Kapitel 2

Nichtdeterministische Endliche Automaten

Grundidee

- Wir lassen bei festem Eingabezeichen mehrere (oder keine) Optionen zum Übergang in einen neuen Zustand zu.

Grundidee

- Wir lassen bei festem Eingabezeichen mehrere (oder keine) Optionen zum Übergang in einen neuen Zustand zu.
- Damit hat ein Automat im Allgemeinen auf jedem Eingabewort mehrere mögliche Läufe.

Grundidee

- Wir lassen bei festem Eingabezeichen mehrere (oder keine) Optionen zum Übergang in einen neuen Zustand zu.
- Damit hat ein Automat im Allgemeinen auf jedem Eingabewort mehrere mögliche Läufe.
- Zum Akzeptieren des Eingabewortes genügt ein erfolgreicher ("akzeptierender") Lauf.

Grundidee

- Wir lassen bei festem Eingabezeichen mehrere (oder keine) Optionen zum Übergang in einen neuen Zustand zu.
- Damit hat ein Automat im Allgemeinen auf jedem Eingabewort mehrere mögliche Läufe.
- Zum Akzeptieren des Eingabewortes genügt ein erfolgreicher ("akzeptierender") Lauf.
- In einem zweiten Schritt werden wir unser Automatenmodell noch um Transitionen erweitern, die von keinem Eingabesymbol ausgelöst werden.

Beispiele 2.1

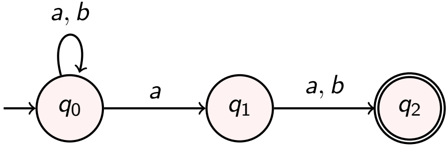
Alphabet $\Sigma := \{a, b\}$.

- $L_1 := \{w \in \Sigma^* \mid \text{das vorletzte Zeichen von } w \text{ ist ein } a\}$.

Beispiele 2.1

Alphabet $\Sigma := \{a, b\}$.

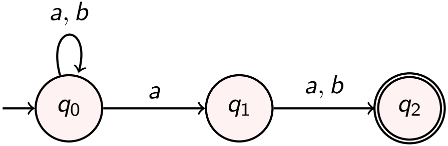
- $L_1 := \{w \in \Sigma^* \mid \text{das vorletzte Zeichen von } w \text{ ist ein } a\}$.



Beispiele 2.1

Alphabet $\Sigma := \{a, b\}$.

- $L_1 := \{w \in \Sigma^* \mid \text{das vorletzte Zeichen von } w \text{ ist ein } a\}$.

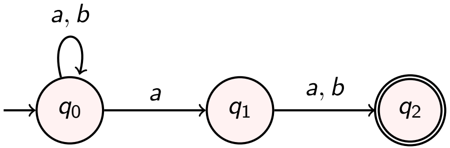


- $L_2 := \{w \in \Sigma^* \mid w \text{ enthält } abb \text{ als Infix}\}$.

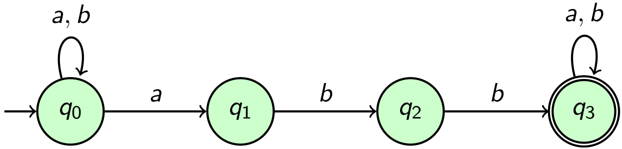
Beispiele 2.1

Alphabet $\Sigma := \{a, b\}$.

- $L_1 := \{w \in \Sigma^* \mid \text{das vorletzte Zeichen von } w \text{ ist ein } a\}$.



- $L_2 := \{w \in \Sigma^* \mid w \text{ enthält } abb \text{ als Infix}\}$.



Abschnitt 2.1

Nichtdeterministische Endliche Automaten (Formal)

Definition 2.2

Ein nichtdeterministischer endlicher Automat (kurz: NFA) ist ein 5-Tupel

$$(Q, \Sigma, \Delta, q_0, F);$$

dabei ist

- $\Delta \subseteq Q \times \Sigma \times Q$ die Transitionsrelation,

(q, a, q')
(q, a, q'')

und wie bei einem DFA sind

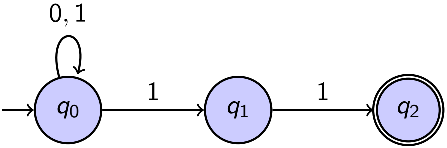
- Q eine endliche Menge, deren Elemente wir als die Zustände bezeichnen,
- Σ ein Alphabet, das Eingabealphabet,
- $q_0 \in Q$ der Anfangszustand,
- $F \subseteq Q$ die Menge der Endzustände oder akzeptierenden Zustände.

Graphische Darstellung

NFAs lassen sich wie DFAs graphisch darstellen; eine mit $a \in \Sigma$ beschriftete Kante vom Zustand $q \in Q$ zum Zustand $r \in Q$ zeigt an, dass $(q, a, r) \in \Delta$.

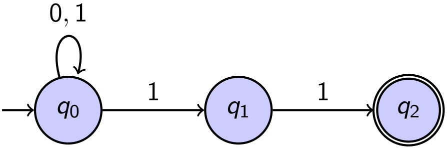
NFAs lassen sich wie DFAs graphisch darstellen; eine mit $a \in \Sigma$ beschriftete Kante vom Zustand $q \in Q$ zum Zustand $r \in Q$ zeigt an, dass $(q, a, r) \in \Delta$.

Beispiel 2.3



NFAs lassen sich wie DFAs graphisch darstellen; eine mit $a \in \Sigma$ beschriftete Kante vom Zustand $q \in Q$ zum Zustand $r \in Q$ zeigt an, dass $(q, a, r) \in \Delta$.

Beispiel 2.3



stellt den Automaten $\big(\{q_0, q_1, q_2\}, \{0, 1\}, \Delta, q_0, \{q_2\}\big)$ mit

$$\Delta = \big\{(q_0, 0, q_0), (q_0, 1, q_0), (q_0, 1, q_1), (q_1, 1, q_2)\big\}$$

dar.

Definition 2.4

Sei $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ ein NFA.

1. Ein Lauf von $\mathcal{A}$ ist eine Folge $(r_0, a_1, r_1, a_2, \ldots, r_{n-1}, a_n, r_n)$, für ein $n \geq 0$, wobei $r_0, \ldots, r_n \in Q$ und $a_1, \ldots, a_n \in \Sigma$, so dass

   (i) $r_0 = q_0$,

   (ii) $(r_{i-1}, a_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

Definition 2.4

Sei $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ ein NFA.

1. Ein Lauf von $\mathcal{A}$ ist eine Folge $(r_0, a_1, r_1, a_2, \ldots, r_{n-1}, a_n, r_n)$, für ein $n \geq 0$, wobei $r_0, \ldots, r_n \in Q$ und $a_1, \ldots, a_n \in \Sigma$, so dass

   (i) $r_0 = q_0$,

   (ii) $(r_{i-1}, a_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

   Wir bezeichnen $(r_0, a_1, r_1, a_2, \ldots, r_{n-1}, a_n, r_n)$ oder einfach die Zustandsfolge $(r_0, r_1, \ldots, r_n)$ auch als einen Lauf von $\mathcal{A}$ auf dem Wort $w = a_1 \ldots a_n$.

Definition 2.4

Sei $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ ein NFA.

1. Ein Lauf von $\mathcal{A}$ ist eine Folge $(r_0, a_1, r_1, a_2, \dots, r_{n-1}, a_n, r_n)$, für ein $n \geq 0$, wobei $r_0, \dots, r_n \in Q$ und $a_1, \dots, a_n \in \Sigma$, so dass

   (i) $r_0 = q_0$,

   (ii) $(r_{i-1}, a_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

   zum Vergleich: $\delta(r_{i-1}, a_i) = r_i$ bei DFAs

   Wir bezeichnen $(r_0, a_1, r_1, a_2, \dots, r_{n-1}, a_n, r_n)$ oder einfach die Zustandsfolge $(r_0, r_1, \dots, r_n)$ auch als einen Lauf von $\mathcal{A}$ auf dem Wort $w = a_1 \dots a_n$.

2. Der Lauf $(r_0, a_1, r_1, a_2, \dots, r_{n-1}, a_n, r_n)$ ist akzeptierend, wenn $r_n \in F$.

Definition 2.4

Sei $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ ein NFA.

1. Ein Lauf von $\mathcal{A}$ ist eine Folge $(r_0, a_1, r_1, a_2, \dots, r_{n-1}, a_n, r_n)$, für ein $n \geq 0$, wobei $r_0, \dots, r_n \in Q$ und $a_1, \dots, a_n \in \Sigma$, so dass

   (i) $r_0 = q_0$,

   (ii) $(r_{i-1}, a_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

   Wir bezeichnen $(r_0, a_1, r_1, a_2, \dots, r_{n-1}, a_n, r_n)$ oder einfach die Zustandsfolge $(r_0, r_1, \dots, r_n)$ auch als einen Lauf von $\mathcal{A}$ auf dem Wort $w = a_1 \dots a_n$.
2. Der Lauf $(r_0, a_1, r_1, a_2, \dots, r_{n-1}, a_n, r_n)$ ist akzeptierend, wenn $r_n \in F$.

Bemerkung 2.5

Zu einem Wort $w \in \Sigma^*$ kann es mehrere oder keinen Lauf von $\mathcal{A}$ auf w geben.

Definition 2.6

1. Ein NFA $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ akzeptiert ein Wort $w \in \Sigma^*$, wenn es einen akzeptierenden Lauf von $\mathcal{A}$ auf w gibt.

Definition 2.6

1. Ein NFA $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ akzeptiert ein Wort $w \in \Sigma^*$, wenn es einen akzeptierenden Lauf von $\mathcal{A}$ auf w gibt.
2. Die von einem NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ erkannte Sprache ist

$$L(\mathcal{A}) := \{w \in \Sigma^* \mid \mathcal{A} \text{ akzeptiert } w\}.$$

Semantik von NFAs

Definition 2.6

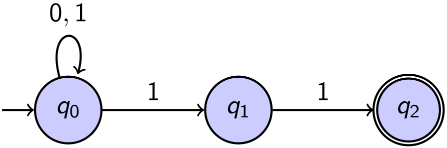
1. Ein NFA $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ akzeptiert ein Wort $w \in \Sigma^*$, wenn es einen akzeptierenden Lauf von $\mathcal{A}$ auf w gibt.
2. Die von einem NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ erkannte Sprache ist

$$L(\mathcal{A}) := \{w \in \Sigma^* \mid \mathcal{A} \text{ akzeptiert } w\}.$$

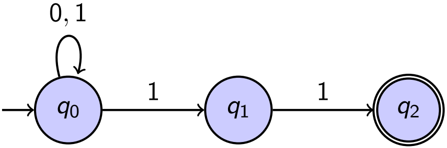
3. Eine Sprache L heißt NFA-erkennbar, wenn es einen ~~D~~NFA $\mathcal{A}$ gibt, so dass $L = L(\mathcal{A})$.

Beispiel 2.3 (Forts.)

Sei $\mathcal{A}$ folgender NFA:



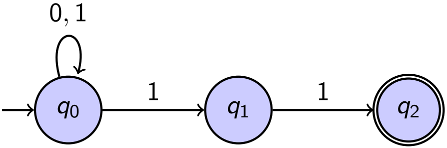
Sei $\mathcal{A}$ folgender NFA:



Akzeptierender Lauf auf dem Wort 0010111

$$\big(q_0, 0, q_0, 0, q_0, 1, q_0, 0, q_0, 1, q_0, 1, q_1, 1, q_2\big).$$

Sei $\mathcal{A}$ folgender NFA:



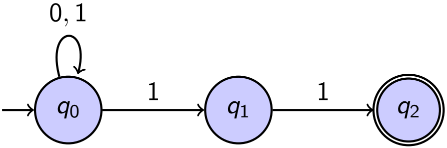
Akzeptierender Lauf auf dem Wort 0010111

$$\big(q_0, 0, q_0, 0, q_0, 1, q_0, 0, q_0, 1, q_0, 1, q_1, 1, q_2\big).$$

Nichtakzeptierender Lauf auf dem selben Wort

$$\big(q_0, 0, q_0, 0, q_0, 1, q_0, 0, q_0, 1, q_0, 1, q_0, 1, q_1\big).$$

Sei $\mathcal{A}$ folgender NFA:



Akzeptierender Lauf auf dem Wort 0010111

$$\big(q_0, 0, q_0, 0, q_0, 1, q_0, 0, q_0, 1, q_0, 1, q_1, 1, q_2\big).$$

Nichtakzeptierender Lauf auf dem selben Wort

$$\big(q_0, 0, q_0, 0, q_0, 1, q_0, 0, q_0, 1, q_0, 1, q_0, 1, q_1\big).$$

Behauptung

$$L(\mathcal{A}) = \big\{w \in \{0,1\}^* \;\big|\; 11 \textit{ ist Suffix von } w\big\}.$$

Behauptung 1

Sei $w \in \{0,1\}^*$. Dann gibt es einen Lauf von $\mathcal{A}$ auf w, der im Zustand q_0 endet.

Beweis.

Sei $w = a_1 \dots a_n$. Dann ist

$$\big(q_0, a_1, q_0, a_2, q_0, \dots, q_0, a_n, q_0\big)$$

ein Lauf von $\mathcal{A}$ auf w. □

Behauptung 2

Sei $w = a_1 \dots a_n \in \{0,1\}^*$. Dann gibt es genau dann einen Lauf von $\mathcal{A}$ auf w, der im Zustand q_1 endet, wenn $n \geq 1$ und $a_n = 1$.

Beweis.

„$\Longrightarrow$“: Sei

$$(r_0, a_1, r_1, a_2, \dots, a_n, r_n)$$

ein Lauf von $\mathcal{A}$ auf w mit $r_n = q_1$. Dann gilt $n \geq 1$, weil $r_0 = q_0$.
Weil $(r_{n-1}, a_n, q_1) \in \Delta$ gilt außerdem $r_{n-1} = q_0$ und $a_n = 1$, denn $(q_0, 1, q_1)$ ist die einzige Transition $(q, a, q') \in \Delta$ mit $q' = q_1$.

„$\Longleftarrow$“: Gelte $n \geq 1$ und $a_n = 1$.
Nach Behauptung 1 gibt es einen Lauf

$$(r_0, a_1, r_1, a_2, \dots, a_{n-1}, r_{n-1})$$

von $\mathcal{A}$ auf $a_1 \dots a_{n-1}$ mit $r_{n-1} = q_0$. Dann ist

$$(r_0, a_1, r_1, a_2, \dots, a_{n-1}, r_{n-1}, 1, q_1)$$

ein Lauf von $\mathcal{A}$ auf w. □

Behauptung 3

Sei $w = a_1 \dots a_n \in \{0,1\}^*$. Dann gibt es genau dann einen Lauf von $\mathcal{A}$ auf w, der im Zustand q_2 endet, wenn $n \geq 2$ und $a_{n-1} = 1$ und $a_n = 1$.

Beweis.

