

Automaten und Formale Sprachen

Vorlesung 3:

Endliche Automaten und rechtslineare Sprachen (1)



Prof. Dr. Dietrich Kuske



FG Automaten und Logik, TU Ilmenau

Wintersemester 2026/27

Einführung

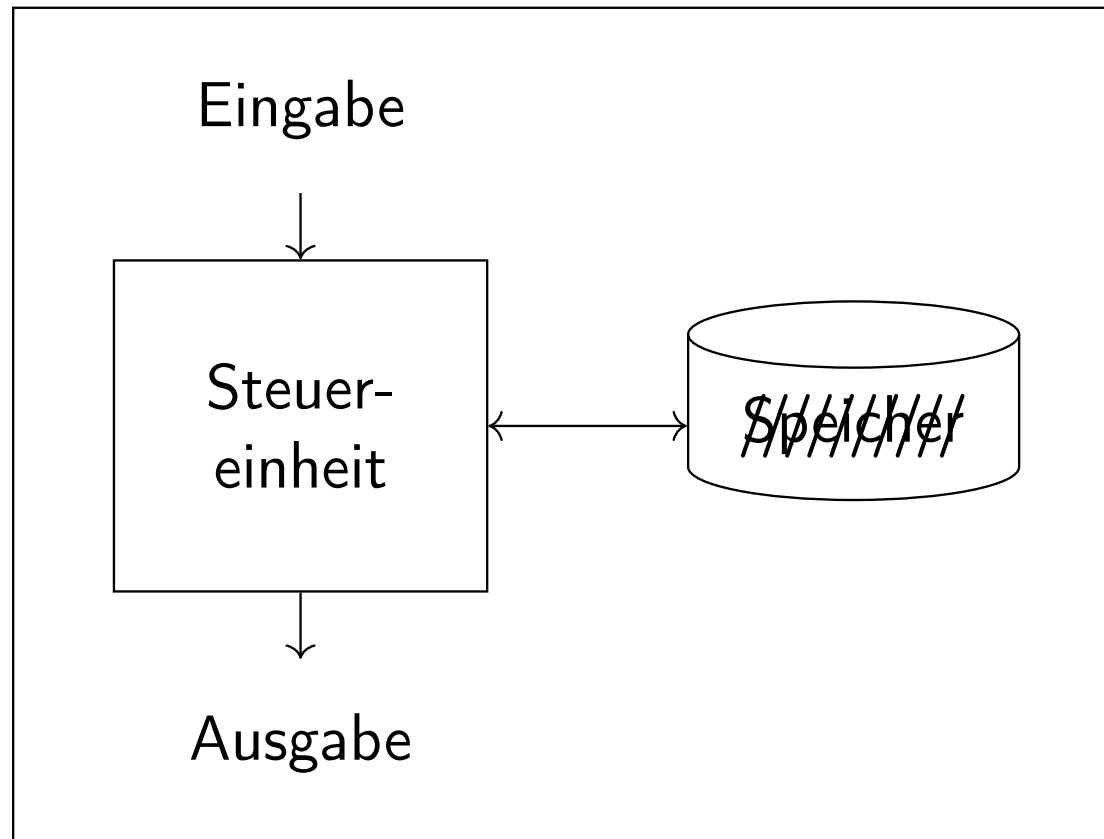
Wörter, Sprachen und Grammatiken

Rechtslineare Sprachen

Kontextfreie Sprachen

Maschinen: endliche Automaten

In diesem Abschnitt beschäftigen wir uns mit rechtslinearen Sprachen, aber zunächst unter einem anderen Blickwinkel.



Steuereinheit:

- endlich viele Zustände
- ändern sich in Abhängigkeit von Eingabe ~~und Speicherkonfiguration~~
- produziert Ausgaben
- ~~gibt Speicherbefehle~~

Eingabe: Folge von „Buchstaben“

Ausgabe: gut/schlecht

Deterministische endliche Automaten

Definition

Ein **deterministischer endlicher Automat** M ist ein 5-Tupel

$M = (Z, \Sigma, z_0, \delta, E)$, wobei:

- Z eine **endliche** Menge von **Zuständen** ist,
- Σ das **Eingabealphabet** (mit $Z \cap \Sigma = \emptyset$) ist,
- $z_0 \in Z$ der **Startzustand** ist,
- $\delta: Z \times \Sigma \rightarrow Z$ die **Überführungs-** oder **Übergangsfunktion** ist und
- $E \subseteq Z$ die Menge der **Endzustände** ist.

