

# Theoretische Informatik: Endliche Automaten, Formale Sprachen und Grammatiken

Marko Livajusic

4. November 2024

# Inhaltsverzeichnis

<b>1</b>	<b>Deterministische Endliche Automaten</b>	<b>2</b>
1.1	Transduktor . . . . .	2
1.1.1	Mealy-Automat . . . . .	2
1.2	Akzeptor . . . . .	2
1.2.1	Moore-Automat . . . . .	3
1.2.2	Minimierung von DEAs . . . . .	4
<b>2</b>	<b>Nichtdeterministische Endliche Automaten</b>	<b>6</b>
2.1	Epsilon-NEAs . . . . .	6
2.1.1	Epsilon-NEA zu NEA . . . . .	6
2.1.2	Epsilon-NEA zu DEA . . . . .	8
2.2	NEA zu DEA mit Potenzmengenkonstruktion . . . . .	11
<b>3</b>	<b>Reguläre Ausdrücke</b>	<b>12</b>
3.1	RegEx zu NEA . . . . .	12
3.1.1	Regulärer Ausdruck: Leere Menge . . . . .	12
3.1.2	Regulärer Ausdruck: Leeres Wort . . . . .	12
3.1.3	Regulärer Ausdruck: Eingabesymbol . . . . .	12
3.1.4	Regulärer Ausdruck: Verkettung . . . . .	12
3.1.5	Regulärer Ausdruck: Alternative . . . . .	13
3.1.6	Regulärer Ausdruck: N-malige Wiederholung . . . . .	13

# 1. Deterministische Endliche Automaten

## 1.1 Transduktor

**Definition 1** Ein Transduktorautomat  $\mathcal{T} : \{\Sigma, A, Z, z_0, \delta, \lambda\}$  ist ein deterministischer endlicher Automat ohne einen Endzustand.

$\Sigma$  : Eingabealphabet

$A$  : Ausgabealphabet

$Z$  : Zustandsmenge

$z_0 \in Z$  : Startzustand

$\delta : \Sigma \times Z \rightarrow Z$  : Überföhrungsfunktion

$\lambda : \Sigma \times Z \rightarrow A^*$  : Ausgabefunktion

### 1.1.1 Mealy-Automat

**Definition 2** Ein Mealy-Automat <sup>1</sup> ist ein Transduktor, dessen Ausgabe von der Überföhrungsfunktion  $\delta$  und vom aktuellen **Zustand**  $z_n$  abhängig ist.

## 1.2 Akzeptor

**Definition 3** Ein Akzeptor  $\mathcal{A} : \{\Sigma, Z, z_0, \delta, F\}$  ist ein deterministischer endlicher Automat, der die Eingabe überprüft und keine Ausgabe besitzt. Er lässt sich wie folgt beschreiben:

---

<sup>1</sup>für die Klausur irrelevant.

