(Trennwort uz, vz) auch u, v

(4') $\forall z \in \Sigma^* : u \equiv_L v \implies uz \equiv_L vz$

folgt aus (4)

Eigenschaften der MN Relation

$(u \equiv_L v :\iff \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

Äquivalenzrelation

(4) $\forall a \in \Sigma : u \equiv_L v \implies ua \equiv_L va$

sonst trennt $a\circ$(Trennwort uz, vz) auch u, v

(4') $\forall z \in \Sigma^* : u \equiv_L v \implies uz \equiv_L vz$

folgt aus (4)

(5) $u \equiv_L v \implies u \in L \Leftrightarrow v \in L$

Eigenschaften der MN Relation

$(u \equiv_L v :\iff \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

Äquivalenzrelation

(4) $\forall a \in \Sigma : u \equiv_L v \implies ua \equiv_L va$

sonst trennt $a\circ$(Trennwort uz, vz) auch u, v

(4') $\forall z \in \Sigma^* : u \equiv_L v \implies uz \equiv_L vz$

folgt aus (4)

(5) $u \equiv_L v \implies u \in L \Leftrightarrow v \in L$

sonst trennbar mit ε

Eigenschaften der MN Relation

$(u \equiv_L v :\iff \forall z \in \Sigma^*: uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

$\boxed{\text{Äquivalenzrelation}}$

(4) $\forall a \in \Sigma: u \equiv_L v \implies ua \equiv_L va$

sonst trennt $a\circ$(Trennwort uz, vz) auch u, v

(4') $\forall z \in \Sigma^*: u \equiv_L v \implies uz \equiv_L vz$

folgt aus (4)

(5) $u \equiv_L v \implies u \in L \Leftrightarrow v \in L$

sonst trennbar mit ε

(1)+(2)+(3)+(4): **Rechtskongruenz**

Eigenschaften der MN Relation

$(u \equiv_L v :\iff \forall z \in \Sigma^* : uz \in L \Leftrightarrow vz \in L)$

(1) $\equiv_L$ ist **symmetrisch**

(2) $\equiv_L$ ist **reflexiv**

(3) $\equiv_L$ ist **transitiv**

Äquivalenzrelation

(4) $\forall a \in \Sigma\colon u \equiv_L v \implies ua \equiv_L va$

sonst trennt $a\circ$(Trennwort uz, vz) auch u, v

(4') $\forall z \in \Sigma^* : u \equiv_L v \implies uz \equiv_L vz$

folgt aus (4)

(5) $u \equiv_L v \implies u \in L \Leftrightarrow v \in L$

sonst trennbar mit ε

(1)+(2)+(3)+(4): **Rechtskongruenz**

(5): $\equiv_L$ **saturiert** L

Verfeinerungen

Verfeinerungen

Definition

Seien $\equiv_A$ und $\equiv_B$ zwei Äquivalenzrelationen.
Wir sagen $\equiv_A$ **verfeinert** $\equiv_B$, gdw.

$$x \equiv_A y \implies x \equiv_B y.$$

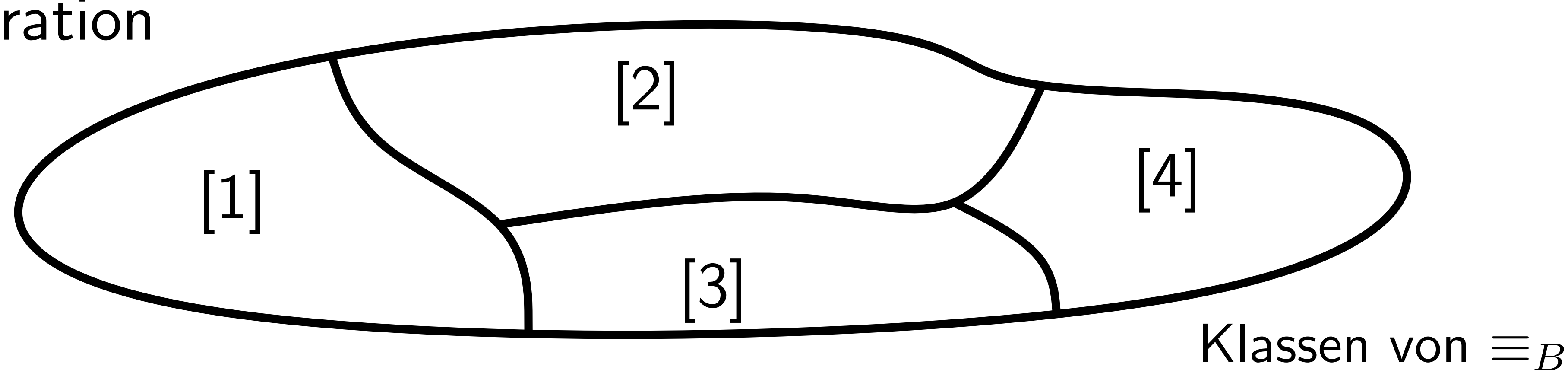
Verfeinerungen

Definition

Seien $\equiv_A$ und $\equiv_B$ zwei Äquivalenzrelationen.
Wir sagen $\equiv_A$ **verfeinert** $\equiv_B$, gdw.