Abkürzung: DFA (deterministic finite automaton)

Beispiel:

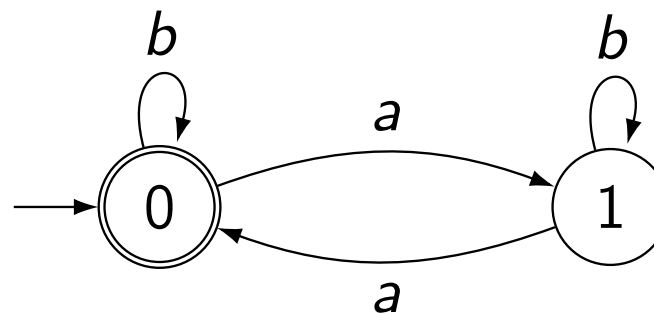
- $Z = \{0, 1\}$
- $\Sigma = \{a, b\}$
- $z_0 = 0$
- $\delta(0, a) = \delta(1, b) = 1, \delta(1, a) = \delta(0, b) = 0$

tabellarisch:

	a	b
0	1	0
1	0	1

- $E = \{0\}$

Intuitiver beschreiben wir diesen DFA graphisch:



Beispiel:

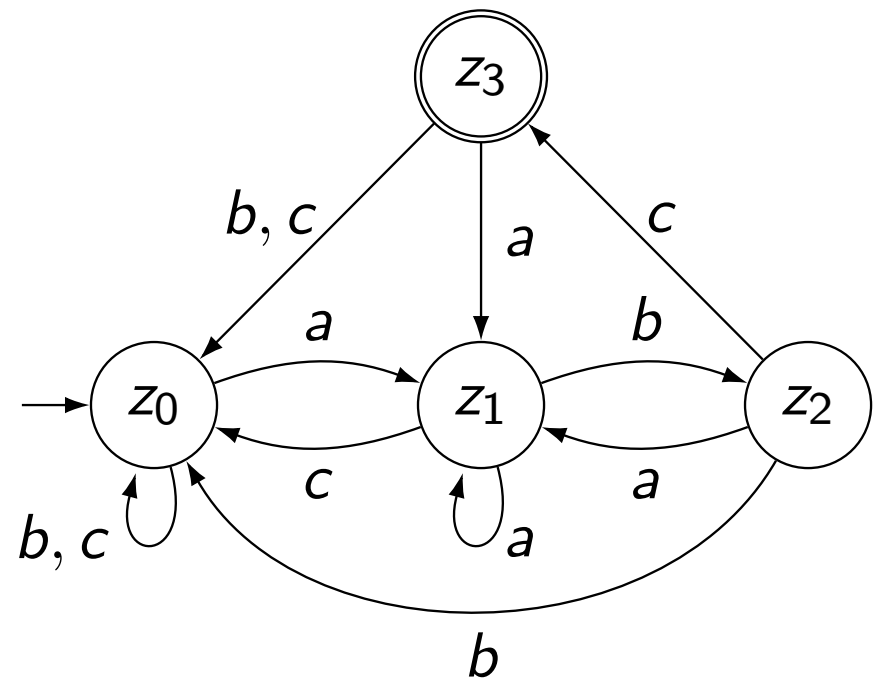
- $Z = \{z_0, z_1, z_2, z_3\}$

- $\Sigma = \{a, b, c\}$

- | | a | b | c |
|-------|-------|-------|-------|
| z_0 | z_1 | z_0 | z_0 |
| z_1 | z_1 | z_2 | z_0 |
| z_2 | z_1 | z_0 | z_3 |
| z_3 | z_1 | z_0 | z_0 |

- $E = \{z_3\}$

graphische Beschreibung:



Die bisherige Übergangsfunktion δ eines DFA liest nur ein Zeichen auf einmal ein. Wir verallgemeinern sie jetzt zu einer Übergangsfunktion $\hat{\delta}$, die die Übergänge für ganze Wörter ermittelt.

