Reguläre Sprachen

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

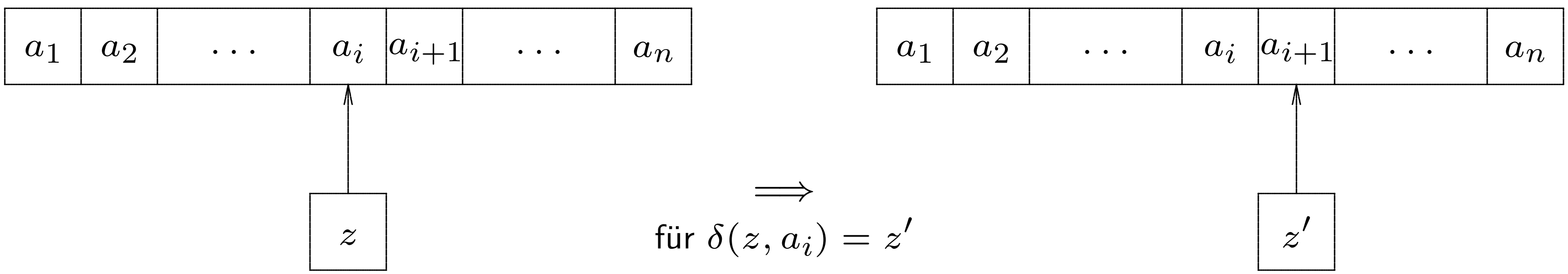
Endliche Automaten

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, welche 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 $z \in Z$, $a \in \Sigma$, $w \in \Sigma^*$.

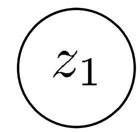
Definition

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

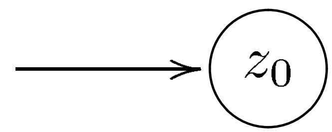
$$T(A) = \{w \in \Sigma^* \mid \hat{\delta}(z_0, w) \in E\}.$$

Überführungsgraphen

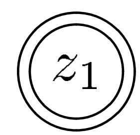
Zustand z_1:



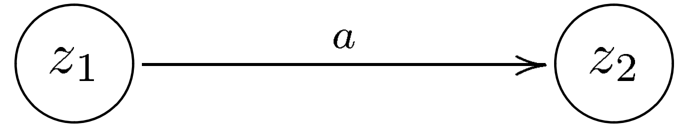
Anfangszustand z_0:



Endzustand $z_1 \in E$:



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



Endliche Automaten – Beispiel

Es sei A der deterministische endliche Automat

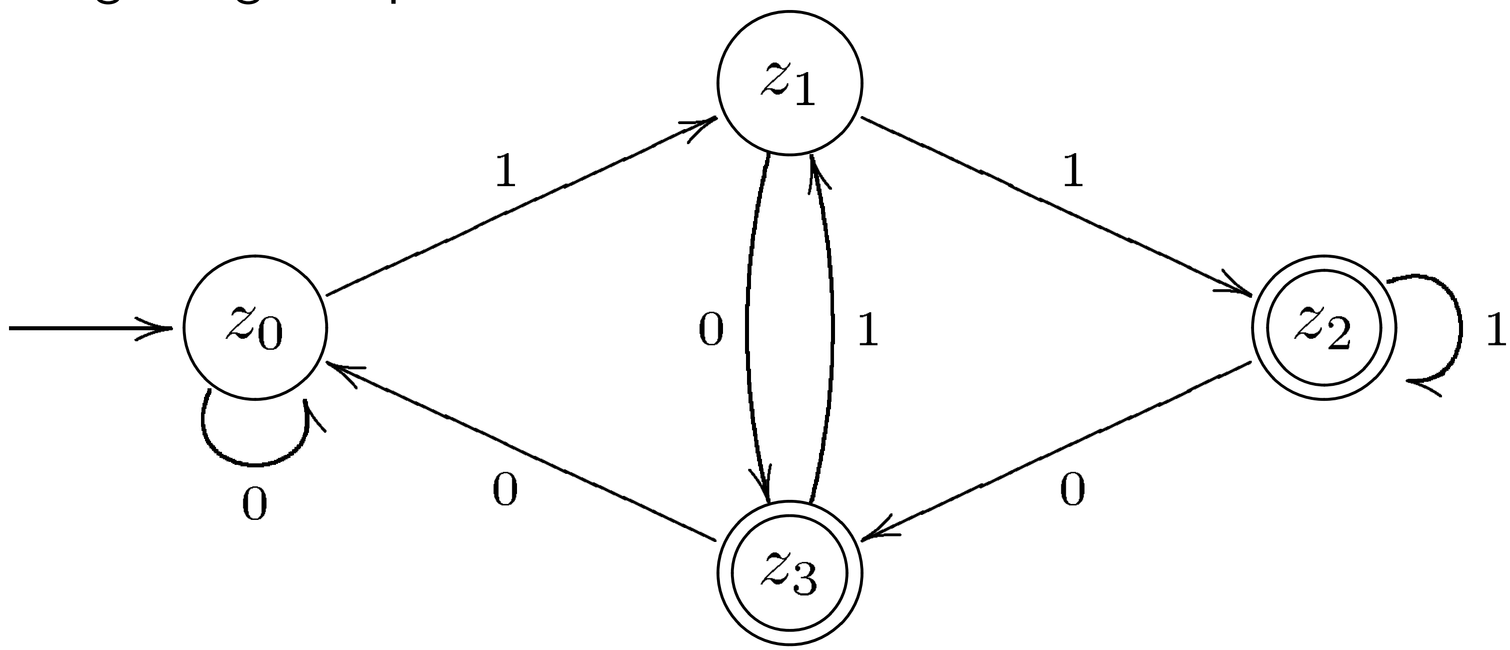
$$A = (\{z_0, z_1, z_2, z_3\}, \{0, 1\}, \delta, z_0, \{z_2, z_3\}),$$