$$x \equiv_A y \implies x \equiv_B y.$$

Illustration



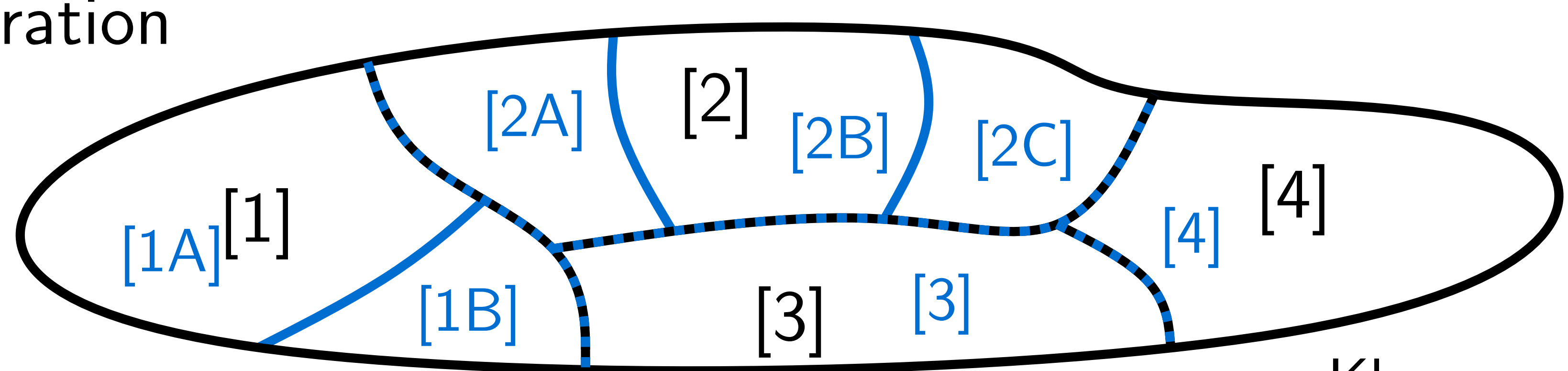
Verfeinerungen

Definition

Seien $\equiv_A$ und $\equiv_B$ zwei Äquivalenzrelationen.
Wir sagen $\equiv_A$ **verfeinert** $\equiv_B$, gdw.

$$x \equiv_A y \implies x \equiv_B y.$$

Illustration



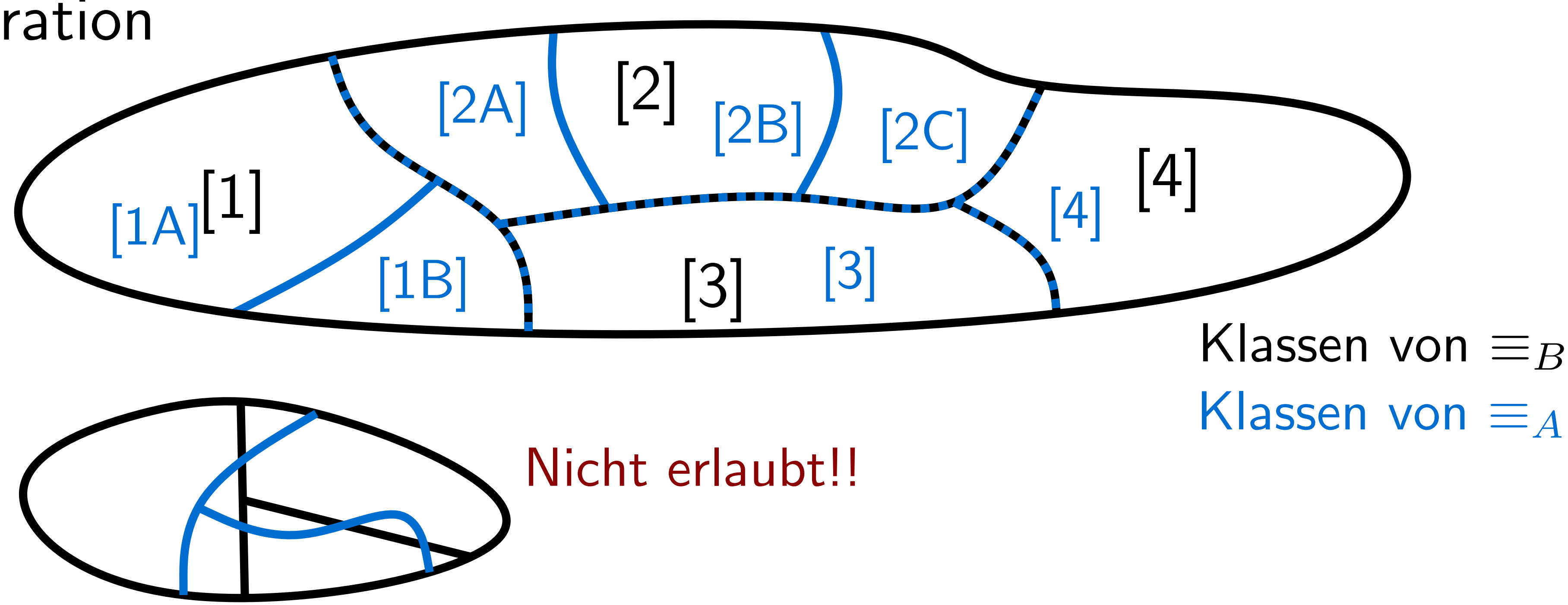
Verfeinerungen

Definition

Seien $\equiv_A$ und $\equiv_B$ zwei Äquivalenzrelationen.
Wir sagen $\equiv_A$ **verfeinert** $\equiv_B$, gdw.