$\Sigma$  : Eingabealphabet

$Z$  : Zustandsmenge

$z_0$  : Startzustand

$\delta$  : Überföhrungsfunktion

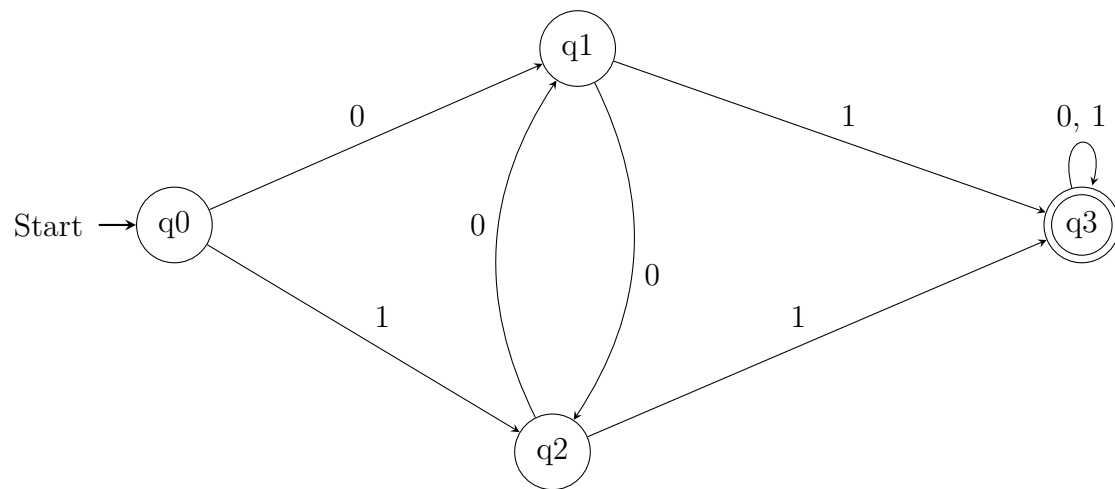
$F$  : Endzustandsmenge

### 1.2.1 Moore-Automat

**Definition 4** *Ein Moore-Automat ist ein Transduktor, dessen Ausgabe vom aktuellen **Zustand**  $z_n$  abhängig ist.*

### 1.2.2 Minimierung von DEAs

Zu minimieren sei folgender DEA:



Diagonale als äquivalent markieren:

Zustand	$q_0$	$q_1$	$q_2$	$q_3$
$q_0$	$\equiv$			
$q_1$		$\equiv$		
$q_2$			$\equiv$	
$q_3$				$\equiv$

Felder, wo ein Zustand auf einen Endzustand trifft, streichen

Zustand	$q_0$	$q_1$	$q_2$	$q_3$
$q_0$	$\equiv$			
$q_1$		$\equiv$		
$q_2$			$\equiv$	
$q_3$	X	X	X	$\equiv$

Eine Übergangstabelle mit übrigen Zuständen erstellen. Die Zustandspaare, die auf einen bereits gestrichenen Zustandspaar abgebildet werden, streichen

Zustand	0	1
<del><math>(q_0, q_1)</math></del>	$(q_1, q_2)$	<del><math>(q_2, q_3)</math></del>
<del><math>(q_0, q_2)</math></del>	$(q_1, q_1)$	<del><math>(q_2, q_3)</math></del>
$(q_1, q_2)$	$(q_2, q_1)$	$(q_3, q_3)$

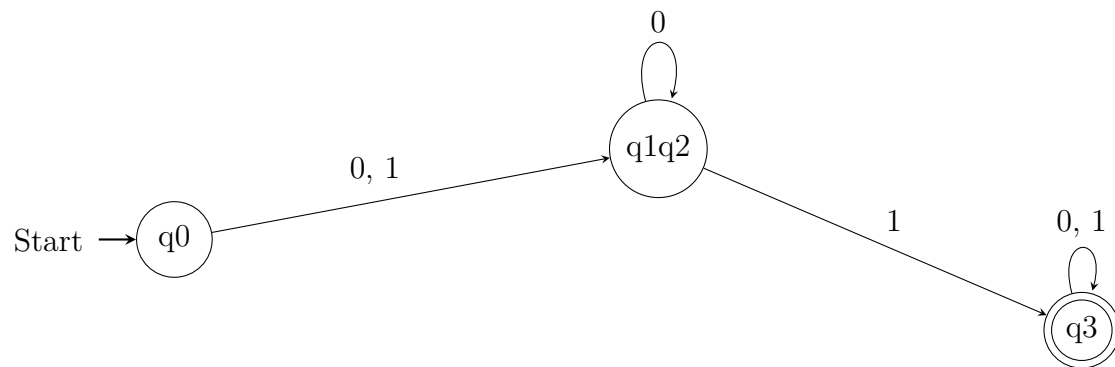
Die neue Tabelle sieht dann so aus:

