

**2c****Minimierung von deterministischen  
endlichen Automaten**

Prof. Dr. Jasmin Blanchette

Lehr- und Forschungseinheit für  
Theoretische Informatik und Theorembeweisen

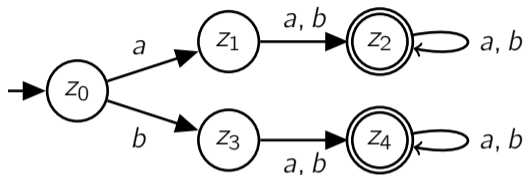
Stand: 23. April 2024

Basiert auf Folien von PD Dr. David Sabel



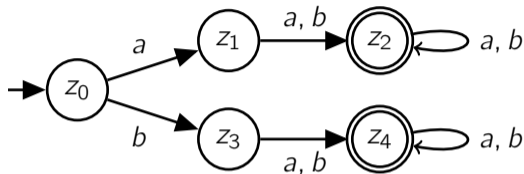
# Beispiel für Minimalität

Nicht minimaler DFA:

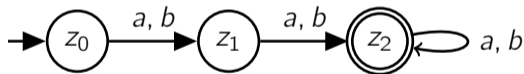


## Beispiel für Minimalität

Nicht minimaler DFA:

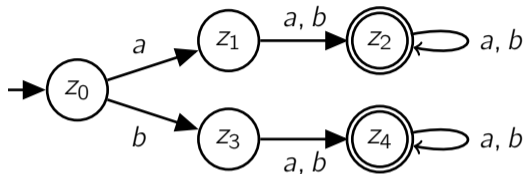


Minimaler DFA:

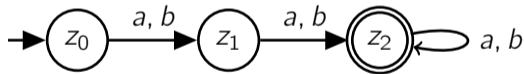


## Beispiel für Minimalität

Nicht minimaler DFA:



Minimaler DFA:



Beide DFAs akzeptieren die **gleiche Sprache**,  
aber der zweite hat **weniger Zustände**.

# Vorteile der Minimierung von DFAs

---

- ▶ Minimale DFAs sind **effizienter zu implementieren**.
- ▶ Minimale DFAs sind oft **einfacher zu verstehen** oder zu analysieren.
- ▶ Da minimale DFAs (bis auf Umbenennung der Zustände) eindeutig sind, können sie **Sprachen exakt darstellen**.

# Grundgedanke der Minimierung von DFAs

---

Schritte:

1. Entferne alle nicht erreichbaren Zustände.
2. Bilde eine Partition (d.h. disjunkte Zerlegung), indem jeweils alle Endzustände und alle Nicht-Endzustände verschmolzen werden.
3. Verfeinere diese Partition schrittweise, bis sie sich nicht mehr verändert.

# Grundgedanke der Minimierung von DFAs

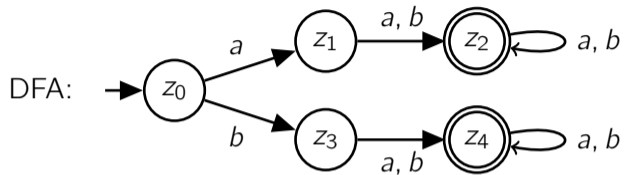
---

Schritte:

1. Entferne alle **nicht erreichbaren Zustände**.
2. Bilde eine **Partition** (d.h. disjunkte Zerlegung), indem jeweils alle Endzustände und alle Nicht-Endzustände verschmolzen werden.
3. **Verfeinere** diese Partition schrittweise, bis sie sich nicht mehr verändert.

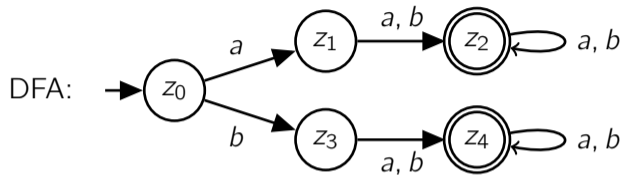
Zwei Ansätze: **graphisch** und **tabellarisch**.

## Beispiel für den graphischen Ansatz

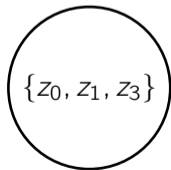




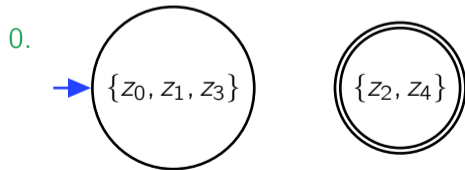
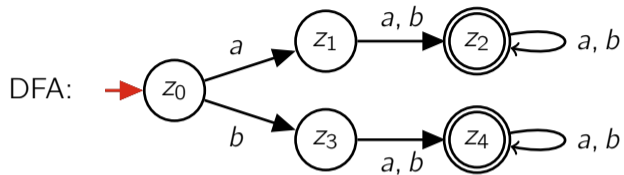
## Beispiel für den graphischen Ansatz



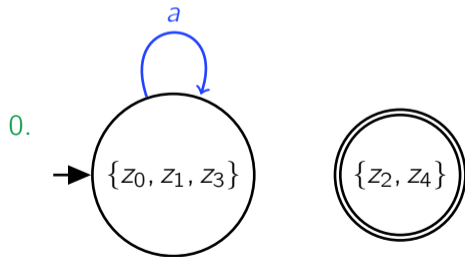
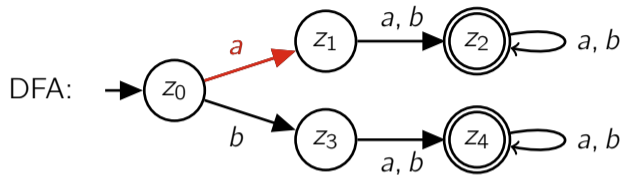
0.



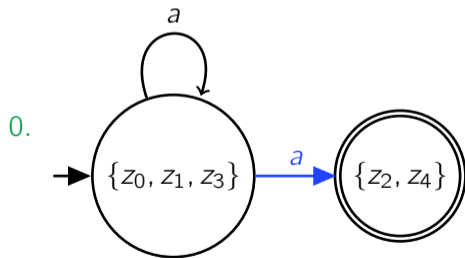
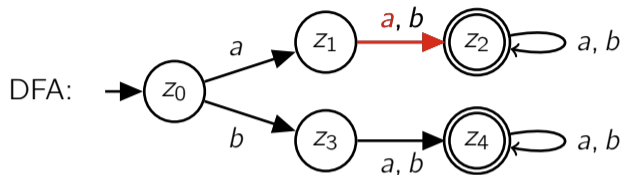
# Beispiel für den graphischen Ansatz



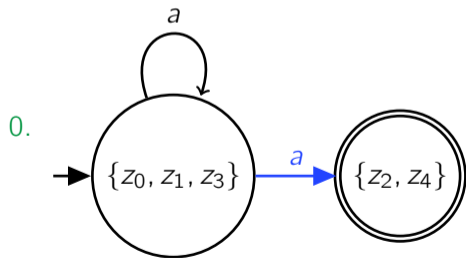
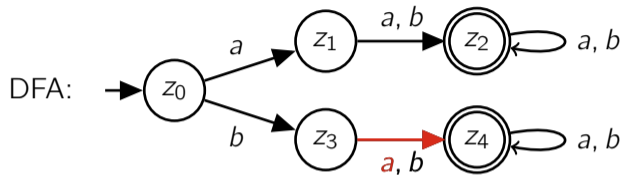
## Beispiel für den graphischen Ansatz