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

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

(ii) $\hat{\delta}(z, wa) = \bigcup_{z' \in \hat{\delta}(z,w)} \delta(z', a)$ das heißt

$= \{z'' \in Z \mid \exists z' \in Z$ mit $z' \in \hat{\delta}(z, w)$ und $z'' \in \delta(z', a)\}$

für $z \in Z$, $a \in \Sigma$, $w \in \Sigma^*$.

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

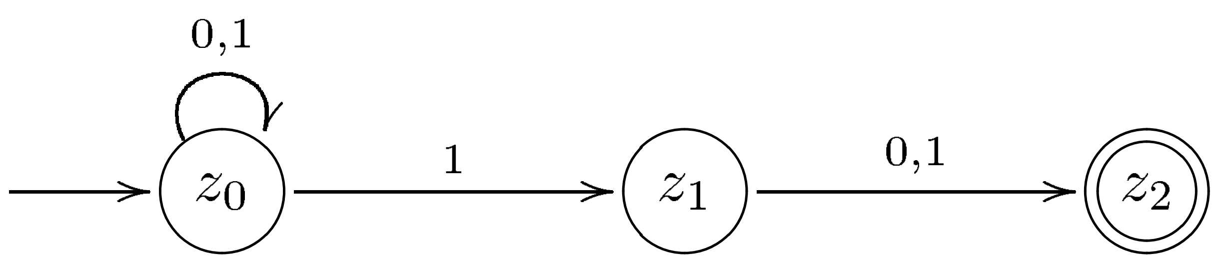
$$T(A) = \{w \in \Sigma^* \mid \hat{\delta}(z_0, w) \cap E \neq \emptyset\}.$$

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

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

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 akzeptierbare 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 nehmen den NEA von Folie 140.

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

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 (vom Typ 3).

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 $z_1 \stackrel{*}{\Longrightarrow}_G wz_2$ 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:

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

Reguläre Grammatik $\longrightarrow$ NEA

Satz

Sei G eine reguläre Grammatik, also vom Typ 3; 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:

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

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

Reguläre Ausdrücke

Definition (Reguläre Ausdrücke)

Sei Σ ein Alphabet, dann gilt:

(i) $\emptyset$ ist ein regulärer Ausdruck über Σ.

(ii) ε ist ein regulärer Ausdruck über Σ.

(iii) Für jedes $a \in \Sigma$ ist a ein regulärer Ausdruck über Σ.

(iv) Wenn α und β reguläre Ausdrücke über Σ sind, so sind auch $\alpha\beta$, $(\alpha \mid \beta)$ und $(\alpha)^*$ reguläre Ausdrücke über Σ.

Sprache eines regulären Ausdrucks

Definition (Sprache eines regulären Ausdrucks)
Sei Σ ein Alphabet und γ ein regulärer Ausdruck über Σ, dann wird die von γ beschriebene Sprache $L(\gamma) \subseteq \Sigma^*$ wie folgt definiert.

(i) Für $\gamma = \emptyset$ gilt $L(\gamma) = \emptyset$.

(ii) Für $\gamma = \varepsilon$ gilt $L(\gamma) = \{\varepsilon\}$.

(iii) Für $\gamma = a$ mit $a \in \Sigma$ gilt $L(\gamma) = \{a\}$.