$$x \equiv_A y \implies x \equiv_B y.$$

Illustration



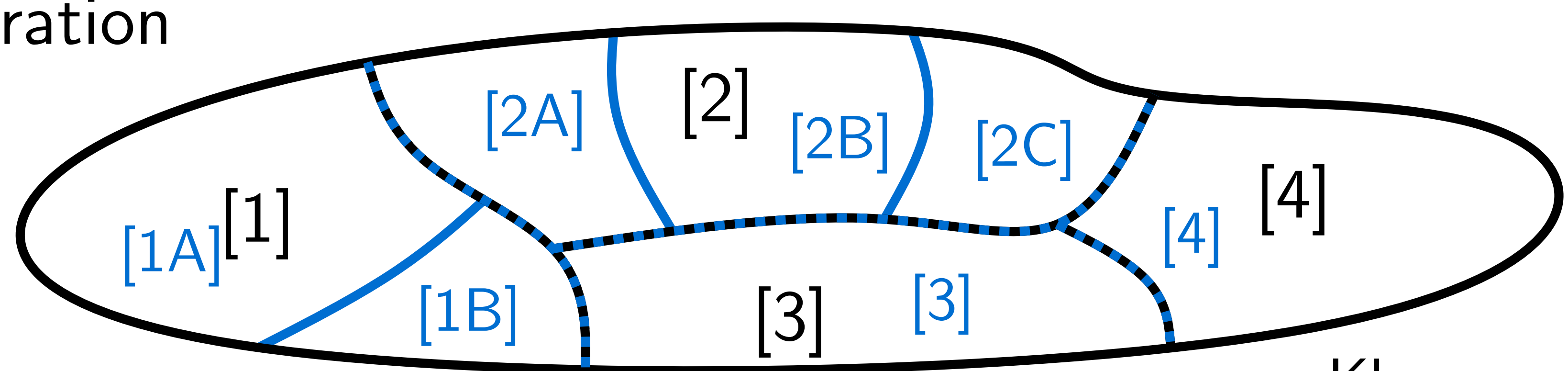
Verfeinerungen

Definition

Seien $\equiv_A$ und $\equiv_B$ zwei Äquivalenzrelationen.
Wir sagen $\equiv_A$ **verfeinert** $\equiv_B$, gdw.

$$x \equiv_A y \implies x \equiv_B y.$$

Illustration



Klassen von $\equiv_B$
Klassen von $\equiv_A$

Bsp. $a\ R_3\ b :\iff a \equiv b \mod 3$
$a\ R_6\ b :\iff a \equiv b \mod 6$

$\to R_6$ verfeinert R_3

Beispiel MN Relation

Beispiel MN Relation

L ist Sprache von $(\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Beispiel MN Relation

L ist Sprache von $(\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

1. Klasse:

$[ab] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ enthält } \mathtt{ab}\}$

Beispiel MN Relation

$$\boxed{L \text{ ist Sprache von } (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*}$$

1. Klasse:

$[ab] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ enthält } \mathtt{ab}\}$

2. Klasse:

$[a] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{a} \text{ und enthält nicht } \mathtt{ab}\}$

Beispiel MN Relation

L ist Sprache von $(\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

1. Klasse:
$[ab] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ enthält } \mathtt{ab}\}$
2. Klasse:
$[a] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{a} \text{ und enthält nicht } \mathtt{ab}\}$
3. Klasse:
$[b] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{b} \text{ und enthält nicht } \mathtt{ab}\} \cup \{\varepsilon\}$

Beispiel MN Relation

$$\boxed{L \text{ ist Sprache von } (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*}$$

1. Klasse:

$[ab] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ enthält } \mathtt{ab}\}$

2. Klasse:

$[a] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{a} \text{ und enthält nicht } \mathtt{ab}\}$

3. Klasse:

$[b] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{b} \text{ und enthält nicht } \mathtt{ab}\} \cup \{\varepsilon\}$

$\hookrightarrow [a] \cup [b] \cup [ab] = \Sigma^*$, d.h. keine weiteren Klassen

Beispiel MN Relation

$\boxed{L \text{ ist Sprache von } (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*}$

1. Klasse:
$[ab] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ enthält } \mathtt{ab}\}$

2. Klasse:
$[a] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{a} \text{ und enthält nicht } \mathtt{ab}\}$

3. Klasse:
$[b] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{b} \text{ und enthält nicht } \mathtt{ab}\} \cup \{\varepsilon\}$

$\hookrightarrow [a] \cup [b] \cup [ab] = \Sigma^*$, d.h. keine weiteren Klassen

Index := Anzahl der Äquivalenzklassen

Beispiel MN Relation

$$\boxed{L \text{ ist Sprache von } (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*}$$

1. Klasse:
$[ab] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ enthält } \mathtt{ab}\}$

2. Klasse:
$[a] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{a} \text{ und enthält nicht } \mathtt{ab}\}$

3. Klasse:
$[b] = \{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w \text{ endet auf } \mathtt{b} \text{ und enthält nicht } \mathtt{ab}\} \cup \{\varepsilon\}$

$\hookrightarrow [a] \cup [b] \cup [ab] = \Sigma^*$, d.h. keine weiteren Klassen

Index $:=$ Anzahl der Äquivalenzklassen

$\hookrightarrow$ L hat also Index 3

Der DEA zur MN Relation

Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

Definition

Der DEA $M(\equiv_L)$ zu $\equiv_L$ ist ein DEA $(Q, \Sigma, \delta, q_0, F)$ mit

- Zustandsmenge $Q = \{[x] \mid x \in \Sigma^*\}$,
- Übergangsfunktion δ, mit $\delta([x], a) = [xa]$,
- Startzustand $[\varepsilon]$,
- akzeptierenden Zuständen $F = \{[w] \mid w \in L\}$.

Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

Definition

Der DEA $M(\equiv_L)$ zu $\equiv_L$ ist ein DEA $(Q, \Sigma, \delta, q_0, F)$ mit

- Zustandsmenge $Q = \{[x] \mid x \in \Sigma^*\}$,
- Übergangsfunktion δ, mit $\delta([x], a) = [xa]$,
- Startzustand $[\varepsilon]$,
- akzeptierenden Zuständen $F = \{[w] \mid w \in L\}$.