„$\Longrightarrow$“: Sei

$$(r_0, a_1, r_1, a_2, \dots, a_n, r_n)$$

ein Lauf von $\mathcal{A}$ auf w mit $r_n = q_2$. Dann gilt $n \geq 1$, weil $r_0 = q_0$.
Weil $(r_{n-1}, a_n, q_2) \in \Delta$ gilt außerdem $r_{n-1} = q_1$ und $a_n = 1$, denn $(q_1, 1, q_2)$ ist die einzige Transition $(q, a, q') \in \Delta$ mit $q' = q_2$.
Also ist

$$(r_0, a_1, r_1, a_2, \dots, a_{n-1}, r_{n-1})$$

ein Lauf von $\mathcal{A}$ auf $a_1 \dots a_{n-1}$, der im Zustand $r_{n-1} = q_1$ endet.
Nach Behauptung 2 gilt also $n - 1 \geq 1$ und $a_{n-1} = 1$.

„$\Longleftarrow$“: Gelte $n \geq 2$ und $a_{n-1} = 1$ und $a_n = 1$.
Nach Behauptung 2 gibt es einen Lauf

$$(r_0, a_1, r_1, a_2, \dots, a_{n-1}, r_{n-1})$$

von $\mathcal{A}$ auf $a_1 \dots a_{n-1}$ mit $r_{n-1} = q_1$. Dann ist

$$(r_0, a_1, r_1, a_2, \dots, a_{n-1}, r_{n-1}, 1, q_2)$$

ein Lauf von $\mathcal{A}$ auf w. □

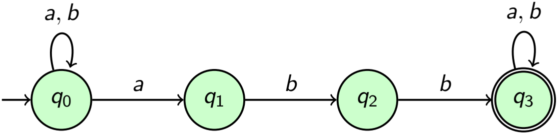
Weil q_2 der einzige akzeptierende Zustand von $\mathcal{A}$ ist impliziert Behauptung 3

$$L(\mathcal{A}) = \big\{ w \in \{0,1\}^* \;\big|\; 11 \text{ ist Suffix von } w \big\},$$

also die ursprüngliche Behauptung.

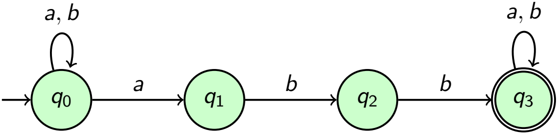
Beispiel 2.7

$\mathcal{A}$ sei folgender NFA:



Beispiel 2.7

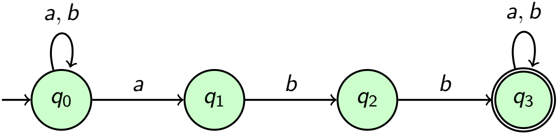
$\mathcal{A}$ sei folgender NFA:



Es gilt $L(\mathcal{A}) = \{w \in \{a, b\}^* \mid w$ enthält *abb* als Infix$\}$ (vgl. Bsp. 2.1)

Beispiel 2.7

$\mathcal{A}$ sei folgender NFA:

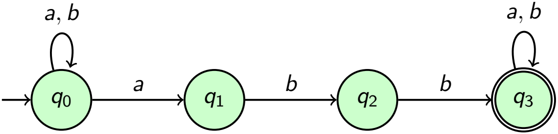


Es gilt $L(\mathcal{A}) = \{w \in \{a,b\}^* \mid w \text{ enthält } abb \text{ als Infix}\}$ (vgl. Bsp. 2.1)

Akzeptierender Lauf von $\mathcal{A}$ auf dem Wort *ababbab*:

$$\big(q_0, a, q_0, b, q_0, a, q_1, b, q_2, b, q_3, a, q_3, b, q_3\big).$$

$\mathcal{A}$ sei folgender NFA:



Es gilt $L(\mathcal{A}) = \{w \in \{a,b\}^* \mid w \text{ enthält } abb \text{ als Infix}\}$ (vgl. Bsp. 2.1)

Akzeptierender Lauf von $\mathcal{A}$ auf dem Wort *ababbab*:

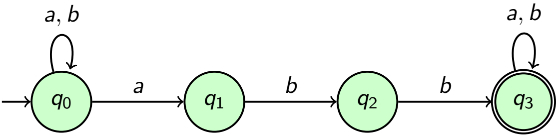
$$\big(q_0, a, q_0, b, q_0, a, q_1, b, q_2, b, q_3, a, q_3, b, q_3\big).$$

Läufe von $\mathcal{A}$ auf dem Wort $w = abab$:

$$\big(q_0, a, q_0, b, q_0, a, q_0, b, q_0\big),$$
$$\big(q_0, a, q_0, b, q_0, a, q_1, b, q_2\big).$$

Beispiel 2.7

$\mathcal{A}$ sei folgender NFA:



Es gilt $L(\mathcal{A}) = \{w \in \{a,b\}^* \mid w \text{ enthält } abb \text{ als Infix}\}$ (vgl. Bsp. 2.1)

Akzeptierender Lauf von $\mathcal{A}$ auf dem Wort *ababbab*:

$$\big(q_0, a, q_0, b, q_0, a, q_1, b, q_2, b, q_3, a, q_3, b, q_3\big).$$

Läufe von $\mathcal{A}$ auf dem Wort $w = abab$:

$$\big(q_0, a, q_0, b, q_0, a, q_0, b, q_0\big),$$
$$\big(q_0, a, q_0, b, q_0, a, q_1, b, q_2\big).$$

Außerdem gibt es noch folgenden Lauf auf dem Präfix ab von w, der sich nicht auf ganz w fortsetzen lässt:

$$\big(q_0, a, q_1, b, q_2\big).$$

Definition 2.8

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA, und seien $q, q' \in Q$ und $w = a_1 \ldots a_n \in \Sigma^*$.

Definition 2.8

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA, und seien $q, q' \in Q$ und $w = a_1 \ldots a_n \in \Sigma^*$.

Dann ist q' in $\mathcal{A}$ von q über w erreichbar (kurz: $\mathcal{A} : q \xrightarrow{w} q'$), wenn es Zustände $r_0, r_1, \ldots, r_n \in Q$ gibt, so dass

(i) $r_0 = q$ und $r_n = q'$ und

(ii) $(r_{i-1}, a_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

Definition 2.8

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA, und seien $q, q' \in Q$ und $w = a_1 \dots a_n \in \Sigma^*$.

Dann ist q' in $\mathcal{A}$ von q über w erreichbar (kurz: $\mathcal{A} : q \xrightarrow{w} q'$), wenn es Zustände $r_0, r_1, \dots, r_n \in Q$ gibt, so dass

(i) $r_0 = q$ und $r_n = q'$ und

(ii) $(r_{i-1}, a_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

Notation

- Wenn $\mathcal{A}$ aus dem Kontext klar ist, schreiben wir auch $q \xrightarrow{w} q'$ statt $\mathcal{A} : q \xrightarrow{w} q'$.

Definition 2.8

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA, und seien $q, q' \in Q$ und $w = a_1 \dots a_n \in \Sigma^*$.

Dann ist q' in $\mathcal{A}$ von q über w erreichbar (kurz: $\mathcal{A} : q \xrightarrow{w} q'$), wenn es Zustände $r_0, r_1, \dots, r_n \in Q$ gibt, so dass

(i) $r_0 = q$ und $r_n = q'$ und

(ii) $(r_{i-1}, a_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

Notation

- Wenn $\mathcal{A}$ aus dem Kontext klar ist, schreiben wir auch $q \xrightarrow{w} q'$ statt $\mathcal{A} : q \xrightarrow{w} q'$.
- Wir verwenden auch Schreibweisen wie $q \xrightarrow{w} q' \xrightarrow{w'} q''$ statt $\big(q \xrightarrow{w} q'$ und $q' \xrightarrow{w'} q''\big)$.

Die Erreichbarkeitsrelation

Definition 2.8

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA, und seien $q, q' \in Q$ und $w = a_1 \dots a_n \in \Sigma^*$.

Dann ist q' in $\mathcal{A}$ von q über w erreichbar (kurz: $\mathcal{A} : q \xrightarrow{w} q'$), wenn es Zustände $r_0, r_1, \dots, r_n \in Q$ gibt, so dass

(i) $r_0 = q$ und $r_n = q'$ und

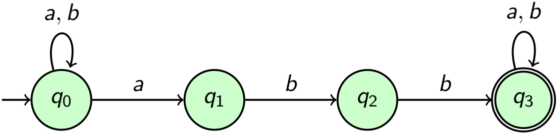
(ii) $(r_{i-1}, a_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

Notation

- Wenn $\mathcal{A}$ aus dem Kontext klar ist, schreiben wir auch $q \xrightarrow{w} q'$ statt $\mathcal{A} : q \xrightarrow{w} q'$.
- Wir verwenden auch Schreibweisen wie $q \xrightarrow{w} q' \xrightarrow{w'} q''$ statt $\big(q \xrightarrow{w} q'$ und $q' \xrightarrow{w'} q''\big)$.
- Wir verwenden die gleiche Notation für DFAs.

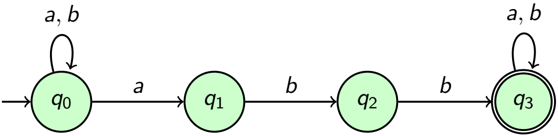
Beispiel 2.7 (Forts.)

$\mathcal{A}$ sei folgender NFA:



Beispiel 2.7 (Forts.)

$\mathcal{A}$ sei folgender NFA:



Es gilt:

$$q_0 \xrightarrow{ababba} q_3,$$

$$q_0 \xrightarrow{ababba} q_0,$$

$$q_0 \xrightarrow{ababba} q_1,$$

$$q_1 \xrightarrow{bbbb} q_3,$$

$$q_3 \xrightarrow{w} q_3 \qquad \text{für alle } w \in \{a, b\}^*.$$

Definition 2.9

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA und $w \in \Sigma^*$. Die Menge der in $\mathcal{A}$ über w erreichbaren Zustände ist

$$E(\mathcal{A}, w) := \{q \in Q \mid \mathcal{A} : q_0 \xrightarrow{w} q\}.$$

Definition 2.9

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA und $w \in \Sigma^*$. Die Menge der in $\mathcal{A}$ über w erreichbaren Zustände ist

$$E(\mathcal{A}, w) := \{q \in Q \mid \mathcal{A} : q_0 \xrightarrow{w} q\}.$$

Lemma 2.10

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA und $w \in \Sigma^*$.
Dann gilt

$$w \in L(\mathcal{A}) \iff E(\mathcal{A}, w) \cap F \neq \emptyset.$$

Definition 2.9

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA und $w \in \Sigma^*$. Die Menge der in $\mathcal{A}$ über w erreichbaren Zustände ist

$$E(\mathcal{A}, w) := \{q \in Q \mid \mathcal{A} : q_0 \xrightarrow{w} q\}.$$

Lemma 2.10

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA und $w \in \Sigma^*$.
Dann gilt

$$w \in L(\mathcal{A}) \iff E(\mathcal{A}, w) \cap F \neq \emptyset.$$

Lemma 2.11

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA.

1. $E(\mathcal{A}, \varepsilon) = \{q_0\}$.
2. *Für alle $w \in \Sigma^*$ und $a \in \Sigma$ gilt*

$$E(\mathcal{A}, wa) = \bigcup_{q \in E(\mathcal{A}, w)} \left\{q' \in Q \;\middle|\; (q, a, q') \in \Delta\right\}.$$

Lemma 2.10

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA und $w \in \Sigma^*$.
Dann gilt

$$w \in L(\mathcal{A}) \iff E(\mathcal{A}, w) \cap F \neq \emptyset.$$

Beweis.

$$\begin{aligned} w \in L(\mathcal{A}) &\iff \exists q \in F : \ q_0 \xrightarrow{w} q \\ &\iff \exists q \in F : \ q \in E(\mathcal{A}, w) \\ &\iff E(\mathcal{A}, w) \cap F \neq \emptyset. \end{aligned}$$

□

Lemma 2.11

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA.

1. $E(\mathcal{A}, \varepsilon) = \{q_0\}$.
2. *Für alle $w \in \Sigma^*$ und $a \in \Sigma$ gilt*

$$E(\mathcal{A}, wa) = \bigcup_{q \in E(\mathcal{A}, w)} \{q' \in Q \mid (q, a, q') \in \Delta\}.$$

Beweis.

1. Für alle $q \in Q$ gilt

$$q \in E(\mathcal{A}, \varepsilon) \iff q_0 \xrightarrow{\varepsilon} q \iff q = q_0.$$

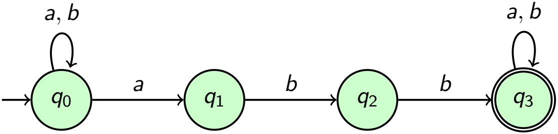
2. Seien $w \in \Sigma^*$ und $a \in \Sigma$. Dann gilt für alle $q \in Q$

$$\begin{aligned} q \in E(\mathcal{A}, wa) &\iff q_0 \xrightarrow{wa} q \\ &\iff \exists q' \in Q : \ q_0 \xrightarrow{w} q' \xrightarrow{a} q \\ &\iff \exists q' \in Q : \ q_0 \xrightarrow{w} q' \text{ und } (q', a, q) \in \Delta \\ &\iff \exists q' \in Q : \ q' \in E(\mathcal{A}, w) \text{ und } (q', a, q) \in \Delta \\ &\iff q \in \bigcup_{q' \in E(\mathcal{A}, w)} \{q'' \in Q \mid (q', a, q'') \in \Delta\}. \end{aligned}$$

□

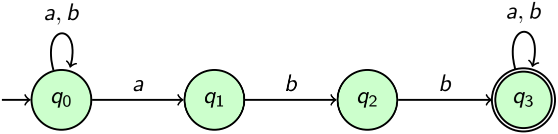
Beispiel 2.7 (Forts.)

$\mathcal{A}$ sei folgender NFA:



Beispiel 2.7 (Forts.)

$\mathcal{A}$ sei folgender NFA:



Die Erreichbarkeitsmengen auf den Präfixen von *ababba* sind:

$$
\begin{aligned}
E(\mathcal{A}, \varepsilon) &= \{q_0\},\\
E(\mathcal{A}, a) &= \{q_0, q_1\},\\
E(\mathcal{A}, ab) &= \{q_0, q_2\},\\
E(\mathcal{A}, aba) &= \{q_0, q_1\},\\
E(\mathcal{A}, abab) &= \{q_0, q_2\},\\
E(\mathcal{A}, ababb) &= \{q_0, q_3\},\\
E(\mathcal{A}, ababba) &= \{q_0, q_1, q_3\}.
\end{aligned}
$$

Algorithmus für die NFA-Akzeptanz

Eingabe: NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ und Wort $w = a_1 \ldots a_n \in \Sigma^*$.

Problem: Entscheide, ob $w \in L(\mathcal{A})$.

Algorithmus für die NFA-Akzeptanz

Eingabe: NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ und Wort $w = a_1 \dots a_n \in \Sigma^*$.

Problem: Entscheide, ob $w \in L(\mathcal{A})$.