Definition

Zu einem gegebenen DFA $M = (Z, \Sigma, z_0, \delta, E)$ definieren wir die Funktion $\hat{\delta}: Z \times \Sigma^* \rightarrow Z$ induktiv wie folgt, wobei $z \in Z$, $w \in \Sigma^*$ und $a \in \Sigma$:

$$\begin{aligned}\hat{\delta}(z, \varepsilon) &= z \\ \hat{\delta}(z, aw) &= \hat{\delta}(\delta(z, a), w)\end{aligned}$$

Intuition: der Zustand $\hat{\delta}(z, w)$ ergibt sich, indem man vom Zustand z aus dem Pfad folgt, der mit w beschriftet ist.

Wenn diese Intuition stimmt, so müßte gelten: $\hat{\delta}(z, uv) = \hat{\delta}(\hat{\delta}(z, u), v)$.

Tut es das?

Lemma

Seien $M = (Z, \Sigma, z_0, \delta, E)$ ein DFA, $z \in Z$ und $v, w \in \Sigma^*$.
Es gilt $\widehat{\delta}(z, vw) = \widehat{\delta}(\widehat{\delta}(z, v), w)$.

Beweis: per Induktion über $|v|$:

IA $v = \varepsilon$: $\widehat{\delta}(z, \varepsilon w) = \widehat{\delta}(z, w) = \widehat{\delta}(\widehat{\delta}(z, \varepsilon), w)$

IV Gelte die Behauptung immer dann, wenn $|v| \leq n$ gilt.

IS $v = au$ mit $a \in \Sigma$, $u \in \Sigma^*$ mit $|u| = n$:

Dann gilt

$$\begin{aligned} \widehat{\delta}(z, auw) &\stackrel{\text{Def.}}{=} \widehat{\delta}(\delta(z, a), uw) \\ &\stackrel{IV}{=} \widehat{\delta}(\widehat{\delta}(\delta(z, a), u), w) \\ &\stackrel{\text{Def.}}{=} \widehat{\delta}(\widehat{\delta}(z, au), w) \end{aligned}$$



Definition

Die von dem DFA $M = (Z, \Sigma, z_0, \delta, E)$ **akzeptierte Sprache** ist

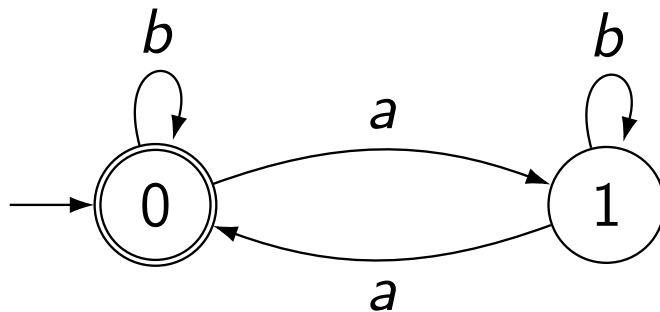
$$L(M) = \{ w \in \Sigma^* \mid \hat{\delta}(z_0, w) \in E \} .$$

Mit anderen Worten: Ein Wort w wird genau dann akzeptiert, wenn derjenige Pfad, der im Anfangszustand beginnt und dessen Übergänge mit den Zeichen von w markiert sind, in einem Endzustand endet.

Definition

Eine Sprache $L \subseteq \Sigma^*$ ist **regulär**, wenn es einen DFA M mit $L(M) = L$ gibt.

