0

Formale Sprachen

Syntaxanalyse Motivation
Tokenerkennung

$L \subseteq \Sigma^*$ $\Sigma = \{0,1\}$ Wörter $w \in \Sigma^*$ erzeugen!

Grammatik

$G = (\Sigma, V, S, P)$

$L(G) := \{ w \in \Sigma^* \mid w$ ist ableitbar aus S mit den Regeln aus $P \}$

Komplexität der Regeln (Chomsky-Einteilung)

2. "kontextfrei" $A \to v,\ A \in V,\ v \in (V \cup \Sigma)^*$

3. "regulär" (rechtslinear/linkslinear) $A \to v,\ A \in V,\ v = aB$ oder $v = \varepsilon,\ a \in \Sigma,\ B \in V$

1

1. $L = \{ 1^n 0^n \mid n \in \mathbb{N}_0 \} \overset{\vee}{\doteq} L(G)$

$S \to \varepsilon \mid 1S0$ <u>kontextfrei</u> (regulär?)

2. $L = \{ 1^n 0^m \mid n, m \in \mathbb{N}_0 \} = L(G)$ <u>regulär</u>

$S \to \varepsilon \mid 1S \mid 0B \qquad B \to \varepsilon \mid 0B$

$\left(S \to \varepsilon \mid S0 \mid B1 \qquad B \to \varepsilon \mid B1 \right)$

"linkslinear"

Wortproblem für reguläre Sprachen lösen?

3.2 Endliche Automaten

Definition 3.6 Ein Endlicher Deterministischer Automat (DFA)

deterministic finite Automat

ein 5-Tupel

$$M = (Q, \Sigma, \delta, q_0, F)$$ derart, dass

Q: endliche Menge von Zustände

Σ: Eingabealphabet

δ: Übergangsfunktion $\qquad \delta : Q \times \Sigma \to Q$

q_0: Startzustand

$F \subseteq Q$: Menge der akzeptierenden Zustände

wobei δ: $\underbrace{Q}_{\text{aktuelles Zustand}} \times \underbrace{\Sigma}_{\text{eingelesenes Zeichen}} \longrightarrow \underbrace{Q}_{\text{Folgezustand}}$.

Arbeitsweise des Automaten

③ a)

DFA M Eingabewort $w = w_1 w_2 \cdots w_n \in \Sigma^*$

$\rightarrow$ $w_i \in \Sigma \quad i = 1, \ldots, n$

Starte in q_0 Startzustand

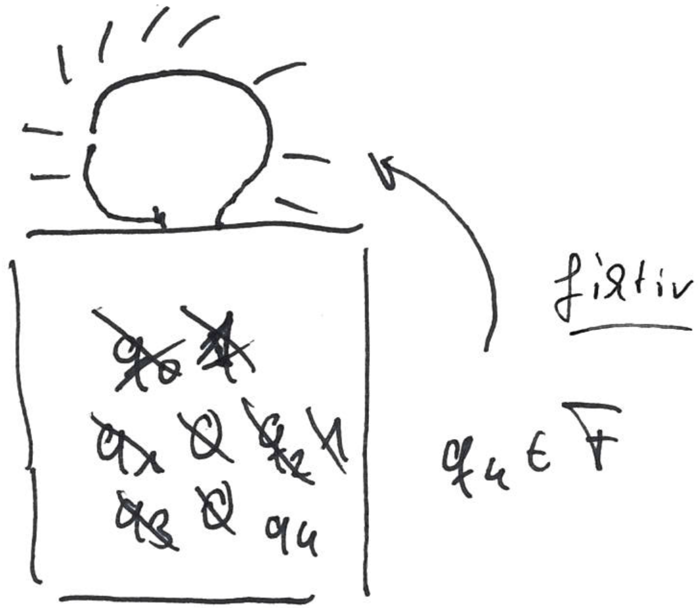
Lese Wort von links nach rechts ein,

- Werte $\delta(q_0, w_1) = q_1$ aus
- Werte $\delta(q_1, w_2) = q_2$ aus

$\vdots$

- Werte $\delta(q_{n-2}, w_{n-1}) = q_{n-1}$ aus
- Werte $\delta(q_{n-1}, w_n) = q_n$ aus