Algorithmus:

1. Berechne iterativ

$$E(\mathcal{A}, \varepsilon), E(\mathcal{A}, a_1), E(\mathcal{A}, a_1 a_2), \dots, E(\mathcal{A}, w)$$

   unter Verwendung von Lemma 2.11.
2. Teste, ob $E(\mathcal{A}, w) \cap F \neq \emptyset$.

Algorithmus für die NFA-Akzeptanz

Eingabe: NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ und Wort $w = a_1 \dots a_n \in \Sigma^*$.

Problem: Entscheide, ob $w \in L(\mathcal{A})$.

Algorithmus:

1. Berechne iterativ

$$E(\mathcal{A}, \varepsilon), E(\mathcal{A}, a_1), E(\mathcal{A}, a_1 a_2), \dots, E(\mathcal{A}, w)$$

   unter Verwendung von Lemma 2.11.
2. Teste, ob $E(\mathcal{A}, w) \cap F \neq \emptyset$.

Analyse

Korrektheit: Folgt aus Lemma 2.10.

Algorithmus für die NFA-Akzeptanz

Eingabe: NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ und Wort $w = a_1 \dots a_n \in \Sigma^*$.

Problem: Entscheide, ob $w \in L(\mathcal{A})$.

Algorithmus:

1. Berechne iterativ

$$E(\mathcal{A}, \varepsilon), E(\mathcal{A}, a_1), E(\mathcal{A}, a_1 a_2), \dots, E(\mathcal{A}, w)$$

   unter Verwendung von Lemma 2.11.
2. Teste, ob $E(\mathcal{A}, w) \cap F \neq \emptyset$.

Analyse

Korrektheit: Folgt aus Lemma 2.10.

Laufzeit: Bei guter Implementierung unter Verwendung geeigneter Datenstrukturen $O(n|\Delta|)$

Algorithmus für die NFA-Akzeptanz

Eingabe: NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ und Wort $w = a_1 \dots a_n \in \Sigma^*$.

Problem: Entscheide, ob $w \in L(\mathcal{A})$.

Algorithmus:

1. Berechne iterativ
$$E(\mathcal{A}, \varepsilon), E(\mathcal{A}, a_1), E(\mathcal{A}, a_1 a_2), \dots, E(\mathcal{A}, w)$$
unter Verwendung von Lemma 2.11.
2. Teste, ob $E(\mathcal{A}, w) \cap F \neq \emptyset$.

Analyse

Korrektheit: Folgt aus Lemma 2.10.

Laufzeit: Bei guter Implementierung unter Verwendung geeigneter Datenstrukturen $O(n|\Delta|)$

(siehe dazu Vorlesung „Algorithmen & Datenstrukturen").

Abschnitt 2.2

Äquivalenz von DFAs und NFAs

Äquivalenz von Automaten

Definition 2.12

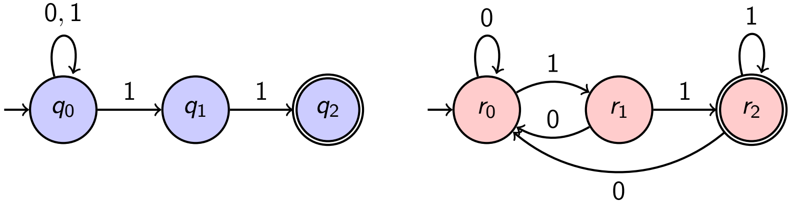
Zwei endliche Automaten (DFA oder NFA) heißen äquivalent, wenn sie die gleiche Sprache erkennen.

Definition 2.12

Zwei endliche Automaten (DFA oder NFA) heißen äquivalent, wenn sie die gleiche Sprache erkennen.

Beispiel 2.13

Folgender NFA und DFA sind äquivalent:

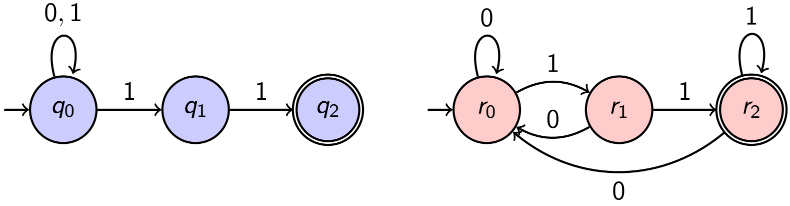


Definition 2.12

Zwei endliche Automaten (DFA oder NFA) heißen äquivalent, wenn sie die gleiche Sprache erkennen.

Beispiel 2.13

Folgender NFA und DFA sind äquivalent:



Beide erkennen die Sprache $\{w \in \{0,1\}^* \mid 11 \text{ ist Suffix von } w\}$.

Satz 2.14

Zu jedem DFA gibt es einen äquivalenten NFA.

Beweis von Satz 2.14

Sei $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ ein DFA.

Wir konstruieren einen zu $\mathcal{A}$ äquivalenten NFA $\mathcal{A}'$.

Dazu sei

$$\Delta := \left\{(q, a, q') \in Q \times \Sigma \times Q \;\middle|\; \delta(q, a) = q'\right\}.$$

Wir setzen $\mathcal{A}' := (Q, \Sigma, \Delta, q_0, F)$.

Behauptung

$\mathcal{A}$ und $\mathcal{A}'$ sind äquivalent.

Beweis.

Betrachte eine Folge

$$\rho := (r_0, a_1, r_1, a_2, \ldots, a_n, r_n)$$

mit $r_0, \ldots, r_n \in Q$ und $a_1, \ldots, a_n \in \Sigma$. Dann gilt für $1 \leq i \leq n$:

$$\delta(r_{i-1}, a_i) = r_i \iff (r_{i-1}, a_i, r_i) \in \Delta.$$

Weil $\mathcal{A}$ und $\mathcal{A}'$ den gleichen Anfangszustand haben, folgt:

$$\rho \text{ ist ein Lauf von } \mathcal{A} \iff \rho \text{ ist ein Lauf von } \mathcal{A}'.$$

Weil $\mathcal{A}$ und $\mathcal{A}'$ die gleichen Endzustände haben, folgt:

$$\rho \text{ ist ein akzeptierender Lauf von } \mathcal{A} \iff \rho \text{ ist ein akzeptierender Lauf von } \mathcal{A}'.$$

Also

$$\mathcal{A} \text{ akzeptiert } a_1 \ldots a_n \iff \mathcal{A}' \text{ akzeptiert } a_1 \ldots a_n.$$

Das impliziert $L(\mathcal{A}) = L(\mathcal{A}')$. $\square$

Von NFAs zu DFAs

Satz 2.15

Zu jedem NFA gibt es einen äquivalenten DFA.

Von NFAs zu DFAs

Satz 2.15

Zu jedem NFA gibt es einen äquivalenten DFA.

Korollar 2.16

Eine Sprache ist genau dann DFA-erkennbar, wenn sie NFA-erkennbar ist.

Erinnerung und Notation

Die Potenzmenge 2^M einer Menge M ist die Menge aller Teilmengen von M, also

$$2^M := \{N \mid N \subseteq M\}.$$

Erinnerung und Notation

Die Potenzmenge 2^M einer Menge M ist die Menge aller Teilmengen von M, also

$$2^M := \{N \mid N \subseteq M\}.$$

Definition 2.17

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA.

Der Potenzmengenautomat von $\mathcal{A}$ ist der DFA $\mathcal{A}' = (Q', \Sigma, \delta, q_0', F')$

Erinnerung und Notation

Die Potenzmenge 2^M einer Menge M ist die Menge aller Teilmengen von M, also

$$2^M := \{N \mid N \subseteq M\}.$$

Definition 2.17

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA.

Der Potenzmengenautomat von $\mathcal{A}$ ist der DFA $\mathcal{A}' = (Q', \Sigma, \delta, q_0', F')$ mit

- $Q' := 2^Q$,

Erinnerung und Notation

Die Potenzmenge 2^M einer Menge M ist die Menge aller Teilmengen von M, also

$$2^M := \{N \mid N \subseteq M\}.$$

Definition 2.17

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA.

Der Potenzmengenautomat von $\mathcal{A}$ ist der DFA $\mathcal{A}' = (Q', \Sigma, \delta, q'_0, F')$ mit

- $Q' := 2^Q$,
- $\delta(q', a) = \{q \in Q \mid \exists p \in q' : (p, a, q) \in \Delta\}$ für alle $q' \in Q', a \in \Sigma$,

2^Q, also eine Menge von Zuständen von $\mathcal{A}$

Die Potenzmengenkonstruktion

Erinnerung und Notation

Die Potenzmenge 2^M einer Menge M ist die Menge aller Teilmengen von M, also

$$2^M := \{N \mid N \subseteq M\}.$$

Definition 2.17

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA.

Der Potenzmengenautomat von $\mathcal{A}$ ist der DFA $\mathcal{A}' = (Q', \Sigma, \delta, q_0', F')$ mit

- $Q' := 2^Q$,
- $\delta(q', a) = \{q \in Q \mid \exists p \in q' : \ (p, a, q) \in \Delta\}$ für alle $q' \in Q', a \in \Sigma$,
- $q_0' := \{q_0\}$,

Die Potenzmengenkonstruktion

Erinnerung und Notation

Die Potenzmenge 2^M einer Menge M ist die Menge aller Teilmengen von M, also

$$2^M := \{N \mid N \subseteq M\}.$$

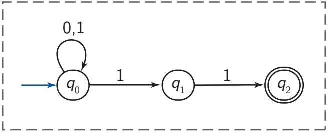
Definition 2.17

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA.

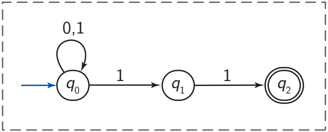
Der Potenzmengenautomat von $\mathcal{A}$ ist der DFA $\mathcal{A}' = (Q', \Sigma, \delta, q_0', F')$ mit

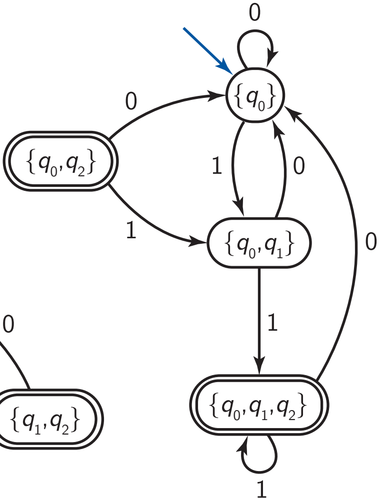
- $Q' := 2^Q$,
- $\delta(q', a) = \{q \in Q \mid \exists p \in q' : \ (p, a, q) \in \Delta\}$ für alle $q' \in Q', a \in \Sigma$,
- $q_0' := \{q_0\}$,
- $F' := \{q' \in Q' \mid q' \cap F \neq \emptyset\}$.

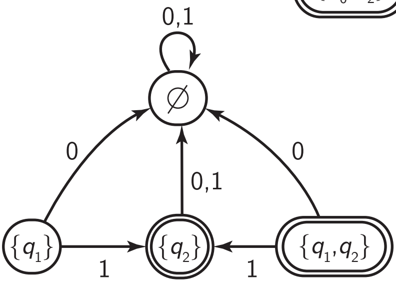
Beispiel 2.18



Beispiel 2.18







Äquivalenzlemma

Lemma 2.19

Jeder NFA ist äquivalent zu seinem Potenzmengenautomaten.

Äquivalenzlemma

Lemma 2.19

Jeder NFA ist äquivalent zu seinem Potenzmengenautomaten.

Das Lemma impliziert sofort Satz 2.15.

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein NFA und $\mathcal{A}' = (Q', \Sigma, \delta, q_0', F')$ sein Potenzmengenautomat.

Wir zeigen:

$$L(\mathcal{A}) = L(\mathcal{A}'). \tag{$\star$}$$

Behauptung 1

Sei $w \in \Sigma^*$ und $q' \in Q'$ mit $\mathcal{A}' : q_0' \xrightarrow{w} q'$. Dann gilt $q' = E(\mathcal{A}, w)$.

Beweis.

Induktion über $n := |w|$.

Induktionsanfang: $n = 0$.
Dann ist $w = \varepsilon$, und es gilt $q' = q_0' = \{q_0\} = E(\mathcal{A}, \varepsilon)$ nach Lemma 2.11.

Induktionsschritt: $n \to n+1$.
Dann ist $w = w'a$ für ein $w' \in \Sigma^*$ mit $|w'| = n$ und $a \in \Sigma$.

Sei $q'' \in Q'$, so dass $\mathcal{A}' : q_0' \xrightarrow{w'} q''$.

Induktionsannahme $\implies$ $q'' = E(\mathcal{A}, w')$.

Damit

$$\begin{aligned}
q' &= \delta(q'', a) \\
&= \{q \in Q \mid \exists p \in q'' : \ (p, a, q) \in \Delta\} && \text{(Definition von } \delta) \\
&= \bigcup_{p \in E(\mathcal{A}, w')} \{q \in Q \mid (p, a, q) \in \Delta\} && (q'' = E(\mathcal{A}, w')) \\
&= E(\mathcal{A}, w) && \text{(Lemma 2.11).}
\end{aligned}$$

$\square$

Beweis von $(\star)$.

Sei $w \in \Sigma^*$.
Sei $q' \in Q'$, so dass $\mathcal{A}' : q_0' \xrightarrow{w} q'$.
Dann gilt

$$\begin{aligned}
w \in L(\mathcal{A}) &\iff E(\mathcal{A}, w) \cap F \neq \emptyset && \text{(Lemma 2.10)} \\
&\iff q' \cap F \neq \emptyset && \text{(Behauptung 1)} \\
&\iff q' \in F' && \text{(Definition von } F') \\
&\iff \mathcal{A}' \text{ akzeptiert } w && (\mathcal{A}' : q_0' \xrightarrow{w} q') \\
&\iff w \in L(\mathcal{A}').
\end{aligned}$$

Definition 2.20

Sei $\mathcal{A} = \big(Q, \Sigma, \genfrac{}{}{0pt}{}{\delta}{\Delta}, q_0, F\big)$ ein $\genfrac{}{}{0pt}{}{\text{DFA}}{\text{NFA}}$.

1. Ein Zustand $q \in Q$ ist erreichbar, wenn es ein $w \in \Sigma^*$ gibt, so dass $\mathcal{A} : q_0 \xrightarrow{w} q$.

Definition 2.20

Sei $\mathcal{A} = \left(Q, \Sigma, \begin{matrix}\delta \\ \Delta\end{matrix}, q_0, F\right)$ ein $\begin{matrix}\text{DFA} \\ \text{NFA}\end{matrix}$.

1. Ein Zustand $q \in Q$ ist erreichbar, wenn es ein $w \in \Sigma^*$ gibt, so dass $\mathcal{A} : q_0 \xrightarrow{w} q$.
2. Die Reduktion von $\mathcal{A}$ auf die erreichbaren Zustände ist der $\begin{matrix}\text{DFA} \\ \text{NFA}\end{matrix}$

$$\mathcal{A}' = \left(Q', \Sigma, \begin{matrix}\delta' \\ \Delta'\end{matrix}, q_0, F'\right),$$

wobei

- $Q' := \{q \in Q \mid q \text{ erreichbar}\}$,
- $\delta' := \delta|_{Q' \times \Sigma}$ bzw. $\Delta' := \Delta \cap (Q' \times \Sigma \times Q')$,
- $F' := F \cap Q'$.

d.h. $\delta'(q, a) = \delta(q, a)$, aber nur für $q \in Q'$

Definition 2.20

Sei $\mathcal{A} = \left(Q, \Sigma, \genfrac{}{}{0pt}{}{\delta}{\Delta}, q_0, F\right)$ ein $\genfrac{}{}{0pt}{}{\text{DFA}}{\text{NFA}}$.