Beispiel: Betrachte wieder den folgenden DFA

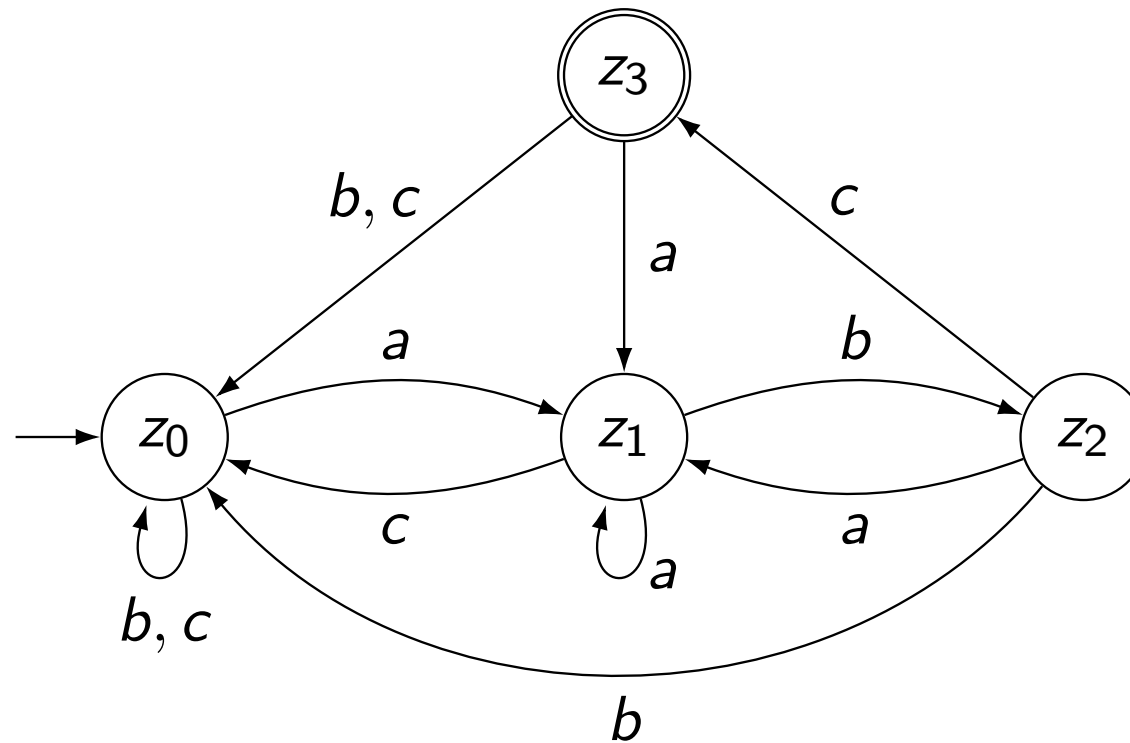


Für ein Wort $u \in \{a, b\}^*$ gilt

$$\hat{\delta}(0, u) = \begin{cases} 0 & \text{falls} \\ 1 & \text{falls} \end{cases}$$

Also erhalten wir $L(M) = \{u \in \Sigma^* : ???\}$, diese Sprache ist also regulär.

Beispiel:



Ziel

Wir wollen zeigen, daß eine Sprache genau dann rechtslinear ist, wenn sie regulär ist. Hierfür drei Schritte:

- jede reguläre Sprache ist rechtslinear (Folie 3.13)
- jede rechtslineare Sprache wird von einem „NFA“ akzeptiert (Folie 4.2)
- jede „NFA-Sprache“ ist regulär (Folie 4.4)

Damit werden wir dann die Äquivalenz eines Erzeugungsmechanismus („rechtslineare Grammatik“) mit einem Akzeptanzmechanismus („DFA“) erhalten.

Proposition

Aus einem DFA M kann eine rechtslineare Grammatik G berechnet werden mit $L(M) = L(G)$.

Insbes. ist also jede reguläre Sprache rechtslinear.

Beweis:

Sei $M = (Z, \Sigma, z_0, \delta, E)$ ein DFA.

Wir definieren eine Typ-3 Grammatik $G = (V, \Sigma, P, S)$ wie folgt:

$$V = Z$$

$$S = z_0$$

$$P = \{z \rightarrow a \delta(z, a) \mid z \in Z, a \in \Sigma\} \cup \{z \rightarrow \varepsilon \mid z \in E\}$$

Behauptung: Für alle $z, z' \in Z$ und $w \in \Sigma^*$ gilt:

$$z \Rightarrow_G^* wz' \iff \hat{\delta}(z, w) = z'.$$

Wir zeigen diese Behauptung durch Induktion über $|w|$ (siehe Zusatzmaterial Folie 3.27).

Für $w \in \Sigma^*$ gilt dann:

$$\begin{aligned} w \in L(G) &\iff \exists z \in V: z_0 \Rightarrow_G^* wz \Rightarrow_G w \\ &\stackrel{\text{Beh.}}{\iff} \exists z \in Z: \hat{\delta}(z_0, w) = z \text{ und } (z \rightarrow \varepsilon) \in P \\ &\iff \hat{\delta}(z_0, w) \in E \\ &\iff w \in L(M) \end{aligned}$$

q.e.d.

Also haben wir $L(M) = L(G)$.



