TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Informatik, Institut für Künstliche Intelligenz, Professur für Computational Logic

FORMALE SYSTEME

4. Vorlesung: Nichtdeterministische Endliche Automaten

Hannes Straß

TU Dresden, 21. Oktober 2021

Rückblick

Wiederholung

- Grammatiken können Sprachen beschreiben und sie grob in Typen unterteilen.
- Typ-3-Grammatiken generieren reguläre Sprachen.
- Deterministische endliche Automaten erkennen reguläre Sprachen.
- Nichtdeterministische endliche Automaten verallgemeinern die Definition der Übergangsfunktion: Der Automat „rät“, welcher Übergang der richtige ist.

Wiederholung: NFA

Ein nichtdeterministischer endlicher Automat (international: „NFA") $\mathcal{M}$ ist ein Tupel $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ mit den folgenden Bestandteilen:

- Q: endliche Menge von Zuständen,
- Σ: Alphabet,
- δ: Übergangsfunktion, eine totale Funktion $Q \times \Sigma \to 2^Q$, wobei 2^Q die Potenzmenge von Q ist;
- Q_0: Menge möglicher Startzustände $Q_0 \subseteq Q$,
- F: Menge von Endzuständen $F \subseteq Q$.

Notation: Wir schreiben statt $q' \in \delta(q, \mathtt{a})$ auch $q \xrightarrow{\mathtt{a}} q'$.

Aufzeichnung startet . . .

Die Sprache eines NFA

Läufe eines NFA

Ein Lauf eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ für ein Wort $w = \sigma_1 \cdots \sigma_n$ ist eine Folge von Zuständen $q_0 \ldots q_m$, so dass gilt:

- $q_0 \in Q_0$,
- $q_{i+1} \in \delta(q_i, \sigma_{i+1})$ für alle $0 \leq i < m$,
- (1) $m = |w| = n$ oder (2) $m < n$ und $\delta(q_m, \sigma_{m+1}) = \emptyset$.

Ein Lauf heißt akzeptierend, falls $m = n$ und $q_n \in F$.
Andernfalls heißt der Lauf verwerfend.

$\rightsquigarrow$ Ein DFA hat genau einen Lauf für jedes Wort.
Er akzeptiert, wenn dieser Lauf akzeptierend ist.

$\rightsquigarrow$ Ein NFA kann für ein Wort mehrere Läufe haben.
Er akzeptiert, wenn einer dieser Läufe akzeptierend ist.

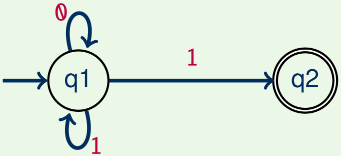
Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ist die Menge aller Wörter $w \in \Sigma^*$, für die $\mathcal{M}$ einen akzeptierenden Lauf hat.

Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ ist die Menge aller Wörter $w \in \Sigma^*$, für die $\mathcal{M}$ einen akzeptierenden Lauf hat.

Beispiel:



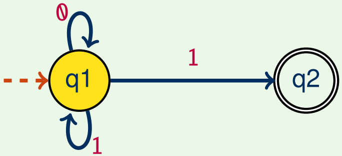
$\delta(q_1, 0) = \{q_1\}$
$\delta(q_1, 1) = \{q_1, q_2\}$
$\delta(q_2, 0) = \emptyset$
$\delta(q_2, 1) = \emptyset$

Wort	Lauf	Ergebnis
011	$q_1\ q_1\ q_2$	verwerfend (zu kurz)
011	$q_1\ q_1\ q_1\ q_2$	akzeptierend
011		

Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ist die Menge aller Wörter $w \in \Sigma^*$, für die $\mathcal{M}$ einen akzeptierenden Lauf hat.

Beispiel:



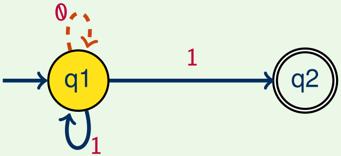
$\delta(q_1, \mathtt{0}) = \{q_1\}$
$\delta(q_1, \mathtt{1}) = \{q_1, q_2\}$
$\delta(q_2, \mathtt{0}) = \emptyset$
$\delta(q_2, \mathtt{1}) = \emptyset$

Wort	**Lauf**	**Ergebnis**
011	$q_1\ q_1\ q_2$	verwerfend (zu kurz)
011	$q_1\ q_1\ q_1\ q_2$	akzeptierend
011	q_1	

Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ ist die Menge aller Wörter $w \in \Sigma^*$, für die $\mathcal{M}$ einen akzeptierenden Lauf hat.

Beispiel:



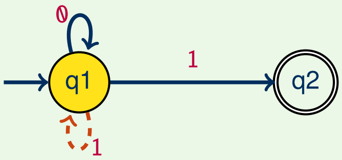
$\delta(q_1, \mathtt{0}) = \{q_1\}$
$\delta(q_1, \mathtt{1}) = \{q_1, q_2\}$
$\delta(q_2, \mathtt{0}) = \emptyset$
$\delta(q_2, \mathtt{1}) = \emptyset$

Wort	Lauf	Ergebnis
011	$q_1\ q_1\ q_2$	verwerfend (zu kurz)
011	$q_1\ q_1\ q_1\ q_2$	akzeptierend
011	$q_1\ q_1$	

Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ist die Menge aller Wörter $w \in \Sigma^*$, für die $\mathcal{M}$ einen akzeptierenden Lauf hat.

Beispiel:



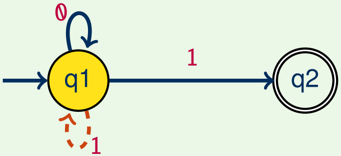
$\delta(q_1, 0) = \{q_1\}$
$\delta(q_1, 1) = \{q_1, q_2\}$
$\delta(q_2, 0) = \emptyset$
$\delta(q_2, 1) = \emptyset$

Wort	Lauf	Ergebnis
011	$q_1\ q_1\ q_2$	verwerfend (zu kurz)
011	$q_1\ q_1\ q_1\ q_2$	akzeptierend
011	$q_1\ q_1\ q_1$	

Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ist die Menge aller Wörter $w \in \Sigma^*$, für die $\mathcal{M}$ einen akzeptierenden Lauf hat.

Beispiel:



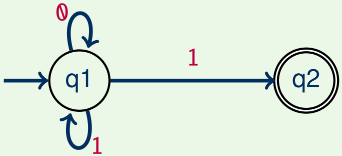
$\delta(q_1, 0) = \{q_1\}$
$\delta(q_1, 1) = \{q_1, q_2\}$
$\delta(q_2, 0) = \emptyset$
$\delta(q_2, 1) = \emptyset$

Wort	Lauf	Ergebnis
011	$q_1\ q_1\ q_2$	verwerfend (zu kurz)
011	$q_1\ q_1\ q_1\ q_2$	akzeptierend
011	$q_1\ q_1\ q_1\ q_1$	

Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ ist die Menge aller Wörter $w \in \Sigma^*$, für die $\mathcal{M}$ einen akzeptierenden Lauf hat.

Beispiel:



$\delta(q_1, 0) = \{q_1\}$

$\delta(q_1, 1) = \{q_1, q_2\}$

$\delta(q_2, 0) = \emptyset$

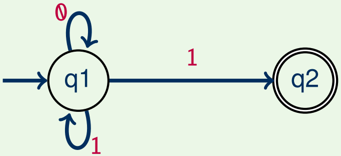
$\delta(q_2, 1) = \emptyset$

Wort	**Lauf**	**Ergebnis**
011	$q_1\ q_1\ q_2$	verwerfend (zu kurz)
011	$q_1\ q_1\ q_1\ q_2$	akzeptierend
011	$q_1\ q_1\ q_1\ q_1$	verwerfend (kein Endzustand)

Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ist die Menge aller Wörter $w \in \Sigma^*$, für die $\mathcal{M}$ einen akzeptierenden Lauf hat.

Beispiel:

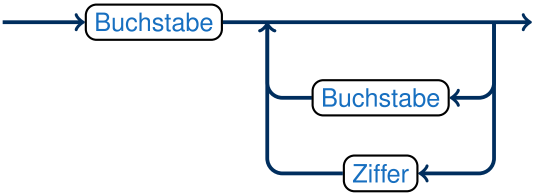


$\delta(q_1, 0) = \{q_1\}$
$\delta(q_1, 1) = \{q_1, q_2\}$
$\delta(q_2, 0) = \emptyset$
$\delta(q_2, 1) = \emptyset$

Wort	**Lauf**	**Ergebnis**
011	$q_1\ q_1\ q_2$	verwerfend (zu kurz)
011	$q_1\ q_1\ q_1\ q_2$	akzeptierend
011	$q_1\ q_1\ q_1\ q_1$	verwerfend (kein Endzustand)

$$\mathbf{L}(\mathcal{M}) = \{0, 1\}^* \circ \{1\}$$

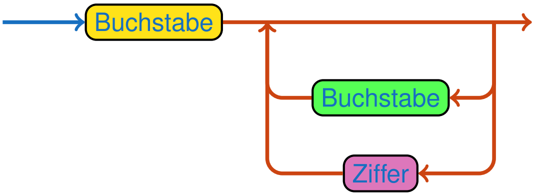
NFA zur Darstellung von Syntaxdiagrammen



Übersetzung in NFA:

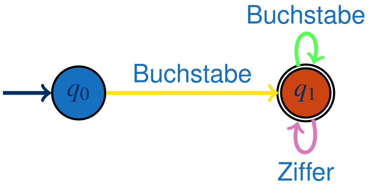
- zusammenhängende Linienbereiche werden Zustände
- Knoten werden Übergänge

NFA zur Darstellung von Syntaxdiagrammen

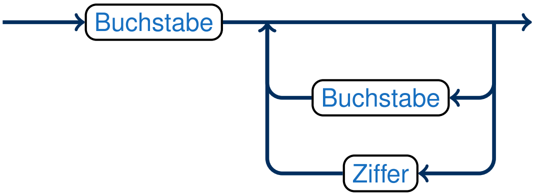


Übersetzung in NFA:

- zusammenhängende Linienbereiche werden Zustände
- Knoten werden Übergänge

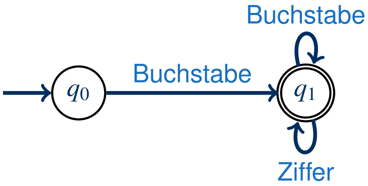


NFA zur Darstellung von Syntaxdiagrammen



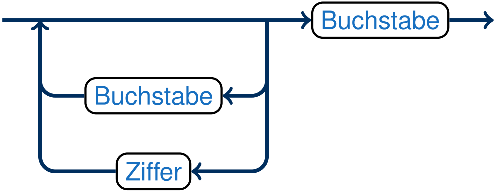
Übersetzung in NFA:

- zusammenhängende Linienbereiche werden Zustände
- Knoten werden Übergänge

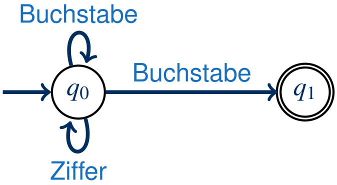


Syntaxdiagramme und Nichtdeterminismus

Das folgende Beispiel führt zu einem NFA, der kein DFA ist:



Entsprechender NFA:



Verallgemeinerte NFA-Übergangsfunktion

Wie beim DFA können wir auch bei einem NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ eine erweiterte Übergangsfunktion definieren, die ganze Wörter einliest.