1. Ein Zustand $q \in Q$ ist erreichbar, wenn es ein $w \in \Sigma^*$ gibt, so dass $\mathcal{A} : q_0 \xrightarrow{w} q$.
2. Die Reduktion von $\mathcal{A}$ auf die erreichbaren Zustände ist der $\genfrac{}{}{0pt}{}{\text{DFA}}{\text{NFA}}$

$$\mathcal{A}' = \left(Q', \Sigma, \genfrac{}{}{0pt}{}{\delta'}{\Delta'}, q_0, F'\right),$$

   wobei
   - $Q' := \{q \in Q \mid q \text{ erreichbar}\}$,
   - $\delta' := \delta|_{Q' \times \Sigma}$ bzw. $\Delta' := \Delta \cap (Q' \times \Sigma \times Q')$,
   - $F' := F \cap Q'$.

Beobachtung 2.21

Die Reduktion eines DFA bzw. NFA auf die erreichbaren Zustände ist wohldefiniert. d.h. $q_0 \in Q'$ und $\delta(Q' \times \Sigma) \subseteq Q'$

Erreichbar Zustände (Forts.)

Beobachtung 2.22

Jeder DFA und NFA ist äquivalent zu seiner Reduktion auf die erreichbaren Zustände.

Erreichbar Zustände (Forts.)

Beobachtung 2.22

Jeder DFA und NFA ist äquivalent zu seiner Reduktion auf die erreichbaren Zustände.

Definition 2.23

1. Ein DFA oder NFA ist reduziert, wenn alle seine Zustände erreichbar sind.

Erreichbar Zustände (Forts.)

Beobachtung 2.22

Jeder DFA und NFA ist äquivalent zu seiner Reduktion auf die erreichbaren Zustände.

Definition 2.23

1. Ein DFA oder NFA ist reduziert, wenn alle seine Zustände erreichbar sind.
2. Sei $\mathcal{A}$ ein NFA. Der reduzierte Potenzmengenautomat von $\mathcal{A}$ ist die Reduktion des Potenzmengenautomaten von $\mathcal{A}$ auf die erreichbaren Zustände.

Erreichbar Zustände (Forts.)

Beobachtung 2.22

Jeder DFA und NFA ist äquivalent zu seiner Reduktion auf die erreichbaren Zustände.

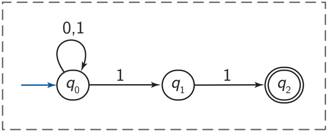
Definition 2.23

1. Ein DFA oder NFA ist reduziert, wenn alle seine Zustände erreichbar sind.
2. Sei $\mathcal{A}$ ein NFA. Der reduzierte Potenzmengenautomat von $\mathcal{A}$ ist die Reduktion des Potenzmengenautomaten von $\mathcal{A}$ auf die erreichbaren Zustände.

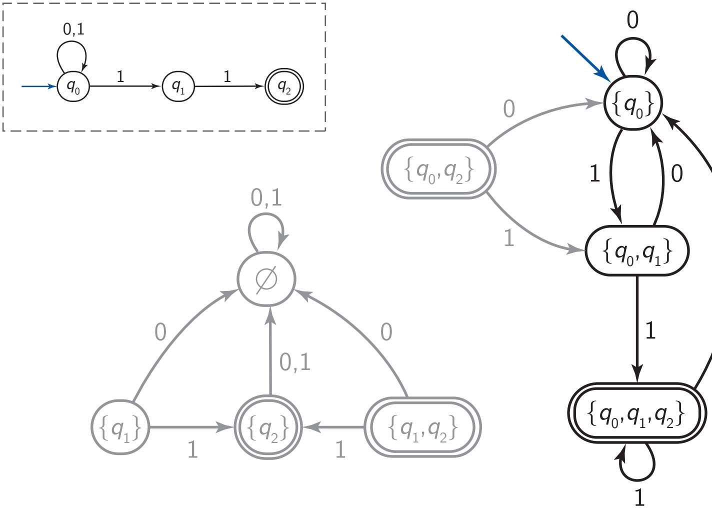
Beobachtung 2.24

Der reduzierte Potenzmengenautomat lässt sich ausgehend vom Anfangszustand schrittweise konstruieren, indem man jeweils die unmittelbar erreichbaren Zustände anfügt.

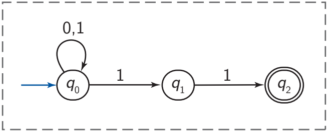
Beispiel 2.18 (Forts.)

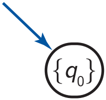


Beispiel 2.18 (Forts.)

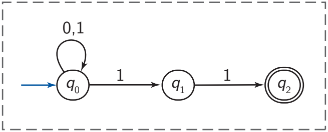


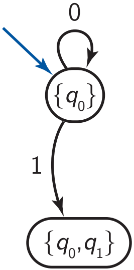
Beispiel 2.18 (Forts.)



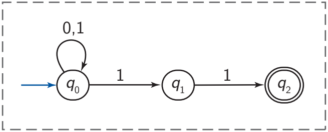


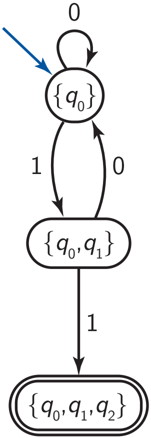
Beispiel 2.18 (Forts.)



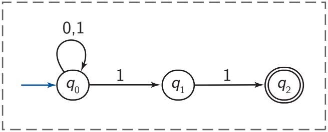


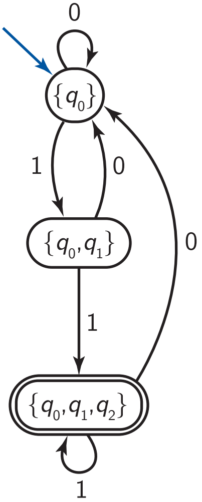
Beispiel 2.18 (Forts.)



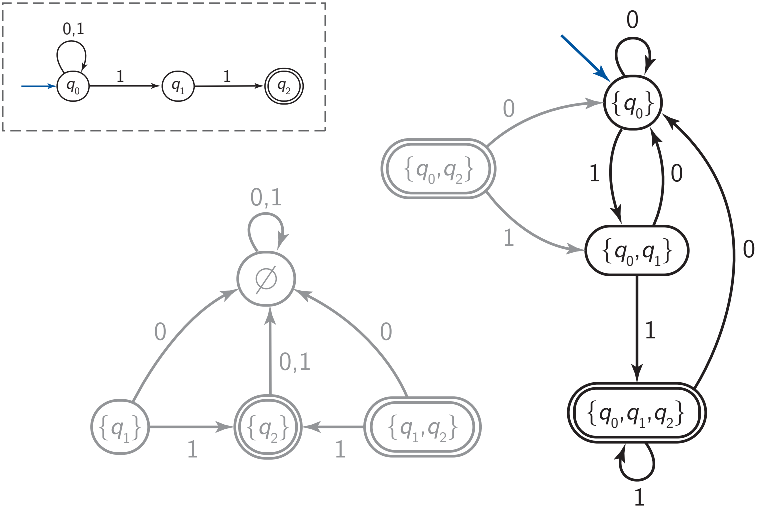


Beispiel 2.18 (Forts.)

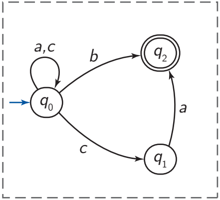




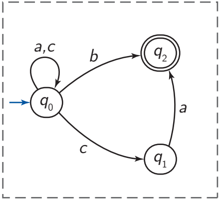
Beispiel 2.18 (Forts.)

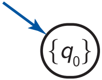


Beispiel 3.5

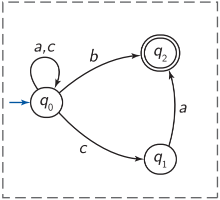


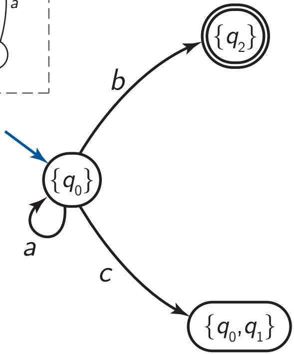
Beispiel 3.5



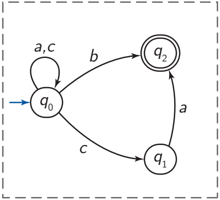


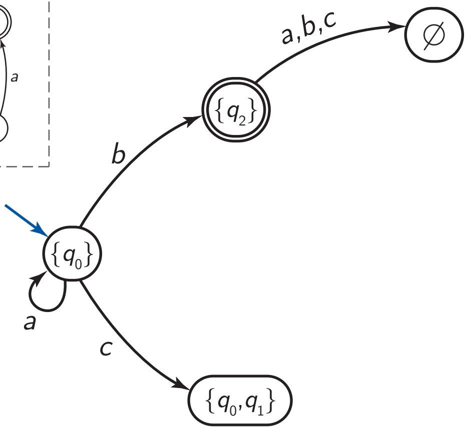
Beispiel 3.5



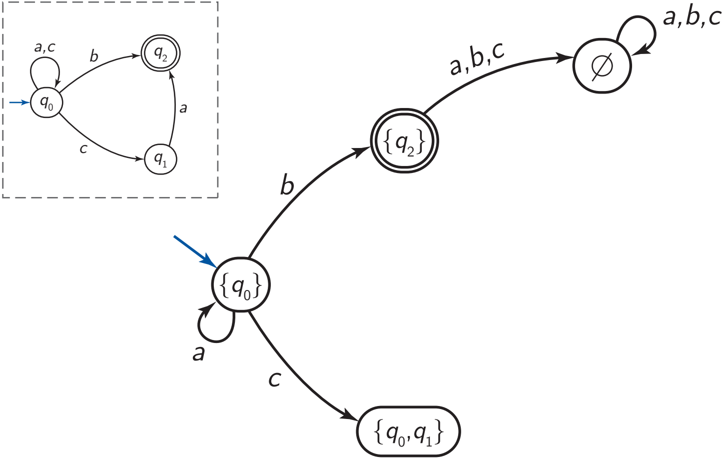


Beispiel 3.5

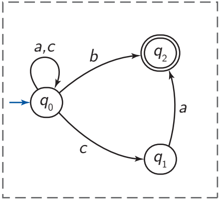


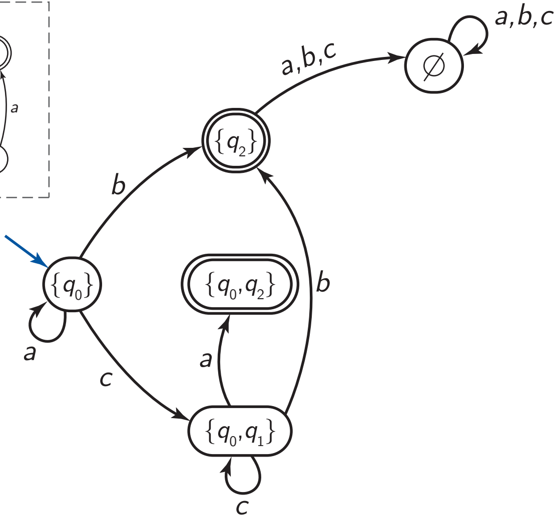


Beispiel 3.5

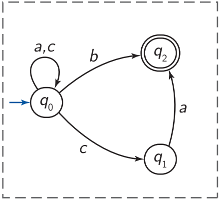


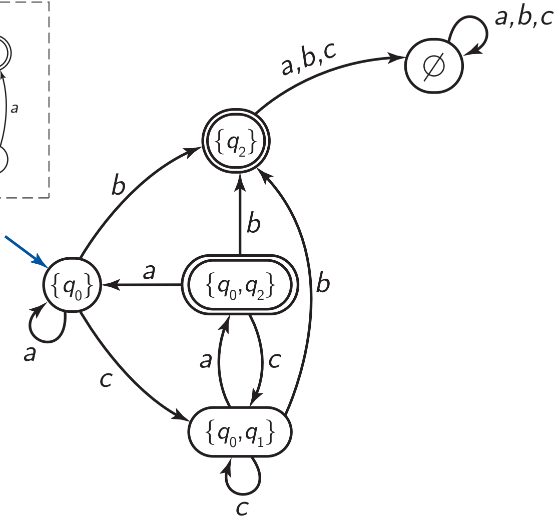
Beispiel 3.5



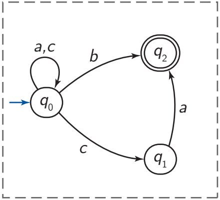


Beispiel 3.5

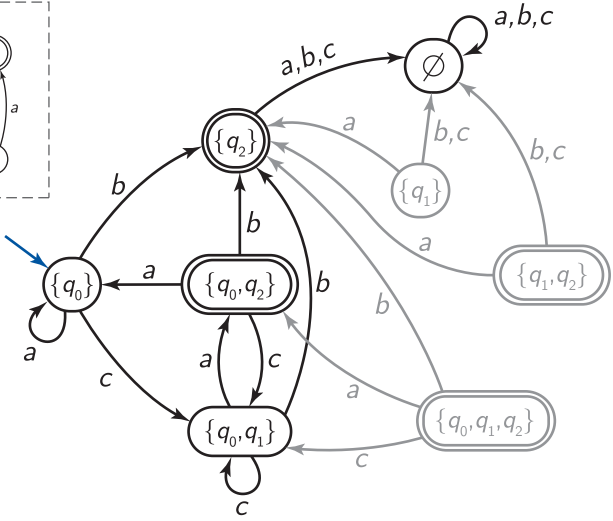




Beispiel 3.5



a,c
b
q_2
q_0
a
c
q_1



a,b,c
a,b,c
$\emptyset$
a
b,c
b,c
$\{q_2\}$
b
$\{q_1\}$
b
a
b
$\{q_1,q_2\}$
a
$\{q_0\}$
$\{q_0,q_2\}$
b
a
a
c
c
a
$\{q_0,q_1,q_2\}$
$\{q_0,q_1\}$
c
c

Zur Effizienz der Konstruktion

- Der Potenzmengenautomat eines NFA mit n Zuständen hat 2^n Zustände.

Zur Effizienz der Konstruktion

- Der Potenzmengenautomat eines NFA mit n Zuständen hat 2^n Zustände.
- Meist ist der reduzierte Potenzmengenautomat deutlich kleiner.