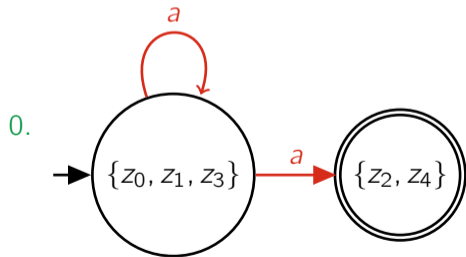
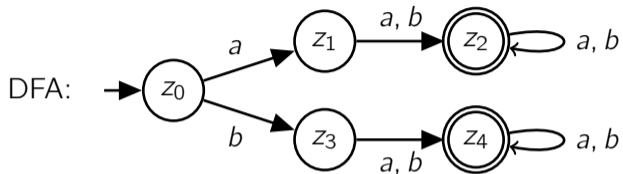
# Beispiel für den graphischen Ansatz



# Beispiel für den graphischen Ansatz



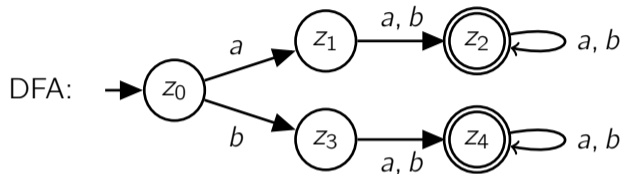
## Beispiel für den graphischen Ansatz



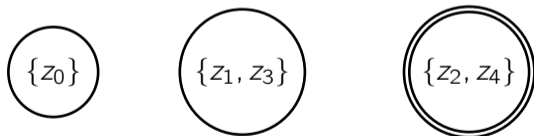
Wird kein DFA

$\{z_0, z_1, z_3\}$  muss gespalten werden,  
damit ein DFA entsteht.

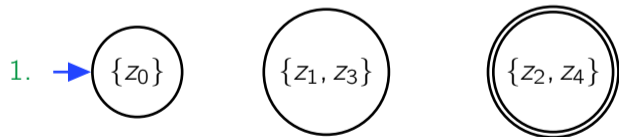
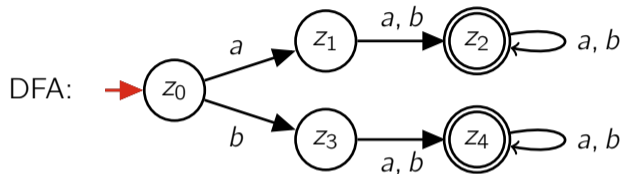
# Beispiel für den graphischen Ansatz



1.

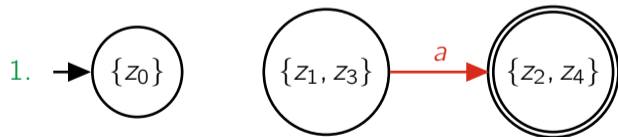
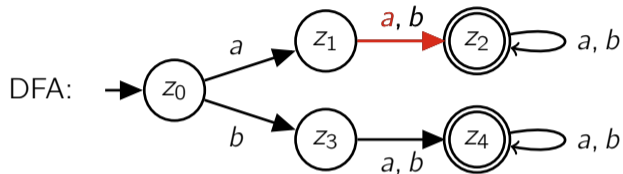


# Beispiel für den graphischen Ansatz

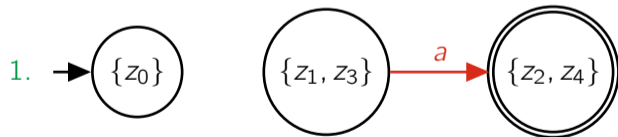
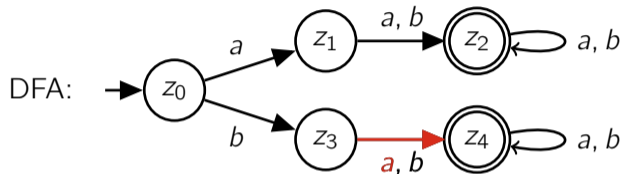




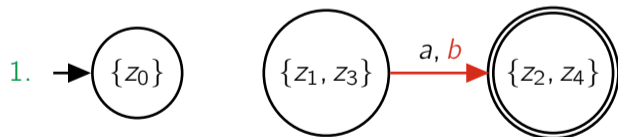
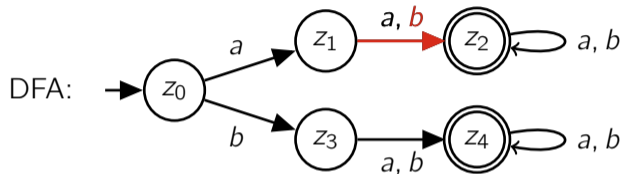
# Beispiel für den graphischen Ansatz



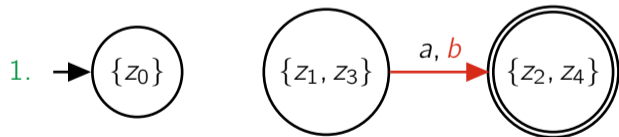
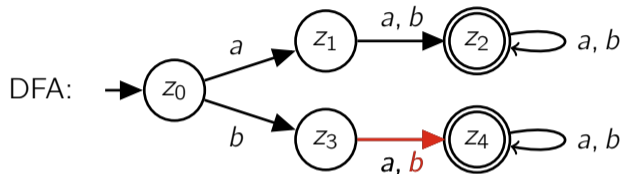
# Beispiel für den graphischen Ansatz



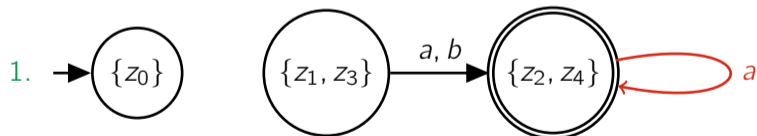
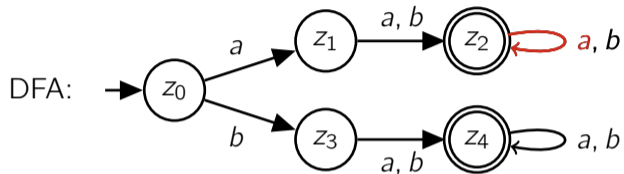
# Beispiel für den graphischen Ansatz



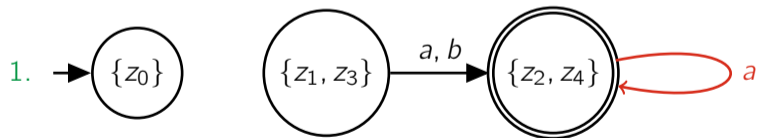
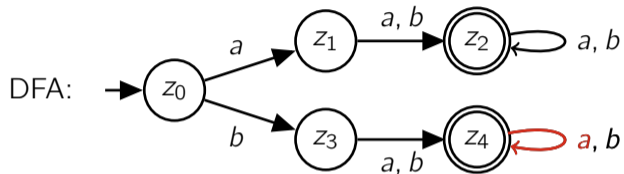
# Beispiel für den graphischen Ansatz