(iv) Für $\gamma = \alpha\beta$ gilt $L(\gamma) = L(\alpha) \cdot L(\beta)$.

(v) Für $\gamma = (\alpha \mid \beta)$ gilt $L(\gamma) = L(\alpha) \cup L(\beta)$.

(vi) Für $\gamma = (\alpha)^*$ gilt $L(\gamma) = (L(\alpha))^*$.

Beispiel eines regulären Ausdrucks

$$
\begin{aligned}
L((0 \mid (0 \mid 1)^*00)) &= L(0) \cup L((0 \mid 1)^*00)) \\
&= L(0) \cup (L((0 \mid 1)^*0)) \cdot L(0)) \\
&= L(0) \cup ((L((0 \mid 1)^*)) \cdot L(0)) \cdot L(0)) \\
&= L(0) \cup (((L((0 \mid 1)))^* \cdot L(0)) \cdot L(0)) \\
&= L(0) \cup (((L(0) \cup L(1))^* \cdot L(0)) \cdot L(0)) \\
&= \{0\} \cup (((\{0\} \cup \{1\})^* \cdot \{0\}) \cdot \{0\}) \\
&= \{0\} \cup ((\{0,1\}^* \cdot \{0\}) \cdot \{0\}) \\
&= \{0\} \cup (\{0,1\}^* \cdot \{00\}),
\end{aligned}
$$

Das heißt, die vom Ausdruck $(0 \mid (0 \mid 1)^*00)$ beschriebene Sprache ist die Menge aller Wörter über dem Alphabet $\{0,1\}$, die gleich 0 sind oder auf 00 enden.

Äquivalenz regulärer Ausdrücke

Definition

Zwei reguläre Ausdrücke β und γ heißen äquivalent, in Zeichen $\beta \equiv \gamma$, wenn $L(\beta) = L(\gamma)$ gilt.

Beispiel:

$$((a \mid b))^* \equiv ((a \mid b)(a \mid b))^*((a \mid b) \mid \varepsilon)$$

Rechenregeln für reguläre Ausdrücke

$$(A \mid (B \mid C)) \equiv ((A \mid B) \mid C)$$
$$A(B \mid C) \equiv (AB \mid AC)$$
$$(B \mid C)A \equiv (BA \mid CA)$$
$$(A \mid B) \equiv (B \mid A)$$
$$(A \mid A) \equiv A$$
$$(A \mid \emptyset) \equiv (\emptyset \mid A) \equiv A$$

$$A\emptyset \equiv \emptyset A \equiv \emptyset$$
$$A\varepsilon \equiv \varepsilon A \equiv A$$
$$((A)^*)^* \equiv (A)^*$$
$$(\emptyset)^* \equiv \varepsilon$$
$$((A \mid \varepsilon))^* \equiv (A)^*$$
$$(A)^* \equiv (\varepsilon \mid A)A^* \equiv ((\varepsilon \mid A))^*A$$

Bemerkung zu regulären Ausdrücken

1. Wir vereinbaren, dass wir Klammern, die nicht notwendigerweise gebraucht werden, weglassen können. Zum Beispiel können wir statt $(\alpha \mid (\beta \mid \gamma))$ auch $(\alpha \mid \beta \mid \gamma)$ schreiben. Wir schreiben auch $L(\alpha \mid \beta)$ statt $L((\alpha \mid \beta))$ sowie a^* statt $(a)^*$ für $a \in \Sigma$.

2. Wir benutzen die abkürzende Schreibweise α^n für $\underbrace{\alpha\alpha\ldots\alpha}_{n\text{-mal}}$.

3. Wir benutzen die abkürzende Schreibweise α^+ für $\alpha^*\alpha$.

4. In der Literatur findet man oft auch abweichende Definitionen der regulären Ausdrücke. Zum Beispiel findet man für $(\alpha \mid \beta)$ auch $(\alpha + \beta)$ oder auch $(\alpha \cup \beta)$. Auch wird natürlich oft $\alpha \cdot \beta$ für $\alpha\beta$ zugelassen.

5. Oft wird in der Literatur zwischen regulärem Ausdruck und beschriebener Sprache nicht unterschieden, das heißt, man identifiziert einen regulären Ausdruck mit der beschriebenen Sprache.

Weitere Beispiele regulärer Ausdrücke

$(a \mid b)^*$ beschreibt die Menge aller Wörter über dem Alphabet $\{a, b\}$.

$(a \mid b)^+$ beschreibt die Menge aller Wörter über dem Alphabet $\{a, b\}$, die nicht dem leeren Wort entsprechen.

