<u>Theorem 3.19</u> DFA entscheidet Sprache L

$\Longleftrightarrow$ L ist regulär

"$\Rightarrow$" DFA M ex. $L(M) = L$

konstruiere G mit $L(G) \overset{!}{=} L(M)$

<u>Jetzt:</u> "$\Leftarrow$"

(1)

"$\Leftarrow$" L ist regulär, $G = (\Sigma, V, S, P)$

reguläre (rechtslineare) Grammatik G mit $L(G) = L$

$\leadsto$ konstruiere NFA (dann baue DFA aus NFA Potenzmengenkonstruktion)

Theorem 3.11

$$\begin{cases} Q = V, \quad \Sigma = \Sigma, \\ q_0 = S, \quad F = \{A \in V \mid (A \to \varepsilon) \in P\} \\ \delta : Q \times \Sigma \to \mathcal{P}(Q) \quad \delta(A, a) := \{B \in V \mid (A \to aB) \in P\} \end{cases}$$

Zeige: $L(G) \overset{!}{=} L(M)$

c) $L(G) \subseteq L(M)$

Betrachte $w = w_1 w_2 \cdots w_n \in L(G)$ $\ulcorner$ $w = \varepsilon \overset{!}{:} S \to \varepsilon \in P$, $\delta(s,\varepsilon) = s$, $s \in F$ $\lrcorner$

$\underset{\text{Def. G.}}{\Rightarrow}$ $\exists$ Ableitung

$$S \to w_1 A_1 \to w_1 w_2 A_2 \to \cdots \to w_1 w_2 \cdots w_{i-1} A_{i-1} \to w_1 w_2 \cdots w_{i-1} w_i A_i \to$$

$$\cdots \to w_1 w_2 \cdots w_n A_n \to w_1 w_2 \cdots w_n \varepsilon = w$$

$\Rightarrow$ Regeln $S \to w_1 A_1$, $A_{i-1} \to w_i A_i$, $A_n \to \varepsilon$

$\underset{\text{via Konstr.}}{\Rightarrow}$ $A \in \delta(s, w_1)$, $A_i \in \delta(A_{i-1}, w_i)$, $A_n \in F$

$\Rightarrow$ $s, A_1, A_2, A_3, \ldots, A_n$ ist erfolgreiche Zustandsfolge für $w = w_1 w_2 \cdots w_n$ in M, $A_n \in F$

$\Rightarrow$ $w \in L(M)$

③

D) $L(M) \subseteq L(G)$ Wort $w = w_1 w_2 \cdots w_n \in L(M)$

Γ $w = \varepsilon$, $\delta(S,\varepsilon) = S \in F$, $\checkmark$ $S \to \varepsilon$ $\lrcorner$

M akzeptiert w

$\Rightarrow$ es ex. Folge $S, A_1, A_2, \ldots, A_n$ mit $\underline{A_n \in F}$

$A_1 \in \delta(S, w_1)$, $A_i \in \delta(A_{i-1}, w_i)$ $i \geq 2$

$\Rightarrow$ es ex. entsprechende Ableitungsregeln in G

via Konstruktion $S \to w_1 A_1$, $A_{i-1} \to w_i A_i$ $i \geq 2$, $A_n \to \varepsilon$

$\Rightarrow$ Wort w kann in G mit genau diesen Regeln abgeleitet werden

$\Rightarrow w \in L(G)$

C) + D) $\Rightarrow L(M) = L(G)$

□

Andere Charakterisierung, ebenfalls äquivalent!

"Schöne" kompakte Bezeichnungsmöglichkeit

Unix: "sed"

Textersetzung

Texte zu ~~F~~iltern!

Suchen in Texten!

Reguläre Ausdrücke!