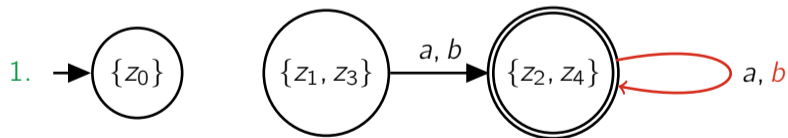
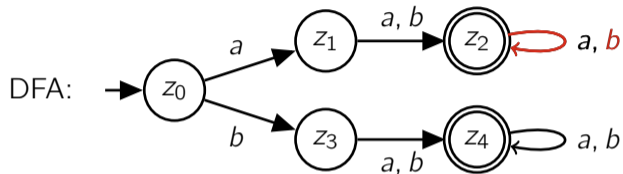
## Beispiel für den graphischen Ansatz



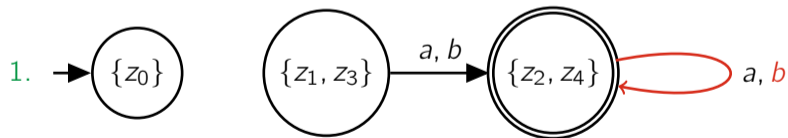
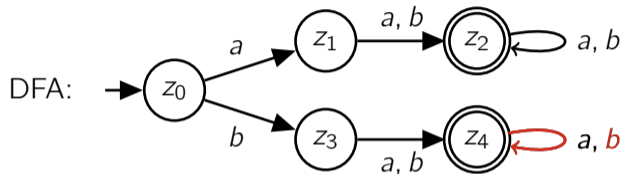
# Beispiel für den graphischen Ansatz



# Beispiel für den graphischen Ansatz

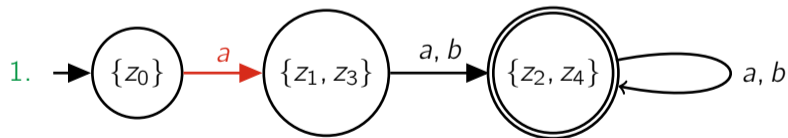
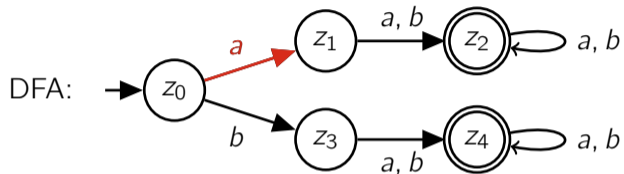


# Beispiel für den graphischen Ansatz

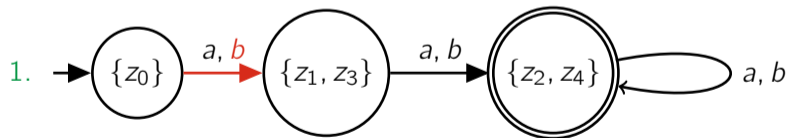
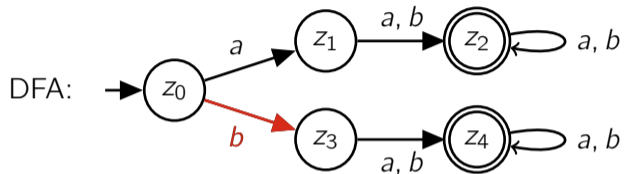




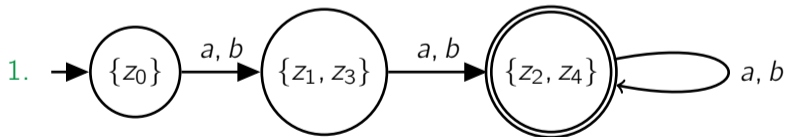
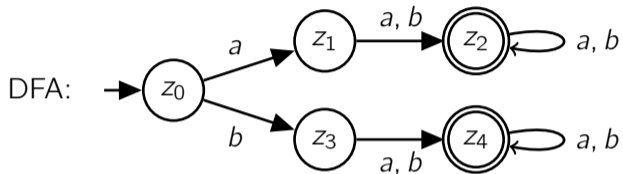
## Beispiel für den graphischen Ansatz



# Beispiel für den graphischen Ansatz



## Beispiel für den graphischen Ansatz



Minimaler DFA

# Minimierung eines DFAs mit dem tabellarischen Ansatz

---

Sei  $M$  ein DFA.

Intuitiver Ansatz:

1. Entferne alle nicht erreichbaren Zustände von  $M$ .
2. Konstruiere die Partitionstabelle.
3. Bilde den minimalen DFA  $M'$ , indem Zustände derselben Klasse verschmolzen werden, basiert auf der letzten Reihe der Partitionstabelle.

# Der tabellarische Ansatz

---

DFA  $M = (Z, \Sigma, z_0, E)$

# Der tabellarische Ansatz

---

DFA  $M = (Z, \Sigma, z_0, E)$

Schritte:

2.1 Bilde eine Partition  $\mathcal{P}$  von  $Z$  mit folgenden Klassen:

Endzustände  $E$  (falls nicht leer) und Nicht-Endzustände  $Z \setminus E$  (falls nicht leer).

# Der tabellarische Ansatz

---

DFA  $M = (Z, \Sigma, z_0, E)$

Schritte:

2.1 Bilde eine Partition  $\mathcal{P}$  von  $Z$  mit folgenden Klassen:

Endzustände  $E$  (falls nicht leer) und Nicht-Endzustände  $Z \setminus E$  (falls nicht leer).

2.2 Wiederhole bis  $\mathcal{P}$  sich nicht mehr verändert:

2.2.1 Für jede Klasse  $K \in \mathcal{P}$  mit  $|K| \geq 2$  und für jedes  $a \in \Sigma$ :

2.2.1.1 Berechne für jeden Zustand  $z \in K$  die Klasse  $L \in \mathcal{P}$ , sodass  $\delta(z, a) \in L$ .

2.2.1.2 Partitioniere  $K$  in Teilklassen je nach  $L$ .

2.2.1.3 Falls es mehrere Teilklassen gibt, ersetze  $K$  in  $\mathcal{P}$  durch die Teilklassen.

# Der tabellarische Ansatz

---

DFA  $M = (Z, \Sigma, z_0, E)$

Schritte:

2.1 Bilde eine Partition  $\mathcal{P}$  von  $Z$  mit folgenden Klassen:

Endzustände  $E$  (falls nicht leer) und Nicht-Endzustände  $Z \setminus E$  (falls nicht leer).

2.2 Wiederhole bis  $\mathcal{P}$  sich nicht mehr verändert:

2.2.1 Für jede Klasse  $K \in \mathcal{P}$  mit  $|K| \geq 2$  und für jedes  $a \in \Sigma$ :

2.2.1.1 Berechne für jeden Zustand  $z \in K$  die Klasse  $L \in \mathcal{P}$ , sodass  $\delta(z, a) \in L$ .

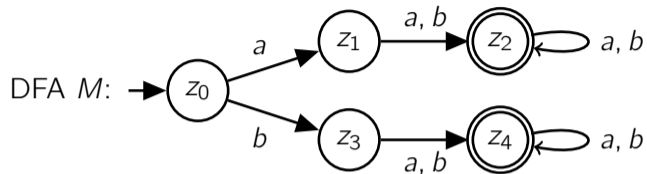
2.2.1.2 Partitioniere  $K$  in Teilklassen je nach  $L$ .