Zuerst erweitern wir δ auf Mengen von Zuständen:

Für eine Zustandsmenge $R \subseteq Q$ und ein Terminalsymbol $\mathtt{a}$ sei

$$\delta^{\cup}(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a}).$$

Verallgemeinerte NFA-Übergangsfunktion

Wie beim DFA können wir auch bei einem NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ eine erweiterte Übergangsfunktion definieren, die ganze Wörter einliest.

Zuerst erweitern wir δ auf Mengen von Zuständen:

Für eine Zustandsmenge $R \subseteq Q$ und ein Terminalsymbol $\mathtt{a}$ sei

$$\delta^{\cup}(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a}).$$

Dann erweitern wir $\delta^{\cup}$ von einzelnen Symbolen zu beliebigen Wörtern:

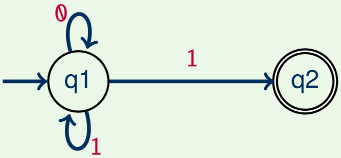
Für eine Zustandsmenge $R \subseteq Q$ und ein Wort $w \in \Sigma^*$ sei $\delta^*(R, w)$ die Menge aller Zustände, die man erreichen kann, wenn man in einem Zustand aus R beginnt und das Wort w einliest. Formal:

- $\delta^*(R, \epsilon) = R$
- $\delta^*(R, \mathtt{a}v) = \delta^*(\delta^{\cup}(R, \mathtt{a}), v)$

Wir notieren künftig wieder δ^* als δ, wenn die gemeinte Funktion aus dem Kontext klar ist.

Beispiel

Beispiel:



$\delta(q_1, 0) = \{q_1\}$
$\delta(q_1, 1) = \{q_1, q_2\}$
$\delta(q_2, 0) = \emptyset$
$\delta(q_2, 1) = \emptyset$

Die Menge der Startzustände ist $Q_0 = \{q_1\}$.

Dann gilt:

$$\delta(Q_0, 0) = \delta(q_1, 0) = \{q_1\}$$

$$\delta(Q_0, 1) = \delta(q_1, 1) = \{q_1, q_2\}$$

$$\begin{aligned}\delta(Q_0, 10) &= \delta(\delta(Q_0, 1), 0) = \delta(\{q_1, q_2\}, 0) \\ &= \delta(q_1, 0) \cup \delta(q_2, 0) = \{q_1\} \cup \emptyset = \{q_1\}\end{aligned}$$

$$\delta(Q_0, 01) = \delta(\delta(Q_0, 0), 1) = \delta(\{q_1\}, 1) = \{q_1, q_2\}$$

Sprache eines NFA (2. Version)

Die erweiterte Übergangsfunktion hilft bei der Definition der Sprache, die ein NFA akzeptiert:

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ ist die Menge

$$\mathbf{L}(\mathcal{M}) = \{w \in \Sigma^* \mid \delta(Q_0, w) \cap F \neq \emptyset\}.$$

Die Bedingung „$\delta(Q_0, w) \cap F \neq \emptyset$“ bedeutet:

„Mindestens einer der Zustände, die man durch Einlesen von w von einem Startzustand aus erreichen kann, ist ein Endzustand.“

Behauptung: Diese Variante stimmt mit der vorherigen (mit akzeptierenden Läufen) überein.

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ein NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ein NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Wdh.: Wir beweisen $P \Leftrightarrow Q$ („P gdw. Q“) durch Zeigen von (1) $P \Rightarrow Q$ und (2) $Q \Rightarrow P$.

Beweis „$\Rightarrow$“: Angenommen, es gibt einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ein NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Wdh.: Wir beweisen $P \Leftrightarrow Q$ („P gdw. Q“) durch Zeigen von (1) $P \Rightarrow Q$ und (2) $Q \Rightarrow P$.

Beweis „$\Rightarrow$“: Angenommen, es gibt einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.

- Dann ist $q_n \in F$.

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ein NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

> Wdh.: Wir beweisen $P \Leftrightarrow Q$ („P gdw. Q“) durch Zeigen von (1) $P \Rightarrow Q$ und (2) $Q \Rightarrow P$.

Beweis „⇒“: Angenommen, es gibt einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.

- Dann ist $q_n \in F$.
- Wir behaupten $q_n \in \delta(Q_0, w)$. (Damit folgt $\delta(Q_0, w) \cap F \neq \emptyset$.)

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ ein NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

> Wdh.: Wir beweisen $P \Leftrightarrow Q$ („P gdw. Q“) durch Zeigen von (1) $P \Rightarrow Q$ und (2) $Q \Rightarrow P$.

Beweis „⇒“: Angenommen, es gibt einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.

- Dann ist $q_n \in F$.
- Wir behaupten $q_n \in \delta(Q_0, w)$. (Damit folgt $\delta(Q_0, w) \cap F \neq \emptyset$.)
- Wir zeigen die stärkere Behauptung $q_i \in \delta(Q_0, \sigma_1 \cdots \sigma_i)$ für alle $0 \leq i \leq n$ mittels Induktion über i:

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ein NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Wdh.: Wir beweisen $P \Leftrightarrow Q$ („P gdw. Q“) durch Zeigen von (1) $P \Rightarrow Q$ und (2) $Q \Rightarrow P$.

Beweis „⇒“: Angenommen, es gibt einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.

- Dann ist $q_n \in F$.
- Wir behaupten $q_n \in \delta(Q_0, w)$. (Damit folgt $\delta(Q_0, w) \cap F \neq \emptyset$.)
- Wir zeigen die stärkere Behauptung $q_i \in \delta(Q_0, \sigma_1 \cdots \sigma_i)$ für alle $0 \leq i \leq n$ mittels Induktion über i:
  - Induktionsanfang: Für $i = 0$ gilt $q_0 \in Q_0 = \delta(Q_0, \epsilon)$.

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ein NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Wdh.: Wir beweisen $P \Leftrightarrow Q$ („P gdw. Q“) durch Zeigen von (1) $P \Rightarrow Q$ und (2) $Q \Rightarrow P$.

Beweis „⇒“: Angenommen, es gibt einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.

- Dann ist $q_n \in F$.
- Wir behaupten $q_n \in \delta(Q_0, w)$. (Damit folgt $\delta(Q_0, w) \cap F \neq \emptyset$.)
- Wir zeigen die stärkere Behauptung $q_i \in \delta(Q_0, \sigma_1 \cdots \sigma_i)$ für alle $0 \leq i \leq n$ mittels Induktion über i:
  - Induktionsanfang: Für $i = 0$ gilt $q_0 \in Q_0 = \delta(Q_0, \epsilon)$.
  - Induktionsvoraussetzung: Die Behauptung gelte für i.

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ ein NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

> Wdh.: Wir beweisen $P \Leftrightarrow Q$ („P gdw. Q“) durch Zeigen von (1) $P \Rightarrow Q$ und (2) $Q \Rightarrow P$.

Beweis „⇒“: Angenommen, es gibt einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.

- Dann ist $q_n \in F$.
- Wir behaupten $q_n \in \delta(Q_0, w)$. (Damit folgt $\delta(Q_0, w) \cap F \neq \emptyset$.)
- Wir zeigen die stärkere Behauptung $q_i \in \delta(Q_0, \sigma_1 \cdots \sigma_i)$ für alle $0 \leq i \leq n$ mittels Induktion über i:
  - Induktionsanfang: Für $i = 0$ gilt $q_0 \in Q_0 = \delta(Q_0, \epsilon)$.
  - Induktionsvoraussetzung: Die Behauptung gelte für i.
  - Induktionsschritt: Für $i + 1$ gilt:

    $q_i \in \delta(Q_0, \sigma_1 \cdots \sigma_i)$ (Induktionsvoraussetzung)

    $q_{i+1} \in \delta(q_i, \sigma_{i+1})$ (laut Definition eines Laufs)

    $q_{i+1} \in \delta(\delta(Q_0, \sigma_1 \cdots \sigma_i), \sigma_{i+1}) = \delta(Q_0, \sigma_1 \cdots \sigma_i \sigma_{i+1})$

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Beweis „$\Leftarrow$“: Angenommen, $\delta(Q_0, w) \cap F \neq \emptyset$.

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Beweis „$\Leftarrow$“: Angenommen, $\delta(Q_0, w) \cap F \neq \emptyset$.

- Wir ermitteln einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Beweis „$\Leftarrow$“: Angenommen, $\delta(Q_0, w) \cap F \neq \emptyset$.

- Wir ermitteln einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.
- Dazu gehen wir rückwärts vor:
  - Wähle $q_n \in F \cap \delta(Q_0, w)$. (Existiert nach Voraussetzung.)
  - Für alle $i = n, \ldots, 1$: (Annahme: $q_i \in \delta(Q_0, \sigma_1 \cdots \sigma_i)$ ist bereits gewählt.)
    Wähle $q_{i-1} \in \delta(Q_0, \sigma_1 \cdots \sigma_{i-1})$, so dass $q_i \in \delta(q_{i-1}, \sigma_i)$.
    (Solch ein q_{i-1} existiert immer:
    $\delta(Q_0, \sigma_1 \cdots \sigma_i) = \delta(\delta(\cdots \delta(\delta(Q_0, \sigma_1), \sigma_2) \cdots, \sigma_{i-1}), \sigma_i) = \delta(\delta(Q_0, \sigma_1 \cdots \sigma_{i-1}), \sigma_i)$.)

Äquivalenz der Sprachdefinitionen für NFAs

Sei $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F\rangle$ NFA und $w = \sigma_1 \cdots \sigma_n \in \Sigma^*$ ein Wort.

Behauptung: Es gibt einen akzeptierenden Lauf für w genau dann, wenn $\delta(Q_0, w) \cap F \neq \emptyset$.

Beweis „$\Leftarrow$": Angenommen, $\delta(Q_0, w) \cap F \neq \emptyset$.