Nichtdeterministische endliche Automaten

Ziel: Eine Sprache ist regulär gdw. sie rechtslinear ist.

erreicht: „ $\implies$ “, wobei die Zustände zu Nichtterminalen wurden

Problem bei „ $\Leftarrow$ “: DFAs sind deterministisch, Grammatiken nichtdeterministisch (es kann z.B. die Produktionen $S \rightarrow aA$ und $S \rightarrow aB$ geben)

Ausweg: erweitere DFAs um Nichtdeterminismus zu „NFAs“ und zeige:

1. jeder „NFA“ kann durch einen DFA simuliert werden (Folie 4.4)
2. jede rechtslineare Sprache kann durch einen „NFA“ akzeptiert werden (Folie 4.2)

Für eine Menge Z ist $\mathcal{P}(Z) = \{Y \mid Y \subseteq Z\}$ die **Potenzmenge** von Z .

Definition

Ein **nichtdeterministischer endlicher Automat mit ε -Transitionen** M ist ein 5-Tupel $M = (Z, \Sigma, S, \delta, E)$, wobei

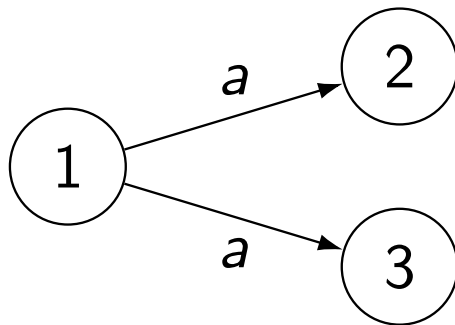
- Z eine **endliche** Menge von **Zuständen**,
- Σ das **Eingabealphabet** (mit $Z \cap \Sigma = \emptyset$),
- $S \subseteq Z$ die Menge der **Startzustände**,
- $\delta: Z \times (\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Z)$ die **Überföhrungsfunktion** und
- $E \subseteq Z$ die Menge der **Endzustände** ist.

Ein **nichtdeterministischer endlicher Automat** ist ein ε -NFA mit $\delta(z, \varepsilon) = \emptyset$ für alle $z \in Z$.

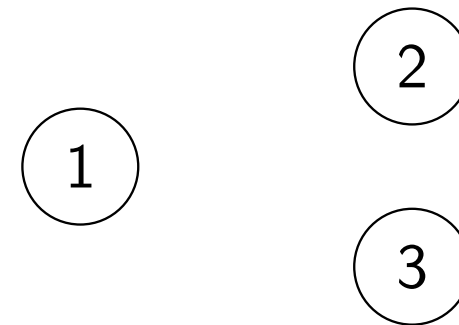
Abkürzung: ε -NFA bzw. NFA (nondeterministic finite automaton)

Beispiel:

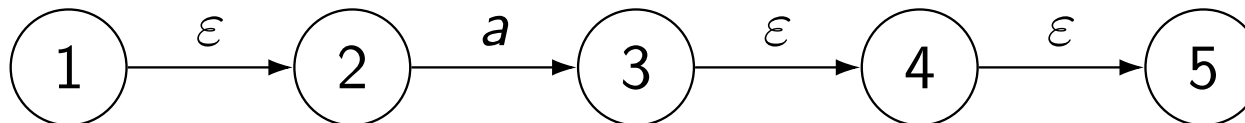
$\delta(1, a) = \{2, 3\}$: ε -NFA kann mit a von 1 nach 2 gehen oder nach 3.



$\delta(1, a) = \delta(1, \varepsilon) = \emptyset$: ε -NFA kann mit a zu keinem Zustand wechseln, er blockiert also.



der folgende ε -NFA kann mit a von 1 nach 3, 4 oder 5 gehen.



Die Übergangsfunktion $\delta: Z \times (\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Z)$ soll jetzt zu einer Mehr-Schritt-Übergangsfunktion $\hat{\delta}: \mathcal{P}(Z) \times \Sigma^* \rightarrow \mathcal{P}(Z)$ erweitert werden.

Ziel: $z \in \hat{\delta}(Y, w) \subseteq Z$ gdw. es w -beschrifteten Pfad von einem der Zustände $y \in Y$ nach z gibt.

