SS 2005

Einführung in die Informatik IV

Ernst W. Mayr

Fakultät für Informatik
TU München

http://www14.in.tum.de/lehre/2005SS/info4/index.html.de

22. April 2005

3. Reguläre Sprachen

3.1 Deterministische endliche Automaten

Definition 21

Ein deterministischer endlicher Automat (englisch: deterministic finite automaton, kurz DFA) wird durch ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $q_0 \in Q$ ist der Startzustand.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to Q$ heißt Übergangsfunktion.

3. Reguläre Sprachen

3.1 Deterministische endliche Automaten

Definition 21

Ein deterministischer endlicher Automat (englisch: deterministic finite automaton, kurz DFA) wird durch ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $q_0 \in Q$ ist der Startzustand.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \rightarrow Q$ heißt Übergangsfunktion.

3. Reguläre Sprachen

3.1 Deterministische endliche Automaten

Definition 21

Ein deterministischer endlicher Automat (englisch: deterministic finite automaton, kurz DFA) wird durch ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $q_0 \in Q$ ist der Startzustand.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to Q$ heißt Übergangsfunktion.

3. Reguläre Sprachen

3.1 Deterministische endliche Automaten

Definition 21

Ein deterministischer endlicher Automat (englisch: deterministic finite automaton, kurz DFA) wird durch ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $q_0 \in Q$ ist der Startzustand.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to Q$ heißt Übergangsfunktion.

3. Reguläre Sprachen

3.1 Deterministische endliche Automaten

Definition 21

Ein deterministischer endlicher Automat (englisch: deterministic finite automaton, kurz DFA) wird durch ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $q_0 \in Q$ ist der Startzustand.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to Q$ heißt Übergangsfunktion.

3. Reguläre Sprachen

3.1 Deterministische endliche Automaten

Definition 21

Ein deterministischer endlicher Automat (englisch: deterministic finite automaton, kurz DFA) wird durch ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $q_0 \in Q$ ist der Startzustand.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to Q$ heißt Übergangsfunktion.

3. Reguläre Sprachen

3.1 Deterministische endliche Automaten

Definition 21

Ein deterministischer endlicher Automat (englisch: deterministic finite automaton, kurz DFA) wird durch ein 5-Tupel $M = (Q, \Sigma, \delta, q_0, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $q_0 \in Q$ ist der Startzustand.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to Q$ heißt Übergangsfunktion.

Die von M akzeptierte Sprache ist

$$L(M) := \{w \in \Sigma^*;\ \hat{\delta}(q_0, w) \in F\}\ ,$$

wobei $\hat{\delta} : Q \times \Sigma^* \to Q$ induktiv definiert ist durch

$$\begin{array}{rcll} \hat{\delta}(q,\epsilon) & = & q & \text{für alle } q \in Q \\ \hat{\delta}(q,ax) & = & \hat{\delta}(\delta(q,a),x) & \text{für alle } q \in Q, a \in \Sigma \\ & & & \qquad\qquad \text{und } x \in \Sigma^* \end{array}$$

Bemerkung: Endliche Automaten können durch (gerichtete und markierte) Zustandsgraphen veranschaulicht werden:

- Knoten $\hat{=}$ Zuständen
- Kanten $\hat{=}$ Übergängen

  genauer: die mit $a \in \Sigma$ markierte Kante (u, v) entspricht $\delta(u, a) = v$

Der Anfangszustand wird durch einen Pfeil, Endzustände werden durch doppelte Kreise gekennzeichnet.

Die von M akzeptierte Sprache ist

$$L(M) := \{w \in \Sigma^*;\ \hat{\delta}(q_0, w) \in F\}\ ,$$

wobei $\hat{\delta} : Q \times \Sigma^* \to Q$ induktiv definiert ist durch

$$\begin{array}{rcll} \hat{\delta}(q, \epsilon) & = & q & \text{für alle } q \in Q \\ \hat{\delta}(q, ax) & = & \hat{\delta}(\delta(q, a), x) & \text{für alle } q \in Q, a \in \Sigma \\ & & & \qquad\text{und } x \in \Sigma^* \end{array}$$

Bemerkung: Endliche Automaten können durch (gerichtete und markierte) Zustandsgraphen veranschaulicht werden:

- Knoten $\widehat{=}$ Zuständen
- Kanten $\widehat{=}$ Übergängen
- genauer: die mit $a \in \Sigma$ markierte Kante (u, v) entspricht: $\delta(u, a) = v$

Der Anfangszustand wird durch einen Pfeil, Endzustände werden durch doppelte Kreise gekennzeichnet.