- Wir ermitteln einen akzeptierenden Lauf $q_0 \ldots q_n$ für w.
- Dazu gehen wir rückwärts vor:
  - Wähle $q_n \in F \cap \delta(Q_0, w)$. (Existiert nach Voraussetzung.)
  - Für alle $i = n, \ldots, 1$: (Annahme: $q_i \in \delta(Q_0, \sigma_1 \cdots \sigma_i)$ ist bereits gewählt.)
    Wähle $q_{i-1} \in \delta(Q_0, \sigma_1 \cdots \sigma_{i-1})$, so dass $q_i \in \delta(q_{i-1}, \sigma_i)$.
    (Solch ein q_{i-1} existiert immer:
    $\delta(Q_0, \sigma_1 \cdots \sigma_i) = \delta(\delta(\cdots\delta(\delta(Q_0, \sigma_1), \sigma_2)\cdots, \sigma_{i-1}), \sigma_i) = \delta(\delta(Q_0, \sigma_1 \cdots \sigma_{i-1}), \sigma_i)$.)
- Dies ist ein Lauf, da $q_0 \in \delta(Q_0, \epsilon) = Q_0$ und alle Übergänge erlaubt sind.
- Es ist ein akzeptierender Lauf, da $q_n \in F$. $\square$

Quiz: Sprache eines NFA

Die Sprache eines NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ ist $\mathbf{L}(\mathcal{M}) = \{w \in \Sigma^* \mid \delta(Q_0, w) \cap F \neq \emptyset\}$.

Quiz: Wir betrachten den grafisch gegebenen NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ über $\Sigma = \{\mathtt{0}, \mathtt{1}\}$:

…

NFA vs. DFA

Vergleich DFA – NFA

Ein endlicher Automat $\mathcal{M}$ ist genau dann äquivalent zu einem endlichen Automaten $\mathcal{M}'$, wenn $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}')$ gilt.

Offensichtlich sind NFAs mindestens so allgemein wie DFAs:

Satz: Jeder DFA kann als NFA aufgefasst werden. Daher wird jede von einem DFA akzeptierbare Sprache auch von einen NFA akzeptiert.

Beweis: Für jeden DFA $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F\rangle$ gibt es einen entsprechenden NFA $\mathcal{M}' = \langle Q, \Sigma, \delta_{\mathsf{NFA}}, \{q_0\}, F\rangle$ mit $\delta_{\mathsf{NFA}}(q, \mathtt{a}) = \{\delta(q, \mathtt{a})\}$. Offensichtlich ist $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}')$. $\square$

Vergleich DFA – NFA

Ein endlicher Automat $\mathcal{M}$ ist genau dann äquivalent zu einem endlichen Automaten $\mathcal{M}'$, wenn $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}')$ gilt.

Offensichtlich sind NFAs mindestens so allgemein wie DFAs:

Satz: Jeder DFA kann als NFA aufgefasst werden. Daher wird jede von einem DFA akzeptierbare Sprache auch von einen NFA akzeptiert.

Beweis: Für jeden DFA $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F\rangle$ gibt es einen entsprechenden NFA $\mathcal{M}' = \langle Q, \Sigma, \delta_{\mathsf{NFA}}, \{q_0\}, F\rangle$ mit $\delta_{\mathsf{NFA}}(q, \mathtt{a}) = \{\delta(q, \mathtt{a})\}$. Offensichtlich ist $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}')$. $\square$

Die Umkehrung dieses Satzes gilt allerdings auch:

Satz: Jede von einem NFA akzeptierbare Sprache wird auch von einen DFA akzeptiert.

In diesem Sinne sind NFA nicht ausdrucksstärker als DFA – wie kann das sein?

NFAs als DFAs – Idee

Die verallgemeinerte NFA-Übergangsfunktion bildet Mengen von Zuständen auf Mengen von Zuständen ab:

$$\delta(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a}).$$

„Wenn der Automat in einem der Zustände R ist und $\mathtt{a}$ liest, so ist er anschließend in einem der Zustände der Menge $\delta(R, \mathtt{a})$."

Dieser Übergang zwischen Mengen möglicher Zustände ist an sich deterministisch.

$\rightsquigarrow$ Wir können einen NFA deterministisch simulieren, indem wir die Menge der möglichen Zustände berechnen.

Die Potenzmengenkonstruktion

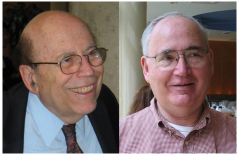
Für einen NFA $\mathcal{M} = \langle Q, \Sigma, \delta, Q_0, F \rangle$ definieren wir den Potenzmengen-DFA $\mathcal{M}_{\text{DFA}} = \langle Q_{\text{DFA}}, \Sigma, \delta_{\text{DFA}}, q_0, F_{\text{DFA}} \rangle$ wie folgt:

- $Q_{\text{DFA}} = 2^Q$ (Potenzmenge von Q)
- $\delta_{\text{DFA}}(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a})$
- $q_0 = Q_0$
- $F_{\text{DFA}} = \{R \in 2^Q \mid R \cap F \neq \emptyset\}$

Der resultierende DFA $\mathcal{M}_{\text{DFA}}$ ist äquivalent zu $\mathcal{M}$:

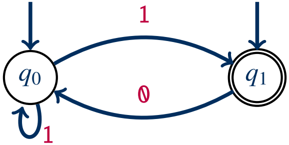
Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\text{DFA}})$

(Beweis später)

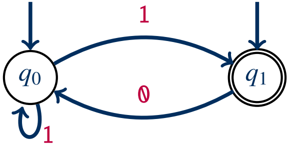


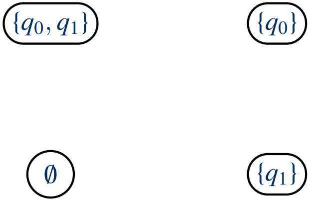
Michael Oser Rabin Dana Scott

Beispiel Potenzmengenkonstruktion

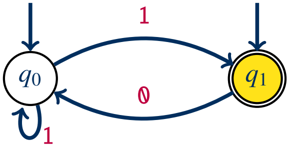


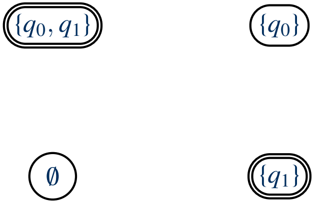
Beispiel Potenzmengenkonstruktion



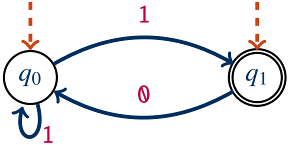


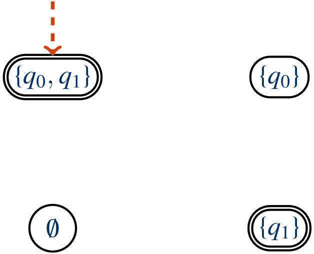
Beispiel Potenzmengenkonstruktion



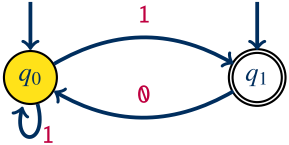


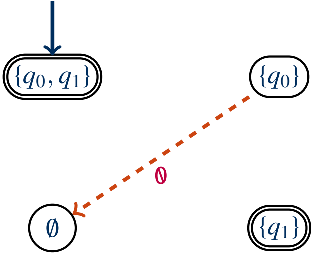
Beispiel Potenzmengenkonstruktion





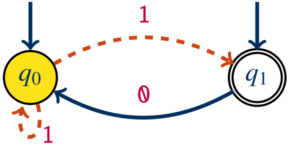
Beispiel Potenzmengenkonstruktion

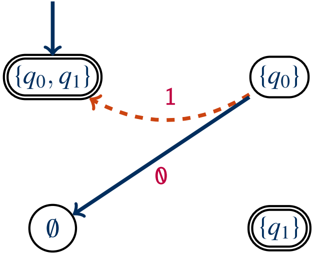




$$\delta_{\mathsf{DFA}}(\{q_0\}, 0) = \emptyset$$

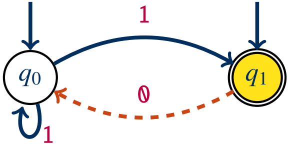
Beispiel Potenzmengenkonstruktion

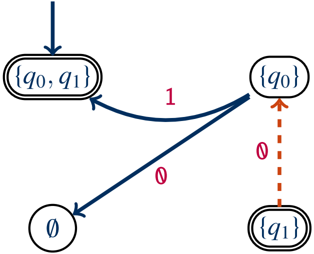




$$\delta_{\mathsf{DFA}}(\{q_0\}, 0) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\{q_0\}, 1) = \{q_0, q_1\}$$

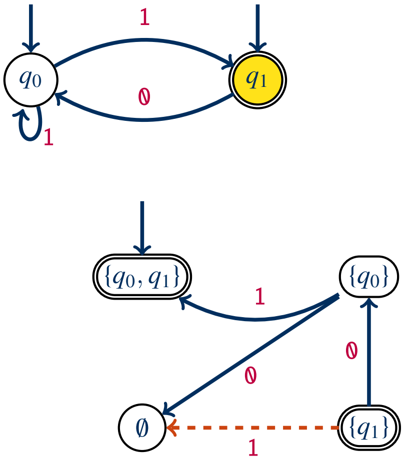
Beispiel Potenzmengenkonstruktion





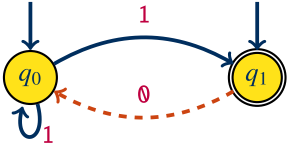
$$\delta_{\mathsf{DFA}}(\{q_0\}, 0) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\{q_0\}, 1) = \{q_0, q_1\}$$
$$\delta_{\mathsf{DFA}}(\{q_1\}, 0) = \{q_0\}$$

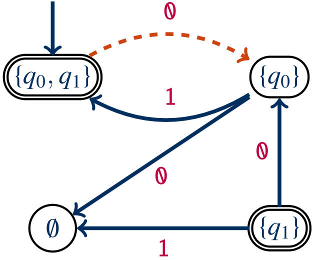
Beispiel Potenzmengenkonstruktion



$\delta_{\mathsf{DFA}}(\{q_0\}, 0) = \emptyset$
$\delta_{\mathsf{DFA}}(\{q_0\}, 1) = \{q_0, q_1\}$
$\delta_{\mathsf{DFA}}(\{q_1\}, 0) = \{q_0\}$
$\delta_{\mathsf{DFA}}(\{q_1\}, 1) = \emptyset$

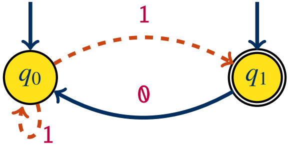
Beispiel Potenzmengenkonstruktion

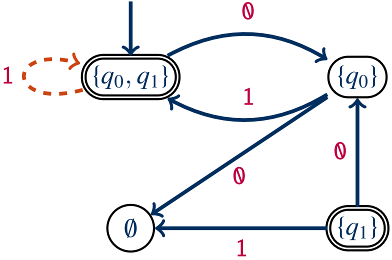




$$\delta_{\mathsf{DFA}}(\{q_0\}, 0) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\{q_0\}, 1) = \{q_0, q_1\}$$
$$\delta_{\mathsf{DFA}}(\{q_1\}, 0) = \{q_0\}$$
$$\delta_{\mathsf{DFA}}(\{q_1\}, 1) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\{q_0, q_1\}, 0) = \{q_0\}$$