Zunächst der Fall $w = \varepsilon$:

Definition

Sei $M = (Z, \Sigma, S, \delta, E)$ ein ε -NFA und $Y_0 \subseteq Z$. Für $n \geq 0$ setze

$$Y_{n+1} = Y_n \cup \bigcup_{y \in Y_n} \delta(y, \varepsilon).$$

Der **Abschluß** von Y_0 ist die Menge $\overline{Y_0} = \bigcup_{n \geq 0} Y_n$.

Intuition: $z \in Y_n$ gdw. es Pfad der Länge $\leq n$ mit ε -Transitionen von einem $y \in Y_0$ nach z gibt.

$z \in \overline{Y_0}$ gdw. es Pfad mit ε -Transitionen von einem $y \in Y_0$ nach z gibt.

Definition

Zu einem gegebenen ε -NFA $M = (Z, \Sigma, S, \delta, E)$ definieren wir die Funktion

$$\hat{\delta}: \mathcal{P}(Z) \times \Sigma^* \rightarrow \mathcal{P}(Z)$$

induktiv wie folgt, wobei $Y \subseteq Z$, $w \in \Sigma^*$ und $a \in \Sigma$:

$$\begin{aligned}\hat{\delta}(Y, \varepsilon) &= \overline{Y} \\ \hat{\delta}(Y, aw) &= \hat{\delta}\left(\bigcup_{z \in \overline{Y}} \delta(z, a), w\right)\end{aligned}$$

Beobachtung

Ist M NFA, so gilt $Y = \overline{Y}$, also $\hat{\delta}(Y, \varepsilon) = Y$ und $\hat{\delta}(Y, aw) = \hat{\delta}(\bigcup_{y \in Y} \delta(y, a), w)$.

Intuition: $z \in \hat{\delta}(Y, w)$ gdw. es w -beschrifteten Pfad von einem $y \in Y$ nach z gibt.

Wenn diese Intuition stimmt, so müßte gelten:

Lemma

Seien $M = (Z, \Sigma, S, \delta, E)$ ein ε -NFA, $X, Y \subseteq Z$ und $v, w \in \Sigma^*$.

Dann gelten

- (1) $\hat{\delta}(Y, w) = \hat{\delta}(\overline{Y}, w)$
- (2) $\hat{\delta}(Y, vw) = \hat{\delta}(\hat{\delta}(Y, v), w)$
- (3) $\hat{\delta}(Y, w) = \overline{\hat{\delta}(Y, w)}$, d.h. $\hat{\delta}(Y, w)$ ist abgeschlossen
- (4) $\hat{\delta}(X \cup Y, w) = \hat{\delta}(X, w) \cup \hat{\delta}(Y, w)$

was es auch tut - Beweis siehe Zusatzmaterial Folie 3.28 ff.

Definition

Die von dem ε -NFA $M = (Z, \Sigma, S, \delta, E)$ **akzeptierte Sprache** ist

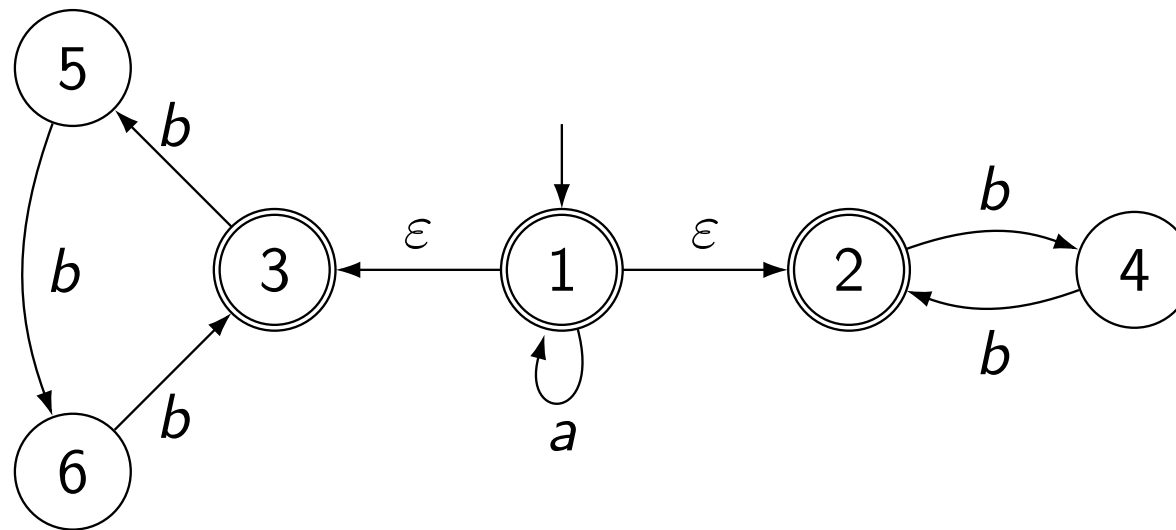
$$L(M) = \{w \in \Sigma^* \mid \hat{\delta}(S, w) \cap E \neq \emptyset\}.$$