Falls $q_n \in F \Rightarrow$ "akzeptiert"



$q_4 \in F$

$\Sigma = \{0, 1\}$

$1010 \in \Sigma^*$

3 b)

$\delta : Q \times \Sigma \to Q$

$\delta(q, a) = q'$ Folgezustand q' beim Lesen von Buchstabe a im Zustand q

BSP M $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{0, 1\}$, q_0 Startzustand

$F = \{q_1\}$

$Q \backslash \Sigma$	0	1
q_0	q_2	q_1
q_1	q_1	q_1
q_2	q_2	q_2

<u>Arbeitsweise:</u> $w \in \{0,1\}^*$

$q_0 011 \to q_2 11 \to q_2 1 \to q_2 \notin F$

$q_0 101 \to q_1 01 \to q_1 1 \to q_1 \in F$

$\{1w \mid w \in \Sigma^*\} = L(M)$

Nur der letzte Zustand interessant! (Durchlauf)

M ist geg. / δ geg.

$$\delta^*: Q \times \Sigma^* \to Q$$

(q, w) als Eingabe: DFA M terminiert im Zustand q' nach sukzessiver Abarbeitung

$$\delta^*(q, w) := q'$$

Definition von δ^*:

$$\delta^*(q, \varepsilon) := q \qquad \text{(i)}$$

$$\delta^*(q, a) := \delta(q, a) \qquad \text{(ii)}$$

$a \in \Sigma, w \in \Sigma^*$ $$\delta^*(q, aw) := \delta^*(\delta(q, a), w) \qquad \text{(iii)}$$

Dann ist: $\delta^*(q, w) = \delta^*(\delta^*(q, w_1), w_2)$

$w = w_1 w_2$ wohldefiniert

$w_1, w_2 \in \Sigma^*$

Begriffe: M DFA

Für jedes $w \in \Sigma^*$ terminiert M in Zustand $q' = \delta^*(q_0, w)$

$q' \notin F$: M verwirft Wort w

$q' \in F$: M akzeptiert Wort w

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

die von M akzeptierte Sprache

M akzeptiert / entscheidet $L(M)$

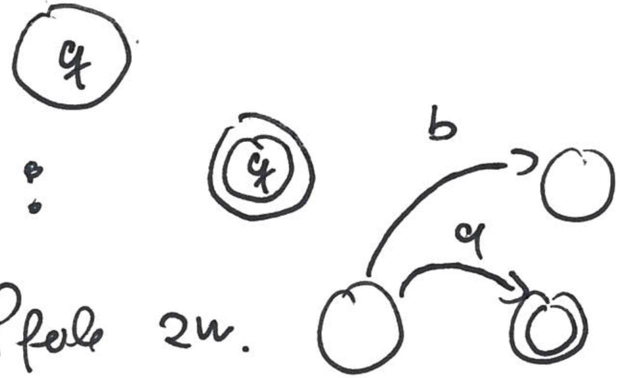
(6)

Anschauliche Darstellung:

Startzustand markieren: ↙ Pfeil

Zustände: q

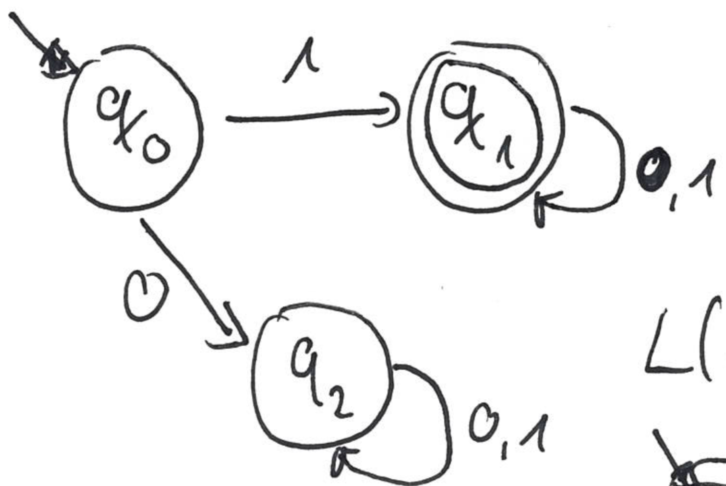
Zustände aus F: $\circledcirc q$



Übergänge: Pfeile zw. Zuständen versehen mit Buchstaben

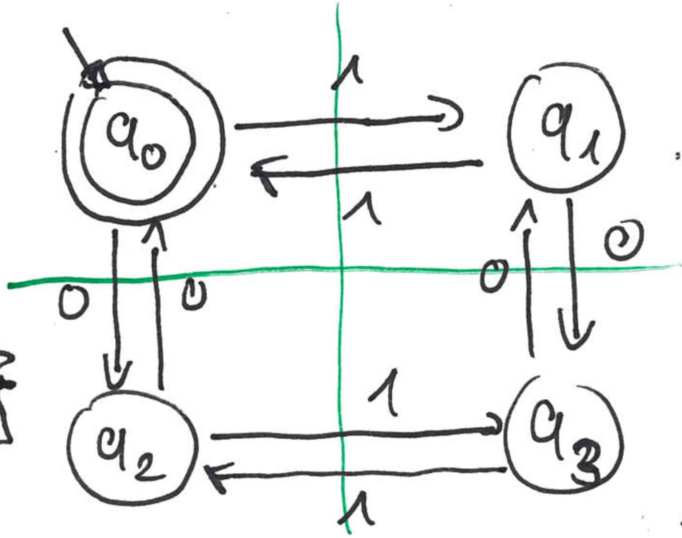
BSP

I.



$L(M) = \{1w \mid w \in \Sigma^*\}$

II.



$L(M) \overset{!}{=}$

$\{w \in \Sigma^* \mid |w|_0 \text{ ist gerade}, |w|_1 \text{ ist gerade}\}$

1. w' mit $|w'|_1$ ist gerade
$\iff \delta^*(q_0, w') \in \{q_0, q_2\}$

2. w' mit $|w'|_0$ ist gerade
$\iff \delta^*(q_0, w') \in \{q_0, q_1\}$

1+2. $\Rightarrow$ Beh.

7

Frage: Was können die DFAs?

Welche Sprachen lassen sich akzeptieren / entscheiden?

$L = \{0^n 1^n \mid n \in \mathbb{N}\}$ wird nicht entschieden durch DFA

$S \to \varepsilon \mid 0S1$ („nicht regulär")

Theorem 3.7 $L = \{0^n 1^n \mid n \in \mathbb{N}_0\}$ wird nicht durch einen DFA entschieden.

Beweis: Indirekt! Widerspruch!

Ann: es ex. DFA M mit $L(M) = L$ / der L entscheidet.

Sei Q endliche Zustandsmenge M $|Q| = n$.

Betrachte $\{\varepsilon, 0, 0^2, 0^3, \ldots, 0^n\}$ $n+1$ Worte (8)

$\delta(q_0, 0^i)$ anwende

$i = 0, \ldots, n$

$\Rightarrow$ es muss Worte $0^i, 0^j$

$\Uparrow$ mit $i \neq j$, $0 \leq i, j \leq n$

q_0	q_1	…	q_{n-1}

<u>Schubfachprinzip</u> geben

mit $\delta^*(q_0, 0^i) \overset{(*)}{=} \delta^*(q_0, 0^j)$

M entscheidet $L \Rightarrow \delta^*(q_0, 0^i 1^i) \in F$ Ann.

$$\delta^*(q_0, \underbrace{0^j 1^i}_{\notin L}) \overset{\downarrow}{=} \delta^*(\delta^*(q_0, 0^j), 1^i) \overset{(*)}{=} \delta^*(\delta^*(q_0, 0^i), 1^i) \overset{\downarrow}{=} \delta^*(q_0, 0^i 1^i) \in F$$

$\Rightarrow$ M akzeptiert <u>auch</u> $0^j 1^i$ $i \neq j$ ↯

□

3.2.1 Pumping Lemma für endl. Automaten

<u>Allgemeines Werkzeug</u> zum Nachweis, das Sprache nicht von DFA akzeptiert wird.