Die von M akzeptierte Sprache ist

$$L(M) := \{w \in \Sigma^*;\ \hat{\delta}(q_0, w) \in F\}\ ,$$

wobei $\hat{\delta} : Q \times \Sigma^* \to Q$ induktiv definiert ist durch

$$\begin{array}{rcll} \hat{\delta}(q, \epsilon) & = & q & \text{für alle } q \in Q \\ \hat{\delta}(q, ax) & = & \hat{\delta}(\delta(q, a), x) & \text{für alle } q \in Q, a \in \Sigma \\ & & & \qquad \text{und } x \in \Sigma^* \end{array}$$

Bemerkung: Endliche Automaten können durch (gerichtete und markierte) Zustandsgraphen veranschaulicht werden:

- Knoten $\widehat{=}$ Zuständen
- Kanten $\widehat{=}$ Übergängen
- genauer: die mit $a \in \Sigma$ markierte Kante (u, v) entspricht $\delta(u, a) = v$

Der Anfangszustand wird durch einen Pfeil, Endzustände werden durch doppelte Kreise gekennzeichnet.

Die von M akzeptierte Sprache ist

$$L(M) := \{w \in \Sigma^*;\ \hat{\delta}(q_0, w) \in F\}\ ,$$

wobei $\hat{\delta} : Q \times \Sigma^* \to Q$ induktiv definiert ist durch

$$\begin{array}{rcll} \hat{\delta}(q,\epsilon) & = & q & \text{für alle } q \in Q \\ \hat{\delta}(q,ax) & = & \hat{\delta}(\delta(q,a),x) & \text{für alle } q \in Q, a \in \Sigma \\ & & & \qquad\text{und } x \in \Sigma^* \end{array}$$

Bemerkung: Endliche Automaten können durch (gerichtete und markierte) Zustandsgraphen veranschaulicht werden:

- Knoten $\widehat{=}$ Zuständen
- Kanten $\widehat{=}$ Übergängen
- genauer: die mit $a \in \Sigma$ markierte Kante (u, v) entspricht $\delta(u, a) = v$

Der Anfangszustand wird durch einen Pfeil, Endzustände werden durch doppelte Kreise gekennzeichnet.

Die von M akzeptierte Sprache ist

$$L(M) := \{w \in \Sigma^*;\ \hat{\delta}(q_0, w) \in F\}\ ,$$

wobei $\hat{\delta} : Q \times \Sigma^* \to Q$ induktiv definiert ist durch

$$\begin{array}{rcll} \hat{\delta}(q, \epsilon) & = & q & \text{für alle } q \in Q \\ \hat{\delta}(q, ax) & = & \hat{\delta}(\delta(q, a), x) & \text{für alle } q \in Q, a \in \Sigma \\ & & & \quad\text{und } x \in \Sigma^* \end{array}$$

Bemerkung: Endliche Automaten können durch (gerichtete und markierte) Zustandsgraphen veranschaulicht werden:

- Knoten $\widehat{=}$ Zuständen
- Kanten $\widehat{=}$ Übergängen
- genauer: die mit $a \in \Sigma$ markierte Kante (u, v) entspricht $\delta(u, a) = v$

Der Anfangszustand wird durch einen Pfeil, Endzustände werden durch doppelte Kreise gekennzeichnet.