Zur Effizienz der Konstruktion

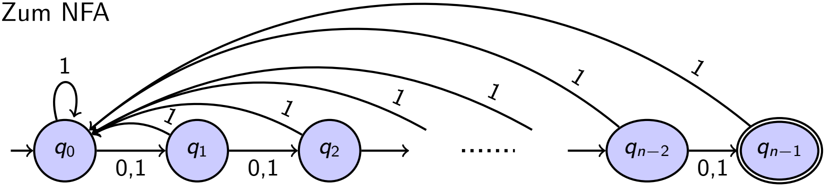
- Der Potenzmengenautomat eines NFA mit n Zuständen hat 2^n Zustände.
- Meist ist der reduzierte Potenzmengenautomat deutlich kleiner.
- Es gibt aber auch Beispiele von NFAs mit n Zuständen, für die jeder äquivalente DFA 2^n Zustände hat.

Zur Effizienz der Konstruktion

- Der Potenzmengenautomat eines NFA mit n Zuständen hat 2^n Zustände.
- Meist ist der reduzierte Potenzmengenautomat deutlich kleiner.
- Es gibt aber auch Beispiele von NFAs mit n Zuständen, für die jeder äquivalente DFA 2^n Zustände hat.

Beispiel 2.26

Zum NFA



mit n Zuständen gibt es keinen DFA mit weniger als 2^n Zuständen.

Beweis als (schwierige) Übung.

Abschnitt 2.3

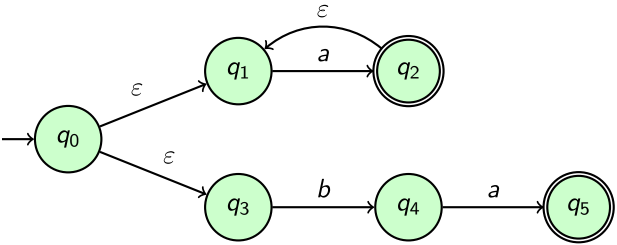
ε-Transitionen

Grundidee

Wir lassen in einem NFA auch Transitionen zu, die von keinem Eingabezeichen ausgelöst werden, sogenannte ε-Transitionen.

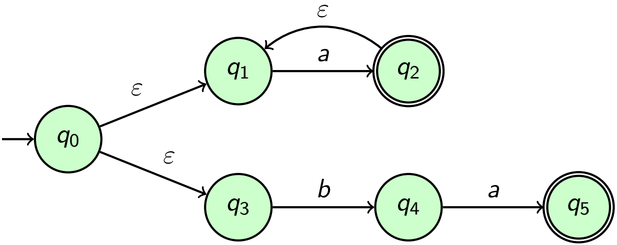
Wir lassen in einem NFA auch Transitionen zu, die von keinem Eingabezeichen ausgelöst werden, sogenannte ε-Transitionen.

Beispiel 2.27



Wir lassen in einem NFA auch Transitionen zu, die von keinem Eingabezeichen ausgelöst werden, sogenannte ε-Transitionen.

Beispiel 2.27



Erkennt die Sprache

$$\{\overbrace{a \dots a}^{n \text{ mal}} \mid n \geq 1\} \cup \{ba\}.$$

Definition 2.28

Ein nichtdeterministischer endlicher Automat mit ε-Transitionen (kurz: ε-NFA) ist ein 5-Tupel

$$(Q, \Sigma, \Delta, q_0, F);$$

dabei ist

$$\Delta \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times Q,$$

und Q, Σ, q_0, F sind wie bei NFAs.

Definition 2.29

Sei $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ ein ε-NFA.

1. Ein Lauf von $\mathcal{A}$ ist eine Folge $(r_0, \sigma_1, r_1, \sigma_2, \ldots, r_{n-1}, \sigma_n, r_n)$, für ein $n \geq 0$, wobei $r_0, \ldots, r_n \in Q$ und $\sigma_1, \ldots, \sigma_n \in \Sigma \cup \{\varepsilon\}$, so dass

   (i) $r_0 = q_0$,

   (ii) $(r_{i-1}, \sigma_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

Definition 2.29

Sei $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ ein ε-NFA.

1. Ein Lauf von $\mathcal{A}$ ist eine Folge $(r_0, \sigma_1, r_1, \sigma_2, \dots, r_{n-1}, \sigma_n, r_n)$, für ein $n \geq 0$, wobei $r_0, \dots, r_n \in Q$ und $\sigma_1, \dots, \sigma_n \in \Sigma \cup \{\varepsilon\}$, so dass

   (i) $r_0 = q_0$,

   (ii) $(r_{i-1}, \sigma_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

   Wir bezeichnen $(r_0, \sigma_1, r_1, \sigma_2, \dots, r_{n-1}, \sigma_n, r_n)$ oder einfach die Zustandsfolge $(r_0, r_1, \dots, r_n)$ auch als einen Lauf von $\mathcal{A}$ auf dem Wort $w \in \Sigma^*$, dass aus $\sigma_1 \dots \sigma_n$ durch Streichen aller εs entsteht.

Definition 2.29

Sei $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ ein ε-NFA.

1. Ein Lauf von $\mathcal{A}$ ist eine Folge $(r_0, \sigma_1, r_1, \sigma_2, \dots, r_{n-1}, \sigma_n, r_n)$, für ein $n \geq 0$, wobei $r_0, \dots, r_n \in Q$ und $\sigma_1, \dots, \sigma_n \in \Sigma \cup \{\varepsilon\}$, so dass

   (i) $r_0 = q_0$,

   (ii) $(r_{i-1}, \sigma_i, r_i) \in \Delta$ für $1 \leq i \leq n$.

   Wir bezeichnen $(r_0, \sigma_1, r_1, \sigma_2, \dots, r_{n-1}, \sigma_n, r_n)$ oder einfach die Zustandsfolge $(r_0, r_1, \dots, r_n)$ auch als einen Lauf von $\mathcal{A}$ auf dem Wort $w \in \Sigma^*$, dass aus $\sigma_1 \dots \sigma_n$ durch Streichen aller εs entsteht.
2. Der Lauf $(r_0, \sigma_1, r_1, \sigma_2, \dots, r_{n-1}, \sigma_n, r_n)$ ist akzeptierend, wenn $r_n \in F$.

Definition 2.30

1. Ein ε-NFA $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ akzeptiert ein Wort $w \in \Sigma^*$, wenn es einen akzeptierenden Lauf von $\mathcal{A}$ auf w gibt.

Definition 2.30

1. Ein ε-NFA $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ akzeptiert ein Wort $w \in \Sigma^*$, wenn es einen akzeptierenden Lauf von $\mathcal{A}$ auf w gibt.
2. Die von einem ε-NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ erkannte Sprache ist

$$L(\mathcal{A}) := \{w \in \Sigma^* \mid \mathcal{A} \text{ akzeptiert } w\}.$$

Semantik von ε-NFAs

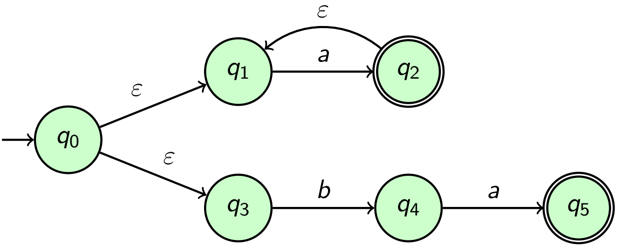
Definition 2.30

1. Ein ε-NFA $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ akzeptiert ein Wort $w \in \Sigma^*$, wenn es einen akzeptierenden Lauf von $\mathcal{A}$ auf w gibt.
2. Die von einem ε-NFA $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ erkannte Sprache ist

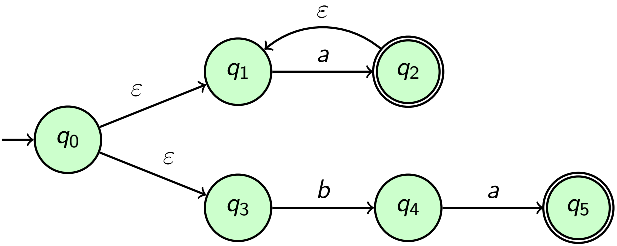
$$L(\mathcal{A}) := \{w \in \Sigma^* \mid \mathcal{A} \text{ akzeptiert } w\}.$$

3. Eine Sprache L heißt ε-NFA-erkennbar, wenn es einen ε-NFA $\mathcal{A}$ gibt, so dass $L = L(\mathcal{A})$.

Sei $\mathcal{A}$ folgender ε-NFA:



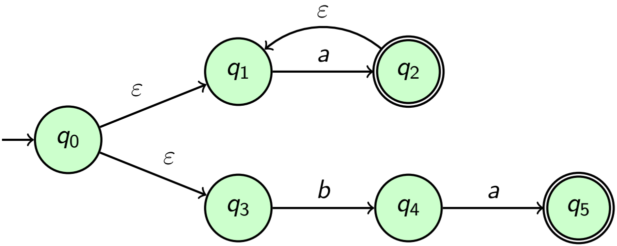
Sei $\mathcal{A}$ folgender ε-NFA:



Dann ist

$$L(\mathcal{A}) = \{\overbrace{a \dots a}^{n \text{ mal}} \mid n \geq 1\} \cup \{ba\}.$$

Sei $\mathcal{A}$ folgender ε-NFA:



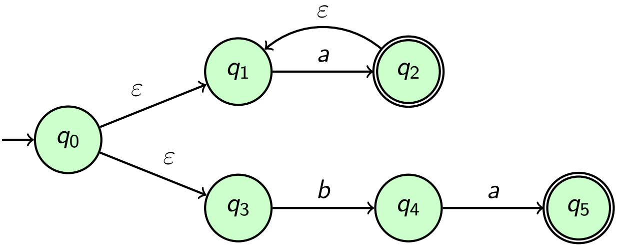
Dann ist

$$L(\mathcal{A}) = \{\overbrace{a\ldots a}^{n \text{ mal}} \mid n \geq 1\} \cup \{ba\}.$$

Akzeptierender Lauf auf dem Wort *aaa*:

$$\big(q_0, \varepsilon, q_1, a, q_2, \varepsilon, q_1, a, q_2, \varepsilon, q_1, a, q_2\big).$$

Sei $\mathcal{A}$ folgender ε-NFA:



Dann ist

$$L(\mathcal{A}) = \{\overbrace{a \dots a}^{n \text{ mal}} \mid n \geq 1\} \cup \{ba\}.$$

Akzeptierender Lauf auf dem Wort *aaa*:

$$\big(q_0, \varepsilon, q_1, a, q_2, \varepsilon, q_1, a, q_2, \varepsilon, q_1, a, q_2\big).$$

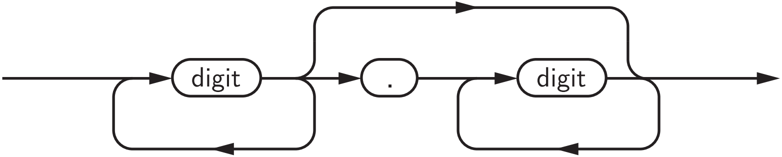
Akzeptierender Lauf auf dem Wort *ba*:

$$\big(q_0, \varepsilon, q_3, b, q_4, a, q_5\big).$$

Ein Syntaxdiagramm (ohne rekursive Aufrufe) ist direkt als ε-NFA lesbar.

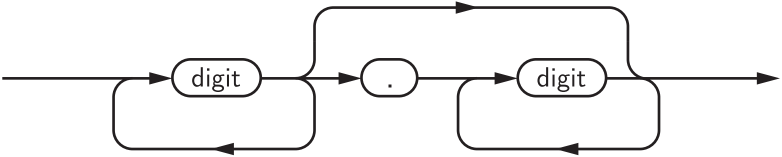
Ein Syntaxdiagramm (ohne rekursive Aufrufe) ist direkt als ε-NFA lesbar.

Beispiel 2.31

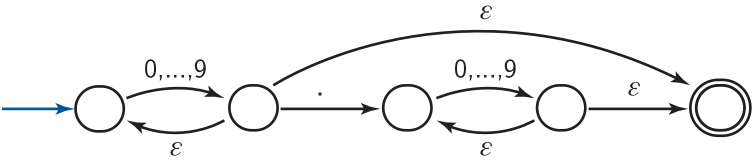


Ein Syntaxdiagramm (ohne rekursive Aufrufe) ist direkt als ε-NFA lesbar.

Beispiel 2.31



Entsprechender ε-NFA:



Beobachtung 2.32

Jeder NFA $\mathcal{A}$ ist auch ein ε-NFA.

Beobachtung 2.32

Jeder NFA $\mathcal{A}$ ist auch ein ε-NFA.

Außerdem hat $\mathcal{A}$ als NFA dieselben (akzeptierenden) Läufe wie als ε-NFA und akzeptiert demzufolge dieselbe Sprache als NFA wie als ε-NFA.

Beobachtung 2.32

Jeder NFA $\mathcal{A}$ ist auch ein ε-NFA.

Außerdem hat $\mathcal{A}$ als NFA dieselben (akzeptierenden) Läufe wie als ε-NFA und akzeptiert demzufolge dieselbe Sprache als NFA wie als ε-NFA.

(Wir sagen, das Modell des ε-NFAs ist eine konservative Erweiterung des Modells des NFAs.)

Beobachtung 2.32

Jeder NFA $\mathcal{A}$ ist auch ein ε-NFA.

Außerdem hat $\mathcal{A}$ als NFA dieselben (akzeptierenden) Läufe wie als ε-NFA und akzeptiert demzufolge dieselbe Sprache als NFA wie als ε-NFA.

(Wir sagen, das Modell des ε-NFAs ist eine konservative Erweiterung des Modells des NFAs.)

Jede NFA erkennbare Spreche ist also auch ε-NFA erkennbar.

Satz 2.33

Zu jeden ε-NFA gibt es einen äquivalenten NFA.

Satz 2.33

Zu jeden ε-NFA gibt es einen äquivalenten NFA.

Korollar 2.34

Für jede Sprache L sind folgende Aussagen äquivalent:

(i) L ist DFA-erkennbar.

(ii) L ist NFA-erkennbar.

(iii) L ist ε-NFA-erkennbar.

Definition 2.35

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA, und seien $q, q' \in Q$ und $w = a_1 \dots a_n \in \Sigma^*$.

Definition 2.35

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA, und seien $q, q' \in Q$ und $w = a_1 \dots a_n \in \Sigma^*$.

Dann ist q' in $\mathcal{A}$ von q über w erreichbar (kurz: $\mathcal{A} : q \xrightarrow{w} q'$), wenn es ein $m \geq n$, Indizes $1 \leq i_1 < i_2 < \dots < i_n \leq m$ und Zustände $r_0, r_1, \dots, r_m \in Q$ gibt, so dass

(i) $r_0 = q$ und $r_m = q'$,

(ii) $(r_{i_j - 1}, a_{i_j}, r_{i_j}) \in \Delta$ für $1 \leq j \leq n$,

(iii) $(r_{i-1}, \varepsilon, r_i) \in \Delta$ für alle $i \in \{1, \dots, m\} \setminus \{i_1, \dots, i_n\}$.

Definition 2.35

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA, und seien $q, q' \in Q$ und $w = a_1 \dots a_n \in \Sigma^*$.