Zustand	$q_0$	$q_1$	$q_2$	$q_3$
$q_0$	$\equiv$			
$q_1$	X	$\equiv$		
$q_2$	X		$\equiv$	
$q_3$	X	X	X	$\equiv$

Die leeren Felder als äquivalent markieren:

Zustand	$q_0$	$q_1$	$q_2$	$q_3$
$q_0$	$\equiv$			
$q_1$	X	$\equiv$		
$q_2$	X	$\equiv$	$\equiv$	
$q_3$	X	X	X	$\equiv$

Spaltenweise die Zustände zusammenfassen:



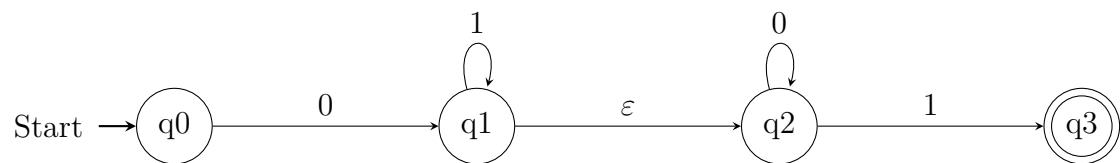
## 2. Nichtdeterministische Endliche Automaten

### 2.1 $\epsilon$ -NEAs

**Definition 5** Ein  $\epsilon$ -NEA ist ein Akzeptor, der  $\epsilon$ -Übergänge besitzt und deshalb mit dem leeren Wort Zustände wechseln kann.

#### 2.1.1 $\epsilon$ -NEA $\rightarrow$ NEA

Gegeben sei folgendes Zustandsdiagramm eines  $\epsilon$ -NEA, welches in einen NEA umgewandelt werden soll:



Zuerst wird eine leere Übergangstabelle erstellt:

Zustand	0	1
$q_0$		
$q_1$		
$q_2$		
$q_3$		

Danach wird für jedes Eingabesymbol eine Tabelle mit der  $\epsilon$ -Hülle erstellt:

Zustand	$\epsilon^*$	0	$\epsilon^*$
$q_0$			

Wie oben zu sehen ist, wird zuerst der Startzustand  $q_0$  eingetragen. Danach wird die  $\epsilon$ -Hülle des Zustands  $q_0$  berechnet und eingetragen.

**Definition 6** Eine  $\epsilon$ -Hülle ist die Menge aller Zustände, die ein Zustand  $q_n$  mit dem leeren Wort  $\epsilon$  erreichen kann.

Da im vorigen Beispiel  $q_0$  mit dem leeren Wort keinen anderen Zustand als sich selbst erreichen kann, wird für dessen  $\epsilon$ -Hülle  $q_0$  eingetragen.

Die nächste Spalte steht für den Zustand, der erreicht wird, wenn bei  $q_0$  das Eingabesymbol 0 eingegeben wird. Dies ist in diesem Beispiel der Zustand  $q_1$ :

Zustand	$\epsilon^*$	0	$\epsilon^*$
$q_0$	$q_0$	<b><math>q_1</math></b>	

Die letzte Spalte bezieht sich auf die  $\epsilon$ -Hülle des Zustands aus der mittleren Spalte, welcher hier fettgedruckt steht. Die  $\epsilon$ -Hülle von  $q_1$  ist dabei  $\{q_1, q_2\}$ . Diese wird ebenfalls eingetragen:

Zustand	$\epsilon^*$	0	$\epsilon^*$
$q_0$	$q_0$	$q_1$	$\{q_1, q_2\}$

Diese  $\epsilon$ -Hülle  $\{q_1, q_2\}$  repräsentiert dabei die Zustände, die  $q_0$  bei der Eingabe von 0 erreicht werden. Deshalb können diese in die Übergangstabelle eingetragen werden:

Zustand	0	1
$q_0$	$\{q_1, q_2\}$	
$q_1$		
$q_2$		
$q_3$		

Dieser Vorgang wird für alle Zustände durchgeführt, sowohl für die Eingabe von 0 als auch von 1. Die Tabellen sehen nach dem Algorithmus wie folgt aus:

Zustand	$\epsilon^*$	0	$\epsilon^*$
$\{q_0\}$	$\{q_0\}$	$\{q_1\}$	$\{q_1, q_2\}$
$\{q_1\}$	$\{q_1\}$ $\{q_1, q_2\}$	$\emptyset$ $\{q_2\}$	$\emptyset$ $\{q_2\}$
$\{q_2\}$	$\{q_2\}$	$\{q_2\}$	$\{q_2\}$
$\{q_3\}$	$\{q_3\}$	$\emptyset$	$\emptyset$



Zustand	$\epsilon^*$	1	$\epsilon^*$
$\{q_0\}$	$\{q_0\}$	$\emptyset$	$\emptyset$
$\{q_1\}$	$\{q_1\}$ $\{q_2\}$	$\{q_1\}$ $\{q_3\}$	$\{q_1, q_2\}$ $\{q_3\}$
$\{q_2\}$	$\{q_2\}$	$\{q_3\}$	$\{q_3\}$
$\{q_3\}$	$\{q_3\}$	$\emptyset$	$\emptyset$

Zustand	0	1
$\{q_0\}$	$\{q_1, q_2\}$	$\emptyset$
$\{q_1\}$	$\{q_2\}$	$\{q_1, q_2, q_3\}$
$\{q_2\}$	$\{q_2\}$	$\{q_3\}$
$\{q_3\}$	$\emptyset$	$\emptyset$

Noch sollen die Endzustände ermittelt werden. Zu den Endzuständen gehört der Endzustand aus dem  $\epsilon$ -NEA und die Zustände, die durch das leere Wort  $\epsilon$  in den ursprünglichen Endzustand gelangen können. Deshalb wird in diesem Fall nur  $q_3$  der Endzustand. Gezeichnet sieht das neue Zustandsdiagramm wie folgt aus:

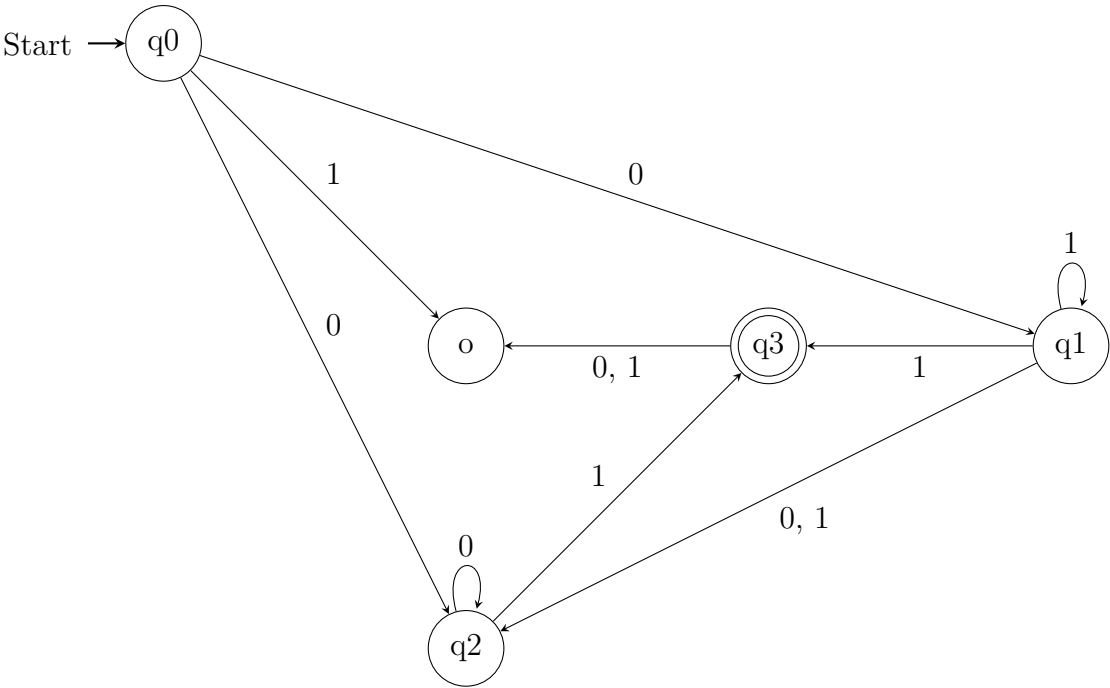
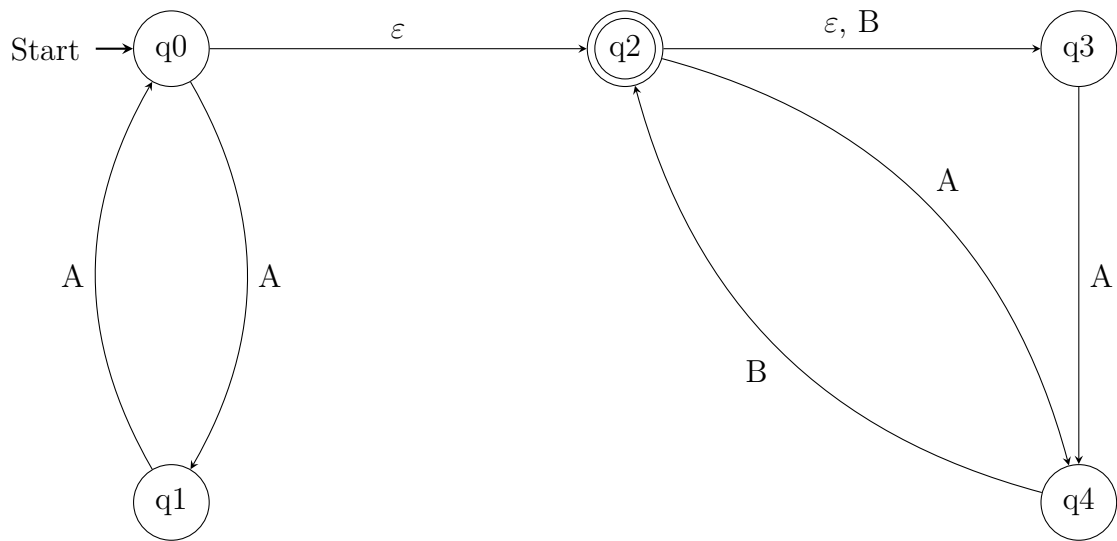


Abbildung 2.1: Der neue NEA, ohne  $\epsilon$ -Übergänge.

„o“ steht hier für die leere Menge  $\emptyset$ .

2.1.2  $\epsilon$ -NEA  $\rightarrow$  DEA

Es sei folgendes Zustandsdiagramm eines  $\epsilon$ -NEAs gegeben:



Die Umwandlung in ein DEA geschieht wie üblich mit der Potenzmengenkonstruktion:

Zustand	A	B
$\rightarrow \{q_0\}$	$\{q_1, q_4\}$	$\{q_3\}$
$\{q_1, q_4\}$	$\{q_0\}$	$\{q_2^*\}$
$\{q_3\}$	$\{q_4\}$	$\emptyset$
$\{q_2^*\}$	$\{q_4\}$	$\{q_3\}$
$\{q_4\}$	$\emptyset$	$\{q_2\}$
$\emptyset$	$\emptyset$	$\emptyset$

Anschließend wird das neue Zustandsdiagramm des DEAs gezeichnet.  $qE$  repräsentiert dabei die leere Menge  $\emptyset$ .