2.2.1.3 Falls es mehrere Teilklassen gibt, ersetze  $K$  in  $\mathcal{P}$  durch die Teilklassen.

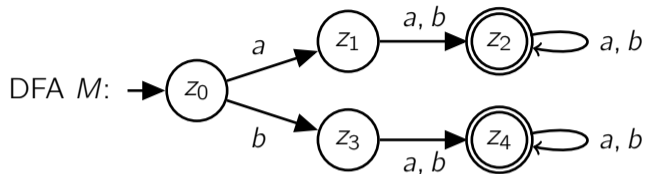
Am Ende besteht  $\mathcal{P}$  aus den verschmolzenen Zuständen.



## Erstes Beispiel für den tabellarischen Ansatz



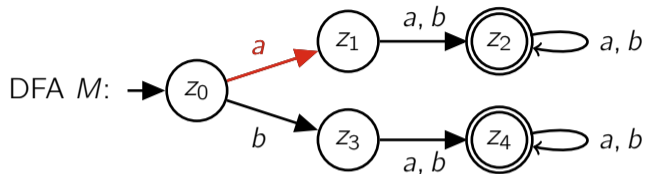
# Erstes Beispiel für den tabellarischen Ansatz



0. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

# Erstes Beispiel für den tabellarischen Ansatz

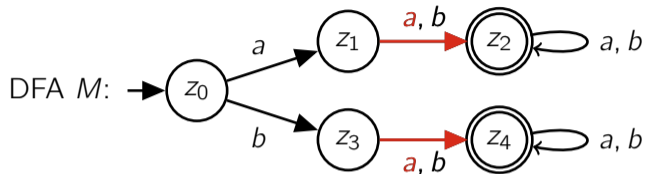


0. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

▶  $z_0$  mit  $a$  landet in der ersten Klasse.

# Erstes Beispiel für den tabellarischen Ansatz

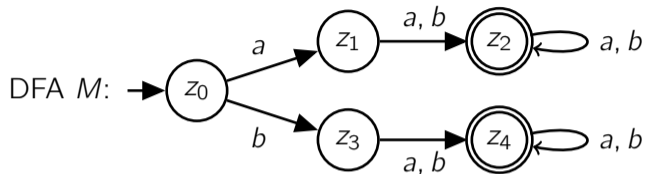


0. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

- ▶  $z_0$  mit  $a$  landet in der ersten Klasse.
- ▶  $z_1$  und  $z_3$  mit  $a$  landen in der zweiten Klasse.

# Erstes Beispiel für den tabellarischen Ansatz

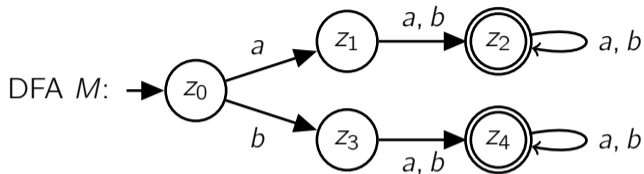


0. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

- ▶  $z_0$  mit  $a$  landet in der ersten Klasse.
- ▶  $z_1$  und  $z_3$  mit  $a$  landen in der zweiten Klasse.
- ▶  $z_0$  muss daher von  $z_1$  und  $z_3$  getrennt werden.

# Erstes Beispiel für den tabellarischen Ansatz



0. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

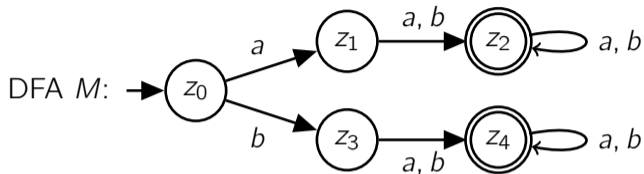
- ▶  $z_0$  mit  $a$  landet in der ersten Klasse.
- ▶  $z_1$  und  $z_3$  mit  $a$  landen in der zweiten Klasse.
- ▶  $z_0$  muss daher von  $z_1$  und  $z_3$  getrennt werden.

1. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

 mit  $a$

# Erstes Beispiel für den tabellarischen Ansatz



0. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

- ▶  $z_0$  mit  $a$  landet in der ersten Klasse.
- ▶  $z_1$  und  $z_3$  mit  $a$  landen in der zweiten Klasse.
- ▶  $z_0$  muss daher von  $z_1$  und  $z_3$  getrennt werden.

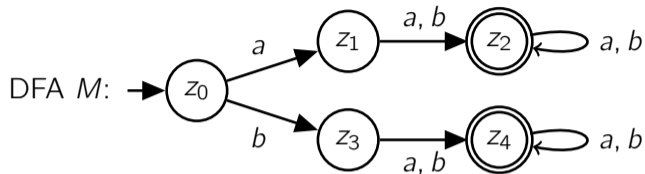
1. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

 mit  $a$

- ▶ Keine weitere Partition möglich.

# Erstes Beispiel für den tabellarischen Ansatz



0. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

- ▶  $z_0$  mit  $a$  landet in der ersten Klasse.
- ▶  $z_1$  und  $z_3$  mit  $a$  landen in der zweiten Klasse.
- ▶  $z_0$  muss daher von  $z_1$  und  $z_3$  getrennt werden.

1. 

$z_0$	$z_1$	$z_3$	$z_2$	$z_4$
-------	-------	-------	-------	-------

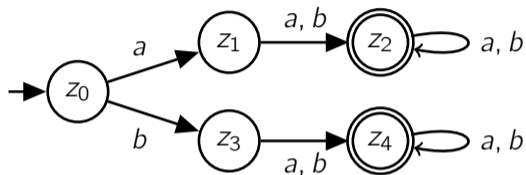
 mit  $a$

- ▶ Keine weitere Partition möglich.
- ▶ Der minimale DFA hat drei Zustände:  $\{z_0\}$ ,  $\{z_1, z_3\}$ ,  $\{z_2, z_4\}$ .

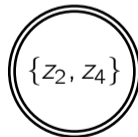
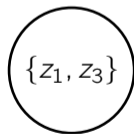
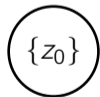


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

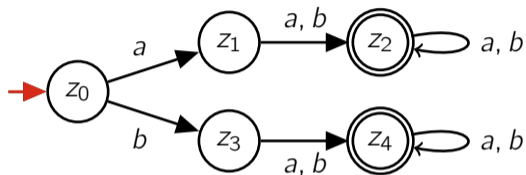


Minimaler DFA  $M'$ :

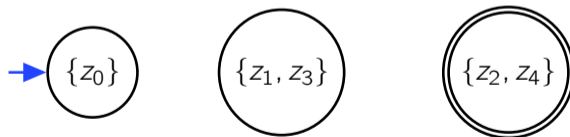


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

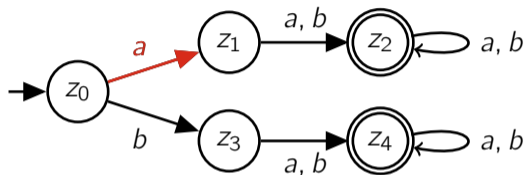


Minimaler DFA  $M'$ :