- Q wohldefiniert, da Index endlich

Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

Definition

Der DEA $M(\equiv_L)$ zu $\equiv_L$ ist ein DEA $(Q, \Sigma, \delta, q_0, F)$ mit

- Zustandsmenge $Q = \{[x] \mid x \in \Sigma^*\}$,
- Übergangsfunktion δ, mit $\delta([x], a) = [xa]$,
- Startzustand $[\varepsilon]$,
- akzeptierenden Zuständen $F = \{[w] \mid w \in L\}$.

- Q wohldefiniert, da Index endlich
- δ wohldefiniert, da $\equiv_L$ Rechtskongruenz - Eigenschaft (4)

Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

Definition

Der DEA $M(\equiv_L)$ zu $\equiv_L$ ist ein DEA $(Q, \Sigma, \delta, q_0, F)$ mit

- Zustandsmenge $Q = \{[x] \mid x \in \Sigma^*\}$,
- Übergangsfunktion δ, mit $\delta([x], a) = [xa]$,
- Startzustand $[\varepsilon]$,
- akzeptierenden Zuständen $F = \{[w] \mid w \in L\}$.

- Q wohldefiniert, da Index endlich
- δ wohldefiniert, da $\equiv_L$ Rechtskongruenz - Eigenschaft (4)

Bsp. zu $(\mathtt{a} + \mathtt{b})^*\mathtt{ab}(\mathtt{a} + \mathtt{b})^*$

Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

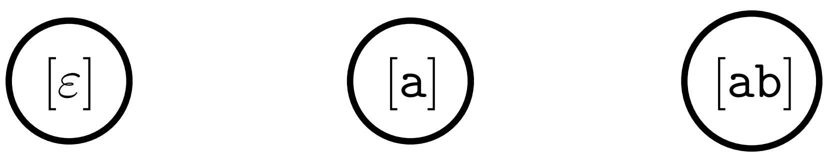
Definition

Der DEA $M(\equiv_L)$ zu $\equiv_L$ ist ein DEA $(Q, \Sigma, \delta, q_0, F)$ mit

- Zustandsmenge $Q = \{[x] \mid x \in \Sigma^*\}$,
- Übergangsfunktion δ, mit $\delta([x], a) = [xa]$,
- Startzustand $[\varepsilon]$,
- akzeptierenden Zuständen $F = \{[w] \mid w \in L\}$.

- Q wohldefiniert, da Index endlich
- δ wohldefiniert, da $\equiv_L$ Rechtskongruenz - Eigenschaft (4)

Bsp. zu $(\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$



Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

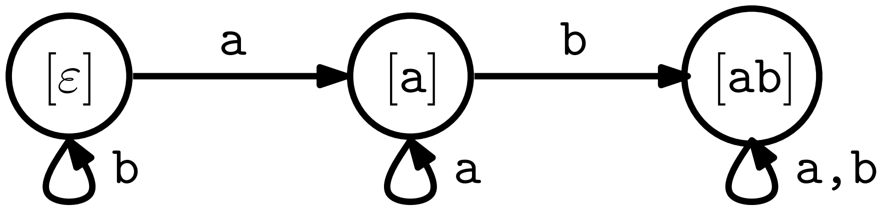
Definition

Der DEA $M(\equiv_L)$ zu $\equiv_L$ ist ein DEA $(Q, \Sigma, \delta, q_0, F)$ mit

- Zustandsmenge $Q = \{[x] \mid x \in \Sigma^*\}$,
- Übergangsfunktion δ, mit $\delta([x], a) = [xa]$,
- Startzustand $[\varepsilon]$,
- akzeptierenden Zuständen $F = \{[w] \mid w \in L\}$.

- Q wohldefiniert, da Index endlich
- δ wohldefiniert, da $\equiv_L$ Rechtskongruenz - Eigenschaft (4)

Bsp. zu $(\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$



Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

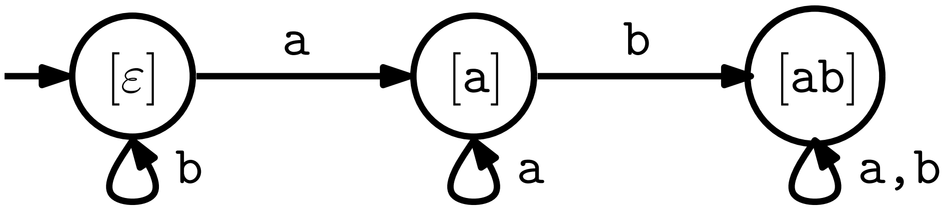
Definition

Der DEA $M(\equiv_L)$ zu $\equiv_L$ ist ein DEA $(Q, \Sigma, \delta, q_0, F)$ mit

- Zustandsmenge $Q = \{[x] \mid x \in \Sigma^*\}$,
- Übergangsfunktion δ, mit $\delta([x], a) = [xa]$,
- Startzustand $[\varepsilon]$,
- akzeptierenden Zuständen $F = \{[w] \mid w \in L\}$.

- Q wohldefiniert, da Index endlich
- δ wohldefiniert, da $\equiv_L$ Rechtskongruenz - Eigenschaft (4)

Bsp. zu $(\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$



Der DEA zur MN Relation

Sei L eine Sprache für die $\equiv_L$ endlichen Index hat.