$(a \mid b)^* aba (a \mid b)^*$ beschreibt die Menge aller Wörter über dem Alphabet $\{a, b\}$, die das Teilwort aba haben.

$(a \mid b)^* a (a \mid b)^2$ beschreibt die Menge aller Wörter über dem Alphabet $\{a, b\}$, deren drittletztes Symbol ein a ist.

$((a \mid b)(a \mid b))^*$ beschreibt die Menge aller Wörter über dem Alphabet $\{a, b\}$, deren Länge gerade ist.

$(b \mid \varepsilon)(ab)^*(a \mid \varepsilon)$ beschreibt die Menge aller Wörter über dem Alphabet $\{a, b\}$, die nicht das Teilwort aa und nicht das Teilwort bb enthalten.

Satz von Kleene

Satz (Kleene)
Die Menge der durch reguläre Ausdrücke beschreibbaren Sprachen ist genau die Menge der regulären Sprachen.

Beweis (Konstruktionen)

- regulärer Ausdruck $\longrightarrow$ NEA: Induktion über Aufbau der Ausdrücke
- NEA $\longrightarrow$ regulärer Ausdruck: Lösung von Gleichungssystemen (siehe Script).

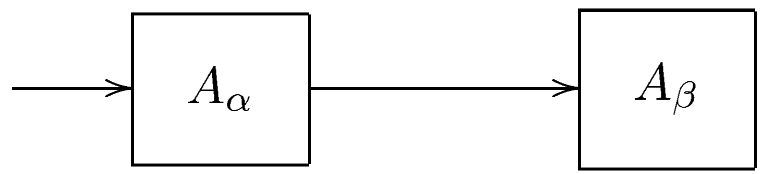
Regulärer Ausdruck $\rightarrow$ NEA

1. Automaten für die Ausdrücke $\emptyset$, ε und a:

2. Seien A_α bzw. A_β NEA mit $T(A_\alpha) = L(\alpha)$ bzw. $T(A_\beta) = L(\beta)$
Konstruiere NEA für $\alpha\beta$, $(\alpha \mid \beta)$ und α^*
(formale Konstruktion: siehe Skript)

Regulärer Ausdruck $\rightarrow$ NEA – Fortsetzung

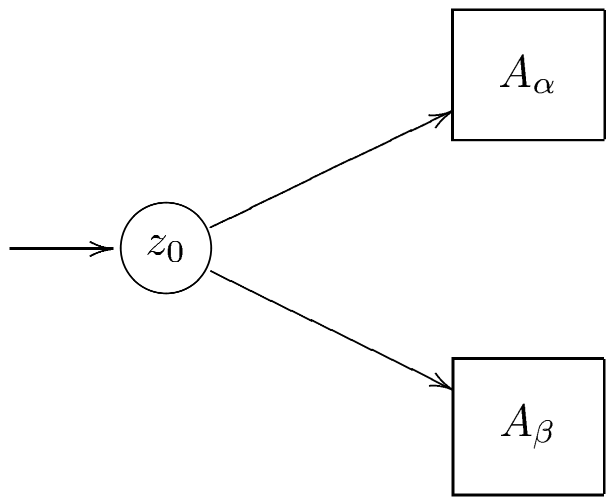
(a) Automat für $\alpha\beta$:



- Verschmelze Endzustände von A_α mit Startzustand von A_β. ("Reihenschaltung")
- neuer Startzustand: Startzustand von A_α
- neue Endzustände: Endzustände von A_β

Regulärer Ausdruck $\rightarrow$ NEA – Fortsetzung

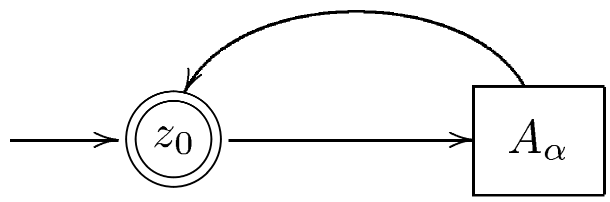
(b) Automat für $(\alpha \mid \beta)$:



- Verschmelze Startzustände von A_α und A_β ("Parallelschaltung").
- neue Endzustände: Endzustände von A_α und A_β

Regulärer Ausdruck $\rightarrow$ NEA – Fortsetzung

(c) Automat für $(\alpha)^*$:



- Füge für jede Kante zu einem Endzustand eine gleiche Kante zum Startzustand ein.
- neuer Endzustand: Startzustand von A_α

Das Pumping Lemma

Satz (Pumping Lemma) Sei L eine reguläre Sprache.
Dann gibt es eine Konstante $k \in \mathbb{N}$, so dass
für alle Wörter $z \in L$ mit $|z| \geq k$
eine Zerlegung $z = uvw$ existiert, so dass gilt:

(i) $|uv| \leq k$ und $|v| \geq 1$,

(ii) für alle $i \in \mathbb{N}$ gilt $uv^iw \in L$.