Die von M akzeptierte Sprache ist

$$L(M) := \{w \in \Sigma^*;\ \hat{\delta}(q_0, w) \in F\}\ ,$$

wobei $\hat{\delta} : Q \times \Sigma^* \to Q$ induktiv definiert ist durch

$$\begin{array}{rcll} \hat{\delta}(q, \epsilon) & = & q & \text{für alle } q \in Q \\ \hat{\delta}(q, ax) & = & \hat{\delta}(\delta(q, a), x) & \text{für alle } q \in Q, a \in \Sigma \\ & & & \quad\text{und } x \in \Sigma^* \end{array}$$

Bemerkung: Endliche Automaten können durch (gerichtete und markierte) Zustandsgraphen veranschaulicht werden:

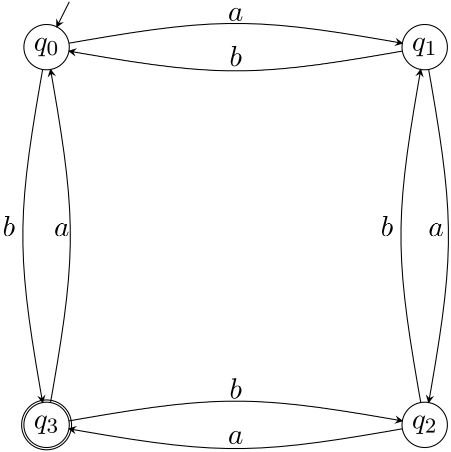
- Knoten $\widehat{=}$ Zuständen
- Kanten $\widehat{=}$ Übergängen
- genauer: die mit $a \in \Sigma$ markierte Kante (u, v) entspricht $\delta(u, a) = v$

Der Anfangszustand wird durch einen Pfeil, Endzustände werden durch doppelte Kreise gekennzeichnet.

Beispiel 22

Sei $M = (Q, \Sigma, \delta, q_0, F)$, wobei

$$
\begin{aligned}
Q &= \{q_0, q_1, q_2, q_3\} \\
\Sigma &= \{a, b\} \\
F &= \{q_3\} \\
\delta(q_0, a) &= q_1 \\
\delta(q_0, b) &= q_3 \\
\delta(q_1, a) &= q_2 \\
\delta(q_1, b) &= q_0 \\
\delta(q_2, a) &= q_3 \\
\delta(q_2, b) &= q_1 \\
\delta(q_3, a) &= q_0 \\
\delta(q_3, b) &= q_2
\end{aligned}
$$



Satz 23

Ist $M = (Q, \Sigma, \delta, q_0, F)$ ein deterministischer endlicher Automat, so ist die durch

$$P \quad := \quad \{q \to aq';\ \delta(q,a) = q'\} \cup \{q \to a;\ \delta(q,a) \in F\}$$

gegebene Grammatik $G = (Q, \Sigma, P, q_0)$ regulär.

Beweis:

Offensichtlich! $\square$

Satz 23

Ist $M = (Q, \Sigma, \delta, q_0, F)$ *ein deterministischer endlicher Automat, so ist die durch*

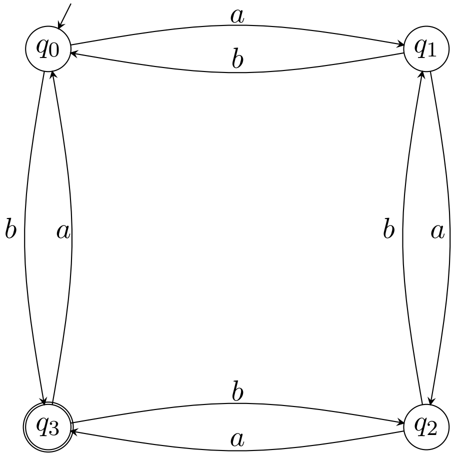
$$P \;:=\; \{q \to aq';\ \delta(q,a) = q'\} \cup \{q \to a;\ \delta(q,a) \in F\}$$

gegebene Grammatik $G = (Q, \Sigma, P, q_0)$ *regulär.*

Beweis:

Offensichtlich! $\square$

Beispiel 24

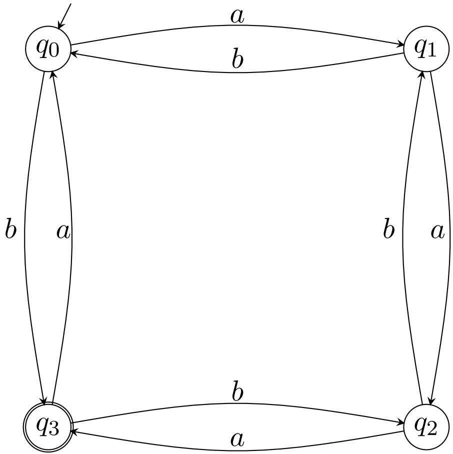


Produktionen:

$$
\begin{array}{llllll}
q_0 & \rightarrow & aq_1 & q_0 & \rightarrow & bq_3 \\
q_1 & \rightarrow & aq_2 & q_1 & \rightarrow & bq_0 \\
q_2 & \rightarrow & aq_3 & q_2 & \rightarrow & bq_1 \\
q_3 & \rightarrow & aq_0 & q_3 & \rightarrow & bq_2 \\
q_2 & \rightarrow & a & q_0 & \rightarrow & b
\end{array}
$$

$$
\begin{array}{lll}
q_0 & \rightarrow & aq_1 \rightarrow abq_0 \\
 & \rightarrow & abaq_1 \rightarrow abaaq_2 \\
 & \rightarrow & abaaa \in L(G)
\end{array}
$$

Beispiel 24



Produktionen:

$$
\begin{array}{lllllll}
q_0 & \rightarrow & aq_1 & q_0 & \rightarrow & bq_3 \\
q_1 & \rightarrow & aq_2 & q_1 & \rightarrow & bq_0 \\
q_2 & \rightarrow & aq_3 & q_2 & \rightarrow & bq_1 \\
q_3 & \rightarrow & aq_0 & q_3 & \rightarrow & bq_2 \\
q_2 & \rightarrow & a & q_0 & \rightarrow & b
\end{array}
$$

$$
\begin{array}{lll}
q_0 & \rightarrow & aq_1 \rightarrow abq_0 \\
 & \rightarrow & abaq_1 \rightarrow abaaq_2 \\
 & \rightarrow & abaaa \in L(G)
\end{array}
$$

Satz 25

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein endlicher deterministischer Automat. Dann gilt für die soeben konstruierte reguläre Grammatik G

$$L(G) = L(M)\ .$$

Beweis:

Sei $w = a_1a_2\cdots a_n \in \Sigma^*$. Dann gilt gemäß Konstruktion:

$$w \in L(M)$$

$$\Leftrightarrow w \in L(G)$$

Satz 25

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein endlicher deterministischer Automat. Dann gilt für die soeben konstruierte reguläre Grammatik G

$$L(G) = L(M)\ .$$

Beweis:

Sei $w = a_1 a_2 \cdots a_n \in \Sigma^*$. Dann gilt gemäß Konstruktion:

$$\begin{array}{ll}
 & w \in L(M) \\
\Leftrightarrow & \exists q_0, q_1, \ldots, q_n \in Q\colon\ q_0 \text{ Startzustand von } M, \\
 & \forall i = 0, \ldots, n-1\colon\ \delta(q_i, a_{i+1}) = q_{i+1},\ q_n \in F \\
\Leftrightarrow & \exists q_0, q_1, \ldots, q_n \in V\colon\ q_0 \text{ Startsymbol von } G \\
 & q_0 \to a_1 q_1 \to a_1 a_2 q_2 \to \cdots \to a_1 \cdots a_{n-1} q_{n-1} \to \\
 & \to a_1 \cdots a_{n-1} a_n \\
\Leftrightarrow & w \in L(G)
\end{array}$$

$\square$

Satz 25

Sei $M = (Q, \Sigma, \delta, q_0, F)$ *ein endlicher deterministischer Automat. Dann gilt für die soeben konstruierte reguläre Grammatik* G

$$L(G) = L(M)\ .$$

Beweis:

Sei $w = a_1 a_2 \cdots a_n \in \Sigma^*$. Dann gilt gemäß Konstruktion:

$$\begin{aligned}
& w \in L(M) \\
\Leftrightarrow\quad & \exists q_0, q_1, \ldots, q_n \in Q\colon\ q_0 \text{ Startzustand von } M, \\
& \forall i = 0, \ldots, n-1\colon\ \delta(q_i, a_{i+1}) = q_{i+1},\ q_n \in F \\
\Leftrightarrow\quad & \exists q_0, q_1, \ldots, q_n \in V\colon\ q_0 \text{ Startsymbol von } G \\
& q_0 \to a_1 q_1 \to a_1 a_2 q_2 \to \cdots \to a_1 \cdots a_{n-1} q_{n-1} \to \\
& \to a_1 \cdots a_{n-1} a_n \\
\Leftrightarrow\quad & w \in L(G)
\end{aligned}$$

$\square$

Satz 25

Sei $M = (Q, \Sigma, \delta, q_0, F)$ *ein endlicher deterministischer Automat. Dann gilt für die soeben konstruierte reguläre Grammatik* G

$$L(G) = L(M)\ .$$

Beweis:

Sei $w = a_1 a_2 \cdots a_n \in \Sigma^*$. Dann gilt gemäß Konstruktion:

$$\begin{aligned}
& w \in L(M) \\
\Leftrightarrow\quad & \exists q_0, q_1, \ldots, q_n \in Q\text{: } q_0 \text{ Startzustand von } M, \\
& \forall i = 0, \ldots, n-1\text{: } \delta(q_i, a_{i+1}) = q_{i+1},\ q_n \in F \\
\Leftrightarrow\quad & \exists q_0, q_1, \ldots, q_n \in V\text{: } q_0 \text{ Startsymbol von } G \\
& q_0 \to a_1 q_1 \to a_1 a_2 q_2 \to \cdots \to a_1 \cdots a_{n-1} q_{n-1} \to \\
& \to a_1 \cdots a_{n-1} a_n \\
\Leftrightarrow\quad & w \in L(G)
\end{aligned}$$

□

Satz 25

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein endlicher deterministischer Automat. Dann gilt für die soeben konstruierte reguläre Grammatik G

$$L(G) = L(M)\ .$$

Beweis:

Sei $w = a_1 a_2 \cdots a_n \in \Sigma^*$. Dann gilt gemäß Konstruktion:

$$\begin{aligned}
& w \in L(M) \\
\Leftrightarrow\quad & \exists q_0, q_1, \ldots, q_n \in Q\colon\ q_0 \text{ Startzustand von } M, \\
& \forall i = 0, \ldots, n-1\colon\ \delta(q_i, a_{i+1}) = q_{i+1},\ q_n \in F \\
\Leftrightarrow\quad & \exists q_0, q_1, \ldots, q_n \in V\colon\ q_0 \text{ Startsymbol von } G \\
& q_0 \to a_1 q_1 \to a_1 a_2 q_2 \to \cdots \to a_1 \cdots a_{n-1} q_{n-1} \to \\
& \to a_1 \cdots a_{n-1} a_n \\
\Leftrightarrow\quad & w \in L(G)
\end{aligned}$$

$\square$

Satz 25

Sei $M = (Q, \Sigma, \delta, q_0, F)$ ein endlicher deterministischer Automat. Dann gilt für die soeben konstruierte reguläre Grammatik G

$$L(G) = L(M)\ .$$

Beweis:

Sei $w = a_1 a_2 \cdots a_n \in \Sigma^*$. Dann gilt gemäß Konstruktion:

$$\begin{array}{ll}
 & w \in L(M) \\
\Leftrightarrow & \exists q_0, q_1, \ldots, q_n \in Q\colon\ q_0 \text{ Startzustand von } M, \\
 & \forall i = 0, \ldots, n-1\colon\ \delta(q_i, a_{i+1}) = q_{i+1},\ q_n \in F \\
\Leftrightarrow & \exists q_0, q_1, \ldots, q_n \in V\colon\ q_0 \text{ Startsymbol von } G \\
 & q_0 \to a_1 q_1 \to a_1 a_2 q_2 \to \cdots \to a_1 \cdots a_{n-1} q_{n-1} \to \\
 & \to a_1 \cdots a_{n-1} a_n \\
\Leftrightarrow & w \in L(G)
\end{array}$$