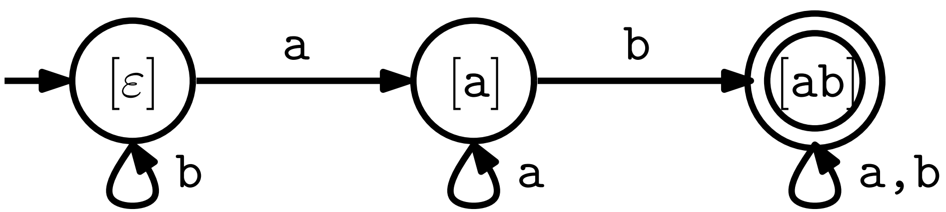
Definition

Der DEA $M(\equiv_L)$ zu $\equiv_L$ ist ein DEA $(Q, \Sigma, \delta, q_0, F)$ mit

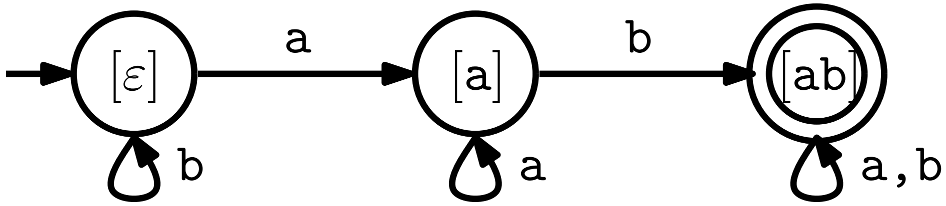
- Zustandsmenge $Q = \{[x] \mid x \in \Sigma^*\}$,
- Übergangsfunktion δ, mit $\delta([x], a) = [xa]$,
- Startzustand $[\varepsilon]$,
- akzeptierenden Zuständen $F = \{[w] \mid w \in L\}$.

- Q wohldefiniert, da Index endlich
- δ wohldefiniert, da $\equiv_L$ Rechtskongruenz - Eigenschaft (4)

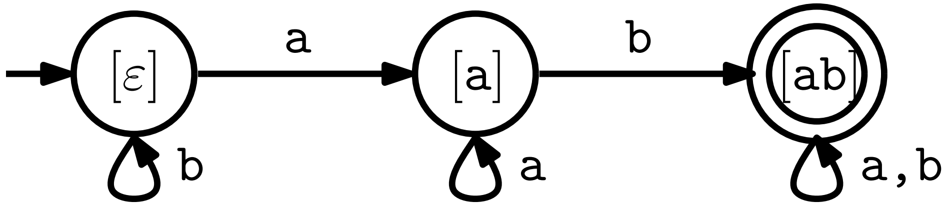
Bsp. zu $(\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$



$M(\equiv_L)$



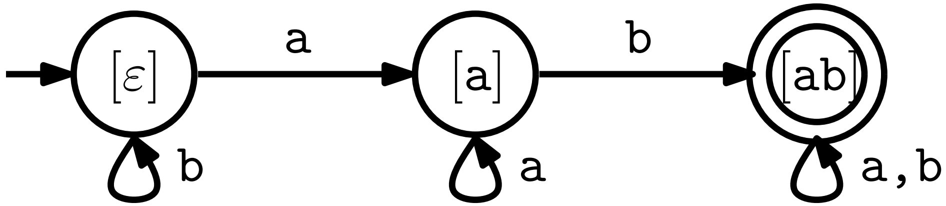
$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$M(\equiv_L)$



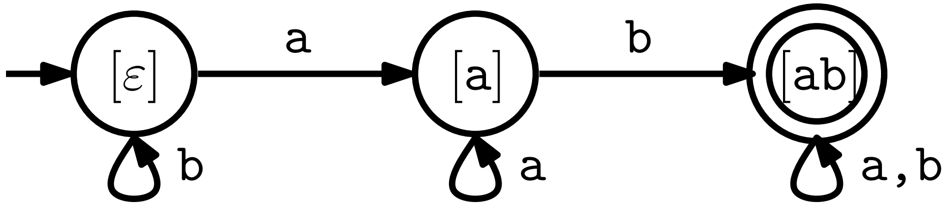
$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

$$L(M(\equiv_L)) = L$$

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

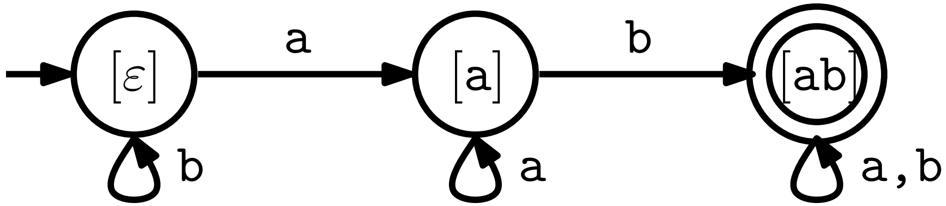
$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