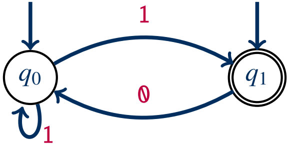
Beispiel Potenzmengenkonstruktion

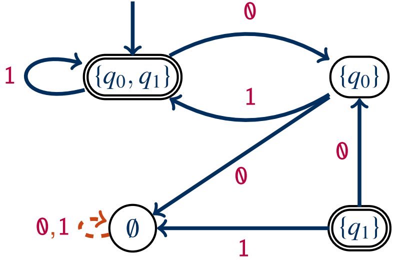




$$\delta_{\mathsf{DFA}}(\{q_0\}, 0) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\{q_0\}, 1) = \{q_0, q_1\}$$
$$\delta_{\mathsf{DFA}}(\{q_1\}, 0) = \{q_0\}$$
$$\delta_{\mathsf{DFA}}(\{q_1\}, 1) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\{q_0, q_1\}, 0) = \{q_0\}$$
$$\delta_{\mathsf{DFA}}(\{q_0, q_1\}, 1) = \{q_0, q_1\}$$

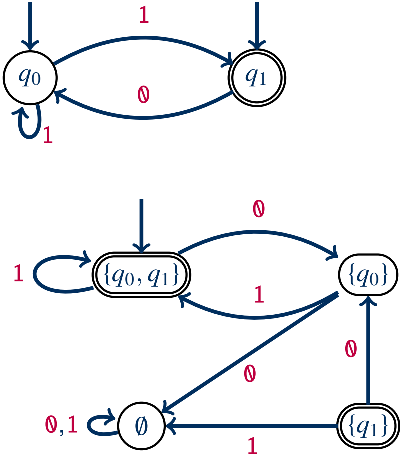
Beispiel Potenzmengenkonstruktion





$$\delta_{\mathsf{DFA}}(\{q_0\}, \mathtt{0}) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\{q_0\}, \mathtt{1}) = \{q_0, q_1\}$$
$$\delta_{\mathsf{DFA}}(\{q_1\}, \mathtt{0}) = \{q_0\}$$
$$\delta_{\mathsf{DFA}}(\{q_1\}, \mathtt{1}) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\{q_0, q_1\}, \mathtt{0}) = \{q_0\}$$
$$\delta_{\mathsf{DFA}}(\{q_0, q_1\}, \mathtt{1}) = \{q_0, q_1\}$$
$$\delta_{\mathsf{DFA}}(\emptyset, \mathtt{0}) = \emptyset$$
$$\delta_{\mathsf{DFA}}(\emptyset, \mathtt{1}) = \emptyset$$

Beispiel Potenzmengenkonstruktion

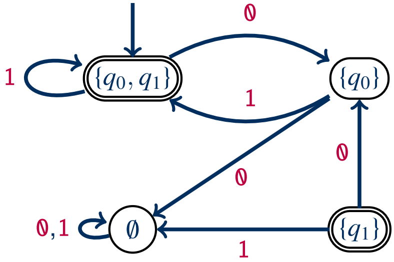


$\delta_{\text{DFA}}(\{q_0\}, 0) = \emptyset$
$\delta_{\text{DFA}}(\{q_0\}, 1) = \{q_0, q_1\}$
$\delta_{\text{DFA}}(\{q_1\}, 0) = \{q_0\}$
$\delta_{\text{DFA}}(\{q_1\}, 1) = \emptyset$
$\delta_{\text{DFA}}(\{q_0, q_1\}, 0) = \{q_0\}$
$\delta_{\text{DFA}}(\{q_0, q_1\}, 1) = \{q_0, q_1\}$
$\delta_{\text{DFA}}(\emptyset, 0) = \emptyset$
$\delta_{\text{DFA}}(\emptyset, 1) = \emptyset$

Erkannte Sprache:
$\{1\}^* \circ (\{0\} \circ \{1\}^+)^*$

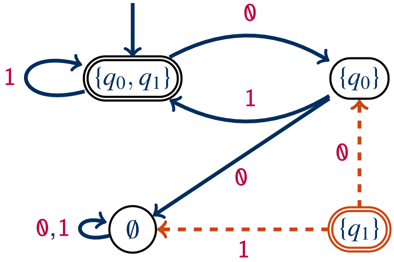
Vereinfachung Potenzmengenkonstruktion

Der Automat aus dem vorherigen Beispiel kann vereinfacht werden:



Vereinfachung Potenzmengenkonstruktion

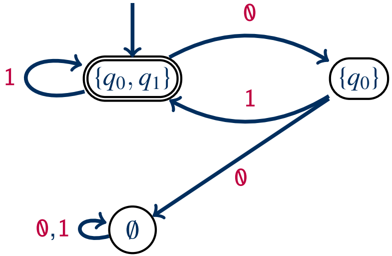
Der Automat aus dem vorherigen Beispiel kann vereinfacht werden:



- Zustand $\{q_1\}$ ist unerreichbar;

Vereinfachung Potenzmengenkonstruktion

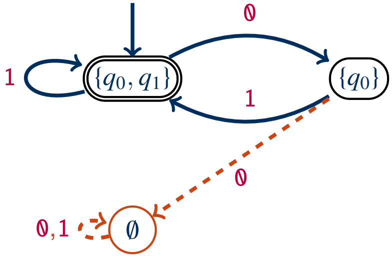
Der Automat aus dem vorherigen Beispiel kann vereinfacht werden:



- Zustand $\{q_1\}$ ist unerreichbar;

Vereinfachung Potenzmengenkonstruktion

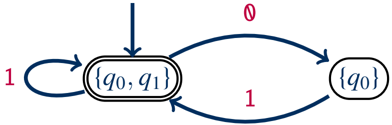
Der Automat aus dem vorherigen Beispiel kann vereinfacht werden:



- Zustand $\{q_1\}$ ist unerreichbar;
- Zustand $\emptyset$ kann nicht verlassen werden (irrelevant für akzeptierende Läufe).

Vereinfachung Potenzmengenkonstruktion

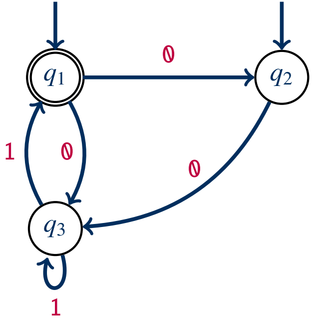
Der Automat aus dem vorherigen Beispiel kann vereinfacht werden:



- Zustand $\{q_1\}$ ist unerreichbar;
- Zustand $\emptyset$ kann nicht verlassen werden (irrelevant für akzeptierende Läufe).

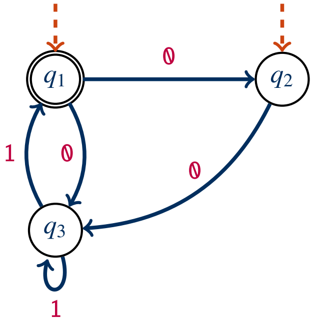
Potenzmengenkonstruktion „on the fly“

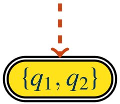
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:



Potenzmengenkonstruktion „on the fly“

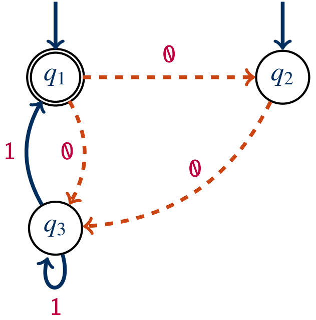
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:

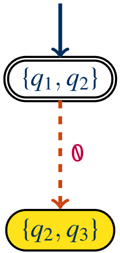




Potenzmengenkonstruktion „on the fly“

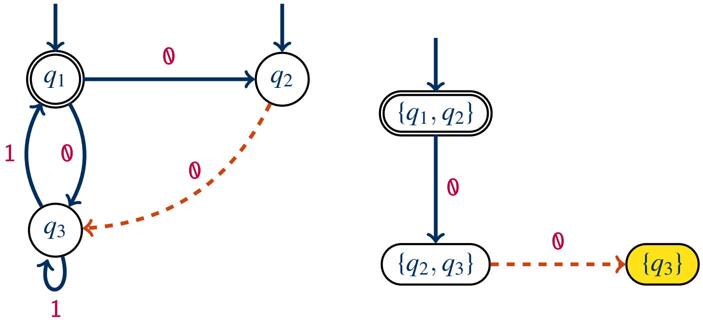
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:





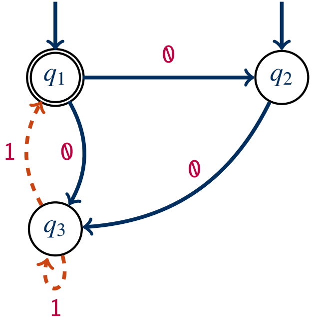
Potenzmengenkonstruktion „on the fly“

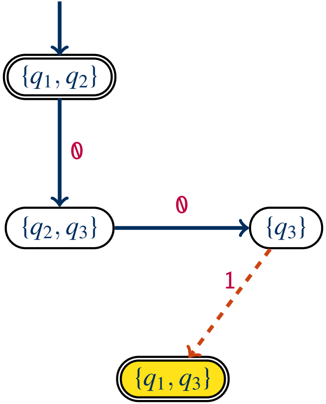
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:



Potenzmengenkonstruktion „on the fly“

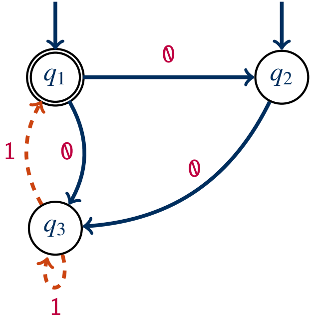
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:

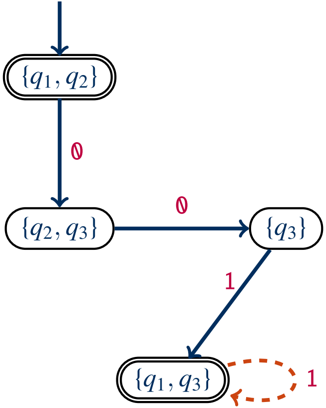




Potenzmengenkonstruktion „on the fly“

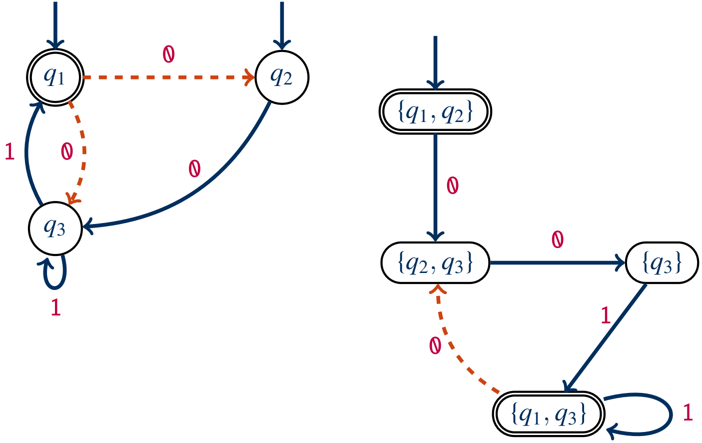
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:





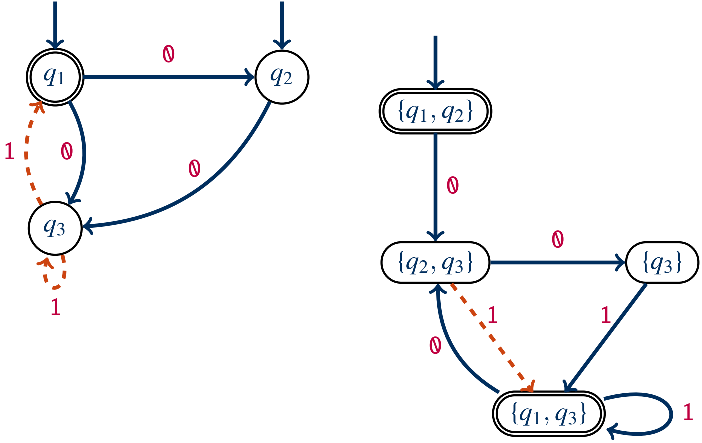
Potenzmengenkonstruktion „on the fly“

Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:



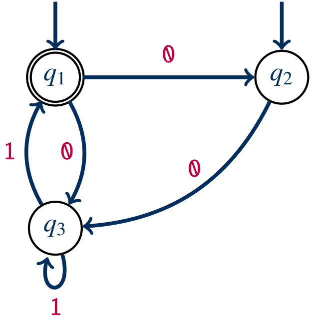
Potenzmengenkonstruktion „on the fly“

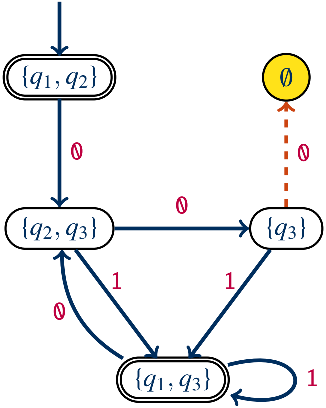
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:



Potenzmengenkonstruktion „on the fly“

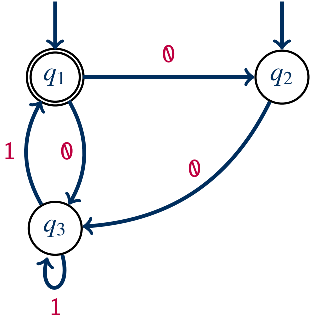
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:

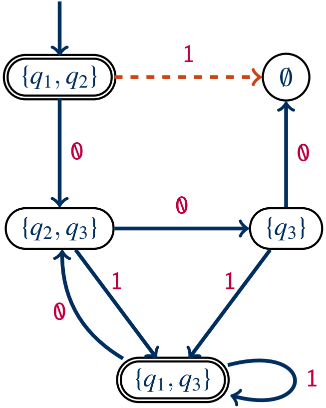




Potenzmengenkonstruktion „on the fly“

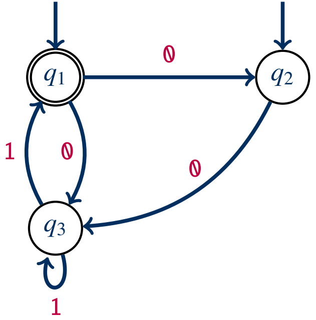
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:

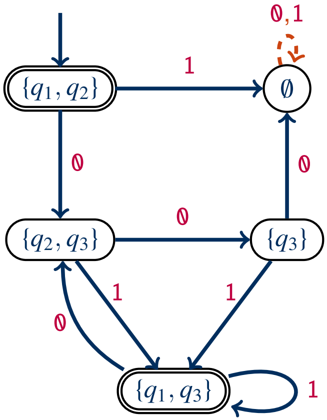




Potenzmengenkonstruktion „on the fly“

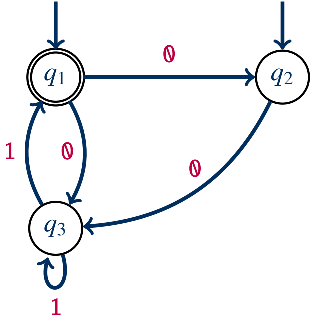
Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:

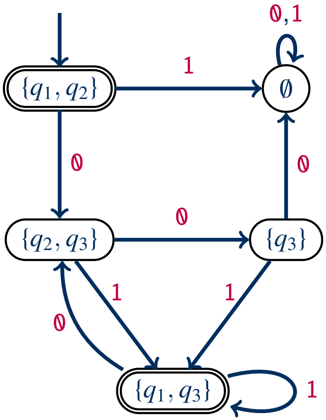




Potenzmengenkonstruktion „on the fly“

Vermeidung unnötiger Zustände durch schrittweise Konstruktion vom Startzustand:





- Erreichbarer Teil spart drei Zustände ein
- Zustand $\emptyset$ wie zuvor unnötig

Potenzmengenkonstruktion: Korrektheit

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis: Wir nutzen die Korrespondenz der verallgemeinerten Übergangsfunktionen aus.
Zuerst zeigen wir, dass für jedes Wort $w \in \Sigma^*$ und jede Zustandsmenge R gilt:
$\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Beweis per Induktion über die Länge von w.)

Potenzmengenkonstruktion: Korrektheit

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis: Wir nutzen die Korrespondenz der verallgemeinerten Übergangsfunktionen aus. Zuerst zeigen wir, dass für jedes Wort $w \in \Sigma^*$ und jede Zustandsmenge R gilt:
$\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Beweis per Induktion über die Länge von w.)

Induktionsanfang:

(1) $\delta_{\mathsf{DFA}}(R, \epsilon) = R = \delta(R, \epsilon)$

Potenzmengenkonstruktion: Korrektheit

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis: Wir nutzen die Korrespondenz der verallgemeinerten Übergangsfunktionen aus.
Zuerst zeigen wir, dass für jedes Wort $w \in \Sigma^*$ und jede Zustandsmenge R gilt:
$\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Beweis per Induktion über die Länge von w.)

Induktionsanfang:

(1) $\delta_{\mathsf{DFA}}(R, \epsilon) = R = \delta(R, \epsilon)$

(2) $\delta_{\mathsf{DFA}}(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a}) = \delta(R, \mathtt{a})$

Potenzmengenkonstruktion: Korrektheit

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis: Wir nutzen die Korrespondenz der verallgemeinerten Übergangsfunktionen aus. Zuerst zeigen wir, dass für jedes Wort $w \in \Sigma^*$ und jede Zustandsmenge R gilt:
$\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Beweis per Induktion über die Länge von w.)

Induktionsanfang:

(1) $\delta_{\mathsf{DFA}}(R, \epsilon) = R = \delta(R, \epsilon)$

(2) $\delta_{\mathsf{DFA}}(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a}) = \delta(R, \mathtt{a})$

Induktionsvoraussetzung: Es gelte $\delta_{\mathsf{DFA}}(R, v) = \delta(R, v)$ für alle Wörter v der Länge ℓ.

Induktionsschritt: Wir zeigen $\delta_{\mathsf{DFA}}(R, \mathtt{a}v) = \delta(R, \mathtt{a}v)$ für ein beliebiges Wort $\mathtt{a}v$ der Länge $\ell + 1$.

(3) $\delta_{\mathsf{DFA}}(R, \mathtt{a}v) = \delta_{\mathsf{DFA}}(\delta_{\mathsf{DFA}}(R, \mathtt{a}), v)$

Potenzmengenkonstruktion: Korrektheit

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis: Wir nutzen die Korrespondenz der verallgemeinerten Übergangsfunktionen aus. Zuerst zeigen wir, dass für jedes Wort $w \in \Sigma^*$ und jede Zustandsmenge R gilt:
$\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Beweis per Induktion über die Länge von w.)

Induktionsanfang:

(1) $\delta_{\mathsf{DFA}}(R, \epsilon) = R = \delta(R, \epsilon)$

(2) $\delta_{\mathsf{DFA}}(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a}) = \delta(R, \mathtt{a})$

Induktionsvoraussetzung: Es gelte $\delta_{\mathsf{DFA}}(R, v) = \delta(R, v)$ für alle Wörter v der Länge ℓ.

Induktionsschritt: Wir zeigen $\delta_{\mathsf{DFA}}(R, \mathtt{a}v) = \delta(R, \mathtt{a}v)$ für ein beliebiges Wort $\mathtt{a}v$ der Länge $\ell + 1$.

(3) $$\begin{aligned}\delta_{\mathsf{DFA}}(R, \mathtt{a}v) &= \delta_{\mathsf{DFA}}(\delta_{\mathsf{DFA}}(R, \mathtt{a}), v) \\ &= \delta_{\mathsf{DFA}}(\delta(R, \mathtt{a}), v) \qquad \text{(wegen (2))}\end{aligned}$$

Potenzmengenkonstruktion: Korrektheit

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis: Wir nutzen die Korrespondenz der verallgemeinerten Übergangsfunktionen aus. Zuerst zeigen wir, dass für jedes Wort $w \in \Sigma^*$ und jede Zustandsmenge R gilt:
$\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Beweis per Induktion über die Länge von w.)

Induktionsanfang:

(1) $\delta_{\mathsf{DFA}}(R, \epsilon) = R = \delta(R, \epsilon)$

(2) $\delta_{\mathsf{DFA}}(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a}) = \delta(R, \mathtt{a})$

Induktionsvoraussetzung: Es gelte $\delta_{\mathsf{DFA}}(R, v) = \delta(R, v)$ für alle Wörter v der Länge ℓ.

Induktionsschritt: Wir zeigen $\delta_{\mathsf{DFA}}(R, \mathtt{a}v) = \delta(R, \mathtt{a}v)$ für ein beliebiges Wort $\mathtt{a}v$ der Länge $\ell + 1$.

(3)
$$\begin{aligned}
\delta_{\mathsf{DFA}}(R, \mathtt{a}v) &= \delta_{\mathsf{DFA}}(\delta_{\mathsf{DFA}}(R, \mathtt{a}), v) \\
&= \delta_{\mathsf{DFA}}(\delta(R, \mathtt{a}), v) && \text{(wegen (2))} \\
&= \delta(\delta(R, \mathtt{a}), v) && \text{(Induktionsvoraussetzung)}
\end{aligned}$$

Potenzmengenkonstruktion: Korrektheit

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis: Wir nutzen die Korrespondenz der verallgemeinerten Übergangsfunktionen aus.
Zuerst zeigen wir, dass für jedes Wort $w \in \Sigma^*$ und jede Zustandsmenge R gilt:
$\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Beweis per Induktion über die Länge von w.)