Definition 3.20, Regulärer Ausdruck R über Alphabet Σ beschreibt Sprache $L(R) \subseteq \Sigma^*$. Entsteht rekursiv über endliche Anwendungen folgender Regeln.

a) i) ϕ ist regulärer Ausdruck für die leere Sprache $\{\} \subseteq \Sigma^*$; $L(\phi) = \phi$

ii) ε ist regulärer Ausdruck für die Sprache $\{\varepsilon\} \subseteq \Sigma^*$; $L(\varepsilon) = \{\varepsilon\}$

iii) $a \in \Sigma$ ist regulärer Ausdruck für die Sprache $\{a\} \subseteq \Sigma^*$: $L(a) = \{a\}$

(5)

b) Seien R_1 und R_2 reguläre Ausdrücke für die Sprachen L_1 und L_2 also $L(R_1) = L_1$ und $L(R_2) = L_2$

dann gilt:

i) $(R_1) + (R_2)$ ist regulärer Ausdruck für $L((R_1)+(R_2)) = L_1 + L_2 := L_1 \cup L_2$

"R_1 oder R_2" $\ulcorner L((a)+(b)) = \{a, b\} = \{a\} + \{b\} \lrcorner$

ii) $(R_1) \cdot (R_2)$ ist regulärer Ausdruck für $L((R_1)\cdot(R_2))$

$= L_1 \cdot L_2 := \{w_1 w_2 \in \Sigma^* \mid w_1 \in L_1, w_2 \in L_2\}$

"Konkatenation von R_1 und R_2"

iii) $(R_1)^*$ ist regulärer Ausdruck für den "Kleen'schen Abschluss L_1^* von L_1"

$L((R_1)^*) = L_1^* := \bigcup_{i \geq 0} L_1^i$ mit $L_1^0 := \{\varepsilon\}$ $L_1^i := L_1 \circ L_1^{i-1}$ für $i \geq 1$

$\ulcorner$ alle endlichen Konkatenationen von L_1, $L((a)^*) = \{\varepsilon, a, a^2, a^3, \ldots\}$ $\lrcorner$

Bsp $\Sigma = \{0, 1\}$ Verknüpfungen: 6

Konkatenation, Oder, Kleenischer Abschluss

1) $L_1 = \{\varepsilon, 111\}$ $L_2 = \{\varepsilon, 01\}$

$L_1 \circ L_2 = \{\varepsilon, 01, 111, 11101\}$ $\varepsilon + 111$

$\underline{L_1 = L((\varepsilon) + (((1)\cdot(1))\cdot(1)))}$

2) $L_1 = \{0^n 1^n \mid n \in \mathbb{N}_0\}$ $L_2 = \{1^n 0^n \mid n \in \mathbb{N}_0\}$

$L_1 \cdot L_2 = \{0^n 1^{n+m} 0^m \mid n, m \in \mathbb{N}_0\}$

3) $L = \{1, 00\}$ $L^0 = \{\varepsilon\}$ $L^1 = L \cdot L^0 = L$ $L^* = \bigcup_{i \geq 0} L^i$

$L^2 = L \cdot L^1 = \{11, 100, 001, 0000\}$ $L^3 = L \cdot L^2$

L^* alle Wörter als endliche Hintereinanderreihung von 1 und 00 in bel. Reihenfolge

7

Konvention Klammerung !

I. $*$ vor $\circ$ vor $+$

II. 01 statt $(0)\cdot(1)$

$$L\left((\varepsilon)+\left(\left((1)\cdot(1)\right)\cdot(1)\right)\right) = \{\varepsilon, 111\}$$

$\varepsilon + 111(0^*)$

$\varepsilon + 1110^*$

Schreibweise:

$\varepsilon + 111$ $\quad$ $\varepsilon \mid 111$

$\underline{\varepsilon + (11)^*}$ $\quad$ $\varepsilon \mid (11)^*$

redundant

BSP Kompakte Darstellung von Sprachen

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

1) Alle Worte aus Σ^*, die mit Wort 101 enden:

$(0+1)^* \circ 101$

2) Alle Wörter bei denen sich 0 und 1 sukzessive abwechseln
Beginn darf auch 1 sein, Ende darf auch 0 sein

$(\varepsilon+1)(01)^*(\varepsilon+0)$ $\qquad$ $(\varepsilon|1)(01)^*(\varepsilon|0)$

3) $R^+ := RR^*$ $\qquad$ $L(R^+)$ ohne das leere Wort

"positive Hülle"

Ziel

Theorem 3.23

Sei $L \subseteq \Sigma^*$ Sprache.