$$L(M(\equiv_L)) = L$$

1. Überlegung $w \in L \Leftrightarrow [w] \in F$

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

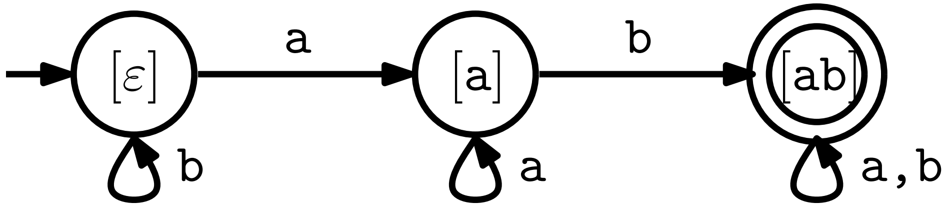
Lemma 3

$$L(M(\equiv_L)) = L$$

1. Überlegung $w \in L \Leftrightarrow [w] \in F$

nach Definition $F = \{[w] \mid w \in L\}$ und weil $\equiv_L$ L saturiert

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

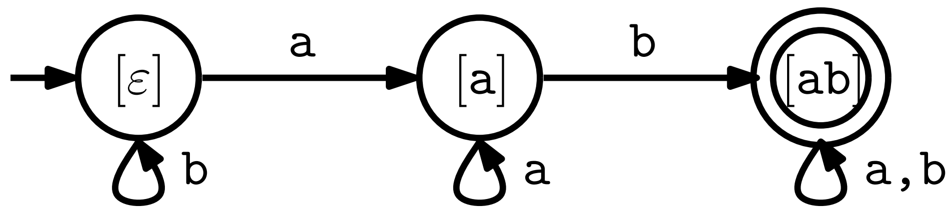
$$L(M(\equiv_L)) = L$$

1. Überlegung $w \in L \Leftrightarrow [w] \in F$

nach Definition $F = \{[w] \mid w \in L\}$ und weil $\equiv_L$ L saturiert

2. Überlegung $\delta^*([x], w) = [xw]$, $w \in \Sigma^*$

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

$$L(M(\equiv_L)) = L$$

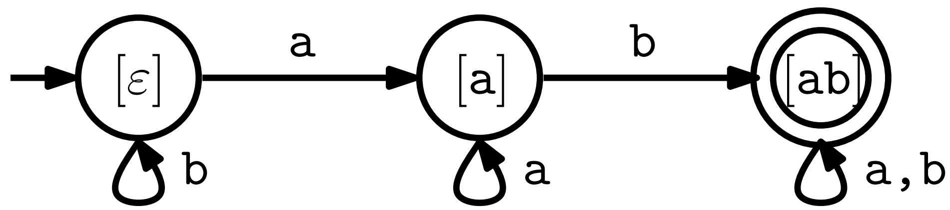
1. Überlegung $w \in L \Leftrightarrow [w] \in F$

nach Definition $F = \{[w] \mid w \in L\}$ und weil $\equiv_L$ L saturiert

2. Überlegung $\delta^*([x], w) = [xw]$, $w \in \Sigma^*$

per Induktion über $|w|$, für $|w| = 1$ folgt Aussage aus Def. von δ

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

$$L(M(\equiv_L)) = L$$

1. Überlegung $w \in L \Leftrightarrow [w] \in F$

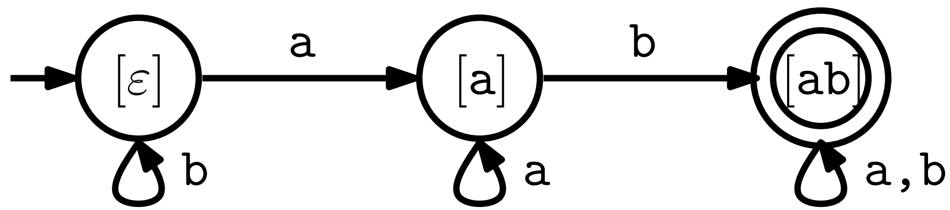
nach Definition $F = \{[w] \mid w \in L\}$ und weil $\equiv_L$ L saturiert

2. Überlegung $\delta^*([x], w) = [xw]$, $w \in \Sigma^*$

per Induktion über $|w|$, für $|w| = 1$ folgt Aussage aus Def. von δ

Beweis

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

$$L(M(\equiv_L)) = L$$

1. Überlegung $w \in L \Leftrightarrow [w] \in F$

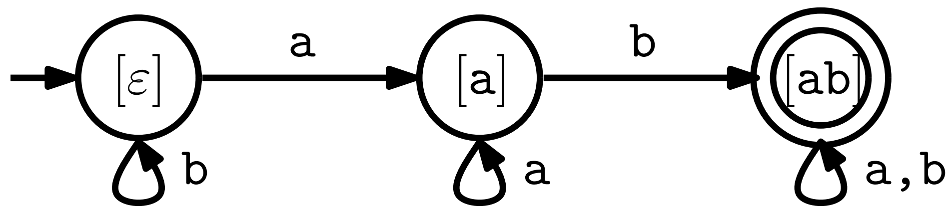
nach Definition $F = \{[w] \mid w \in L\}$ und weil $\equiv_L$ L saturiert

2. Überlegung $\delta^*([x], w) = [xw]$, $w \in \Sigma^*$

per Induktion über $|w|$, für $|w| = 1$ folgt Aussage aus Def. von δ

Beweis $\quad w \in L(M(\equiv_L)) \iff \delta^*([\varepsilon], w) \in F$

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

$$L(M(\equiv_L)) = L$$

1. Überlegung $w \in L \Leftrightarrow [w] \in F$

nach Definition $F = \{[w] \mid w \in L\}$ und weil $\equiv_L$ L saturiert

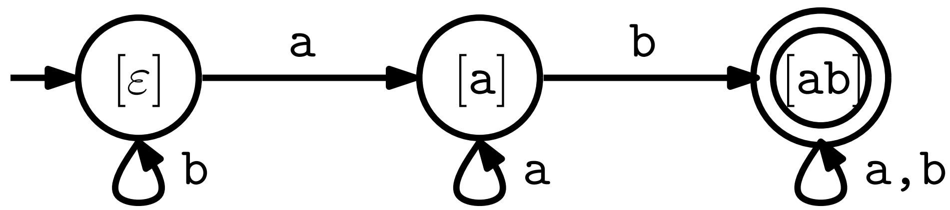
2. Überlegung $\delta^*([x], w) = [xw]$, $w \in \Sigma^*$

per Induktion über $|w|$, für $|w| = 1$ folgt Aussage aus Def. von δ

Beweis $w \in L(M(\equiv_L)) \iff \delta^*([\varepsilon], w) \in F$

$\iff [w] \in F$ nach 2.