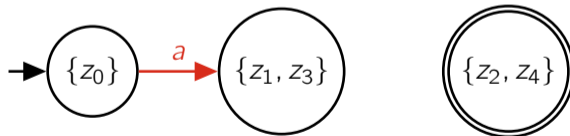


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

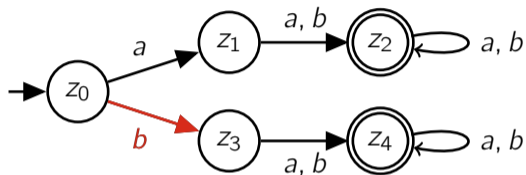


Minimaler DFA  $M'$ :

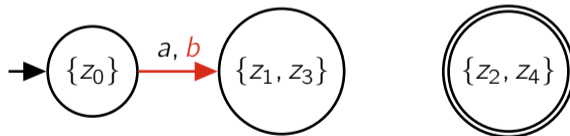


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

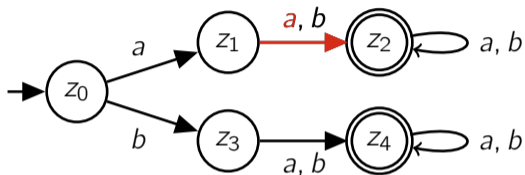


Minimaler DFA  $M'$ :

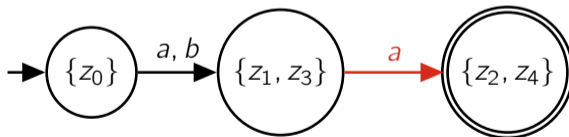


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

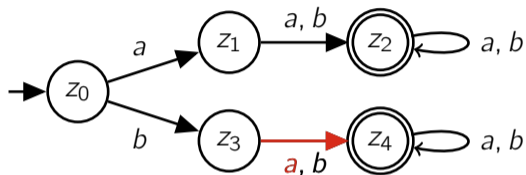


Minimaler DFA  $M'$ :

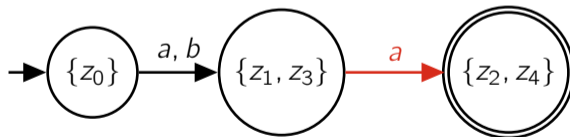


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

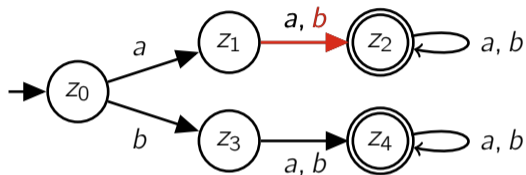


Minimaler DFA  $M'$ :

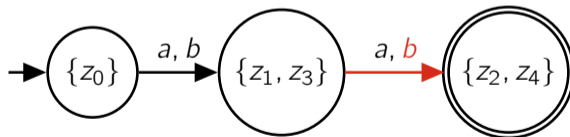


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

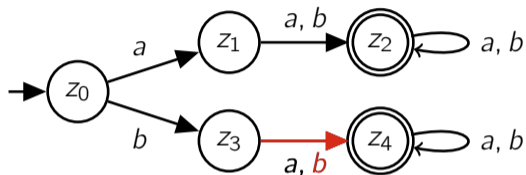


Minimaler DFA  $M'$ :

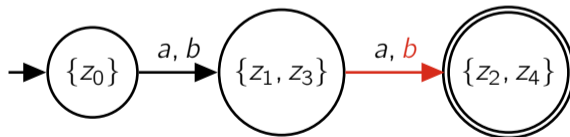


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :



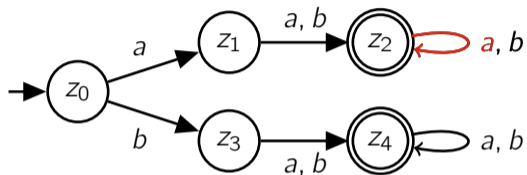
Minimaler DFA  $M'$ :



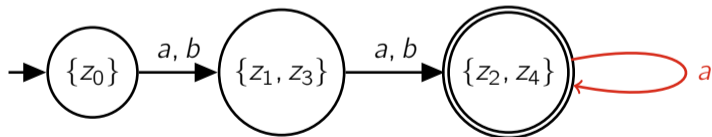


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

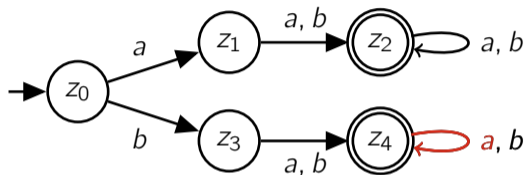


Minimaler DFA  $M'$ :

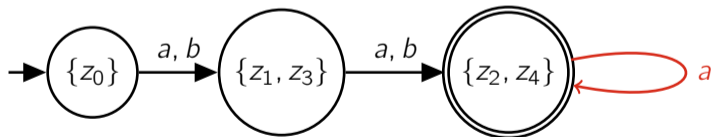


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

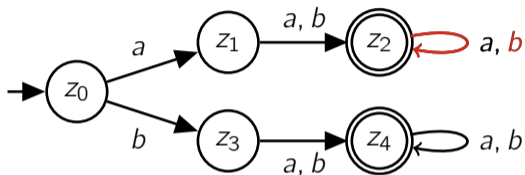


Minimaler DFA  $M'$ :

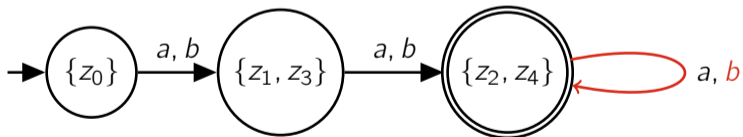


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

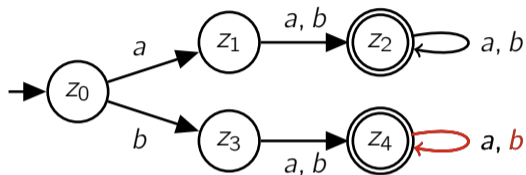


Minimaler DFA  $M'$ :

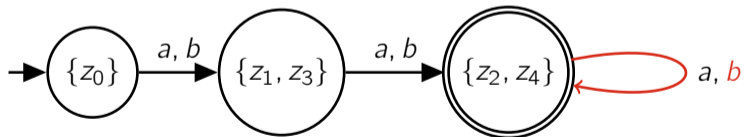


# Beispiel für die Konstruktion des minimalen DFA

DFA  $M$ :

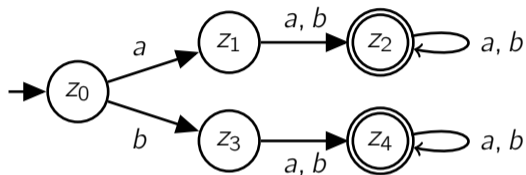


Minimaler DFA  $M'$ :

