Reguläre Sprachen

- Reguläre Sprachen (Typ-3-Sprachen)
  - haben große Bedeutung in Textverarbeitung und Programmierung (z.B. lexikalische Analyse)
  - besitzen für viele Entscheidungsprobleme effiziente Algorithmen
- Äquivalenz zu endlichen Automaten
- Äquivalenz zu regulären Ausdrücken
- Grenzen der regulären Sprachen (Pumping-Lemma)

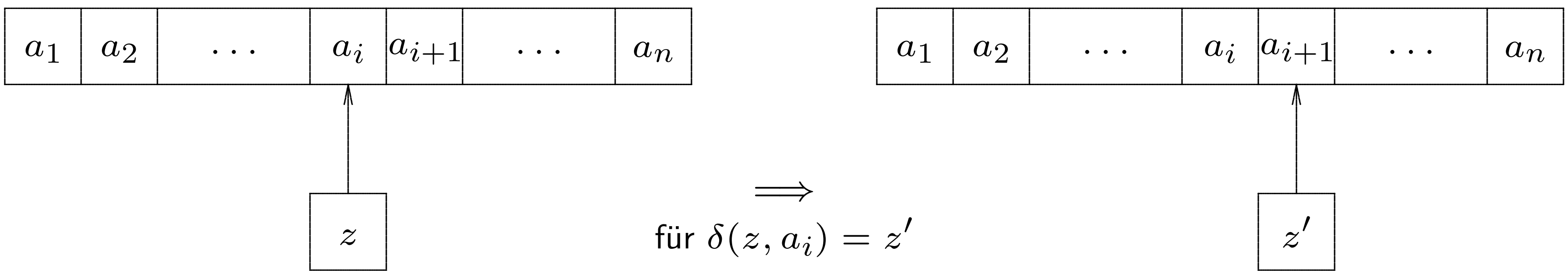
Endliche Automaten

Definition

Ein deterministischer endlicher Automat (DEA) A ist ein 5-Tupel $A = (Z, \Sigma, \delta, z_0, E)$. Dabei sind

- Z das Zustandsalphabet,
- Σ das Eingabealphabet mit $Z \cap \Sigma = \emptyset$,
- $\delta \colon Z \times \Sigma \to Z$ die Zustandsüberführungsfunktion,
- $z_0 \in Z$ der Anfangszustand und
- $E \subseteq Z$ die Menge der Endzustände.

Interpretation der Arbeitsweise des Endlichen Automaten



“Turingmaschine, die die Eingabe einmal von links nach rechts liest”

Akzeptierte Sprache des Endlichen Automaten

Definition

Sei A ein deterministischer endlicher Automat $A = (Z, \Sigma, \delta, z_0, E)$; die erweiterte Zustandsüberführungsfunktion $\hat{\delta} : Z \times \Sigma^* \to Z$ wird definiert durch

(i) $\hat{\delta}(z, \varepsilon) = z$ für alle $z \in Z$,

(ii) $\hat{\delta}(z, wa) = \delta(\hat{\delta}(z, w), a)$ für alle $z \in Z$, $a \in \Sigma$, $w \in \Sigma^*$.

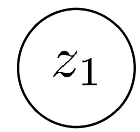
Definition

Für einen DEA $A = (Z, \Sigma, \delta, z_0, E)$ ist die von ihm akzeptierte Sprache $T(A)$ definiert durch

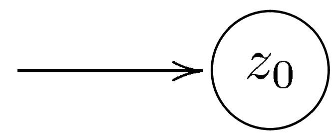
$$T(A) = \{w \in \Sigma^* \mid \hat{\delta}(z_0, w) \in E\}.$$

Überführungsgraphen

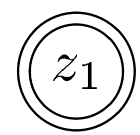
Zustand z_1:



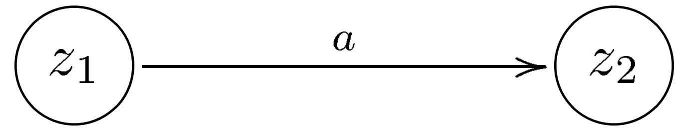
Anfangszustand z_0:



Endzustand $z_1 \in E$:



$\delta(z_1, a) = z_2$:



Endliche Automaten – Beispiel

Es sei A der deterministische endliche Automat

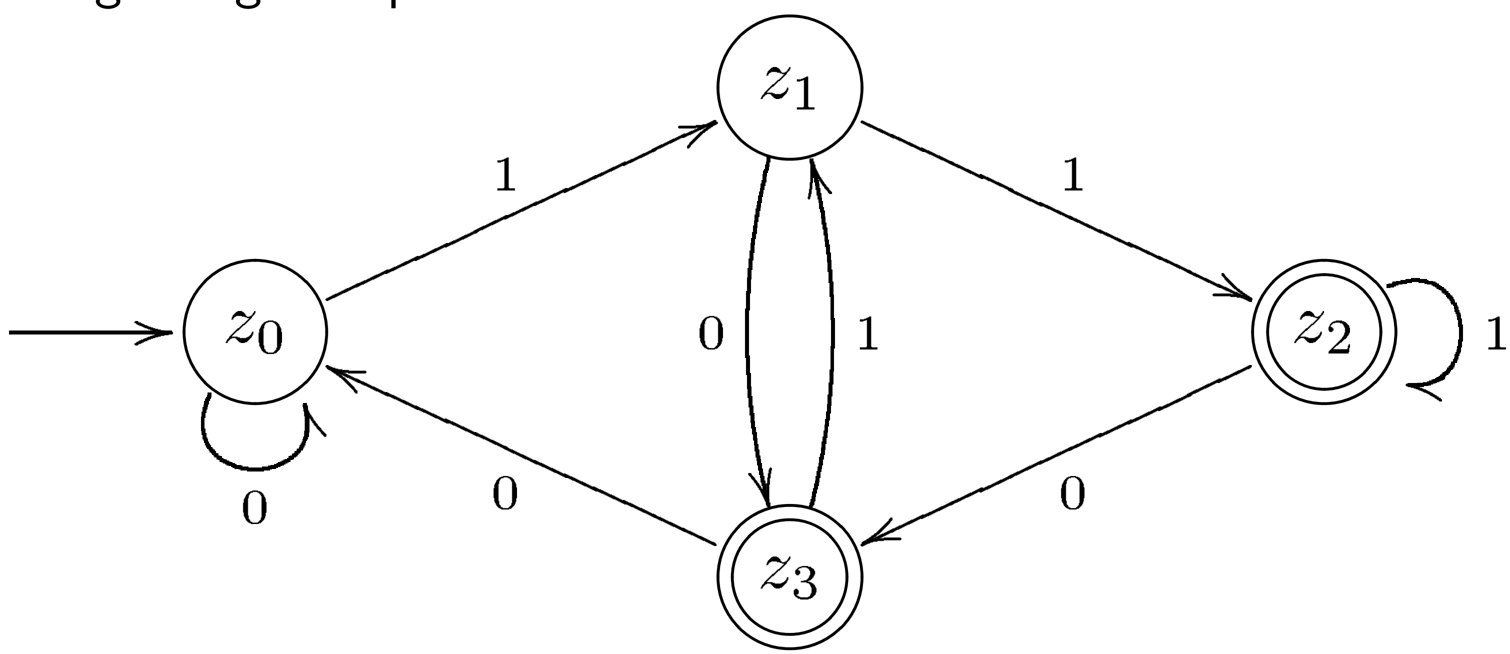
$$A = (\{z_0, z_1, z_2, z_3\}, \{0, 1\}, \delta, z_0, \{z_2, z_3\}),$$

mit der Überführungsfunktion δ, gegeben durch folgende Tabelle.

δ	z_0	z_1	z_2	z_3
0	z_0	z_3	z_3	z_0
1	z_1	z_2	z_2	z_1

DEA – Beispiel (Fortsetzung)

Der dazugehörige Graph lautet



Nichtdeterministische endliche Automaten

Definition

Ein nichtdeterministischer endlicher Automat (NEA) A ist ein 5-Tupel $A = (Z, \Sigma, \delta, z_0, E)$. Dabei sind

- Z das Zustandsalphabet,
- Σ das Eingabealphabet mit $Z \cap \Sigma = \emptyset$,
- $\delta\colon Z \times \Sigma \to 2^Z$ die Zustandsüberführungsfunktion,
- $z_0 \in Z$ der Anfangszustand und
- $E \subseteq Z$ die Menge der Endzustände.