Induktionsanfang:

(1) $\delta_{\mathsf{DFA}}(R, \epsilon) = R = \delta(R, \epsilon)$

(2) $\delta_{\mathsf{DFA}}(R, \mathtt{a}) = \bigcup_{q \in R} \delta(q, \mathtt{a}) = \delta(R, \mathtt{a})$

Induktionsvoraussetzung: Es gelte $\delta_{\mathsf{DFA}}(R, v) = \delta(R, v)$ für alle Wörter v der Länge ℓ.

Induktionsschritt: Wir zeigen $\delta_{\mathsf{DFA}}(R, \mathtt{a}v) = \delta(R, \mathtt{a}v)$ für ein beliebiges Wort $\mathtt{a}v$ der Länge $\ell + 1$.

(3)
$$
\begin{aligned}
\delta_{\mathsf{DFA}}(R, \mathtt{a}v) &= \delta_{\mathsf{DFA}}(\delta_{\mathsf{DFA}}(R, \mathtt{a}), v) \\
&= \delta_{\mathsf{DFA}}(\delta(R, \mathtt{a}), v) && \text{(wegen (2))} \\
&= \delta(\delta(R, \mathtt{a}), v) && \text{(Induktionsvoraussetzung)} \\
&= \delta(R, \mathtt{a}v)
\end{aligned}
$$

Potenzmengenkonstruktion: Korrektheit (2)

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis (Fortsetzung): Wir haben gezeigt: $\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Für alle $R \subseteq Q$, $w \in \Sigma^*$.)

Potenzmengenkonstruktion: Korrektheit (2)

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis (Fortsetzung): Wir haben gezeigt: $\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Für alle $R \subseteq Q$, $w \in \Sigma^*$.)

Damit ergibt sich für beliebige Wörter $w \in \Sigma^*$:

$w \in \mathbf{L}(\mathcal{M})$ gdw.

Potenzmengenkonstruktion: Korrektheit (2)

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis (Fortsetzung): Wir haben gezeigt: $\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Für alle $R \subseteq Q$, $w \in \Sigma^*$.)

Damit ergibt sich für beliebige Wörter $w \in \Sigma^*$:

$w \in \mathbf{L}(\mathcal{M})$ gdw. $\delta(Q_0, w) \cap F \neq \emptyset$

Potenzmengenkonstruktion: Korrektheit (2)

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis (Fortsetzung): Wir haben gezeigt: $\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Für alle $R \subseteq Q$, $w \in \Sigma^*$.)

Damit ergibt sich für beliebige Wörter $w \in \Sigma^*$:

$$
\begin{aligned}
w \in \mathbf{L}(\mathcal{M}) \quad & \text{gdw.} \quad \delta(Q_0, w) \cap F \neq \emptyset \\
& \text{gdw.} \quad \delta_{\mathsf{DFA}}(Q_0, w) \cap F \neq \emptyset
\end{aligned}
$$

Potenzmengenkonstruktion: Korrektheit (2)

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\text{DFA}})$

Beweis (Fortsetzung): Wir haben gezeigt: $\delta_{\text{DFA}}(R, w) = \delta(R, w)$. (Für alle $R \subseteq Q$, $w \in \Sigma^*$.)

Damit ergibt sich für beliebige Wörter $w \in \Sigma^*$:

$$
\begin{aligned}
w \in \mathbf{L}(\mathcal{M}) \quad &\text{gdw.} \quad \delta(Q_0, w) \cap F \neq \emptyset \\
&\text{gdw.} \quad \delta_{\text{DFA}}(Q_0, w) \cap F \neq \emptyset \\
&\text{gdw.} \quad \delta_{\text{DFA}}(Q_0, w) \in F_{\text{DFA}}
\end{aligned}
$$

Potenzmengenkonstruktion: Korrektheit (2)

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis (Fortsetzung): Wir haben gezeigt: $\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Für alle $R \subseteq Q$, $w \in \Sigma^*$.)

Damit ergibt sich für beliebige Wörter $w \in \Sigma^*$:

$$
\begin{aligned}
w \in \mathbf{L}(\mathcal{M}) \quad & \text{gdw.} \quad \delta(Q_0, w) \cap F \neq \emptyset \\
& \text{gdw.} \quad \delta_{\mathsf{DFA}}(Q_0, w) \cap F \neq \emptyset \\
& \text{gdw.} \quad \delta_{\mathsf{DFA}}(Q_0, w) \in F_{\mathsf{DFA}} \\
& \text{gdw.} \quad w \in \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})
\end{aligned}
$$

□

Potenzmengenkonstruktion: Korrektheit (2)

Satz (Rabin/Scott): $\mathbf{L}(\mathcal{M}) = \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})$

Beweis (Fortsetzung): Wir haben gezeigt: $\delta_{\mathsf{DFA}}(R, w) = \delta(R, w)$. (Für alle $R \subseteq Q$, $w \in \Sigma^*$.)

Damit ergibt sich für beliebige Wörter $w \in \Sigma^*$:

$$
\begin{aligned}
w \in \mathbf{L}(\mathcal{M}) \quad & \text{gdw.} \quad \delta(Q_0, w) \cap F \neq \emptyset \\
& \text{gdw.} \quad \delta_{\mathsf{DFA}}(Q_0, w) \cap F \neq \emptyset \\
& \text{gdw.} \quad \delta_{\mathsf{DFA}}(Q_0, w) \in F_{\mathsf{DFA}} \\
& \text{gdw.} \quad w \in \mathbf{L}(\mathcal{M}_{\mathsf{DFA}})
\end{aligned}
$$

$\square$

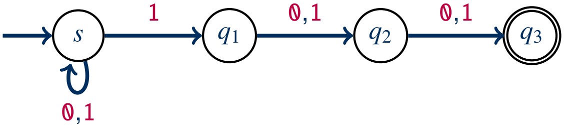
Die beiden Automatenmodelle DFA und NFA charakterisieren also die gleiche Klasse von Sprachen.
$\rightsquigarrow$ Benötigen wir den Nichtdeterminismus der NFAs überhaupt noch?

Größenvergleich

Der DFA eines NFA hat $2^{|Q|}$ – also exponentiell viele – Zustände.
Auch „on the fly“ lässt sich das im Allgemeinen nicht vermeiden.

Beispiel: „Wörter mit 1 an drittletzter Stelle“

NFA:

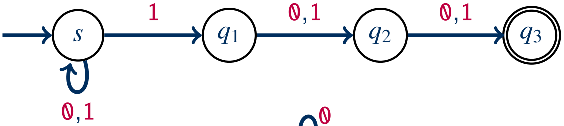


Größenvergleich

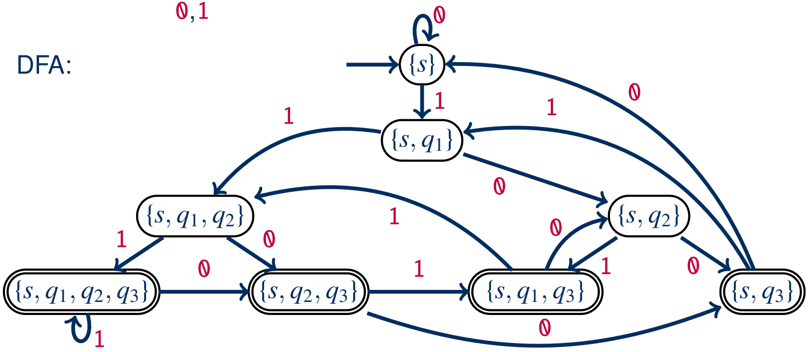
Der DFA eines NFA hat $2^{|Q|}$ – also exponentiell viele – Zustände.
Auch „on the fly“ lässt sich das im Allgemeinen nicht vermeiden.

Beispiel: „Wörter mit 1 an drittletzter Stelle“

NFA:



DFA:



Größenvergleich (2)

Allgemein kann man für jede Zahl $n \geq 1$ die Sprache $\mathbf{L}_n = \{\mathtt{0}, \mathtt{1}\}^* \mathtt{1} \{\mathtt{0}, \mathtt{1}\}^{n-1}$ betrachten („Wörter mit $\mathtt{1}$ an n-letzter Stelle“).

Es gilt:

- Es gibt einen NFA mit $n + 1$ Zuständen, der $\mathbf{L}_n$ erkennt.
- Jeder DFA, der $\mathbf{L}_n$ erkennt, hat mindestens 2^n Zustände.

Größenvergleich (2)

Allgemein kann man für jede Zahl $n \geq 1$ die Sprache $\mathbf{L}_n = \{\mathtt{0}, \mathtt{1}\}^* \mathtt{1} \{\mathtt{0}, \mathtt{1}\}^{n-1}$ betrachten („Wörter mit $\mathtt{1}$ an n-letzter Stelle“).

Es gilt:

- Es gibt einen NFA mit $n + 1$ Zuständen, der $\mathbf{L}_n$ erkennt.
- Jeder DFA, der $\mathbf{L}_n$ erkennt, hat mindestens 2^n Zustände.

Beweis (per Widerspruch): Angenommen, DFA $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F \rangle$ erkennt $\mathbf{L}_n$ und hat weniger als 2^n Zustände. Über $\{0, 1\}$ gibt es 2^n verschiedene Wörter der Länge n. Nach dem Schubfachprinzip gibt es $w_1, w_2 \in \{0, 1\}^n$ mit $w_1 \neq w_2$, sodass $\delta(q_0, w_1) = \delta(q_0, w_2)$ gilt. Da $w_1 \neq w_2$ gibt es ein $k \leq n$, so dass w_1 und w_2 sich an k-letzter Stelle unterscheiden. Wir definieren nun $v = 0^{n-k}$ und betrachten die Wörter $w_1 v$ und $w_2 v$, die sich an n-letzter Stelle unterscheiden. Aus $\delta(q_0, w_1) = \delta(q_0, w_2)$ folgt $\delta(q_0, w_1 v) = \delta(q_0, w_2 v)$, also gilt $w_1 v \in \mathbf{L}(\mathcal{M})$ gdw. $w_2 v \in \mathbf{L}(\mathcal{M})$. Dies steht im Widerspruch dazu, dass $\mathcal{M}$ die Sprache $\mathbf{L}_n$ erkennt, da sich die Wörter an n-letzter Stelle unterscheiden. $\square$

Größenvergleich (2)

Allgemein kann man für jede Zahl $n \geq 1$ die Sprache $\mathbf{L}_n = \{\mathtt{0}, \mathtt{1}\}^* \mathtt{1} \{\mathtt{0}, \mathtt{1}\}^{n-1}$ betrachten („Wörter mit $\mathtt{1}$ an n-letzter Stelle").