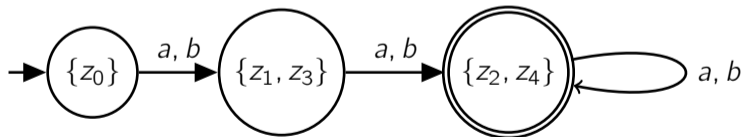


# Beispiel für die Konstruktion des minimalen DFA

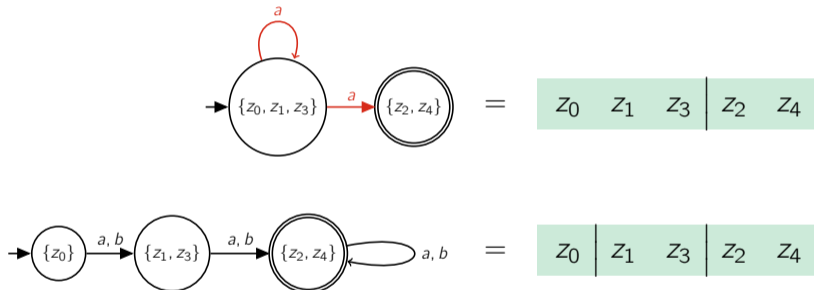
DFA  $M$ :



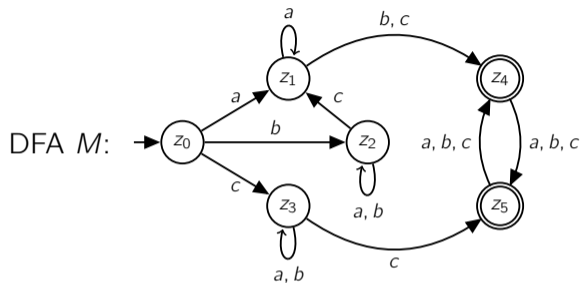
Minimaler DFA  $M'$ :



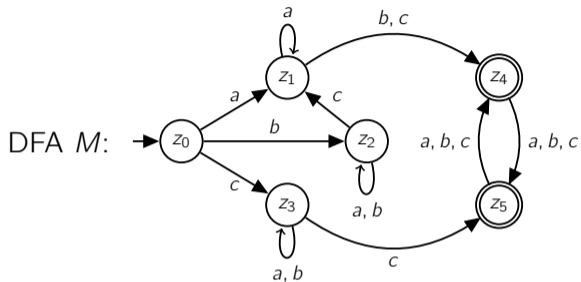
# Äquivalenz des graphischen und des tabellarischen Ansatzes



## Zweites Beispiel für den tabellarischen Ansatz



## Zweites Beispiel für den tabellarischen Ansatz

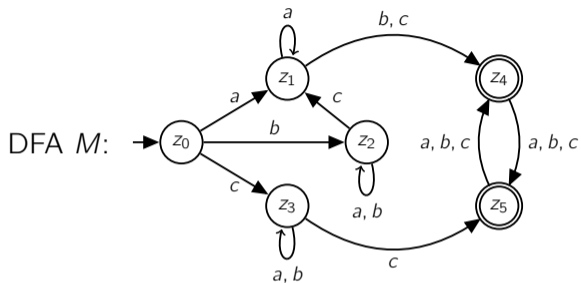


Partitionstabelle:

0.	z <sub>0</sub>	z <sub>1</sub>	z <sub>2</sub>	z <sub>3</sub>		z <sub>4</sub>	z <sub>5</sub>
----	----------------	----------------	----------------	----------------	--	----------------	----------------



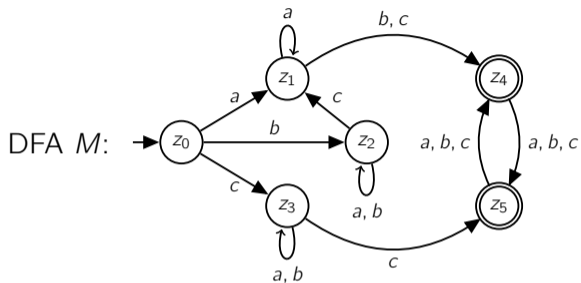
## Zweites Beispiel für den tabellarischen Ansatz



Partitionstabelle:

0.	<table border="1"><tr><td><math>z_0</math></td><td><math>z_1</math></td><td><math>z_2</math></td><td><math>z_3</math></td><td><math>z_4</math></td><td><math>z_5</math></td></tr></table>	$z_0$	$z_1$	$z_2$	$z_3$	$z_4$	$z_5$
$z_0$	$z_1$	$z_2$	$z_3$	$z_4$	$z_5$		
1.	<table border="1"><tr><td><math>z_1</math></td><td><math>z_0</math></td><td><math>z_2</math></td><td><math>z_3</math></td><td><math>z_4</math></td><td><math>z_5</math></td></tr></table> mit $b$	$z_1$	$z_0$	$z_2$	$z_3$	$z_4$	$z_5$
$z_1$	$z_0$	$z_2$	$z_3$	$z_4$	$z_5$		

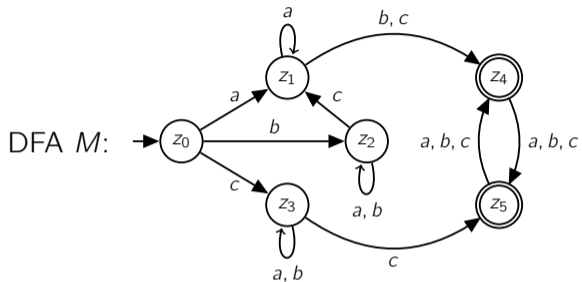
## Zweites Beispiel für den tabellarischen Ansatz



Partitionstabelle:

0.	$z_0$	$z_1$	$z_2$	$z_3$	$z_4$	$z_5$	
1.	$z_1$	$z_0$	$z_2$	$z_3$	$z_4$	$z_5$	mit $b$
2.	$z_1$	$z_0$	$z_2$	$z_3$	$z_4$	$z_5$	mit $c$

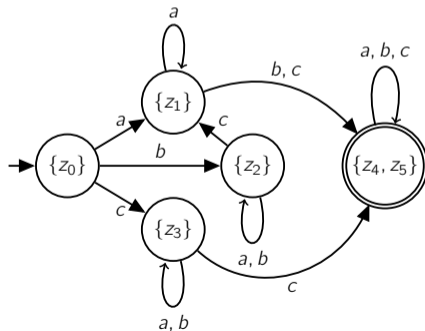
# Zweites Beispiel für den tabellarischen Ansatz



Partitionstabelle:

0.	$z_0$	$z_1$	$z_2$	$z_3$	$z_4$	$z_5$	
1.	$z_1$	$z_0$	$z_2$	$z_3$	$z_4$	$z_5$	mit $b$
2.	$z_1$	$z_0$	$z_2$	$z_3$	$z_4$	$z_5$	mit $c$

Minimaler DFA  $M'$ :



# Äquivalenzklassenautomat

Beide Ansätze basieren auf dem Äquivalenzklassenautomaten.

## Definition

Sei  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA.

Wir nennen zwei Zustände  $z, z' \in Z$  äquivalent und schreiben  $z \equiv_M z'$  (alternativ  $z \equiv z'$ ), falls gilt:  $\tilde{\delta}(z, w) \in E$  g.d.w.  $\tilde{\delta}(z', w) \in E$  für alle  $w \in \Sigma^*$ .

# Äquivalenzklassenautomat

Beide Ansätze basieren auf dem Äquivalenzklassenautomaten.

## Definition

Sei  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA.

Wir nennen zwei Zustände  $z, z' \in Z$  äquivalent und schreiben  $z \equiv_M z'$  (alternativ  $z \equiv z'$ ), falls gilt:  $\tilde{\delta}(z, w) \in E$  g.d.w.  $\tilde{\delta}(z', w) \in E$  für alle  $w \in \Sigma^*$ .