Dann ist q' in $\mathcal{A}$ von q über w erreichbar (kurz: $\mathcal{A} : q \xrightarrow{w} q'$), wenn es ein $m \geq n$, Indizes $1 \leq i_1 < i_2 < \dots < i_n \leq m$ und Zustände $r_0, r_1, \dots, r_m \in Q$ gibt, so dass

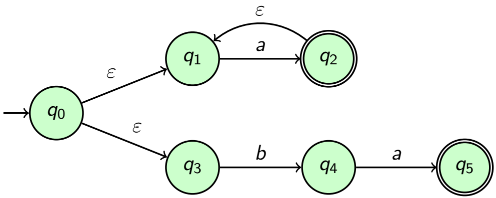
(i) $r_0 = q$ und $r_m = q'$,

(ii) $(r_{i_j - 1}, a_{i_j}, r_{i_j}) \in \Delta$ für $1 \leq j \leq n$,

(iii) $(r_{i-1}, \varepsilon, r_i) \in \Delta$ für alle $i \in \{1, \dots, m\} \setminus \{i_1, \dots, i_n\}$.

Beispiel 2.27 (Forts.)

Betrachte wieder den ε-NFA



Definition 2.35

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA, und seien $q, q' \in Q$ und $w = a_1 \dots a_n \in \Sigma^*$.

Dann ist q' in $\mathcal{A}$ von q über w erreichbar (kurz: $\mathcal{A} : q \xrightarrow{w} q'$), wenn es ein $m \geq n$, Indizes $1 \leq i_1 < i_2 < \dots < i_n \leq m$ und Zustände $r_0, r_1, \dots, r_m \in Q$ gibt, so dass

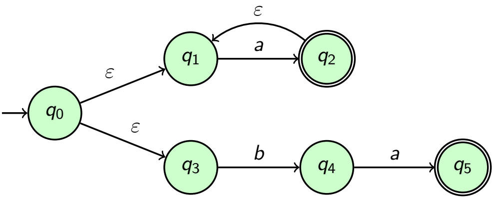
(i) $r_0 = q$ und $r_m = q'$,

(ii) $(r_{i_j - 1}, a_{i_j}, r_{i_j}) \in \Delta$ für $1 \leq j \leq n$,

(iii) $(r_{i-1}, \varepsilon, r_i) \in \Delta$ für alle $i \in \{1, \dots, m\} \setminus \{i_1, \dots, i_n\}$.

Beispiel 2.27 (Forts.)

Betrachte wieder den ε-NFA



Es gilt $q_0 \xrightarrow{\varepsilon} q_1$,
$q_0 \xrightarrow{a} q_1$,
$q_2 \xrightarrow{aa} q_2$.

Eigenschaften der Erreichbarkeitsrelation

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA.

- Wie bei NFAs gilt

$$\mathcal{A} \text{ akzeptiert } w \iff \mathcal{A} : q_0 \xrightarrow{w} q \text{ für ein } q \in F.$$

Eigenschaften der Erreichbarkeitsrelation

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA.

- Wie bei NFAs gilt

$$\mathcal{A} \text{ akzeptiert } w \iff \mathcal{A} : q_0 \xrightarrow{w} q \text{ für ein } q \in F.$$

- Anders als bei NFAs bedeutet $q \xrightarrow{\varepsilon} q'$ nicht $q = q'$, sondern dass $r_0 = q, r_1, \ldots, r_m = q'$ existieren mit $(r_{i-1}, \varepsilon, r_i) \in \Delta$ für $1 \leq i \leq m$.

Eigenschaften der Erreichbarkeitsrelation

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA.

- Wie bei NFAs gilt

$$\mathcal{A} \text{ akzeptiert } w \iff \mathcal{A} : q_0 \xrightarrow{w} q \text{ für ein } q \in F.$$

- Anders als bei NFAs bedeutet $q \xrightarrow{\varepsilon} q'$ nicht $q = q'$, sondern dass $r_0 = q, r_1, \ldots, r_m = q'$ existieren mit $(r_{i-1}, \varepsilon, r_i) \in \Delta$ für $1 \leq i \leq m$.

- Ebenfalls anders als bei NFAs bedeutet $q \xrightarrow{a} q'$ für $a \in \Sigma$ nicht $(q, a, q') \in \Delta$, sondern dass $r, r' \in Q$ existieren mit

$$q \xrightarrow{\varepsilon} r \text{ und } (r, a, r') \in \Delta \text{ und } r' \xrightarrow{\varepsilon} q'.$$

Eigenschaften der Erreichbarkeitsrelation

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA.

- Wie bei NFAs gilt

$$\mathcal{A} \text{ akzeptiert } w \iff \mathcal{A} : q_0 \xrightarrow{w} q \text{ für ein } q \in F.$$

- Anders als bei NFAs bedeutet $q \xrightarrow{\varepsilon} q'$ nicht $q = q'$, sondern dass $r_0 = q, r_1, \ldots, r_m = q'$ existieren mit $(r_{i-1}, \varepsilon, r_i) \in \Delta$ für $1 \leq i \leq m$.

- Ebenfalls anders als bei NFAs bedeutet $q \xrightarrow{a} q'$ für $a \in \Sigma$ nicht $(q, a, q') \in \Delta$, sondern dass $r, r' \in Q$ existieren mit

$$q \xrightarrow{\varepsilon} r \text{ und } (r, a, r') \in \Delta \text{ und } r' \xrightarrow{\varepsilon} q'.$$

- Wie bei NFAs gilt aber für alle Wörter $w = a_1 \ldots a_n$ und Zustände $q, q' \in Q$:

$$q \xrightarrow{w} q' \iff \exists r_1, \ldots, r_{n-1} \in Q : \; q \xrightarrow{a_1} r_1 \xrightarrow{a_2} r_2 \xrightarrow{a_3} \cdots \xrightarrow{a_{n-1}} r_{n-1} \xrightarrow{a_n} q'.$$

Satz 2.33
Zu jedem ε-NFA gibt es einen äquivalenten NFA.

Beweis.
Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA.

Wir definieren einen NFA $\mathcal{A}' = (Q, \Sigma, \Delta', q_0, F')$ mit

- $\Delta' := \left\{ (q, a, q') \in Q \times \Sigma \times Q \;\middle|\; \mathcal{A} : q \xrightarrow{a} q' \right\}$,
- $F' := \{ q \in Q \mid \exists q' \in F : \ \mathcal{A} : q \xrightarrow{\varepsilon} q' \}$.

Wir zeigen:

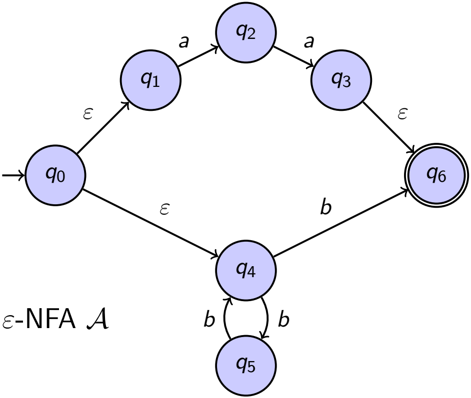
$$L(\mathcal{A}) = L(\mathcal{A}'). \qquad (\star)$$

Sei dazu $w = a_1 \dots a_n \in \Sigma^*$ (möglicherweise $n = 0$ und $w = \varepsilon$).
Dann gilt

$$\begin{aligned}
w \in L(\mathcal{A}) &\iff \exists q \in F : \ \mathcal{A} : q_0 \xrightarrow{w} q \\
&\iff \exists q \in F, r_0, \dots, r_n \in Q : \ \mathcal{A} : q_0 = r_0 \xrightarrow{a_1} r_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} r_n \xrightarrow{\varepsilon} q \\
&\iff \exists q \in F, r_0, \dots, r_n \in Q : \ \mathcal{A}' : q_0 = r_0 \xrightarrow{a_1} r_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} r_n \\
&\qquad\qquad \text{und } \mathcal{A} : r_n \xrightarrow{\varepsilon} q \\
&\iff \exists q' \in F' : \ \mathcal{A}' : q_0 \xrightarrow{w} q' \\
&\iff w \in L(\mathcal{A}')
\end{aligned}$$

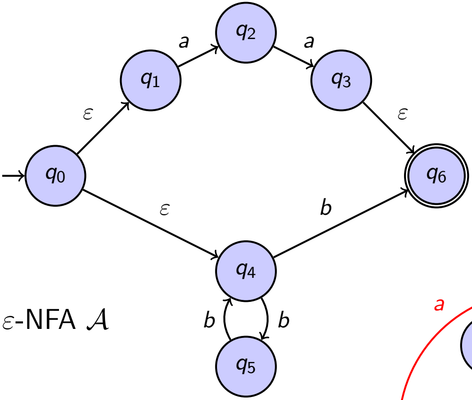
Damit ist $(\star)$ bewiesen, also sind $\mathcal{A}$ und $\mathcal{A}'$ äquivalent. $\square$

Beispiel 2.36

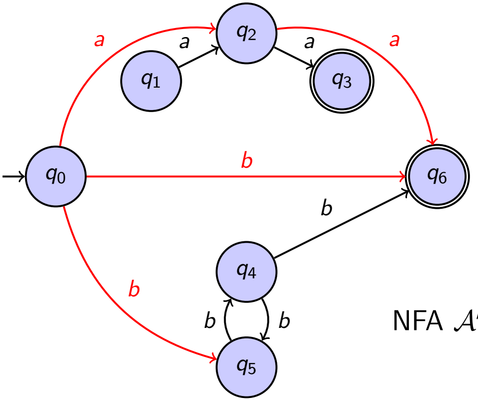


ε-NFA $\mathcal{A}$

Beispiel 2.36



ε-NFA $\mathcal{A}$



NFA $\mathcal{A}'$

Abschnitt 2.4

Abschlusseigenschaften FA-erkennbarer Sprachen

- Wir wissen, dass die selben Sprachen DFA-, NFA-, ε-NFA-erkennbar sind.

FA-erkennbare Sprachen

- Wir wissen, dass die selben Sprachen DFA-, NFA-, ε-NFA-erkennbar sind.

  Der Einfachheit halber nennen wir diese von nun an FA-erkennbare Sprachen.

FA-erkennbare Sprachen

- Wir wissen, dass die selben Sprachen DFA-, NFA-, ε-NFA-erkennbar sind.

  Der Einfachheit halber nennen wir diese von nun an FA-erkennbare Sprachen.

- Wir wissen bereits, dass die Klasse der FA-erkennbaren Sprachen unter den Mengenoperationen Komplement, Durchschnitt, Vereinigung abgeschlossen sind.

FA-erkennbare Sprachen

- Wir wissen, dass die selben Sprachen DFA-, NFA-, ε-NFA-erkennbar sind.

  Der Einfachheit halber nennen wir diese von nun an FA-erkennbare Sprachen.

- Wir wissen bereits, dass die Klasse der FA-erkennbaren Sprachen unter den Mengenoperationen Komplement, Durchschnitt, Vereinigung abgeschlossen sind.

  Wir werden aber noch einmal neue Konstruktionen dafür kennenlernen.

FA-erkennbare Sprachen

- Wir wissen, dass die selben Sprachen DFA-, NFA-, ε-NFA-erkennbar sind.

  Der Einfachheit halber nennen wir diese von nun an FA-erkennbare Sprachen.

- Wir wissen bereits, dass die Klasse der FA-erkennbaren Sprachen unter den Mengenoperationen Komplement, Durchschnitt, Vereinigung abgeschlossen sind.

  Wir werden aber noch einmal neue Konstruktionen dafür kennenlernen.

- Anschließend werden wir zeigen, dass die FA-erkennbaren Sprachen unter Verkettung und Iteration abgeschlossen sind.

Vereinigung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Vereinigung

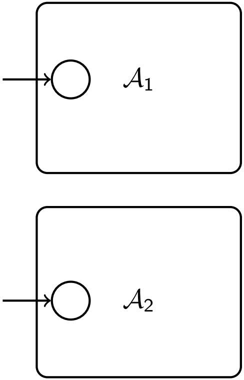
Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$ erkennt.

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$ erkennt.

Graphisch

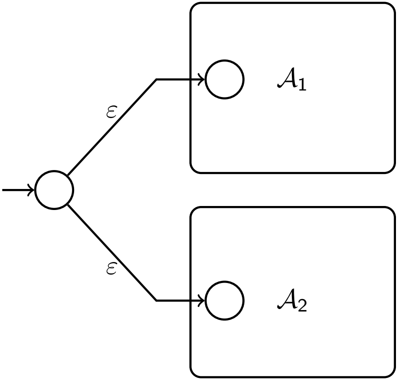


Vereinigung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$ erkennt.

Graphisch

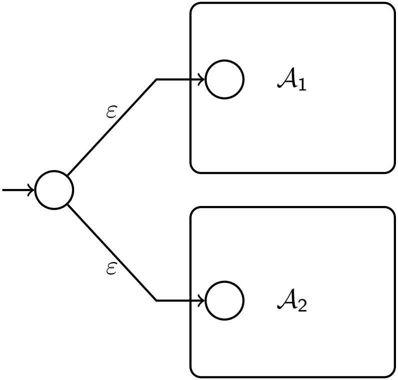


Vereinigung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$ erkennt.

Graphisch



Formal

Wir nehmen an: $Q_1 \cap Q_2 = \emptyset$.

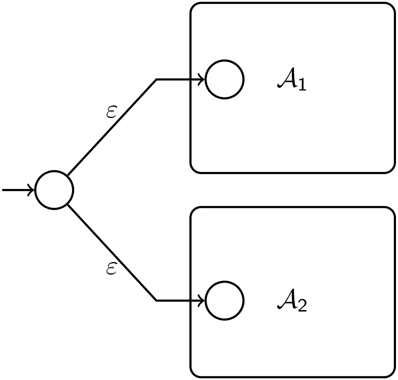
Sei $q_0 \notin Q_1 \cup Q_2$ ein neuer Zustand.

Vereinigung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$ erkennt.

Graphisch



Formal

Wir nehmen an: $Q_1 \cap Q_2 = \emptyset$.

Sei $q_0 \notin Q_1 \cup Q_2$ ein neuer Zustand.

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ mit

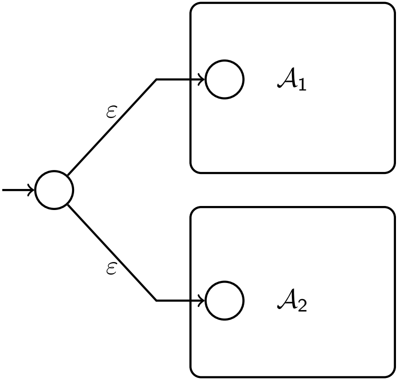
- $Q := Q_1 \cup Q_2 \cup \{q_0\}$;
- $\Delta := \Delta_1 \cup \Delta_2 \cup \{(q_0, \varepsilon, q_{10}), (q_0, \varepsilon, q_{20})\}$;
- $F := F_1 \cup F_2$.

Vereinigung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$ erkennt.

Graphisch



Formal

Wir nehmen an: $Q_1 \cap Q_2 = \emptyset$.