<u>Lemma 3.8</u> (Pumping Lemma) [positive Aussage]

Sei L Sprache die von DFA akzeptiert wird. Dann gilt folgende Aussage

$$\exists n \in \mathbb{N} : \forall z \in L, \Big(|z| \geq n \implies$$

$$\exists u, v, w \in \Sigma^* \text{ mit } z = uvw \text{ und } |uv| \leq n \text{ und } |v| \geq 1$$

$$\text{und } \underline{\forall i \geq 0 \text{ gilt: } uv^iw \in L} \Big)$$

<u>Beweis:</u> L wird von DFA M entschieden

Zustandsmenge Q und $|Q| = n$

Sei $z \in L$ $|z| \geq n$

$\Rightarrow$ DFA durchläuft bei Abarbeitung von z $(n+1)$ Zustände (q_0 dazu)

$\Rightarrow$ mind. ein Zustand kommt doppelt vor

<u>Schubfachprinzip</u>

Nehme den ersten solchen Zustand q und sei u <u>Präfix von z</u> bei dem q zum ersten Mal vorkommt.

<u>also:</u> $z = u z'$ $\quad \delta^*(q_0, z) = \delta^*(\delta^*(q_0, u), z') = \delta^*(q, z)$

Sei uv Präfix von z bei dem q das zweite Mal vorkommt.

$\Rightarrow |v| \geq 1$ $\quad \underline{\delta^*(q, v) = q}$ $\qquad z = u v z''$, $z'' = w$

<u>Zeige:</u> $\delta^*(q, w) \in F$

$$\underset{\in F}{\delta^*(q_0, uvw)} = \delta^*(\delta^*(q_0,u), vw) = \delta^*(q, vw) = \delta^*(\delta^*(q,v), w) = \delta^*(q,w) \in F$$

$\Rightarrow$ Jetzt kann ich v aufpumpen! $v^i = vv\cdots v$ i-mal

$$\delta^*(q_0, \underline{uv^i w}) = \delta^*(\delta^*(q_0,u), v^i w) = \delta^*(q, v^i w) \qquad \delta^*(q,v) = q$$

$$= \delta^*(\delta^*(q,v), v^{i-1} w) = \delta^*(q, v^{i-1} w) = \cdots = \delta^*(q, vw) = \delta^*(q,w) \in F$$

$\underline{\text{gilt}}$ $\forall i \geq 0$

$\square$

PL- anwenden   wichtig

BSP $L = \{0^i 1^i \mid i \in \mathbb{N}_0\}$

Ann: L wird von DFA A entschieden

$\Rightarrow$ PL gilt $\exists n \in \mathbb{N} : \forall z \in L :$

$(|z| \geq n \Rightarrow \exists$ Zerlegung $z = uvw$ mit $|uv| \leq n$, $|v| \geq 1$ und $\forall i \geq 0$ gilt $uv^iw \in L)$

Nehme dieses n wähle $z = 0^n 1^n$ $|z| = 2n \geq n$

$\Rightarrow$ Zerlegung ex $z = uvw$ $(|uv| \leq n$ $|v| \geq 1$

$\Rightarrow$ uv besteht nur aus Nullen also auch $|v| \geq 1$

$\Rightarrow$ $v = 0^k$ $k \geq 1$

(13)

$\Rightarrow$ wähle $i=2$

$$uv^2w = 0^n 0^k 1^n \in L \quad ↯$$

$$n+k > n$$

$$k \geq 1$$

$$uvvw = \underset{0^{g-k}}{u}\,\underset{0^k}{v}\,\underset{0^k}{v} \mid \underset{0^{n-g}1^n}{w} \qquad 1 \leq g \leq n \qquad \square$$