Der Äquivalenzklassenautomat zu  $M$  ist der DFA  $M' = (Z', \Sigma, \delta', z'_0, E')$  mit

$$Z' := \{[z]_{\equiv} \mid z \in Z\}$$

$$z'_0 := [z_0]_{\equiv}$$

$$E' := \{[z]_{\equiv} \mid z \in E\}$$

$$\delta'([z]_{\equiv}, a) := [\delta(z, a)]_{\equiv}$$

# Äquivalenzklassenautomat

Beide Ansätze basieren auf dem Äquivalenzklassenautomaten.

## Definition

Sei  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA.

Wir nennen zwei Zustände  $z, z' \in Z$  äquivalent und schreiben  $z \equiv_M z'$  (alternativ  $z \equiv z'$ ), falls gilt:  $\tilde{\delta}(z, w) \in E$  g.d.w.  $\tilde{\delta}(z', w) \in E$  für alle  $w \in \Sigma^*$ .

Der Äquivalenzklassenautomat zu  $M$  ist der DFA  $M' = (Z', \Sigma, \delta', z'_0, E')$  mit

$$Z' := \{[z]_{\equiv} \mid z \in Z\}$$

$$z'_0 := [z_0]_{\equiv}$$

$$E' := \{[z]_{\equiv} \mid z \in E\}$$

$$\delta'([z]_{\equiv}, a) := [\delta(z, a)]_{\equiv}$$

Informell: Zwei Zustände sind äquivalent, wenn sie die gleiche „Sprache“ darstellen.

# Korrektheit und Minimalität des Äquivalenzklassenautomaten

## Satz

Seien  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA und  $M' = (Z', \Sigma, \delta', z'_0, E')$  der Äquivalenzklassenautomat zu  $M$ . Dann gilt

1.  $L(M') = L(M)$ .
2. Falls alle Zustände in  $Z$  von  $z_0$  erreichbar sind, dann ist  $M'$  minimal, d.h. jeder DFA  $M''$  mit  $L(M'') = L(M')$  hat mindestens so viele Zustände wie  $M'$ .

# Korrektheit und Minimalität des Äquivalenzklassenautomaten

## Satz

Seien  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA und  $M' = (Z', \Sigma, \delta', z'_0, E')$  der Äquivalenzklassenautomat zu  $M$ . Dann gilt

1.  $L(M') = L(M)$ .
2. Falls alle Zustände in  $Z$  von  $z_0$  erreichbar sind, dann ist  $M'$  minimal, d.h. jeder DFA  $M''$  mit  $L(M'') = L(M')$  hat mindestens so viele Zustände wie  $M'$ .

## Beweis

1. Wir zeigen:  $w \in L(M')$  g.d.w.  $w \in L(M)$ . Sei  $w \in \Sigma^*$ .



# Korrektheit und Minimalität des Äquivalenzklassenautomaten

## Satz

Seien  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA und  $M' = (Z', \Sigma, \delta', z'_0, E')$  der Äquivalenzklassenautomat zu  $M$ . Dann gilt

1.  $L(M') = L(M)$ .
2. Falls alle Zustände in  $Z$  von  $z_0$  erreichbar sind, dann ist  $M'$  minimal, d.h. jeder DFA  $M''$  mit  $L(M'') = L(M')$  hat mindestens so viele Zustände wie  $M'$ .

## Beweis

1. Wir zeigen:  $w \in L(M')$  g.d.w.  $w \in L(M)$ . Sei  $w \in \Sigma^*$ .  
 $M$  durchläuft  $z_0, \dots, z_{|w|}$  entlang  $w$  und akzeptiert  $w$  g.d.w.  $z_{|w|} \in E$ .

# Korrektheit und Minimalität des Äquivalenzklassenautomaten

## Satz

Seien  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA und  $M' = (Z', \Sigma, \delta', z'_0, E')$  der Äquivalenzklassenautomat zu  $M$ . Dann gilt

1.  $L(M') = L(M)$ .
2. Falls alle Zustände in  $Z$  von  $z_0$  erreichbar sind, dann ist  $M'$  minimal, d.h. jeder DFA  $M''$  mit  $L(M'') = L(M')$  hat mindestens so viele Zustände wie  $M'$ .

## Beweis

1. Wir zeigen:  $w \in L(M')$  g.d.w.  $w \in L(M)$ . Sei  $w \in \Sigma^*$ .

$M$  durchläuft  $z_0, \dots, z_{|w|}$  entlang  $w$  und akzeptiert  $w$  g.d.w.  $z_{|w|} \in E$ .

$M'$  durchläuft  $[z_0]_{\equiv}, \dots, [z_{|w|}]_{\equiv}$  und akzeptiert  $w$  g.d.w.  $[z_{|w|}]_{\equiv} \in E'$ .

# Korrektheit und Minimalität des Äquivalenzklassenautomaten

## Satz

Seien  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA und  $M' = (Z', \Sigma, \delta', z'_0, E')$  der Äquivalenzklassenautomat zu  $M$ . Dann gilt

1.  $L(M') = L(M)$ .
2. Falls alle Zustände in  $Z$  von  $z_0$  erreichbar sind, dann ist  $M'$  minimal, d.h. jeder DFA  $M''$  mit  $L(M'') = L(M')$  hat mindestens so viele Zustände wie  $M'$ .

## Beweis

1. Wir zeigen:  $w \in L(M')$  g.d.w.  $w \in L(M)$ . Sei  $w \in \Sigma^*$ .

$M$  durchläuft  $z_0, \dots, z_{|w|}$  entlang  $w$  und akzeptiert  $w$  g.d.w.  $z_{|w|} \in E$ .

$M'$  durchläuft  $[z_0]_{\equiv}, \dots, [z_{|w|}]_{\equiv}$  und akzeptiert  $w$  g.d.w.  $[z_{|w|}]_{\equiv} \in E'$ .

Da per Definition  $[z_{|w|}]_{\equiv} \in E'$  äquivalent zu  $z_{|w|} \in E$  ist, folgt, dass

$L(M') = L(M)$ .

# Korrektheit und Minimalität des Äquivalenzklassenautomaten

## Satz

Seien  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA und  $M' = (Z', \Sigma, \delta', z'_0, E')$  der Äquivalenzklassenautomat zu  $M$ . Dann gilt

1.  $L(M') = L(M)$ .
2. Falls alle Zustände in  $Z$  von  $z_0$  erreichbar sind, dann ist  $M'$  minimal, d.h. jeder DFA  $M''$  mit  $L(M'') = L(M')$  hat mindestens so viele Zustände wie  $M'$ .

**Beweis** (Fortsetzung)

2. Wird in späterer Vorlesung gezeigt (nur FSK). □

## Algorithmus 3: Berechnung aller äquivalenten Zustände

**Eingabe:** DFA  $M = (Z, \Sigma, \delta, z_0, E)$ , der keine unerreichbaren Zustände hat

**Ausgabe:** Partition  $\mathcal{P} = \{[z_1]_{\equiv}, \dots, [z_m]_{\equiv}\}$  von  $Z$

**Beginn**

initialisiere Partition  $\mathcal{P}$  mit  $E$  (falls nicht leer) und  $Z \setminus E$  (falls nicht leer);

**wiederhole**

**für jedes**  $K = \{z_1, \dots, z_m\} \in \mathcal{P}$  mit  $|m| \geq 2$  **tue**

**für jedes**  $a \in \Sigma$  **tue**

            berechne die Partition  $\mathcal{Q}$  von  $\{z_1, \dots, z_m\}$  über  
            dem Wert von  $[\delta(z_i, a)]$  für jedes  $i$ ;

$\mathcal{P} := (\mathcal{P} \setminus \{K\}) \cup \mathcal{Q}$ ;

**Ende**

**Ende**

**bis** sich  $\mathcal{P}$  nicht mehr verändert;

**return**  $\mathcal{P}$

**Ende**

### Satz

Sei  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA, der keine unerreichbaren Zustände hat. Algorithmus 3 berechnet die Äquivalenzklassen bezüglich  $\equiv$ .

