Satz 28

Sei $M = (Q, \Sigma, \delta, q_0, F)$ *ein deterministischer endlicher Automat und*

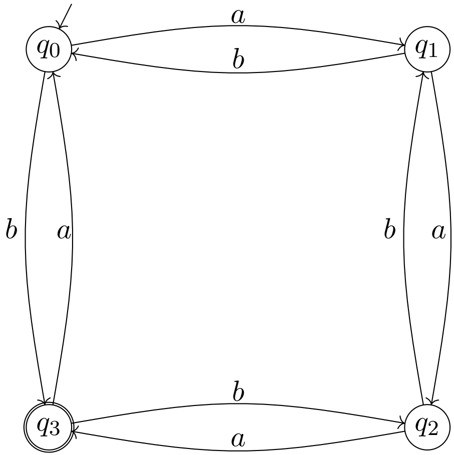
$$P \;:=\; \{q \to aq';\ \delta(q,a) = q'\} \cup \{q \to a;\ \delta(q,a) \in F\}$$

eine Menge von Produktionen, zu der wir, falls $q_0 \in F$*, noch die Produktion* $q_0 \to \epsilon$ *hinzufügen (und dann, falls nötig, nämlich wenn* q_0 *auf der rechten Seite einer Produktion vorkommt, die Monotoniebedingung wiederherstellen).*
Dann ist die Grammatik $G = (Q, \Sigma, P, q_0)$ *regulär.*

Beweis:

Offensichtlich! $\square$

Beispiel 29



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 30

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:

Der Fall $w = \epsilon$ ist klar. Sei nun $w = a_1 a_2 \cdots a_n \in \Sigma^+$. Dann gilt gemäß Konstruktion:

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

$\square$

3.2 Nichtdeterministische endliche Automaten

Definition 31

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. $\delta : Q \times \Sigma \to \mathcal{P}(Q)$ heißt Übergangsrelation.

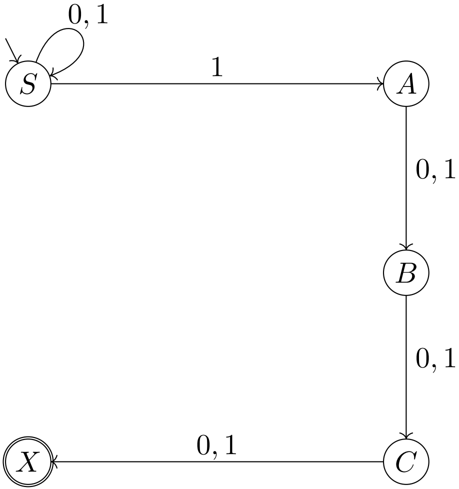
Die von N akzeptierte Sprache ist

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

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

$$\begin{array}{rcl} \hat{\delta}(Q', \epsilon) & = & Q' \quad \forall Q' \subseteq Q \\ \hat{\delta}(Q', ax) & = & \hat{\delta}(\bigcup\limits_{q \in Q'} \delta(q, a), x) \quad \forall Q' \subseteq Q, \forall a \in \Sigma, \forall x \in \Sigma^* \end{array}$$

Beispiel 32



NFA für Binärzeichenreihen, deren viertletztes Zeichen 1 ist

3.3 Äquivalenz von NFA und DFA

Satz 33

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

Also

NFA N:	Q	Σ	δ	S	F
DFA M':	2^Q	Σ	δ'	S	F'

Beweis (Forts.):

Es gilt:

$$
\begin{aligned}
w \in L(N) \quad &\Leftrightarrow \quad \hat{\delta}(S, w) \cap F \neq \emptyset \\
&\Leftrightarrow \quad \widehat{\delta'}(q'_0, w) \in F' \\
&\Leftrightarrow \quad w \in L(M').
\end{aligned}
$$

□

Der zugehörige Algorithmus zur Überführung eines NFA in einen DFA heißt Teilmengenkonstruktion, Potenzmengenkonstruktion oder Myhill-Konstruktion.

