

$L \subset \Sigma^*$ Sprache

$w \in \Sigma^* : F_L(w) = \{ x \in \Sigma^* \mid wx \in L \}$

Fortsetzungssprache von w bezüglich L

$\mathcal{F}_L = \{ F_L(w) \mid w \in \Sigma^* \}$ Menge der Fortsetzungssprachen für L

Satz (Myhill – Nerode):

L DEA-Sprache $\Leftrightarrow \mathcal{F}_L$ ist endlich

Satz (Myhill – Nerode):

L DEA-Sprache $\Leftrightarrow \mathcal{F}_L$ ist endlich

Bew: “ $\Rightarrow$ ”

L DEA-Sprache $\Rightarrow \exists$ DEA $M=(Q,\Sigma,s,F,\Delta)$ mit $L(M)=L$

für $q \in Q$ sei $L_q = \{x \in \Sigma^* \mid (q,x) \vdash_M^* (f,\varepsilon) \text{ für irgendein } f \in F\}$

$\mathcal{F}_L = \{L_q \mid q \in Q \text{ mit } q \text{ von } s \text{ erreichbar}\}$ (plus möglicherweise $\{\}$)

und $\mathcal{F}_L$ ist endlich.

Satz (Myhill – Nerode):

L DEA-Sprache $\Leftrightarrow \mathcal{F}_L$ ist endlich

Bew: “ $\Leftarrow$ ”

$\mathcal{F}_L$ endlich. Definiere NEA $M = (\Sigma, Q, s, F, \Delta)$ mit

$$Q = \mathcal{F}_L$$

$$s = F_L(\varepsilon) = L$$

$$F = \{ S \in \mathcal{F}_L \mid \varepsilon \in S \}$$

$$\Delta = \{ (F_L(w), a, F_L(wa)) \mid w \in \Sigma^*, a \in \Sigma \}$$

Dann gilt

$$(F_L(x_1 \cdots x_k), x_{k+1} \cdots x_n) \vdash_M (F_L(x_1 \cdots x_{k+1}), x_{k+2} \cdots x_n)$$

und M akzeptiert genau L .

Konsequenz 4: (DEA Minimierung) Sei M ein DEA. Man kann effektiv einen DEA M' konstruieren, sodass $L(M')=L(M)$ und die Anzahl der Zustände von M ist minimal. Diese Konstruktion braucht nur Zeit polynomiell in der Beschreibungsgröße von M' .

Beweis: $M=(\Sigma,Q,s,F,\Delta)$

für $q \in Q$ sei $L_q = \{x \in \Sigma^* \mid (q,x) \vdash_M^* (f,\varepsilon) \text{ für irgendein } f \in F\}$

$\mathcal{F}_L = \{L_q \mid q \in Q \text{ mit } q \text{ von } s \text{ erreichbar}\}$ (plus möglicherweise $\{\}$)

Idee: Betrachte Übergangsgraphen G_M

- 1) Eliminiere $q \in Q$, die nicht von s erreichbar von G_M
Vervollständige G_M , d.h. füge Knoten $\perp$ hinzu und
für jedes $(q,a) \in (Q \cup \{\perp\}) \times \Sigma$ ohne Übergangsregel
füge Regel (Kante) $(q,a,\perp)$ hinzu.
- 2) Im neuen Graphen bestimme, für welche Knotenpaare
 $\{p,q\}$ gilt $L_p = L_q$. ($\Rightarrow$ Äquivalenzrelation auf den Knoten)
- 3) Aus den Äquivalenzklassen bilde den Minimalautomaten.

Algorithmus für Schritt 2:

Ziel: Berechne $U = \{ \{p,q\} \mid L_p \neq L_q \}$ unterscheidbare Paare
und $N = \{ \{p,q\} \mid L_p = L_q \}$ nicht unterscheidbare Paare

$U := \{ \{p,q\} \mid p \in F \text{ und } q \in Q \setminus F \}$

$N := \{ \{p,q\} \mid p,q \in F \text{ oder } p,q \in Q \setminus F \}$

while $\exists \{p,q\} \in N$ und $\exists \{p',q'\} \in U$ und $\exists a \in \Sigma$
mit $(p,a,p') \in \Delta$ und $(q,a,q') \in \Delta$

do

verschiebe $\{p,q\}$ aus N nach U

Abschlusseigenschaften von regulären Sprachen (DKA-Sprachen)

Satz: Wenn L und L' reguläre Sprachen sind, dann sind auch folgende Sprachen regulär:

- 1) $L \cup L'$, $L \cap L'$, das Komplement von L
- 2) $L \cdot L' = \{ xy \mid x \in L \text{ und } y \in L' \}$ (Konkatenation)
- 3) $L^R = \{ x^R \mid x \in L \}$ (Umkehrung)
- 4) $L^i = L \cdot L^{i-1}$ für $i > 0$ ($L^0 = \{\varepsilon\}$)
- 5) $L^* = \bigcup_{i \in \mathbb{N}} L^i$ (Kleene Stern Operator)