Sei $q_0 \notin Q_1 \cup Q_2$ ein neuer Zustand.

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ mit

- $Q := Q_1 \cup Q_2 \cup \{q_0\}$;
- $\Delta := \Delta_1 \cup \Delta_2 \cup \{(q_0, \varepsilon, q_{10}), (q_0, \varepsilon, q_{20})\}$;
- $F := F_1 \cup F_2$.

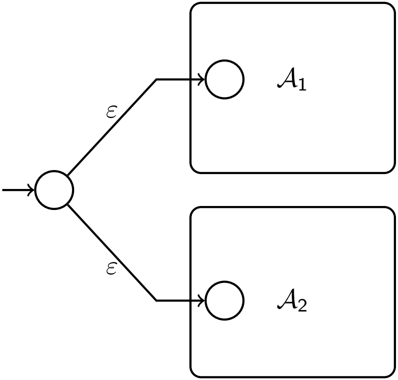
Behauptung

$L(\mathcal{A}) = L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$.

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$ erkennt.

Graphisch



Formal

Wir nehmen an: $Q_1 \cap Q_2 = \emptyset$.

Sei $q_0 \notin Q_1 \cup Q_2$ ein neuer Zustand.

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ mit

- $Q := Q_1 \cup Q_2 \cup \{q_0\}$;
- $\Delta := \Delta_1 \cup \Delta_2 \cup \{(q_0, \varepsilon, q_{10}), (q_0, \varepsilon, q_{20})\}$;
- $F := F_1 \cup F_2$.

Behauptung

$L(\mathcal{A}) = L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$.

Ohne Beweis (ist aber einfach).

Komplement

Sei $\mathcal{A}$ ein ε-NFA.

Konstruktion eines Automaten für $\overline{L(\mathcal{A})}$

1. Schritt: Konstruiere zu $\mathcal{A}$ äquivalenten NFA $\mathcal{A}'$ (vgl. Satz 2.33)
2. Schritt: Konstruiere zu $\mathcal{A}'$ äquivalenten DFA $\mathcal{A}''$ mit der Potenzmengenkonstruktion.
3. Schritt: Konstruiere DFA $\mathcal{A}'''$ mit $L(\mathcal{A}''') = \overline{L(\mathcal{A}'')}$ (vgl. Satz 1.29).

Bemerkung 2.37

Die Konstruktion ist im “worst-case” sehr ineffizient, weil sie die Umwandlung eines NFA in einen DFA erfordert.

Es kann deswegen passieren, dass $\mathcal{A}$ n Zustände hat und $\mathcal{A}'''$ 2^n Zustände.

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir wollen einen ε-NFA konstruieren, der die Sprache $L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$ erkennt.

Durchschnitt

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir wollen einen ε-NFA konstruieren, der die Sprache $L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$ erkennt.

1. Ansatz

Verwendung von Komplement und Vereinigung mittels de Morgan'scher Regel.

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir wollen einen ε-NFA konstruieren, der die Sprache $L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$ erkennt.

1. Ansatz

Verwendung von Komplement und Vereinigung mittels de Morgan'scher Regel.

Ist aber ineffzient, weil es Komplementierungen und damit die Umwandlung von NFAs in DFAs mittels Potenzmengenkonstruktion erfordert.

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir wollen einen ε-NFA konstruieren, der die Sprache $L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$ erkennt.

1. Ansatz

Verwendung von Komplement und Vereinigung mittels de Morgan'scher Regel.

Ist aber ineffzient, weil es Komplementierungen und damit die Umwandlung von NFAs in DFAs mittels Potenzmengenkonstruktion erfordert.

2.Ansatz

Verallgemeinerung der Produktkonstruktion auf NFAs (ist problemlos möglich).

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Verkettung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

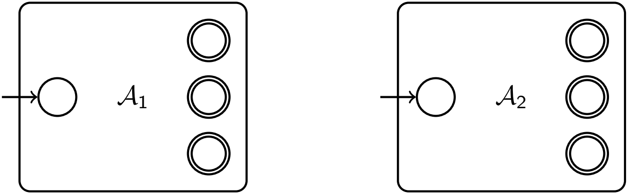
Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1)L(\mathcal{A}_2)$ erkennt.

Verkettung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1)L(\mathcal{A}_2)$ erkennt.

Graphisch

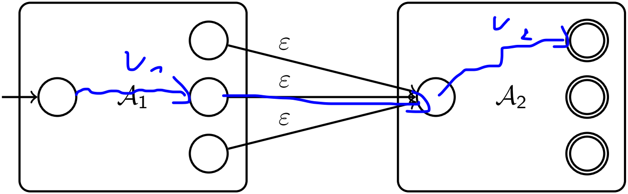


Verkettung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1)L(\mathcal{A}_2)$ erkennt.

Graphisch

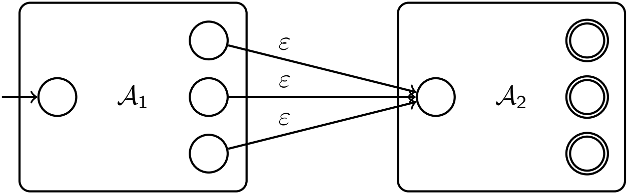


$$w = v_1 \; v_2$$
$$ \cap \quad \cap$$
$$ L(A_1) \; L(A_2)$$

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1)L(\mathcal{A}_2)$ erkennt.

Graphisch



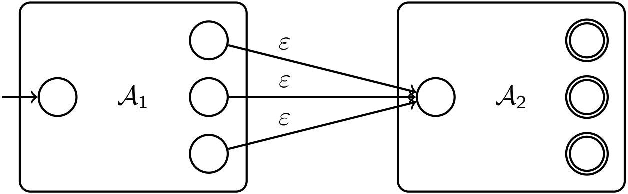
Formal

Wir nehmen an: $Q_1 \cap Q_2 = \emptyset$.

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1)L(\mathcal{A}_2)$ erkennt.

Graphisch



Formal

Wir nehmen an: $Q_1 \cap Q_2 = \emptyset$.

Sei $\mathcal{A} = (Q_1 \cup Q_2, \Sigma, \Delta, q_{10}, F_2)$ mit

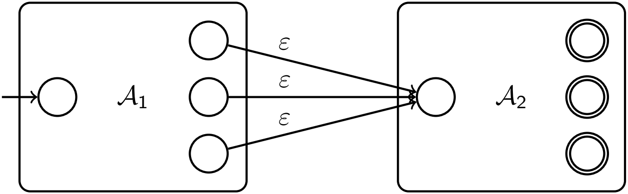
$$\Delta := \Delta_1 \cup \Delta_2 \cup \{(q, \varepsilon, q_{20}) \mid q \in F_1\}.$$

Verkettung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1)L(\mathcal{A}_2)$ erkennt.

Graphisch



Formal

Wir nehmen an: $Q_1 \cap Q_2 = \emptyset$.

Sei $\mathcal{A} = (Q_1 \cup Q_2, \Sigma, \Delta, q_{10}, F_2)$ mit

$$\Delta := \Delta_1 \cup \Delta_2 \cup \{(q, \varepsilon, q_{20}) \mid q \in F_1\}.$$

Behauptung

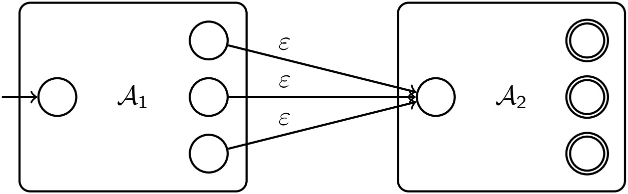
$L(\mathcal{A}) = L(\mathcal{A}_1)L(\mathcal{A}_2)$.

Verkettung

Seien $\mathcal{A}_1 = (Q_1, \Sigma, \Delta_1, q_{10}, F_1)$ und $\mathcal{A}_2 = (Q_2, \Sigma, \Delta_2, q_{20}, F_2)$ ε-NFAs.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A}_1)L(\mathcal{A}_2)$ erkennt.

Graphisch



Formal

Wir nehmen an: $Q_1 \cap Q_2 = \emptyset$.

Sei $\mathcal{A} = (Q_1 \cup Q_2, \Sigma, \Delta, q_{10}, F_2)$ mit

$$\Delta := \Delta_1 \cup \Delta_2 \cup \{(q, \varepsilon, q_{20}) \mid q \in F_1\}.$$

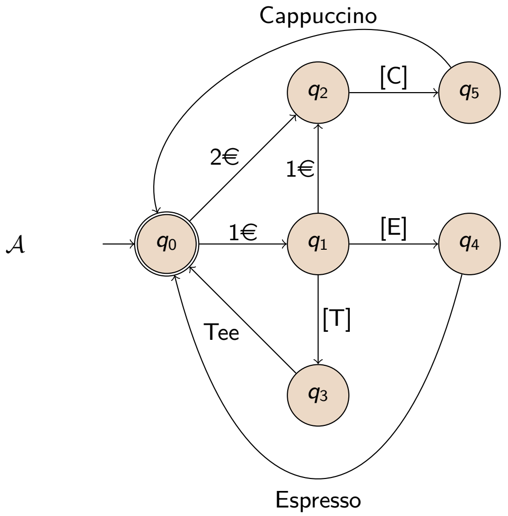
Behauptung

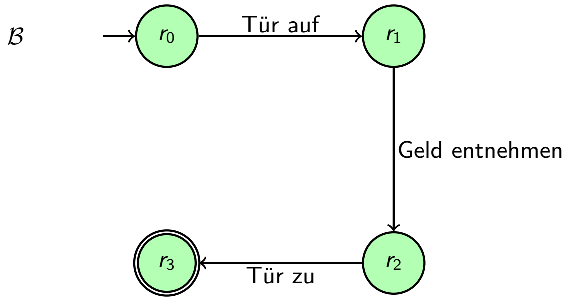
$L(\mathcal{A}) = L(\mathcal{A}_1)L(\mathcal{A}_2)$.

Ohne Beweis (ist aber einfach).

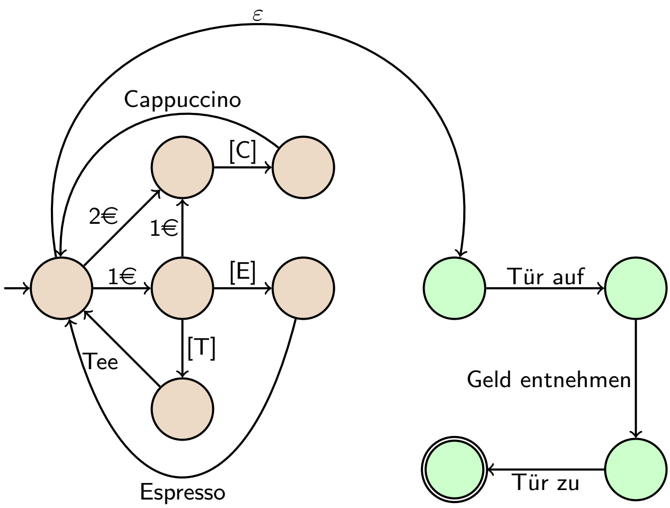
Beispiel 1.36 (Forts.)

In Beispiel 1.36 wollten wir die von folgenden beiden NFAs beschriebenen Prozesse hintereinander ausführen, also die zugehörigen Sprachen verketten.





Unsere Konstruktion liefert folgenden ε-NFA:



Notation

Für ein Zeichen a und ein $n \in \mathbb{N}$ schreiben wir

$$\textcolor{red}{a^n} := \underbrace{a \dots a}_{n \text{ mal}}.$$

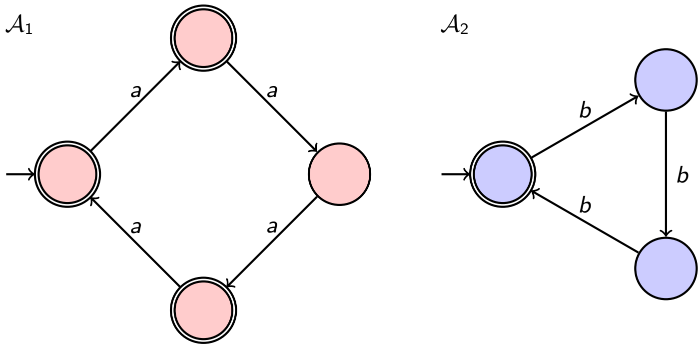
Wir betrachten folgende Sprachen über $\Sigma = \{a, b\}$:

$$L_1 := \{a^k \mid k \not\equiv 2 \bmod 4\},$$
$$L_2 := \{b^\ell \mid \ell \equiv 0 \bmod 3\}.$$

Dann ist

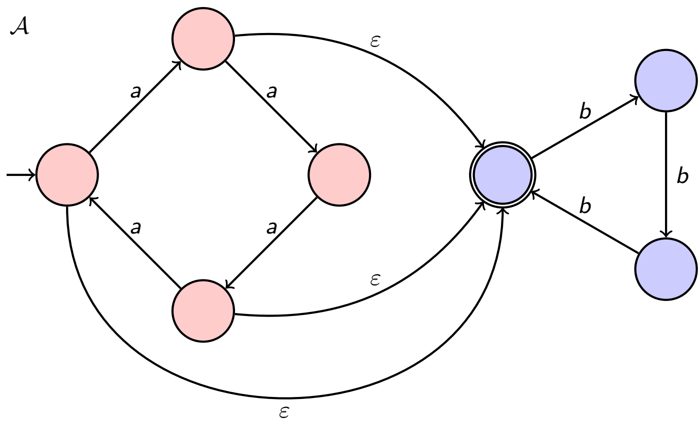
$$L_1L_2 = \{a^k b^\ell \mid k \not\equiv 2 \bmod 4 \text{ und } \ell \equiv 0 \bmod 3\}.$$

Folgende Automaten erkennen die Sprachen L_1 bzw. L_2:



Es gilt $L_1 = L(\mathcal{A}_1)$ und $L_2 = L(\mathcal{A}_2)$.

Wir konstruieren daraus den folgenden Automaten für die Verkettung:



Es gilt $L(\mathcal{A}) = L_1L_2$.

Iteration

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$.

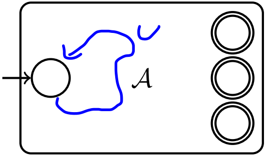
Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A})^*$ erkennt.

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A})^*$ erkennt.

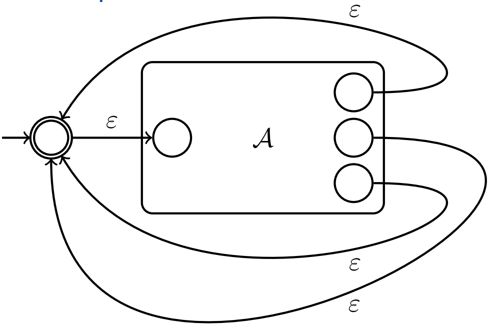
Graphisch



Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A})^*$ erkennt.