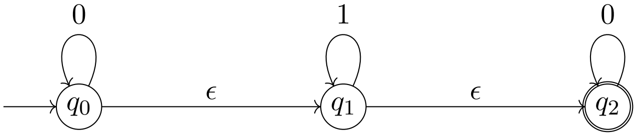
3.4 NFA's mit ϵ-Übergängen

Definition 34

Ein (nichtdeterministischer) endlicher Automat A mit ϵ-Übergängen ist ein 5-Tupel analog zur Definition des NFA mit

$$\delta : Q \times (\Sigma \uplus \{\epsilon\}) \to \mathcal{P}(Q) \; .$$

Ein ϵ-Übergang wird ausgeführt, ohne dass ein Eingabezeichen gelesen wird. Wir setzen o.B.d.A. voraus, dass A nur einen Anfangszustand hat.



Definiere für alle $a \in \Sigma$

$$\bar{\delta}(q, a) := \hat{\delta}(q, \epsilon^* a \epsilon^*) \ .$$

Falls A das leere Wort ϵ mittels ϵ-Übergängen akzeptiert, also $F \cap \hat{\delta}(q_0, \epsilon^*) \neq \emptyset$, dann setze zusätzlich

$$F := F \cup \{q_0\} \ .$$

Satz 35

$$w \in L(A) \Leftrightarrow \hat{\bar{\delta}}(S, w) \cap F \neq \emptyset \ .$$

Beweis:

Hausaufgabe! $\square$

3.5 Entfernen von ϵ-Übergängen

Satz 36

Zu jedem nichtdeterministischen endlichen Automaten A mit ϵ-Übergängen gibt es einen nichtdeterministischen endlichen Automaten A' ohne ϵ-Übergänge, so dass gilt:

$$L(A) = L(A')$$

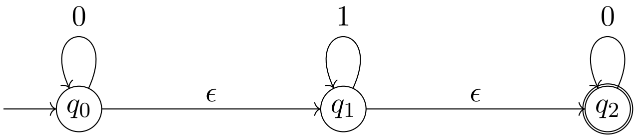
Beweis:

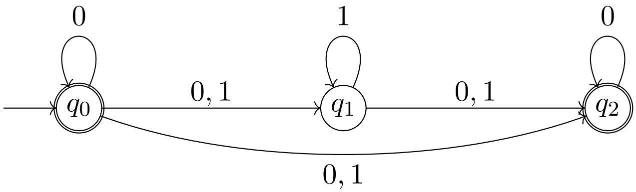
Ersetze δ durch $\bar{\delta}$ und F durch F' mit

$$F' = \begin{cases} F & \epsilon \notin L(A) \\ F \cup \{q_0\} & \epsilon \in L(A) \end{cases}$$

$\square$

Beispiel 37





3.6 Endliche Automaten und reguläre Sprachen

Satz 38

Ist $G = (V, \Sigma, P, S)$ *eine rechtslineare (also reguläre) Grammatik (o.B.d.A. sind die rechten Seiten aller Produktionen aus* $\Sigma \cup \Sigma V$*), so ist* $N = (V \uplus \{X\}, \Sigma, \delta, \{S\}, F)$*, (wobei* X *ein neues Nichtterminal-Symbol ist), mit*

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

und, für alle $A, B \in V$*,* $a \in \Sigma \cup \{\epsilon\}$*,*

$$\begin{array}{lcll} B \in \delta(A, a) & \iff & A \to aB & \text{und} \\ X \in \delta(A, a) & \iff & A \to a & \end{array}$$

ein nichtdeterministischer endlicher Automat, der genau $L(G)$ *akzeptiert.*

Beweis:

Aus der Konstruktion folgt, dass N ein NFA ist (i.A. mit ϵ-Übergängen).
Durch eine einfache Induktion über n zeigt man, dass eine Satzform

$$a_1 a_2 \cdots a_{n-1} A \text{ bzw. } a_1 a_2 \cdots a_n$$

in G genau dann ableitbar ist, wenn für die erweiterte Übergangsfunktion $\hat{\delta}$ des zu N äquivalenten NFA *ohne* ϵ-Übergänge gilt:

$$A \in \hat{\delta}(S, a_1 a_2 \cdots a_{n-1})$$

bzw.

$$X \in \hat{\delta}(S, a_1 a_2 \cdots a_n)$$

(bzw., für $n = 0$, $F \cap \hat{\delta}(S, \epsilon) \neq \emptyset$). $\square$

Zusammenfassend ergibt sich:

Satz 39

Die Klasse der regulären Sprachen (Chomsky-3-Sprachen) ist identisch mit der Klasse der Sprachen, die

- *von DFA's akzeptiert/erkannt werden,*
- *von NFA's akzeptiert werden,*
- *von NFA's mit ϵ-Übergängen akzeptiert werden.*

Beweis:

Wie soeben gezeigt. $\square$

3.7 Reguläre Ausdrücke

Reguläre Ausdrücke sollen eine kompakte Notation für spezielle Sprachen sein, wobei endliche Ausdrücke hier auch unendliche Mengen beschreiben können.

Definition 40

Reguläre Ausdrücke sind induktiv definiert durch:

1. $\emptyset$ ist ein regulärer Ausdruck.
2. ϵ ist ein regulärer Ausdruck.
3. Für jedes $a \in \Sigma$ ist a ist ein regulärer Ausdruck.
4. Wenn α und β reguläre Ausdrücke sind, dann sind auch (α), $\alpha\beta$, $(\alpha|\beta)$ (hierfür wird oft auch $(\alpha + \beta)$ geschrieben) und $(\alpha)^*$ reguläre Ausdrücke.
5. Nichts sonst ist ein regulärer Ausdruck.

Bemerkung: Ist α atomar, so schreiben wir statt $(\alpha)^*$ oft auch nur α^*.

Zu einem regulären Ausdruck γ ist die zugehörige Sprache $L(\gamma)$ induktiv definiert durch:

Definition 41

1. Falls $\gamma = \emptyset$, so gilt $L(\gamma) = \emptyset$.
2. Falls $\gamma = \epsilon$, so gilt $L(\gamma) = \{\epsilon\}$.
3. Falls $\gamma = a$, so gilt $L(\gamma) = \{a\}$.
4. Falls $\gamma = (\alpha)$, so gilt $L(\gamma) = L(\alpha)$.
5. Falls $\gamma = \alpha\beta$, so gilt
$$L(\gamma) = L(\alpha)L(\beta) = \{uv;\ u \in L(\alpha), v \in L(\beta)\}\ .$$
6. Falls $\gamma = (\alpha \mid \beta)$, so gilt
$$L(\gamma) = L(\alpha) \cup L(\beta) = \{u;\ u \in L(\alpha) \vee u \in L(\beta)\}\ .$$
7. Falls $\gamma = (\alpha)^*$, so gilt
$$L(\gamma) = L(\alpha)^* = \{u_1 u_2 \ldots u_n;\ n \in \mathbb{N}_0, u_1, \ldots, u_n \in L(\alpha)\}\ .$$

Beispiel 42

Sei das zugrunde liegende Alphabet $\Sigma = \{0, 1\}$.

- alle Wörter, die gleich 0 sind oder mit 00 enden:

$$(0 \mid (0 \mid 1)^*00)$$

- alle Wörter, die 0110 enthalten:

$$(0|1)^*0110(0|1)^*$$

- alle Wörter, die eine gerade Anzahl von 1'en enthalten:

$$(0^*10^*1)^*0^*$$

- alle Wörter, die die Binärdarstellung einer durch 3 teilbaren Zahl darstellen, also

$$0, 11, 110, 1001, 1100, 1111, 10010, \ldots$$

  Hausaufgabe!