$\square$

Satz 25

Sei $M = (Q, \Sigma, \delta, q_0, F)$ *ein endlicher deterministischer Automat. Dann gilt für die soeben konstruierte reguläre Grammatik* G

$$L(G) = L(M) \,.$$

Beweis:

Sei $w = a_1 a_2 \cdots a_n \in \Sigma^*$. Dann gilt gemäß Konstruktion:

$$\begin{array}{rl}
& w \in L(M) \\
\Leftrightarrow & \exists q_0, q_1, \ldots, q_n \in Q\colon\ q_0 \text{ Startzustand von } M, \\
& \forall i = 0, \ldots, n-1\colon\ \delta(q_i, a_{i+1}) = q_{i+1},\ q_n \in F \\
\Leftrightarrow & \exists q_0, q_1, \ldots, q_n \in V\colon\ q_0 \text{ Startsymbol von } G \\
& q_0 \rightarrow a_1 q_1 \rightarrow a_1 a_2 q_2 \rightarrow \cdots \rightarrow a_1 \cdots a_{n-1} q_{n-1} \rightarrow \\
& \rightarrow a_1 \cdots a_{n-1} a_n \\
\Leftrightarrow & w \in L(G)
\end{array}$$

$\square$

3.2 Nichtdeterministische endliche Automaten

Definition 26

Ein nichtdeterministischer endlicher Automat (englisch: nondeterministic finite automaton, kurz NFA) wird durch ein 5-Tupel $N = (Q, \Sigma, \delta, S, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $S \subseteq Q$ ist die Menge der Startzustände.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to \mathcal{P}(Q) \setminus \{\emptyset\}$ heißt Übergangsrelation.

3.2 Nichtdeterministische endliche Automaten

Definition 26

Ein nichtdeterministischer endlicher Automat (englisch: nondeterministic finite automaton, kurz NFA) wird durch ein 5-Tupel $N = (Q, \Sigma, \delta, S, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $S \subseteq Q$ ist die Menge der Startzustände.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to \mathcal{P}(Q) \setminus \{\emptyset\}$ heißt Übergangsrelation.

3.2 Nichtdeterministische endliche Automaten

Definition 26

Ein nichtdeterministischer endlicher Automat (englisch: nondeterministic finite automaton, kurz NFA) wird durch ein 5-Tupel $N = (Q, \Sigma, \delta, S, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $S \subseteq Q$ ist die Menge der Startzustände.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to \mathcal{P}(Q) \setminus \{\emptyset\}$ heißt Übergangsrelation.

3.2 Nichtdeterministische endliche Automaten

Definition 26

Ein nichtdeterministischer endlicher Automat (englisch: nondeterministic finite automaton, kurz NFA) wird durch ein 5-Tupel $N = (Q, \Sigma, \delta, S, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $S \subseteq Q$ ist die Menge der Startzustände.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to \mathcal{P}(Q) \setminus \{\emptyset\}$ heißt Übergangsrelation.

3.2 Nichtdeterministische endliche Automaten

Definition 26

Ein nichtdeterministischer endlicher Automat (englisch: nondeterministic finite automaton, kurz NFA) wird durch ein 5-Tupel $N = (Q, \Sigma, \delta, S, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $S \subseteq Q$ ist die Menge der Startzustände.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to \mathcal{P}(Q) \setminus \{\emptyset\}$ heißt Übergangsrelation.

3.2 Nichtdeterministische endliche Automaten

Definition 26

Ein nichtdeterministischer endlicher Automat (englisch: nondeterministic finite automaton, kurz NFA) wird durch ein 5-Tupel $N = (Q, \Sigma, \delta, S, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $S \subseteq Q$ ist die Menge der Startzustände.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to \mathcal{P}(Q) \setminus \{\emptyset\}$ heißt Übergangsrelation.