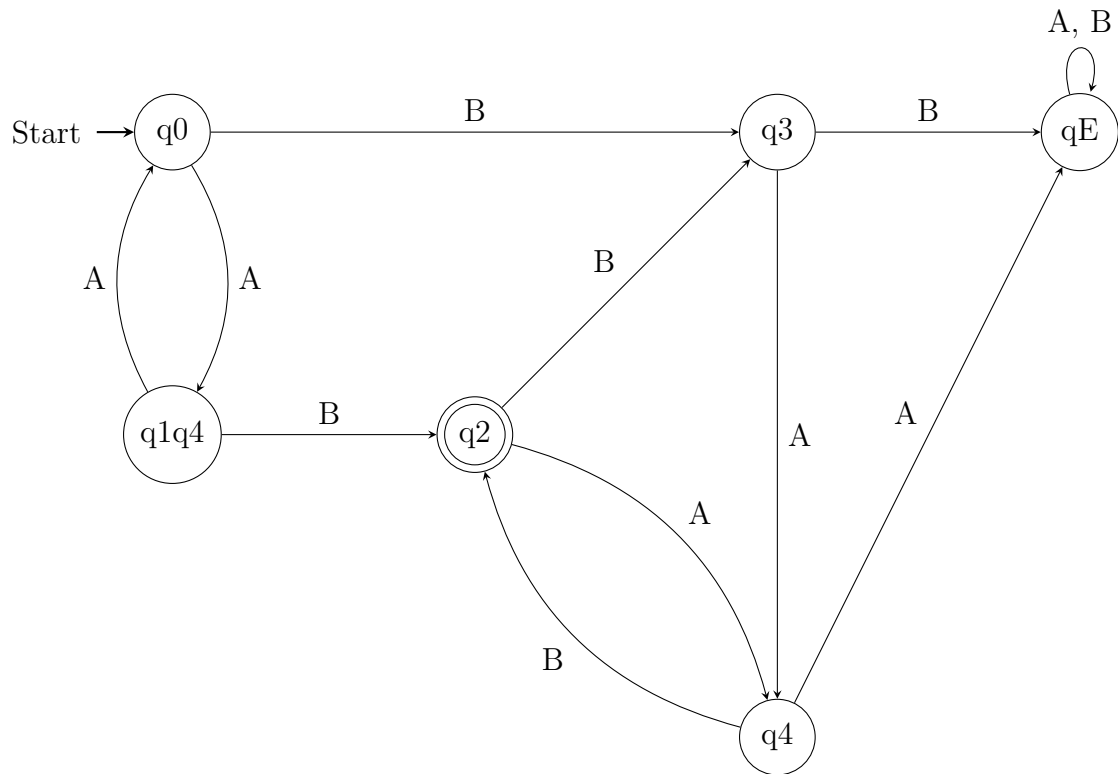
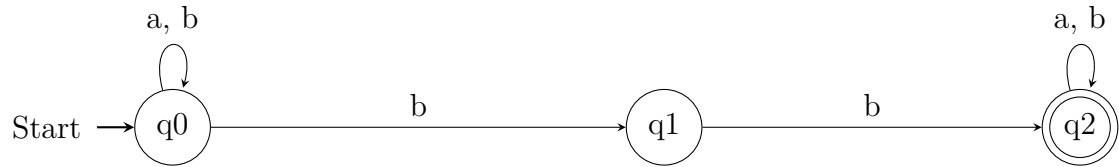


Abbildung 2.2: Umwandlung von  $\epsilon$ -NEA zu DEA. Dieser ist jedoch nicht zwangsläufig optimal bzw. minimal.