Es gilt:

- Es gibt einen NFA mit $n + 1$ Zuständen, der $\mathbf{L}_n$ erkennt.
- Jeder DFA, der $\mathbf{L}_n$ erkennt, hat mindestens 2^n Zustände.

Beweis (per Widerspruch): Angenommen, DFA $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F \rangle$ erkennt $\mathbf{L}_n$ und hat weniger als 2^n Zustände. Über $\{0, 1\}$ gibt es 2^n verschiedene Wörter der Länge n. Nach dem Schubfachprinzip gibt es $w_1, w_2 \in \{0, 1\}^n$ mit $w_1 \neq w_2$, sodass $\delta(q_0, w_1) = \delta(q_0, w_2)$ gilt. Da $w_1 \neq w_2$ gibt es ein $k \leq n$, so dass w_1 und w_2 sich an k-letzter Stelle unterscheiden. Wir definieren nun $v = 0^{n-k}$ und betrachten die Wörter $w_1 v$ und $w_2 v$, die sich an n-letzter Stelle unterscheiden. Aus $\delta(q_0, w_1) = \delta(q_0, w_2)$ folgt $\delta(q_0, w_1 v) = \delta(q_0, w_2 v)$, also gilt $w_1 v \in \mathbf{L}(\mathcal{M})$ gdw. $w_2 v \in \mathbf{L}(\mathcal{M})$. Dies steht im Widerspruch dazu, dass $\mathcal{M}$ die Sprache $\mathbf{L}_n$ erkennt, da sich die Wörter an n-letzter Stelle unterscheiden. $\square$

Schlussfolgerung: NFAs können exponentiell kompakter sein als äquivalente DFAs.

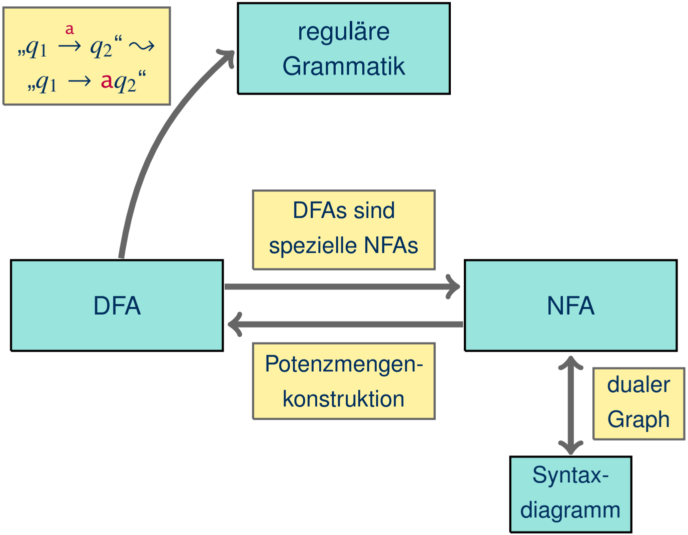
$\rightsquigarrow$ Nichtdeterminismus der NFAs trotzdem nützlich.

Quiz: NFA vs. DFA

Quiz: Sei $\mathcal{M}_\mathrm{N}$ ein NFA und $\mathcal{M}_\mathrm{D}$ ein DFA mit $\mathbf{L}(\mathcal{M}_\mathrm{N}) = \mathbf{L}(\mathcal{M}_\mathrm{D})$.

...

Darstellungen von Typ-3-Sprachen



Von regulären Grammatiken zu NFAs

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir können nun die noch fehlende Richtung dieser Behauptung zeigen:

Für jede reguläre Grammatik G gibt es einen NFA $\mathcal{M}_G$, welcher die selbe Sprache akzeptiert (d.h., $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$).

Von regulären Grammatiken zu NFAs

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir können nun die noch fehlende Richtung dieser Behauptung zeigen:

Für jede reguläre Grammatik G gibt es einen NFA $\mathcal{M}_G$, welcher die selbe Sprache akzeptiert (d.h., $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$).

Für $G = \langle V, \Sigma, P, \mathsf{S} \rangle$ ergibt sich $\mathcal{M}_G = \langle Q, \Sigma, \delta, Q_0, F \rangle$ wie folgt:

- $Q := V \cup \{q_f\}$

Von regulären Grammatiken zu NFAs

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir können nun die noch fehlende Richtung dieser Behauptung zeigen:

Für jede reguläre Grammatik G gibt es einen NFA $\mathcal{M}_G$, welcher die selbe Sprache akzeptiert (d.h., $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$).

Für $G = \langle V, \Sigma, P, \mathsf{S} \rangle$ ergibt sich $\mathcal{M}_G = \langle Q, \Sigma, \delta, Q_0, F \rangle$ wie folgt:

- $Q := V \cup \{q_f\}$
- $Q_0 := \{\mathsf{S}\}$

Von regulären Grammatiken zu NFAs

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir können nun die noch fehlende Richtung dieser Behauptung zeigen:

Für jede reguläre Grammatik G gibt es einen NFA $\mathcal{M}_G$, welcher die selbe Sprache akzeptiert (d.h., $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$).

Für $G = \langle V, \Sigma, P, \mathsf{S} \rangle$ ergibt sich $\mathcal{M}_G = \langle Q, \Sigma, \delta, Q_0, F \rangle$ wie folgt:

- $Q := V \cup \{q_f\}$
- $Q_0 := \{\mathsf{S}\}$
- $F := \{q_f\} \cup \{\mathsf{A} \in V \mid \mathsf{A} \to \epsilon \in P\}$

Von regulären Grammatiken zu NFAs

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir können nun die noch fehlende Richtung dieser Behauptung zeigen:

Für jede reguläre Grammatik G gibt es einen NFA $\mathcal{M}_G$, welcher die selbe Sprache akzeptiert (d.h., $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$).

Für $G = \langle V, \Sigma, P, \mathsf{S}\rangle$ ergibt sich $\mathcal{M}_G = \langle Q, \Sigma, \delta, Q_0, F\rangle$ wie folgt:

- $Q := V \cup \{q_f\}$
- $Q_0 := \{\mathsf{S}\}$
- $F := \{q_f\} \cup \{\mathsf{A} \in V \mid \mathsf{A} \to \epsilon \in P\}$
- $\delta(\mathsf{A}, \mathsf{c}) := \{\mathsf{B} \mid \mathsf{A} \to \mathsf{cB} \in P\} \cup \{q_f \mid \mathsf{A} \to \mathsf{c} \in P\}$

Beispiel

Wir betrachten eine reguläre Grammatik mit den folgenden sechs Regeln:

$$S \to \mathtt{1} \mid \mathtt{0}A$$

$$A \to \mathtt{0} \mid \mathtt{1} \mid \mathtt{1}A \mid \epsilon$$

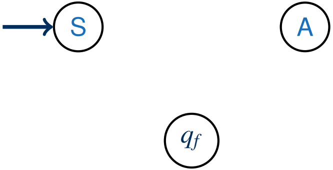
Entsprechender NFA:

Beispiel

Wir betrachten eine reguläre Grammatik mit den folgenden sechs Regeln:

$$S \to \mathtt{1} \mid \mathtt{0}A$$
$$A \to \mathtt{0} \mid \mathtt{1} \mid \mathtt{1}A \mid \epsilon$$

Entsprechender NFA:

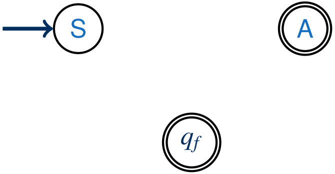


Beispiel

Wir betrachten eine reguläre Grammatik mit den folgenden sechs Regeln:

$$S \to \mathtt{1} \mid \mathtt{0}A$$
$$A \to \mathtt{0} \mid \mathtt{1} \mid \mathtt{1}A \mid \epsilon$$

Entsprechender NFA:



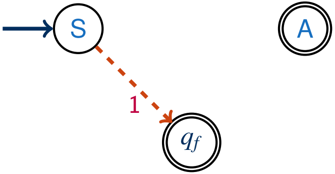
Beispiel

Wir betrachten eine reguläre Grammatik mit den folgenden sechs Regeln:

$$S \to 1 \mid 0A$$

$$A \to 0 \mid 1 \mid 1A \mid \epsilon$$

Entsprechender NFA:



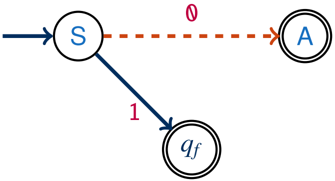
Beispiel

Wir betrachten eine reguläre Grammatik mit den folgenden sechs Regeln:

$$S \to \mathtt{1} \mid \mathtt{0}A$$

$$A \to \mathtt{0} \mid \mathtt{1} \mid \mathtt{1}A \mid \epsilon$$

Entsprechender NFA:

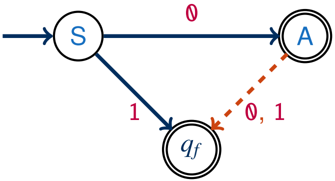


Beispiel

Wir betrachten eine reguläre Grammatik mit den folgenden sechs Regeln:

$$S \to 1 \mid 0A$$
$$A \to 0 \mid 1 \mid 1A \mid \epsilon$$

Entsprechender NFA:



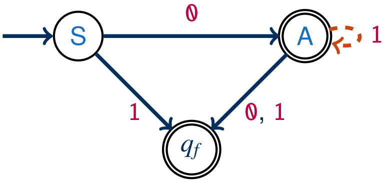
Beispiel

Wir betrachten eine reguläre Grammatik mit den folgenden sechs Regeln:

$$S \to 1 \mid 0A$$

$$A \to 0 \mid 1 \mid 1A \mid \epsilon$$

Entsprechender NFA:



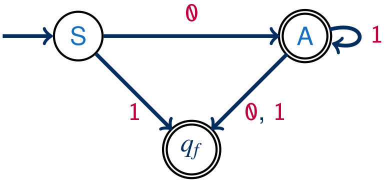
Beispiel

Wir betrachten eine reguläre Grammatik mit den folgenden sechs Regeln:

$$S \to 1 \mid 0A$$

$$A \to 0 \mid 1 \mid 1A \mid \epsilon$$

Entsprechender NFA:



Dargestellte Sprache: $\{1\} \cup (\{0\} \circ \{1\}^* \circ \{\epsilon, 0\})$

Korrektheit

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir behaupten, dass $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$, d.h. für jedes Wort $w \in \Sigma^*$ soll gelten:
$w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$.

Korrektheit

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir behaupten, dass $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$, d.h. für jedes Wort $w \in \Sigma^*$ soll gelten:
$w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$.

Der Sonderfall $w = \epsilon$ ist ziemlich einfach:

Korrektheit

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir behaupten, dass $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$, d.h. für jedes Wort $w \in \Sigma^*$ soll gelten: $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$.