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

$$L(M(\equiv_L)) = L$$

1. Überlegung $w \in L \Leftrightarrow [w] \in F$

nach Definition $F = \{[w] \mid w \in L\}$ und weil $\equiv_L$ L saturiert

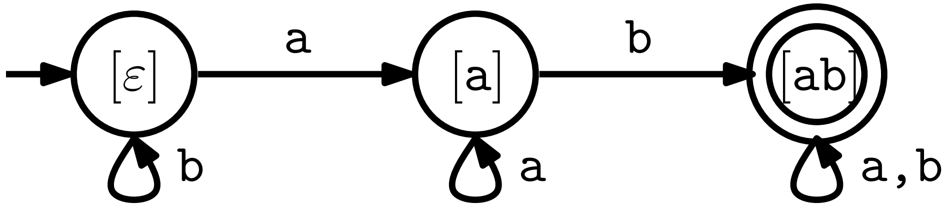
2. Überlegung $\delta^*([x], w) = [xw]$, $w \in \Sigma^*$

per Induktion über $|w|$, für $|w| = 1$ folgt Aussage aus Def. von δ

Beweis

$$\begin{aligned} w \in L(M(\equiv_L)) &\iff \delta^*([\varepsilon], w) \in F \\ &\iff [w] \in F && \text{nach 2.} \\ &\iff w \in L && \text{nach 1.} \end{aligned}$$

$M(\equiv_L)$



$L = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

$L(M(\equiv_L)) = (\mathtt{a}+\mathtt{b})^*\mathtt{ab}(\mathtt{a}+\mathtt{b})^*$

Lemma 3

$$L(M(\equiv_L)) = L$$

1. Überlegung $w \in L \Leftrightarrow [w] \in F$

nach Definition $F = \{[w] \mid w \in L\}$ und weil $\equiv_L$ L saturiert

2. Überlegung $\delta^*([x], w) = [xw]$, $w \in \Sigma^*$

per Induktion über $|w|$, für $|w| = 1$ folgt Aussage aus Def. von δ

Beweis

$$\begin{aligned} w \in L(M(\equiv_L)) &\iff \delta^*([\varepsilon], w) \in F \\ &\iff [w] \in F && \text{nach 2.} \\ &\iff w \in L && \text{nach 1.} \end{aligned}$$

□

Die Relation eines DEA

Die Relation eines DEA

Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA. Die **Relation von M** $\equiv_M$ ist definiert als

$$\forall u, v \in \Sigma^* : \quad u \equiv_M v :\iff \delta^*(q_0, u) = \delta^*(q_0, v).$$

Die Relation eines DEA

Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA. Die **Relation von M** $\equiv_M$ ist definiert als

$$\forall u, v \in \Sigma^* : \quad u \equiv_M v :\Longleftrightarrow \delta^*(q_0, u) = \delta^*(q_0, v).$$

D.h. u, v stehen in Relation, wenn ihre Läufe im gleichen Zustand enden.

Die Relation eines DEA

Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA. Die **Relation von M** $\equiv_M$ ist definiert als

$$\forall u, v \in \Sigma^* : \quad u \equiv_M v :\Longleftrightarrow \delta^*(q_0, u) = \delta^*(q_0, v).$$

D.h. u, v stehen in Relation, wenn ihre Läufe im gleichen Zustand enden.

Eigenschaften

Die Relation eines DEA

Definition

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein DEA. Die **Relation von M** $\equiv_M$ ist definiert als

$$\forall u, v \in \Sigma^* : \quad u \equiv_M v :\Longleftrightarrow \delta^*(q_0, u) = \delta^*(q_0, v).$$

D.h. u, v stehen in Relation, wenn ihre Läufe im gleichen Zustand enden.

Eigenschaften

- $\equiv_M$ ist eine Äquivalenzrelation (transitiv, symmetrisch, reflexiv)
- $\equiv_M$ ist eine Rechtskongruenz
- $\equiv_M$ saturiert L
- $\equiv_M$ hat Index $|Q|$

Lemma 4

$$\equiv_M \text{ ist eine Verfeinerung von } \equiv_{L(M)}$$

Lemma 4

$$\equiv_M \text{ ist eine Verfeinerung von } \equiv_{L(M)}$$

Beweis

Wir zeigen etwas Stärkeres:

Jede Rechtskongruenz $\equiv$, die L saturiert, verfeinert $\equiv_L$

Lemma 4

$\equiv_M$ ist eine Verfeinerung von $\equiv_{L(M)}$

Beweis

Wir zeigen etwas Stärkeres:
Jede Rechtskongruenz $\equiv$, die L saturiert, verfeinert $\equiv_L$

$u \equiv v \implies \forall z \in \Sigma^* \colon (uz \equiv vz)$ Eigenschaft (4')

Lemma 4

$\equiv_M$ ist eine Verfeinerung von $\equiv_{L(M)}$

Beweis

Wir zeigen etwas Stärkeres:
Jede Rechtskongruenz $\equiv$, die L saturiert, verfeinert $\equiv_L$

$$u \equiv v \implies \forall z \in \Sigma^* \colon (uz \equiv vz) \qquad \text{Eigenschaft (4')}$$
$$\implies \forall z \in \Sigma^* \colon (uz \in L \Leftrightarrow vz \in L) \qquad \text{Eigenschaft (5)}$$

Lemma 4

$\equiv_M$ ist eine Verfeinerung von $\equiv_{L(M)}$

Beweis

Wir zeigen etwas Stärkeres:

Jede Rechtskongruenz $\equiv$, die L saturiert, verfeinert $\equiv_L$

$u \equiv v \implies \forall z \in \Sigma^* \colon (uz \equiv vz)$ Eigenschaft (4')

$\implies \forall z \in \Sigma^* \colon (uz \in L \Leftrightarrow vz \in L)$ Eigenschaft (5)

$\implies u \equiv_L v$ Def. $\equiv_L$

$\rightarrow$ $\equiv_M$ verfeinert $\equiv_{L(M)}$ $\square$

Lemma 4

$$\equiv_M \text{ ist eine Verfeinerung von } \equiv_{L(M)}$$

Beweis

Wir zeigen etwas Stärkeres:

Jede Rechtskongruenz $\equiv$, die L saturiert, verfeinert $\equiv_L$

$$u \equiv v \implies \forall z \in \Sigma^* : (uz \equiv vz) \quad \text{Eigenschaft (4')}$$

$$\implies \forall z \in \Sigma^* : (uz \in L \Leftrightarrow vz \in L) \quad \text{Eigenschaft (5)}$$

$$\implies u \equiv_L v \quad \text{Def. } \equiv_L$$

$\rightarrow \quad \equiv_M$ verfeinert $\equiv_{L(M)}$ $\square$

Konsequenz: M hat nicht weniger Zustände als $M(\equiv_{L(M)})$

Lemma 4

$\equiv_M$ ist eine Verfeinerung von $\equiv_{L(M)}$

Beweis

Wir zeigen etwas Stärkeres:
Jede Rechtskongruenz $\equiv$, die L saturiert, verfeinert $\equiv_L$

$$
\begin{aligned}
u \equiv v &\implies \forall z \in \Sigma^* \colon (uz \equiv vz) && \text{Eigenschaft (4')}\\
&\implies \forall z \in \Sigma^* \colon (uz \in L \Leftrightarrow vz \in L) && \text{Eigenschaft (5)}\\
&\implies u \equiv_L v && \text{Def. } \equiv_L
\end{aligned}
$$

$\to$ $\equiv_M$ verfeinert $\equiv_{L(M)}$ $\square$

Konsequenz: M hat nicht weniger Zustände als $M(\equiv_{L(M)})$
$M(\equiv_{L(M)})$ ist ein minimaler DEA

Satz 6 (Satz von Myhill–Nerode)

$$L \in \mathsf{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Satz 6 (Satz von Myhill–Nerode)

$$L \in \text{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

Satz 6 (Satz von Myhill–Nerode)

$$L \in \text{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

$L \in \text{REG} \Rightarrow \exists\text{DEA } M : L(M) = L$

Satz 6 (Satz von Myhill–Nerode)

$$L \in \text{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

$$\begin{aligned} L \in \text{REG} &\Rightarrow \exists \text{DEA } M : L(M) = L \\ &\Rightarrow \equiv_M \text{ saturiert } L \end{aligned}$$

Satz 6 (Satz von Myhill–Nerode)

$$L \in \mathsf{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

$$\begin{aligned} L \in \mathsf{REG} &\Rightarrow \exists \mathsf{DEA}\ M : L(M) = L \\ &\Rightarrow \equiv_M \text{ saturiert } L \\ &\Rightarrow \equiv_M \text{ verfeinert } \equiv_L \end{aligned}$$

Lemma 4

Satz 6 (Satz von Myhill–Nerode)

$$L \in \text{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

$$\begin{aligned} L \in \text{REG} &\Rightarrow \exists \text{DEA } M : L(M) = L \\ &\Rightarrow \equiv_M \text{ saturiert } L \\ &\Rightarrow \equiv_M \text{ verfeinert } \equiv_L \\ &\Rightarrow \text{ Index } \equiv_L \leq \text{ Index } \equiv_M < \infty \end{aligned}$$

Lemma 4

Satz 6 (Satz von Myhill–Nerode)

$$L \in \text{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

$$\begin{aligned} L \in \text{REG} &\Rightarrow \exists \text{DEA } M : L(M) = L \\ &\Rightarrow \equiv_M \text{ saturiert } L \\ &\Rightarrow \equiv_M \text{ verfeinert } \equiv_L \\ &\Rightarrow \text{ Index} \equiv_L \leq \text{ Index} \equiv_M < \infty \end{aligned}$$

Lemma 4

Beweis $(\Leftarrow)$

Satz 6 (Satz von Myhill–Nerode)

$$L \in \text{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

$$\begin{aligned} L \in \text{REG} &\Rightarrow \exists \text{DEA } M : L(M) = L \\ &\Rightarrow \equiv_M \text{ saturiert } L \\ &\Rightarrow \equiv_M \text{ verfeinert } \equiv_L \\ &\Rightarrow \text{ Index } \equiv_L \leq \text{ Index } \equiv_M < \infty \end{aligned}$$

Lemma 4

Beweis $(\Leftarrow)$

$\equiv_L$ hat endlichen Index $\Rightarrow M(\equiv_L)$ ist DEA der L erkennt

(nach Lemma 3)

Satz 6 (Satz von Myhill–Nerode)

$$L \in \text{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

$$\begin{aligned} L \in \text{REG} &\Rightarrow \exists \text{DEA } M : L(M) = L \\ &\Rightarrow \equiv_M \text{ saturiert } L \\ &\Rightarrow \equiv_M \text{ verfeinert } \equiv_L \\ &\Rightarrow \text{ Index } \equiv_L \leq \text{ Index } \equiv_M < \infty \end{aligned}$$

Lemma 4

Beweis $(\Leftarrow)$

$\equiv_L$ hat endlichen Index $\Rightarrow M(\equiv_L)$ ist DEA der L erkennt

(nach Lemma 3)

$\Rightarrow L$ ist regulär

Satz 6 (Satz von Myhill–Nerode)

$$L \in \mathsf{REG} \iff \equiv_L \text{ hat endlichen Index}$$

Beweis $(\Rightarrow)$

$$\begin{aligned} L \in \mathsf{REG} &\Rightarrow \exists \mathsf{DEA}\ M : L(M) = L \\ &\Rightarrow \equiv_M \text{ saturiert } L \\ &\Rightarrow \equiv_M \text{ verfeinert } \equiv_L \\ &\Rightarrow \text{Index} \equiv_L \leq \text{Index} \equiv_M < \infty \end{aligned}$$

Lemma 4

Beweis $(\Leftarrow)$

$\equiv_L$ hat endlichen Index $\Rightarrow M(\equiv_L)$ ist DEA der L erkennt

(nach Lemma 3)

$\Rightarrow L$ ist regulär $\square$