## 2.2 NEA $\rightarrow$ DEA (Potenzmengenkonstruktion)

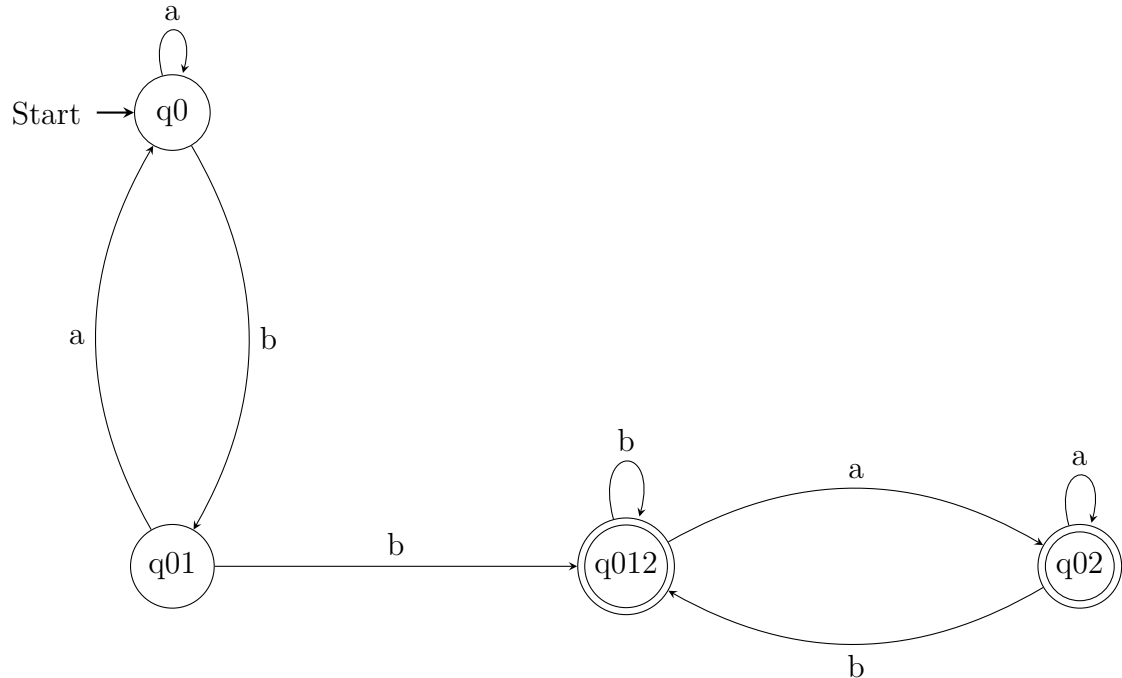
Dieser NEA soll in einen DEA umgewandelt werden:



**Vorgehen:** Es wird zuerst eine Übergangstabelle aufgestellt und geschaut, welche Zustände neu auftreten.

Zustand	$a$	$b$
$\rightarrow \{q_0\}$	$\{q_0\}$	$\{q_0, q_1\}$
$\{q_0, q_1\}$	$\{q_0\}$	$\{q_0, q_1, q_2^*\}$
$\{q_0, q_1, q_2\}^*$	$\{q_0, q_2^*\}$	$\{q_0, q_1, q_2^*\}$
$\{q_0, q_2\}^*$	$\{q_0, q_2^*\}$	$\{q_0, q_1, q_2^*\}$

Danach wird aus dieser Übergangstabelle der DEA gezeichnet:



## 3. Reguläre Ausdrücke

$+$  : wiederhole das Zeichen davor  $n$ -mal, wobei  $n > 0$

$*$  : wiederhole das Zeichen davor  $n$ -mal, wobei  $n \geq 0$

### 3.1 RegEx $\rightarrow$ $\epsilon$ -NEA

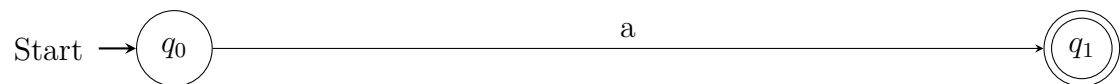
#### 3.1.1 $R = \emptyset$



#### 3.1.2 $R = \epsilon$

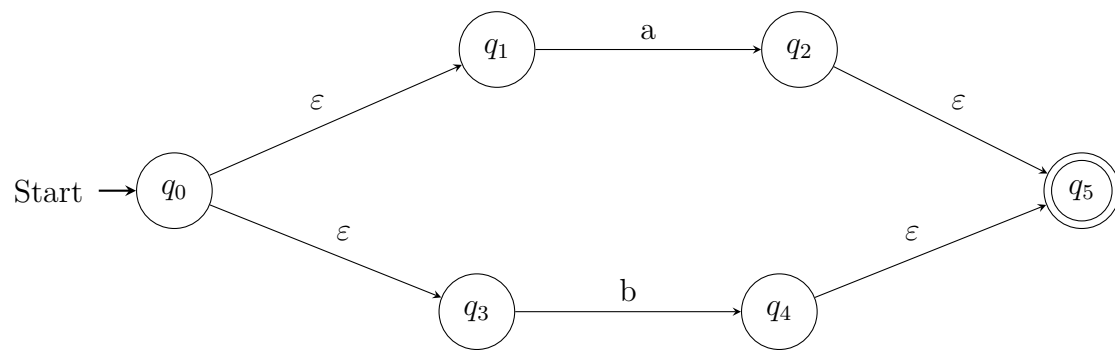
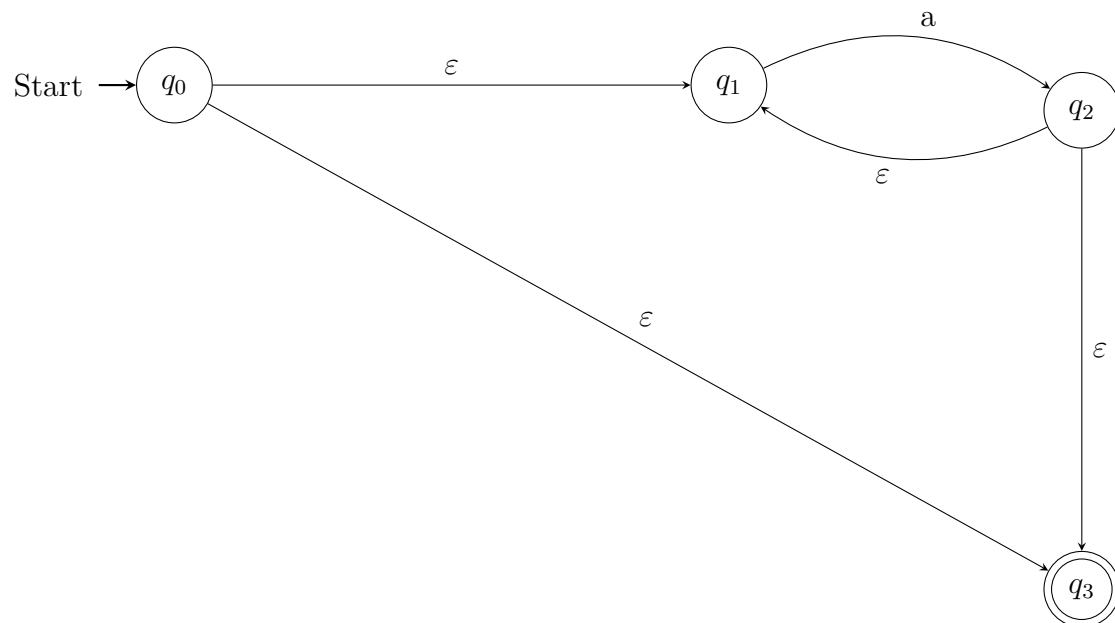


#### 3.1.3 $R = a$



#### 3.1.4 $R = ab$



**3.1.5**  $R = a|b$ **3.1.6**  $R = a^*$ 

**Beispiel 1** Es soll der reguläre Ausdruck  $(0|1)^*01$  in einen  $\epsilon$ -NEA umgewandelt werden.