Akzeptierte Sprache eines NEA

Definition Sei A ein NEA mit $A = (Z, \Sigma, \delta, z_0, E)$; die erweiterte Zustandsfunktion $\hat{\delta}\colon Z \times \Sigma^* \to 2^Z$ wird definiert durch

$$
\begin{aligned}
&\text{(i)} \quad \hat{\delta}(z, \varepsilon) = \{z\} \qquad \text{für alle } z \in Z, \\
&\text{(ii)} \quad \hat{\delta}(z, wa) = \bigcup_{z' \in \hat{\delta}(z,w)} \delta(z', a) \qquad \text{das heißt} \\
&\qquad = \{z'' \in Z \mid \exists z' \in Z \text{ mit } z' \in \hat{\delta}(z, w) \text{ und } z'' \in \delta(z', a)\} \\
&\qquad \text{für } z \in Z,\ a \in \Sigma,\ w \in \Sigma^*.
\end{aligned}
$$

Definition Für einen NEA $A = (Z, \Sigma, \delta, z_0, E)$ sei die von ihm akzeptierte Sprache $T(A)$ definiert durch

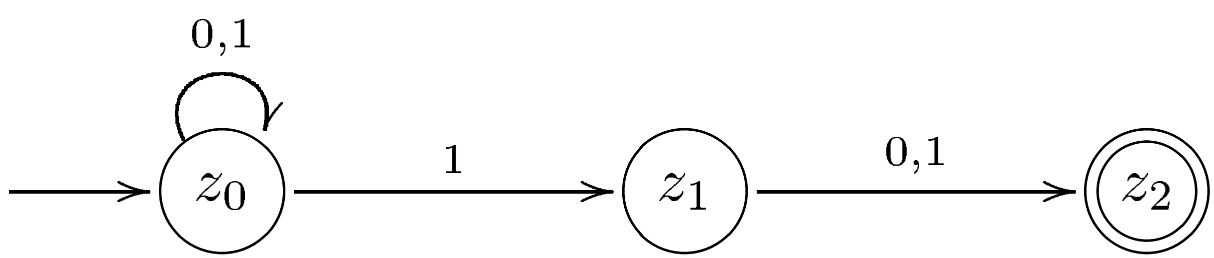
$$T(A) = \{w \in \Sigma^* \mid \hat{\delta}(z_0, w) \cap E \neq \emptyset\}.$$

Beispiel eines NEA

Beispiel Sei $A = (\{z_0, z_1, z_2\}, \{0, 1\}, \delta, z_0, \{z_2\})$ ein NEA, wobei δ durch die folgende Tabelle gegeben ist:

δ	z_0	z_1	z_2
0	$\{z_0\}$	$\{z_2\}$	$\emptyset$
1	$\{z_0, z_1\}$	$\{z_2\}$	$\emptyset$

Der dazugehörige Graph ist:



Beispiel eines NEA – Fortsetzung

$$
\begin{array}{lcll}
\hat{\delta}(z_0, 0) & = & \{z_0\} & \Rightarrow \; 0 \notin T(A) \\
\hat{\delta}(z_0, 1) & = & \{z_0, z_1\} & \Rightarrow \; 1 \notin T(A) \\
\hat{\delta}(z_0, 01) & = & \{z_0, z_1\} & \Rightarrow \; 01 \notin T(A) \\
\hat{\delta}(z_0, 11) & = & \{z_0, z_1, z_2\} & \Rightarrow \; 11 \in T(A) \\
\hat{\delta}(z_0, 001) & = & \{z_0, z_1\} & \Rightarrow \; 001 \notin T(A) \\
\hat{\delta}(z_0, 010) & = & \{z_0, z_2\} & \Rightarrow \; 010 \in T(A)
\end{array}
$$

Die akzeptierte Sprache $T(A)$ ist die Menge aller Wörter über $\{0, 1\}$, deren vorletztes Symbol eine 1 ist, d. h.

$$T(A) = \{w \in \{0,1\}^* \mid w = u1x \text{ mit } u \in \{0,1\}^*,\ x \in \{0,1\}\}.$$

Äquivalenz von NEA und DEA

Satz
Jede von einem NEA akzeptierte Sprache ist auch von einem DEA akzeptierbar.

Beweis (Potenzmengen-Konstruktion)

Aus NEA $A = (Z, \Sigma, \delta, z_0, E)$ konstruiere DEA $A' = (Z', \Sigma, \delta', z_0', E')$:

$$\begin{aligned} Z' &= 2^Z, \\ z_0' &= \{z_0\}, \\ E' &= \{z' \in Z' \mid z' \cap E \neq \emptyset\}, \\ \delta'(z', a) &= \bigcup_{z \in z'} \delta(z, a) \quad \text{für alle } z' \in Z' \text{ und } a \in \Sigma. \end{aligned}$$

Dann gilt

$$\hat{\delta}'(\{z_0\}, w) = \hat{\delta}(z_0, w) \text{ für alle } w \in \Sigma^*, \text{ d.h. } T(A) = T(A').$$

Beispiel NEA $\longrightarrow$ DEA

Wir betrachten den NEA von Folie 146.