3.2 Nichtdeterministische endliche Automaten

Definition 26

Ein nichtdeterministischer endlicher Automat (englisch: nondeterministic finite automaton, kurz NFA) wird durch ein 5-Tupel $N = (Q, \Sigma, \delta, S, F)$ beschrieben, das folgende Bedingungen erfüllt:

1. Q ist eine endliche Menge von Zuständen.
2. Σ ist eine endliche Menge, das Eingabealphabet, wobei $Q \cap \Sigma = \emptyset$.
3. $S \subseteq Q$ ist die Menge der Startzustände.
4. $F \subseteq Q$ ist die Menge der Endzustände.
5. $\delta : Q \times \Sigma \to \mathcal{P}(Q) \setminus \{\emptyset\}$ heißt Übergangsrelation.

Die von N akzeptierte Sprache ist

$$L(N) := \{w \in \Sigma^*;\ \hat{\delta}(S, w) \cap F \neq \emptyset\}\ ,$$

wobei $\hat{\delta} : \mathcal{P}(Q) \times \Sigma^* \to \mathcal{P}(Q)$ wieder induktiv definiert ist durch

$$\begin{aligned}
\hat{\delta}(Q', \epsilon) &= Q' \quad \forall Q' \subseteq Q, Q' \neq \emptyset \\
\hat{\delta}(Q', ax) &= \bigcup_{q \in Q'} \hat{\delta}(\delta(q, a), x) \quad \forall \emptyset \neq Q' \subseteq Q, \forall a \in \Sigma, \forall x \in \Sigma^*
\end{aligned}$$

Die von N akzeptierte Sprache ist

$$L(N) := \{w \in \Sigma^*;\ \hat{\delta}(S, w) \cap F \neq \emptyset\}\ ,$$

wobei $\hat{\delta} : \mathcal{P}(Q) \times \Sigma^* \to \mathcal{P}(Q)$ wieder induktiv definiert ist durch

$$\begin{aligned} \hat{\delta}(Q', \epsilon) &= Q' \quad \forall Q' \subseteq Q, Q' \neq \emptyset \\ \hat{\delta}(Q', ax) &= \bigcup_{q \in Q'} \hat{\delta}(\delta(q, a), x) \quad \forall \emptyset \neq Q' \subseteq Q, \forall a \in \Sigma, \forall x \in \Sigma^* \end{aligned}$$

Satz 27

Ist $G = (V, \Sigma, P, S)$ *eine reguläre (rechtslineare) Grammatik, so ist* $N = (V \cup \{X\}, \Sigma, \delta, \{S\}, F)$, *mit*

$$F := \begin{cases} \{S, X\}, & \textit{falls } S \to \epsilon \in P \\ \{X\}, & \textit{sonst} \end{cases}$$

und, für alle $A, B \in V$, $a \in \Sigma$,

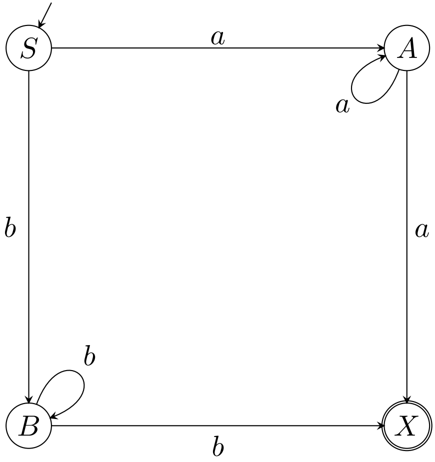
$$\begin{aligned} B \in \delta(A, a) \quad &\Longleftrightarrow \quad A \to aB \qquad \textit{und} \\ X \in \delta(A, e) \quad &\Longleftrightarrow \quad A \to a \end{aligned}$$

ein nichtdeterministischer endlicher Automat.

Beispiel 28

Produktionen:

$$
\begin{array}{lcl}
S & \to & aA \\
S & \to & bB \\
A & \to & aA \\
A & \to & a \\
B & \to & bB \\
B & \to & b
\end{array}
$$



Satz 29

Für jede von einem nichtdeterministischen endlichen Automaten akzeptierte Sprache L gibt es auch einen deterministischen endlichen Automaten M mit

$$L = L(M) .$$

Beweis:

Sei $N = (Q, \Sigma, \delta, S, F)$ ein NFA.