Folgende Aussagen sind äquivalent:

1) L wird von DFA entschieden. (M mit $L(M) = L$)
2) L wird von NFA entschieden. (M' mit $L(M') = L$)
3) L hat endlichen Nerode-Index, $\mathrm{index}(R_L)$.
4) L läßt sich durch regulärer Grammatik darstellen. (G mit $L(G) = L$)
(rechtslinear, linkslinear)
5) Es ex. regulären Ausdruck R ~~...~~ (R mit $L(R) = L$)
der L beschreibt.

Allgemeine Motivation

Sprache $L \subseteq \Sigma^*$ Wortproblem lösen!

Jedes beliebige Berechnungsproblem kann ich als Wortproblem definieren!

BSP TSP Problem Entscheidungsvariante!

So als Wortproblem kodieren!

Sprache der Wörter, die die TSP-Aufgabe mit Kosten $\leq k$ repräsentieren!

Lemma 3.21 + Lemma 3.22 $\rightsquigarrow$ Theorem 3.23

Reguläre Sprachen $\Longleftrightarrow$ Reguläre Ausdrücke

<u>Lemma 3.21</u>: Jeder reguläre Ausdruck R beschreibt eine reguläre Sprache $L(R)$.

<u>Beweisskizze</u>: <u>Strukturelle Induktion</u> über den Aufbau von regulären Ausdrücken

<u>Ind Anfg.</u> Grundbausteine

$\emptyset \quad L(\emptyset) = \emptyset$

$\varepsilon \quad L(\varepsilon) = \{\varepsilon\}$

$a \quad L(a) = \{a\}$

Zeige: NFA für $L(R)$ existiert der $L(R)$ entscheidet.

Ind. Anfang

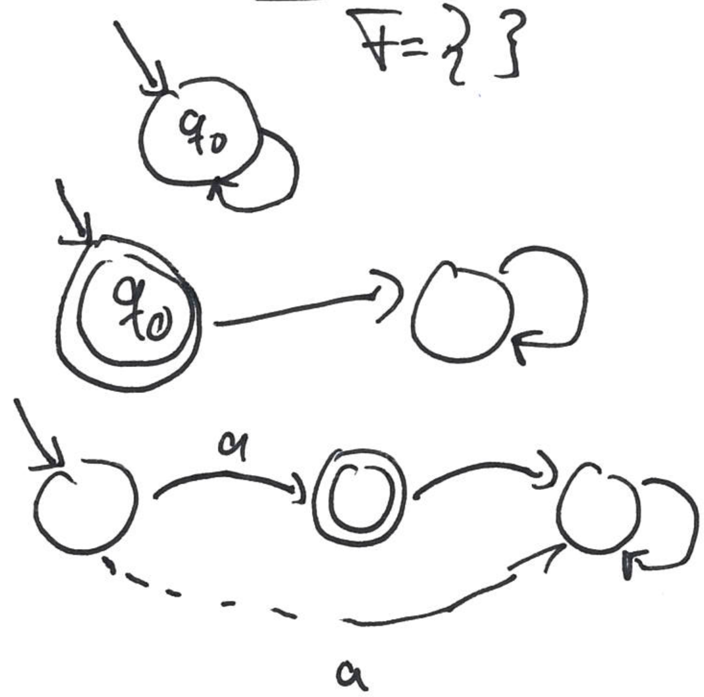
$\emptyset$ $\quad L(\emptyset) = \emptyset$

ε $\quad L(\varepsilon) = \{\varepsilon\}$

a $\quad L(a) = \{a\}$

NFA bauen

$F = \{\}$



$\emptyset$

$\{\varepsilon\}$

$\{a\}$

Ind. Ann: Es ex NFAs M_1 und M_2 die $L(R_1)$ und $L(R_2)$ akzeptieren für die regulären Ausdrücke R_1 und R_2.