Wir konstruieren den äquivalenten DEA $A' = (Z', \{0,1\}, \delta', \{z_0\}, E')$.

Bemerkung: Man braucht nur die Teilmengen von Z zu betrachten, die von $\{z_0\}$ erreichbar sind (sparsame Potenzmengenkonstruktion).

δ'	$\{z_0\}$	$\{z_0, z_1\}$	$\{z_0, z_2\}$	$\{z_0, z_1, z_2\}$
0	$\{z_0\}$	$\{z_0, z_2\}$	$\{z_0\}$	$\{z_0, z_2\}$
1	$\{z_0, z_1\}$	$\{z_0, z_1, z_2\}$	$\{z_0, z_1\}$	$\{z_0, z_1, z_2\}$

$Z' = \{\{z_0\}, \{z_0, z_1\}, \{z_0, z_2\}, \{z_0, z_1, z_2\}\}$

$E' = \{\{z_0, z_2\}, \{z_0, z_1, z_2\}\}$

Beispiel NEA $\longrightarrow$ DEA – Fortsetzung

Zur besseren Lesbarkeit bezeichnen wir die Zustände um:

$\{z_0\} =: q_0$, $\{z_0, z_1\} =: q_1$, $\{z_0, z_2\} =: q_2$ und $\{z_0, z_1, z_2\} =: q_3$.

$$A' = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \delta', q_0, \{q_2, q_3\})$$

mit

δ'	q_0	q_1	q_2	q_3
0	q_0	q_2	q_0	q_2
1	q_1	q_3	q_1	q_3

Endliche Automaten und reguläre Grammatiken

Satz
Sei A ein NEA. Dann ist die von A akzeptierte Sprache $T(A)$ regulär.

Beweis.

Aus NEA $A = (Z, \Sigma, \delta, z_0, E)$ konstruiere Typ-3-Grammatik $G = (V, \Sigma, P, S)$
mit $V = Z$, $S = z_0$ und

$$P = \{z_1 \to az_2 \mid z_2 \in \delta(z_1, a))\} \cup \{z_1 \to \varepsilon \mid z_1 \in E\}.$$

Es gilt $z_1 \overset{*}{\Longrightarrow}_G wz_2$ genau dann, wenn $z_2 \in \hat{\delta}(z_1, w)$.

NEA $\rightarrow$ Grammatik – Beispiel

Sei $A = (\{z_0, z_1, z_2, z_3\}, \{a, b\}, \delta, z_0, \{z_3\})$ der DEA mit der Überführungsfunktion δ:

δ	z_0	z_1	z_2	z_3
a	z_1	z_1	z_3	z_3
b	z_0	z_2	z_0	z_3

Wir konstruieren jetzt die äquivalente Typ-3-Grammatik $G = (V, \Sigma, P, S)$:

$$\begin{array}{rcl} V & = & \{z_0, z_1, z_2, z_3\}, \\ \Sigma & = & \{a, b\}, \\ S & = & z_0, \\ P & = & \{z_0 \rightarrow az_1,\ z_0 \rightarrow bz_0,\ z_1 \rightarrow az_1,\ z_1 \rightarrow bz_2\} \\ & & \cup \{z_2 \rightarrow az_3,\ z_2 \rightarrow bz_0,\ z_3 \rightarrow az_3,\ z_3 \rightarrow bz_3\} \\ & & \cup \{z_3 \rightarrow \varepsilon\}. \end{array}$$

Eine Normalform für reguläre Grammatiken

Lemma

Für jede reguläre Sprache L gibt es eine Typ-3-Grammatik $G = (V, \Sigma, P, S)$ mit $L(G) = L$, deren Regeln der folgenden Form sind:

$$A \to aB,\ A, B \in V, a \in \Sigma \textbf{ oder } A \to \varepsilon,\ A \in V$$

Reguläre Grammatik $\longrightarrow$ NEA

Satz
Sei G eine reguläre Grammatik; dann existiert ein NEA A mit $T(A) = L(G)$.

Beweis

Für eine reguläre Grammatik $G = (V, \Sigma, P, S)$ in Normalform konstruieren wir den NEA $A = (Z, \Sigma, \delta, z_0, E)$ wie folgt:

$$Z = V, \qquad z_0 = S, \qquad E = \{A \in V \mid A \to \varepsilon \in P\}$$

$$\delta(A, a) = \{B \in V \mid A \to aB \in P\} \text{ für } A \in V \text{ und } a \in \Sigma.$$