Definiere

1. $M' := (Q', \Sigma, \delta', q'_0, F')$
2. $Q' := \mathcal{P}(Q)$, die Potenzmenge von Q
3. $\delta'(Q'', a) := \bigcup_{q' \in Q''} \hat{\delta}(q', a)$ für alle $Q'' \subseteq Q$, $a \in \Sigma$
4. $q'_0 := S$
5. $F' := \{Q'' \subseteq Q;\ Q'' \cap F \neq \emptyset\}$

$\square$

Beweis:

Sei $N = (Q, \Sigma, \delta, S, F)$ ein NFA.

Definiere

1. $M' := (Q', \Sigma, \delta', q'_0, F')$
2. $Q' := \mathcal{P}(Q)$, die Potenzmenge von Q
3. $\delta'(Q'', a) := \bigcup_{q' \in Q''} \hat{\delta}(q', a)$ für alle $Q'' \subseteq Q$, $a \in \Sigma$
4. $q'_0 := S$
5. $F' := \{Q'' \subseteq Q;\ Q'' \cap F \neq \emptyset\}$

$\square$

Beweis:
Sei $N = (Q, \Sigma, \delta, S, F)$ ein NFA.

Definiere

1. $M' := (Q', \Sigma, \delta', q'_0, F')$
2. $Q' := \mathcal{P}(Q)$, die Potenzmenge von Q
3. $\delta'(Q'', a) := \bigcup_{q' \in Q''} \hat{\delta}(q', a)$ für alle $Q'' \subseteq Q$, $a \in \Sigma$
4. $q'_0 := S$
5. $F' := \{Q'' \subseteq Q;\ Q'' \cap F \neq \emptyset\}$

$\square$

Beweis:

Sei $N = (Q, \Sigma, \delta, S, F)$ ein NFA.

Definiere

1. $M' := (Q', \Sigma, \delta', q'_0, F')$
2. $Q' := \mathcal{P}(Q)$, die Potenzmenge von Q
3. $\delta'(Q'', a) := \bigcup_{q' \in Q''} \hat{\delta}(q', a)$ für alle $Q'' \subseteq Q$, $a \in \Sigma$
4. $q'_0 := S$
5. $F' := \{Q'' \subseteq Q;\ Q'' \cap F \neq \emptyset\}$

$\square$

Beweis:

Sei $N = (Q, \Sigma, \delta, S, F)$ ein NFA.

Definiere

1. $M' := (Q', \Sigma, \delta', q'_0, F')$
2. $Q' := \mathcal{P}(Q)$, die Potenzmenge von Q
3. $\delta'(Q'', a) := \bigcup_{q' \in Q''} \hat{\delta}(q', a)$ für alle $Q'' \subseteq Q$, $a \in \Sigma$
4. $q'_0 := S$
5. $F' := \{Q'' \subseteq Q;\ Q'' \cap F \neq \emptyset\}$

$\square$

Beweis:

Sei $N = (Q, \Sigma, \delta, S, F)$ ein NFA.

Definiere

1. $M' := (Q', \Sigma, \delta', q'_0, F')$
2. $Q' := \mathcal{P}(Q)$, die Potenzmenge von Q
3. $\delta'(Q'', a) := \bigcup_{q' \in Q''} \hat{\delta}(q', a)$ für alle $Q'' \subseteq Q$, $a \in \Sigma$
4. $q'_0 := S$
5. $F' := \{Q'' \subseteq Q;\ Q'' \cap F \neq \emptyset\}$

$\square$

Beweis:

Sei $N = (Q, \Sigma, \delta, S, F)$ ein NFA.

Definiere

1. $M' := (Q', \Sigma, \delta', q'_0, F')$
2. $Q' := \mathcal{P}(Q)$, die Potenzmenge von Q
3. $\delta'(Q'', a) := \bigcup_{q' \in Q''} \hat{\delta}(q', a)$ für alle $Q'' \subseteq Q$, $a \in \Sigma$
4. $q'_0 := S$
5. $F' := \{Q'' \subseteq Q;\ Q'' \cap F \neq \emptyset\}$

$\square$