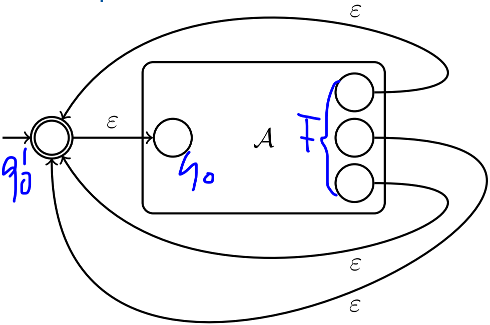
Graphisch



Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A})^*$ erkennt.

Graphisch



Formal

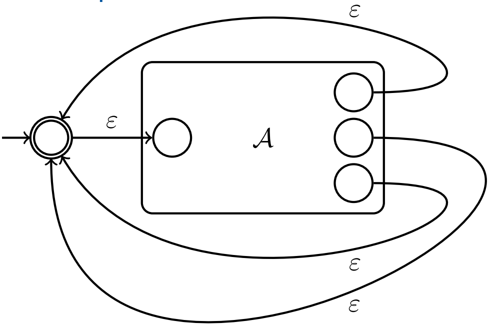
Sei $q_0' \notin Q$ ein neuer Zustand.

Iteration

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A})^*$ erkennt.

Graphisch



Formal

Sei $q_0' \notin Q$ ein neuer Zustand.

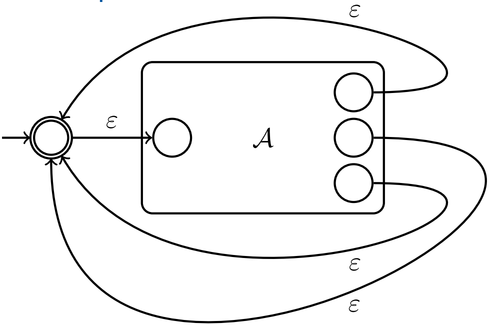
Sei
$\mathcal{A}' = (Q \cup \{q_0'\}, \Sigma, \Delta', q_0', \{q_0'\})$
mit

$$\begin{aligned}\Delta' &:= \Delta \cup \{(q_0', \varepsilon, q_0)\}\\ &\quad \cup \{(q, \varepsilon, q_0') \mid q \in F\}.\end{aligned}$$

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$.

Wir konstruieren einen ε-NFA, der die Sprache $L(\mathcal{A})^*$ erkennt.

Graphisch



Formal

Sei $q_0' \notin Q$ ein neuer Zustand.

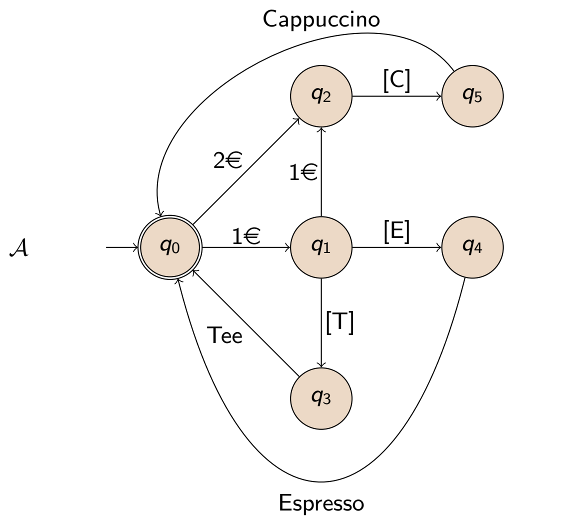
Sei
$\mathcal{A}' = (Q \cup \{q_0'\}, \Sigma, \Delta', q_0', \{q_0'\})$
mit

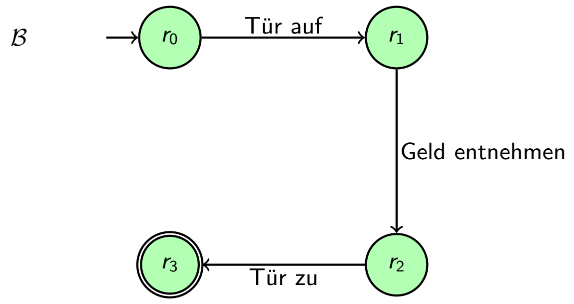
$$\Delta' := \Delta \cup \{(q_0', \varepsilon, q_0)\} \cup \{(q, \varepsilon, q_0') \mid q \in F\}.$$

Behauptung

$L(\mathcal{A}') = L(\mathcal{A})^*$.

Aus den Automaten

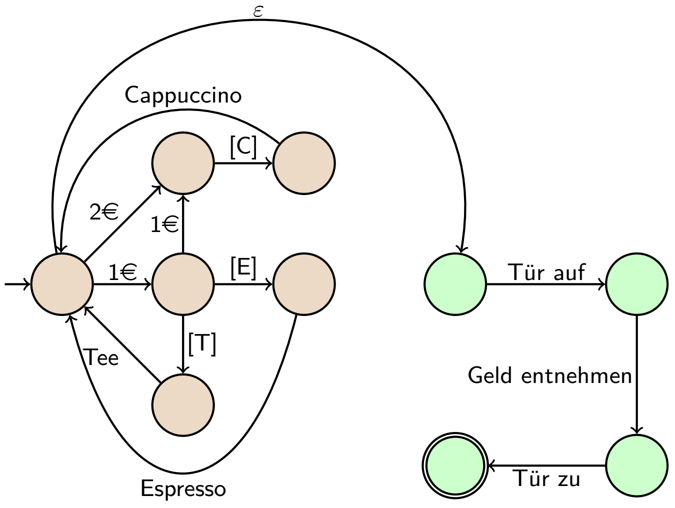




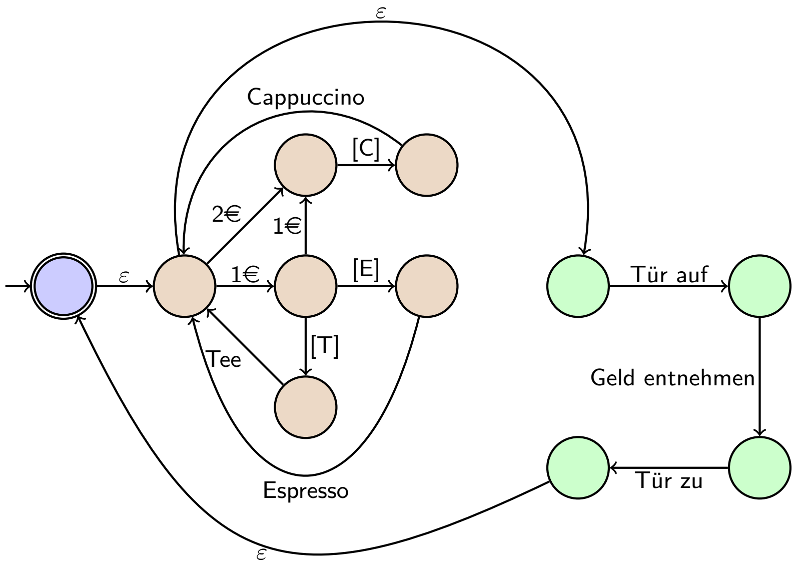
haben wir folgenden Automaten für die Sprache

$$L := L(\mathcal{A})L(\mathcal{B})$$

konstruiert:



Daraus erhalten wir folgenden Automaten für die Sprache L^*:



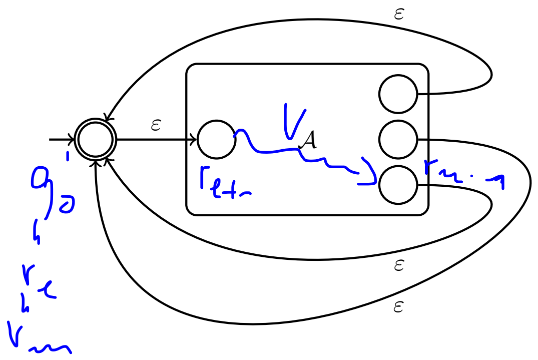
Bemerkungen 2.39

Wir haben damit also einen Automaten für die Sprache $\big(L(\mathcal{A})L(\mathcal{B})\big)^*$ konstruiert.

- Die Konstruktion folgt aus allgemeinen Ergebnissen und ist damit korrekt.
- Aber ist $\big(L(\mathcal{A})L(\mathcal{B})\big)^*$ wirklich die Sprache, die uns interessiert?
- Und wie können wir komplexe Sprachen überhaupt vernünftig beschreiben?

Sei $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ ein ε-NFA und $L := L(\mathcal{A})$.
Wir definieren den $\mathcal{A}' = (Q \cup \{q_0'\}, \Sigma, \Delta', q_0', \{q_0'\})$ mit

$$\Delta' := \Delta \cup \{(q_0', \varepsilon, q_0)\} \cup \{(q, \varepsilon, q_0') \mid q \in F\}.$$



Unser Ziel ist es,

$$L^* = L(\mathcal{A}') \qquad (\star)$$

zu beweisen.

Definition

Sei $n \in \mathbb{N}$. Ein n-Lauf von $\mathcal{A}'$ ist ein Lauf mit folgenden Eigenschaften:

- Der Zustand q_0' kommt genau n mal vor.
- Der letzte Zustand des Laufs ist q_0'.

Beobachtung

Ein Lauf von $\mathcal{A}'$ ist genau dann akzeptierend, wenn er ein n-Lauf für ein $n \geq 1$ ist.

Per Induktion über $n \geq 0$ zeigen wir für alle $w \in \Sigma^*$:

$$w \in L^n \iff \text{es gibt einen } (n+1)\text{-Lauf von } \mathcal{A}' \text{ auf } w. \qquad (\star\star)$$

Das impliziert sofort $(\star)$.

Induktionsanfang: $n = 0$.
Dann ist $w = \varepsilon$.
Es gilt $\varepsilon \in L^0$, und (q_0') ist ein 1-Lauf von $\mathcal{A}'$ auf ε.

Induktionsschritt: $n \to n+1$.

„$\Longrightarrow$“: Gelte $w \in L^{n+1} = L^n L$.
Dann gibt es ein $w' \in L^n$ und ein $v \in L$, so dass $w = w'v$.
Weil $w' \in L^n$ gibt es nach Induktionsannahme einen n-Lauf

$$(r_0, \ldots, r_\ell)$$

von $\mathcal{A}'$ auf w'.

Weil $v \in L$ gibt es einen akzeptierenden Lauf

$$(s_0, \ldots, s_m)$$

von $\mathcal{A}$ auf v.

Es gilt: $r_\ell = q_0'$ und $s_0 = q_0$ und $s_m \in F$.

Also $(r_\ell, \varepsilon, s_0) \in \Delta'$ und $(s_m, \varepsilon, q_0') \in \Delta'$.
Weil $\Delta \subseteq \Delta'$ ist damit

$$(r_0, \ldots, r_\ell, s_0, \ldots, s_m, q_0')$$

ein $(n+1)$-Lauf von $\mathcal{A}'$ auf $w = w'v$.

„$\Longleftarrow$“: Sei $r_\ell = q_0'$

$$(r_0, \sigma_1, r_1, \ldots, \sigma_m, r_m)$$

ein $(n+2)$-Lauf von $\mathcal{A}'$ auf w. (Das Wort w entsteht also aus $\sigma_1 \ldots \sigma_m$ durch Weglassen der εs.)

Sei $\ell < m$ maximal, so dass $r_\ell = q_0'$, und sei w' das Wort, das aus $\sigma_1 \ldots \sigma_\ell$ durch Weglassen der εs entsteht.
Dann ist

$$(r_0, \sigma_1, r_1, \ldots, \sigma_\ell, r_\ell)$$

ein $(n+1)$-Lauf von $\mathcal{A}'$ auf w'.
Nach Induktionsannahme gilt also $w' \in L^n$.

- Weil $r_\ell = q_0'$ und weil (q_0', ε, q_0) die einzige von q_0' ausgehende Transition von $\mathcal{A}'$ ist, gilt $r_\ell + 1 = q_0$ und $\sigma_{\ell+1} = \varepsilon$.
- Weil $r_m = q_0'$ und weil für alle Transitionen (q, σ, q_0') von $\mathcal{A}'$
  $$q \in F \quad \text{und} \quad \sigma = \varepsilon$$
  gilt, ist $r_{m-1} \in F$ und $\sigma_m = \varepsilon$.
- Weil $r_{\ell+1}, \ldots, r_{m-1} \in Q$ und
  $$\Delta' \cap (Q \times (\Sigma \cup \{\varepsilon\}) \times Q) = \Delta$$
  ist
  $$(r_{\ell+1}, \sigma_{\ell+2}, r_{\ell+2}, \ldots, \sigma_{m-1}, r_{m-1})$$
  ein akzeptierender Lauf von $\mathcal{A}$ auf dem Wort v, das aus $\sigma_{\ell+2} \ldots \sigma_{m-1}$ durch Weglassen der εs entsteht.
  Also ist $v \in L$.
- Weil $\sigma_{\ell+1} = \sigma_m = \varepsilon$, ergibt sich $w = w'v$ und damit
  $$w \in L^n L = L^{n+1}.$$

Zusammenfassung

Wir haben gezeigt (Korollar 2.34), dass DFAs, NFAs und ε-NFAs die selben Sprachen erkennen. Wir nennen diese Sprachen FA-erkennbar.

Wir haben gezeigt (Korollar 2.34), dass DFAs, NFAs und ε-NFAs die selben Sprachen erkennen. Wir nennen diese Sprachen FA-erkennbar.

Zusammengefasst ergeben die Ergebnisse dieses Abschnitts folgenden Satz.

Satz 2.40

Sei Σ ein Alphabet.

1. *Ist $L \subseteq \Sigma^*$ FA-erkennbar, so auch $\overline{L}$.*
2. *Sind $K, L \subseteq \Sigma^*$ FA-erkennbar, so auch $K \cup L$.*
3. *Sind $K, L \subseteq \Sigma^*$ FA-erkennbar, so auch $K \cap L$.*
4. *Sind $K, L \subseteq \Sigma^*$ FA-erkennbar, so auch KL.*
5. *Ist $L \subseteq \Sigma^*$ FA-erkennbar, so auch L^*.*

Zusammenfassung

Wir haben gezeigt (Korollar 2.34), dass DFAs, NFAs und ε-NFAs die selben Sprachen erkennen. Wir nennen diese Sprachen FA-erkennbar.

Zusammengefasst ergeben die Ergebnisse dieses Abschnitts folgenden Satz.

Satz 2.40

Sei Σ ein Alphabet.

1. *Ist $L \subseteq \Sigma^*$ FA-erkennbar, so auch $\overline{L}$.*
2. *Sind $K, L \subseteq \Sigma^*$ FA-erkennbar, so auch $K \cup L$.*
3. *Sind $K, L \subseteq \Sigma^*$ FA-erkennbar, so auch $K \cap L$.*
4. *Sind $K, L \subseteq \Sigma^*$ FA-erkennbar, so auch KL.*
5. *Ist $L \subseteq \Sigma^*$ FA-erkennbar, so auch L^*.*

Wir sagen auch, dass die Klasse der FA-erkennbaren Sprachen abgeschlossen ist unter Komplementbildung, Vereinigung, Durchschnitt Verkettung und Iteration.