## Korrektheit von Algorithmus 3

---

### Satz

Sei  $M = (Z, \Sigma, \delta, z_0, E)$  ein DFA, der keine unerreichbaren Zustände hat. Algorithmus 3 berechnet die Äquivalenzklassen bezüglich  $\equiv$ .

**Beweis** Wir führen den Beweis in zwei Schritten:

1. Wenn  $z \equiv z'$ , dann sind  $z$  und  $z'$  in derselben Klasse von  $\mathcal{P}$ .
2. Wenn  $z \not\equiv z'$ , dann sind  $z$  und  $z'$  in verschiedenen Klassen von  $\mathcal{P}$ .

## **Beweis** (Fortsetzung)

1. Wir zeigen die Kontraposition: Wenn  $z$  und  $z'$  in verschiedenen Klassen von  $\mathcal{P}$  sind, dann  $z \not\equiv z'$ .



## **Beweis** (Fortsetzung)

1. Wir zeigen die Kontraposition: Wenn  $z$  und  $z'$  in verschiedenen Klassen von  $\mathcal{P}$  sind, dann  $z \neq z'$ . Und um  $z \neq z'$  zu zeigen, reicht es, ein Wort  $w$  zu finden, sodass  $\tilde{\delta}(z, w) \in E$  und  $\tilde{\delta}(z', w) \notin E$  oder umgekehrt.

## **Beweis** (Fortsetzung)

1. Wir zeigen die Kontraposition: Wenn  $z$  und  $z'$  in verschiedenen Klassen von  $\mathcal{P}$  sind, dann  $z \neq z'$ . Und um  $z \neq z'$  zu zeigen, reicht es, ein Wort  $w$  zu finden, sodass  $\tilde{\delta}(z, w) \in E$  und  $\tilde{\delta}(z', w) \notin E$  oder umgekehrt.

Durch Induktion über die Anzahl  $n$  der Schleifeniterationen bis  $z$  und  $z'$  getrennt wurden.

## Beweis (Fortsetzung)

1. Wir zeigen die Kontraposition: Wenn  $z$  und  $z'$  in verschiedenen Klassen von  $\mathcal{P}$  sind, dann  $z \neq z'$ . Und um  $z \neq z'$  zu zeigen, reicht es, ein Wort  $w$  zu finden, sodass  $\tilde{\delta}(z, w) \in E$  und  $\tilde{\delta}(z', w) \notin E$  oder umgekehrt.

Durch Induktion über die Anzahl  $n$  der Schleifeniterationen bis  $z$  und  $z'$  getrennt wurden.

- Fall  $n = 0$ : Da  $z$  und  $z'$  schon vor der ersten Iteration getrennt wurden, müssen  $z \in E$  und  $z' \notin E$  oder umgekehrt gelten. Wir nehmen  $w = \varepsilon$ :  $\tilde{\delta}(z, \varepsilon) \in E$  und  $\tilde{\delta}(z', \varepsilon) \notin E$  oder umgekehrt.

# Korrektheit des tabellarischen Ansatzes

## Beweis (Fortsetzung)

1. Wir zeigen die Kontraposition: Wenn  $z$  und  $z'$  in verschiedenen Klassen von  $\mathcal{P}$  sind, dann  $z \neq z'$ . Und um  $z \neq z'$  zu zeigen, reicht es, ein Wort  $w$  zu finden, sodass  $\tilde{\delta}(z, w) \in E$  und  $\tilde{\delta}(z', w) \notin E$  oder umgekehrt.

Durch Induktion über die Anzahl  $n$  der Schleifeniterationen bis  $z$  und  $z'$  getrennt wurden.

- ▶ Fall  $n = 0$ : Da  $z$  und  $z'$  schon vor der ersten Iteration getrennt wurden, müssen  $z \in E$  und  $z' \notin E$  oder umgekehrt gelten. Wir nehmen  $w = \varepsilon$ :  $\tilde{\delta}(z, \varepsilon) \in E$  und  $\tilde{\delta}(z', \varepsilon) \notin E$  oder umgekehrt.
- ▶ Fall  $n > 0$ : Da  $z$  und  $z'$  in Iteration  $n$  getrennt wurden, muss es  $a \in \Sigma$  geben, sodass  $\delta(z, a)$  und  $\delta(z', a)$  schon in Iteration  $n - 1$  getrennt waren. Die Induktionshypothese liefert ein Wort  $w'$  mit  $\tilde{\delta}(\delta(z, a), w') \in E$  und  $\tilde{\delta}(\delta(z', a), w') \notin E$  oder umgekehrt. Wir nehmen  $w = aw'$ :  $\tilde{\delta}(z, aw') \in E$  und  $\tilde{\delta}(z', aw') \notin E$  oder umgekehrt.

## Beweis (Fortsetzung)

2. Wir müssen zeigen, dass wenn  $z \neq z'$ , dann sind  $z$  und  $z'$  in zwei verschiedenen Klassen von  $\mathcal{P}$ . Sei  $w$  ein Wort sodass  $\tilde{\delta}(z, w) \in E$  und  $\tilde{\delta}(z', w) \notin E$  oder umgekehrt.

Durch Induktion über  $|w|$ .

- ▶ Fall  $w = \varepsilon$ : Dann haben wir  $\tilde{\delta}(z, \varepsilon) \in E$  und  $\tilde{\delta}(z', \varepsilon) \notin E$  oder umgekehrt. Die Initialisierungsphase sorgt dafür, dass  $z$  und  $z'$  getrennt werden.
- ▶ Fall  $w$  ist von der Form  $aw'$ : Dann haben wir  $\tilde{\delta}(z, aw') = \tilde{\delta}(\delta(z, a), w') \in E$  und  $\tilde{\delta}(z', aw') = \tilde{\delta}(\delta(z', a), w') \notin E$  oder umgekehrt. Per Induktionshypothese müssen  $\delta(z, a)$  und  $\delta(z', a)$  getrennt sein. Der Algorithmus muss  $z$  und  $z'$  trennen, wenn  $a$  betrachtet wird. □

## Algorithmus 4: Minimierung von DFAs

---

**Eingabe:** DFA  $M = (Z, \Sigma, \delta, z_0, E)$

**Ausgabe:** Minimaler DFA  $M'$  mit  $L(M') = L(M)$

**Beginn**

entferne Zustände aus  $M$ , die nicht vom Startzustand aus erreichbar sind;  
berechne äquivalente Zustände mit Algorithmus 3;  
erzeuge den Äquivalenzklassenautomat, indem die berechneten äquivalenten Zustände verschmolzen werden;

**Ende**