Der Sonderfall $w = \epsilon$ ist ziemlich einfach:

$\epsilon \in \mathbf{L}(G)$ gdw. $\mathsf{S} \to \epsilon \in P$

Korrektheit

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir behaupten, dass $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$, d.h. für jedes Wort $w \in \Sigma^*$ soll gelten: $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$.

Der Sonderfall $w = \epsilon$ ist ziemlich einfach:

$$\begin{aligned} \epsilon \in \mathbf{L}(G) \quad &\text{gdw.} \quad \mathsf{S} \to \epsilon \in P \\ &\text{gdw.} \quad \mathsf{S} \in F \end{aligned}$$

Korrektheit

Satz: Die Klasse der Sprachen, die durch DFAs oder NFAs erkannt werden können, ist genau die Klasse der regulären Sprachen.

Beweis: Wir behaupten, dass $\mathbf{L}(G) = \mathbf{L}(\mathcal{M}_G)$, d.h. für jedes Wort $w \in \Sigma^*$ soll gelten: $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$.

Der Sonderfall $w = \epsilon$ ist ziemlich einfach:

$$\begin{aligned} \epsilon \in \mathbf{L}(G) \quad & \text{gdw.} \quad \mathsf{S} \to \epsilon \in P \\ & \text{gdw.} \quad \mathsf{S} \in F \\ & \text{gdw.} \quad \epsilon \in \mathbf{L}(\mathcal{M}_G) \end{aligned}$$

$\mathbf{L}(G) \subseteq \mathbf{L}(\mathcal{M}_G)$

Wir zeigen noch $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$ für den Fall $|w| \geq 1$.

„⇒“ Angenommen $w \in \mathbf{L}(G)$ mit $w = \mathtt{a}_1 \cdots \mathtt{a}_n$ und $n \geq 1$.

$\mathbf{L}(G) \subseteq \mathbf{L}(\mathcal{M}_G)$

Wir zeigen noch $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$ für den Fall $|w| \geq 1$.

„$\Rightarrow$“ Angenommen $w \in \mathbf{L}(G)$ mit $w = \mathtt{a}_1 \cdots \mathtt{a}_n$ und $n \geq 1$.

Es gibt zwei mögliche Herleitungen für w:

(1) $\mathsf{S} \Rightarrow \mathtt{a}_1\mathsf{B}_1 \Rightarrow \ldots \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathsf{B}_{n\text{-}1} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathtt{a}_n$

(2) $\mathsf{S} \Rightarrow \mathtt{a}_1\mathsf{B}_1 \Rightarrow \ldots \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathsf{B}_{n\text{-}1} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathtt{a}_n\mathsf{B}_n \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_n$

$\mathbf{L}(G) \subseteq \mathbf{L}(\mathcal{M}_G)$

Wir zeigen noch $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$ für den Fall $|w| \geq 1$.

„⇒“ Angenommen $w \in \mathbf{L}(G)$ mit $w = \mathtt{a}_1 \cdots \mathtt{a}_n$ und $n \geq 1$.

Es gibt zwei mögliche Herleitungen für w:

(1) $\mathsf{S} \Rightarrow \mathtt{a}_1\mathsf{B}_1 \Rightarrow \ldots \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathsf{B}_{n\text{-}1} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathtt{a}_n$

(2) $\mathsf{S} \Rightarrow \mathtt{a}_1\mathsf{B}_1 \Rightarrow \ldots \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathsf{B}_{n\text{-}1} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathtt{a}_n\mathsf{B}_n \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_n$

In Fall (1) wurden Regeln der folgenden Form angewendet:

$$\mathsf{S} \to \mathtt{a}_1\mathsf{B}_1 \qquad \mathsf{B}_1 \to \mathtt{a}_2\mathsf{B}_2 \qquad \ldots \qquad \mathsf{B}_{n\text{-}1} \to \mathtt{a}_n$$

$\mathbf{L}(G) \subseteq \mathbf{L}(\mathcal{M}_G)$

Wir zeigen noch $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$ für den Fall $|w| \geq 1$.

„⇒“ Angenommen $w \in \mathbf{L}(G)$ mit $w = \mathtt{a}_1 \cdots \mathtt{a}_n$ und $n \geq 1$.

Es gibt zwei mögliche Herleitungen für w:

(1) $\mathsf{S} \Rightarrow \mathtt{a}_1\mathsf{B}_1 \Rightarrow \ldots \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathsf{B}_{n\text{-}1} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathtt{a}_n$

(2) $\mathsf{S} \Rightarrow \mathtt{a}_1\mathsf{B}_1 \Rightarrow \ldots \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathsf{B}_{n\text{-}1} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathtt{a}_n\mathsf{B}_n \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_n$

In Fall (1) wurden Regeln der folgenden Form angewendet:

$$\mathsf{S} \to \mathtt{a}_1\mathsf{B}_1 \qquad \mathsf{B}_1 \to \mathtt{a}_2\mathsf{B}_2 \qquad \ldots \qquad \mathsf{B}_{n\text{-}1} \to \mathtt{a}_n$$

Also hat $\mathcal{M}_G$ die folgenden Übergänge:

$$\mathsf{S} \xrightarrow{\mathtt{a}_1} \mathsf{B}_1 \qquad \mathsf{B}_1 \xrightarrow{\mathtt{a}_2} \mathsf{B}_2 \qquad \ldots \qquad \mathsf{B}_{n\text{-}1} \xrightarrow{\mathtt{a}_n} q_f$$

$\mathbf{L}(G) \subseteq \mathbf{L}(\mathcal{M}_G)$

Wir zeigen noch $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$ für den Fall $|w| \geq 1$.

„⇒“ Angenommen $w \in \mathbf{L}(G)$ mit $w = \mathtt{a}_1 \cdots \mathtt{a}_n$ und $n \geq 1$.

Es gibt zwei mögliche Herleitungen für w:

(1) $\mathsf{S} \Rightarrow \mathtt{a}_1\mathsf{B}_1 \Rightarrow \ldots \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathsf{B}_{\mathsf{n\text{-}1}} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathtt{a}_n$

(2) $\mathsf{S} \Rightarrow \mathtt{a}_1\mathsf{B}_1 \Rightarrow \ldots \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathsf{B}_{\mathsf{n\text{-}1}} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_{n-1}\mathtt{a}_n\mathsf{B}_\mathsf{n} \Rightarrow \mathtt{a}_1 \cdots \mathtt{a}_n$

In Fall (1) wurden Regeln der folgenden Form angewendet:

$$\mathsf{S} \to \mathtt{a}_1\mathsf{B}_1 \qquad \mathsf{B}_1 \to \mathtt{a}_2\mathsf{B}_2 \qquad \ldots \qquad \mathsf{B}_{\mathsf{n\text{-}1}} \to \mathtt{a}_n$$

Also hat $\mathcal{M}_G$ die folgenden Übergänge:

$$\mathsf{S} \xrightarrow{\mathtt{a}_1} \mathsf{B}_1 \qquad \mathsf{B}_1 \xrightarrow{\mathtt{a}_2} \mathsf{B}_2 \qquad \ldots \qquad \mathsf{B}_{\mathsf{n\text{-}1}} \xrightarrow{\mathtt{a}_n} q_f$$

Also ist $\mathsf{S}\mathsf{B}_1\mathsf{B}_2 \ldots \mathsf{B}_{\mathsf{n\text{-}1}} q_f$ ein akzeptierender Lauf von $\mathcal{M}_G$ und $\mathcal{M}_G$ akzeptiert das Wort w.

Fall (2) ist ähnlich, wobei der Lauf auf B_n endet und $\mathsf{B}_\mathsf{n} \in F$.

$\mathbf{L}(G) \supseteq \mathbf{L}(\mathcal{M}_G)$

Wir zeigen noch $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$ für den Fall $|w| \geq 1$.

„$\Leftarrow$“ Angenommen $w \in \mathbf{L}(\mathcal{M}_G)$ mit $w = \sigma_1 \cdots \sigma_n$ und $n \geq 1$.

$\mathbf{L}(G) \supseteq \mathbf{L}(\mathcal{M}_G)$

Wir zeigen noch $w \in \mathbf{L}(G)$ gdw. $w \in \mathbf{L}(\mathcal{M}_G)$ für den Fall $|w| \geq 1$.

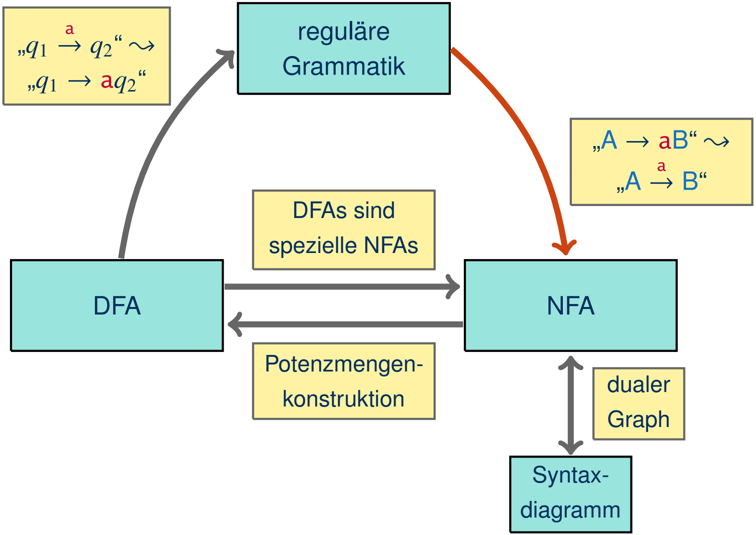
„$\Leftarrow$“ Angenommen $w \in \mathbf{L}(\mathcal{M}_G)$ mit $w = \sigma_1 \cdots \sigma_n$ und $n \geq 1$.

Beweis analog zur vorangegangenen Richtung; grob skizziert:

- w hat einen akzeptierenden Lauf in $\mathcal{M}_G$
- wir betrachten die möglichen Formen solcher Läufe
- in jedem Fall finden wir entsprechende NFA-Übergänge
- daraus ergeben sich geeignete Grammatikregeln, um w abzuleiten

$\square$

Darstellungen von Typ-3-Sprachen



Zusammenfassung und Ausblick

Nichtdeterministische endliche Automaten (NFA) vereinfachen die Modellierung, z.B. die direkte Darstellung von Syntaxdiagrammen.

Rabin/Scott: DFAs und NFAs erkennen die selben Sprachen.
(Potenzmengenkonstruktion)

DFA/NFAs erkennen genau die regulären Sprachen. (Grammatik $\leftrightarrow$ NFA)

Offene Fragen:

- Gibt es noch mehr Darstellungsformen für reguläre Sprachen?
- Was kommt heraus, wenn man Operationen auf reguläre Sprache anwendet?
- Wir haben gesehen, dass man Automaten manchmal vereinfachen kann – geht das noch besser?