Mit anderen Worten: Ein Wort w wird genau dann akzeptiert, wenn es mindestens einen Pfad von einem Anfangszustand zu einem Endzustand gibt, dessen Übergänge mit den Zeichen von w markiert sind (daneben kann es auch Pfade geben, die nicht in einen Endzustand führen).

Beispiel: gesucht ist ein ε -NFA, der die Sprache

$$L = \{a^k b^{2\ell} \mid k, \ell \geq 0\} \cup \{a^k b^{3\ell} \mid k, \ell \geq 0\}$$

akzeptiert.

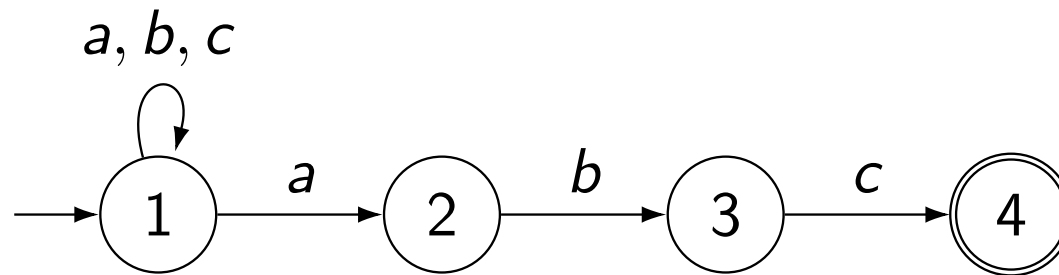


Dieser ε -NFA führt zunächst beliebig viele a aus und springt dann (nichtdeterministisch) mit ε -Transition, um gerade oder durch drei teilbare Anzahl von b auszuführen.

Beispiel: gesucht ist ein NFA, der die Sprache

$$L = \{u \in \{a, b, c\}^* \mid u \text{ endet auf } abc\}$$

akzeptiert.



Dieser NFA rät zu einem beliebigen Zeitpunkt nichtdeterministisch, daß jetzt das Suffix abc beginnt und überprüft dann, daß dies tatsächlich der Fall ist.

Beispielhafte Testfragen aus dieser Vorlesung (andere sind möglich):

- Definition DFA/ ε -NFA/NFA
- Definition $L(M)$ für DFA/ ε -NFA M
- Definition reguläre Sprache

Zusammenfassung 3. Vorlesung

in dieser Vorlesung neu

- Definitionen DFA, ε -NFA, NFA und die von diesen akzeptierte Sprache
- jede reguläre Sprache (d.h. von einem DFA akzeptiert) ist rechtslinear

offen für kommende Vorlesung

- jede rechtslineare Sprache ist „NFA-Sprache“
- jede „NFA-Sprache“ ist regulär

weiterhin in nächster Vorlesung

- Abschlußeigenschaften der Klasse der rechtslinearen Sprachen, d.h. gilt z.B. K, L rechtslinear $\implies K \cap L$ rechtslinear?
- dabei: jede „ ε -NFA-Sprache“ ist „NFA-Sprache“

Zusatzmaterial

Beweis der Behauptung auf Folie 3.14:

IA $|w| = 0$: $z \Rightarrow_G^* z'$ gdw. $z = z'$ gdw. $\hat{\delta}(z, w) = z'$

IV Gelte Behauptung für alle Wörter der Länge $\leq n$

IS $w = av$ mit $a \in \Sigma$, $v \in \Sigma^*$, $|v| = n$

$$z \Rightarrow_G^* avz' \iff \exists z'' \in Z: z \Rightarrow_G az'' \Rightarrow_G^* avz'$$

$$\stackrel{IV}{\iff} \exists z'' \in Z: z'' = \hat{\delta}(z, a) = \delta(z, a) \text{ und } z' = \hat{\delta}(z'', v)$$

$$\iff z' \in \hat{\delta}(z, av)$$

Beweis des Lemmas auf Folie 3.20:

(1) Fallunterscheidung $w = \varepsilon$ und $w \neq \varepsilon$:

$$1. \text{ Fall } w = \varepsilon: \hat{\delta}(Y, \varepsilon) = \overline{Y} = \overline{\overline{Y}} = \hat{\delta}(\overline{Y}, \varepsilon)$$

2. Fall $w = au$ mit $a \in \Sigma$ und $u \in \Sigma^*$:

$$\hat{\delta}(Y, au) = \hat{\delta}\left(\bigcup_{y \in \overline{Y}} \delta(y, a), u\right) \stackrel{\overline{Y} \equiv \overline{\overline{Y}}}{=} \hat{\delta}\left(\bigcup_{y \in \overline{\overline{Y}}} \delta(y, a), u\right) = \hat{\delta}(\overline{Y}, au)$$

(2) per Induktion über $|v|$

IA $v = \varepsilon$:

$$\hat{\delta}(Y, vw) \stackrel{vw \equiv w}{=} \hat{\delta}(Y, w) \stackrel{(1)}{=} \hat{\delta}(\overline{Y}, w) = \hat{\delta}(\hat{\delta}(Y, v), w)$$

IV Gelte (2) für alle (Y, v, w) with $|v| \leq n$

IS $v = au$ mit $a \in \Sigma$, $|u| = n$. Dann gilt

$$\begin{aligned} \hat{\delta}(Y, auw) &\stackrel{Def.}{=} \hat{\delta}\left(\bigcup_{y \in \overline{Y}} \delta(y, a), uw\right) \\ &\stackrel{IV}{=} \hat{\delta}\left(\hat{\delta}\left(\bigcup_{y \in \overline{Y}} \delta(y, a), u\right), w\right) \stackrel{Def.}{=} \hat{\delta}(\hat{\delta}(Y, au), w) \end{aligned}$$

(2) folgt aus (2): $\hat{\delta}(Y, w) = \hat{\delta}(Y, w\varepsilon) \stackrel{(2)}{=} \hat{\delta}(\hat{\delta}(Y, w), \varepsilon) = \overline{\hat{\delta}(Y, w)}$

(3) Induktion über $|w|$:

IA hier betrachten wir die Fälle $|w| = 0$ und $|w| = 1$ separat: $w = \emptyset$:

$$\hat{\delta}(X \cup Y, w) = \overline{X \cup Y} = \overline{X} \cup \overline{Y} = \hat{\delta}(X, w) \cup \hat{\delta}(Y, w)$$

$w = a \in \Sigma$:

$$\begin{aligned} \hat{\delta}(X \cup Y, a) &= \hat{\delta}(X \cup Y, a\varepsilon) \\ &= \bigcup_{z \in \overline{X \cup Y}} \overline{\hat{\delta}(z, a)} \\ &= \bigcup_{z \in \overline{X}} \overline{\hat{\delta}(z, a)} \cup \bigcup_{z \in \overline{Y}} \overline{\hat{\delta}(z, a)} \\ &= \hat{\delta}(X, a\varepsilon) \cup \hat{\delta}(Y, a\varepsilon) \\ &= \hat{\delta}(X, a) \cup \hat{\delta}(Y, a) \end{aligned}$$

IS $w = au$ mit $a \in \Sigma$, $u \in \Sigma^*$

$$\begin{aligned}
 \widehat{\delta}(X \cup Y, au) &\stackrel{(2)}{=} \widehat{\delta}(\widehat{\delta}(X \cup Y, a), u) \\
 &\stackrel{(IA)}{=} \widehat{\delta}(\widehat{\delta}(X, a) \cup \widehat{\delta}(Y, a), u) \\
 &\stackrel{(IV)}{=} \widehat{\delta}(\widehat{\delta}(X, a), u) \cup \widehat{\delta}(\widehat{\delta}(Y, a), u) \\
 &\stackrel{(2)}{=} \widehat{\delta}(X, au) \cup \widehat{\delta}(Y, au)
 \end{aligned}$$