Pumping Lemma – Beweis

- Reguläre Sprache L werde durch DEA $A = (Z, \Sigma, z_0, \delta, E)$ mit k Zuständen akzeptiert.
- Betrachte Wort $x \in L$ mit $|x| \geq k$.
- Nach Einlesen der ersten k Buchstaben von x wurden $k + 1$ Zustände angenommen.
  Nach dem Schubfachprinzip müssen 2 dieser Zustände gleich sein, d.h. es gibt eine Zerlegung $x = uvw$ mit $|uv| \leq k$ $\hat{\delta}(z_0, u) = \hat{\delta}(z_0, uv) = z_u$, $\hat{\delta}(z_0, uvw) = q \in E$.
- Nach Definition von $\hat{\delta}$ gilt:
  $\hat{\delta}(z_u, v) = z_u$, d.h. $\hat{\delta}(z_u, v^i) = z_u$ für alle $i \geq 0$
- Es folgt $\hat{\delta}(z_0, uv^iw) = \hat{\delta}(\hat{\delta}(z_0, uv^i), w) = \hat{\delta}(z_u, w) = q$, d.h. $uv^iw \in L$.

Anwendung des Pumping Lemmas

Erfüllt eine Sprache nicht die Behauptung des Pumping-Lemmas, so kann sie auch nicht regulär sein.

Um zu beweisen, dass eine Sprache L nicht die Behauptung des Pumping-Lemmas erfüllt, muss man Folgendes zeigen:

Für jede natürliche Zahl k
existiert ein Wort z mit $|z| \geq k$,
so dass für jede Zerlegung $z = uvw$ mit $|uv| \leq k$ und $|v| \geq 1$
ein $i \in \mathbb{N}$ existiert, so dass $uv^iw \notin L$ gilt.

Anwendungen des Pumping Lemmas - Beispiel

Satz: Die Menge PAL der Palindrome über $\{a, b\}$ ist nicht regulär.

Beweis

Wir zeigen, dass PAL nicht die Behauptung des Pumping-Lemmas erfüllt.

- Sei k eine beliebige natürliche Zahl.
- Wähle $z = a^k b a^k$. (Offensichtlich gilt $|z| \geq k$ und $z \in PAL$).
- Für jede Zerlegung $z = uvw$ mit $|uv| \leq k$ und $|v| \geq 1$ gilt:
  $u = a^r$, $v = a^s$, $w = a^{k-r-s} b a^k$ mit $r + s \leq k$ und $s \geq 1$.
- Wähle $i = 2$. Es gilt $uv^2w = a^r a^{2s} a^{k-r-s} b a^k = a^{k+s} b a^k \notin PAL$.

Anwendungen des Pumping Lemmas

Mit Hilfe des Pumping Lemmas kann man auch die Nichtregularität der Sprachen

$$L = \{a^{n^2} \mid n \geq 1\},$$

$$L = \{a^{2^n} \mid n \geq 1\},$$

$$L = \{a^p \mid p \text{ Primzahl}\},$$

$$L = \{a^n b^n c^n \mid n \geq 1\},$$

$$L = \{ww^R \mid w \in \{a,b\}^*\},$$

$$L = \{ww \mid w \in \{a,b\}^*\},$$

$$L = \text{Menge der regulären Ausdrücke über } \Sigma$$

und vieler anderer zeigen.

Abschlusseigenschaften regulärer Sprachen

Satz Sind A und B reguläre Sprachen über Σ, dann sind auch

(i) $A \cup B$,

(ii) $A \cap B$,

(iii) $A \cdot B$,

(iv) $\Sigma \setminus A$ und

(v) A^*

reguläre Sprachen.

Wortproblem und andere Entscheidungsprobleme

Satz Das Wortproblem für ein Wort der Länge n und einen NEA mit k Zuständen ist mit einem Zeitaufwand in der Größenordnung von $n \cdot k$ entscheidbar.

Satz Das Leerheitsproblem, das Endlichkeitsproblem, das Schnittproblem für NEA sind in linearer Zeit (Leerheit, Endlichkeit) bzw. in quadratischer Zeit (Schnitt) entscheidbar.

Satz Das Äquivalenzproblem für DEA ist in quadratischer Zeit entscheidbar.