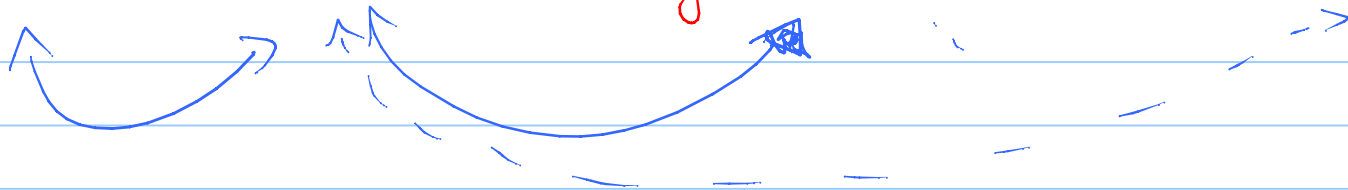


Tutorium, 26.11.2008

Notiztitel

11/5/2008

DEA - NEA - Reg Ausdrücke - Grammatiken



$$L(A) = L((a+b)^*)$$

n.d.V. gut
abc verständlich

[• Äquivalenzbeweise für abstrakte Ausdrücke
mit konkreten Symbolen (z.B. 4.7.2)] ✓

• AP-NBA's ✓

• Aufgabe 4.5

• U 5.6 nicht optimieren während der Umwandlung
nur Teil 1
bei Teil 2 vorher optimieren

$$4.7.2 \quad (RS + R)^* R \equiv R (SR + R)^*$$

$$\begin{aligned} 0 & \\ 1 & \quad (RS + R) R = \underbrace{RS}_R \underbrace{R}_R + \underbrace{RR}_R = R(SR + R) \\ & \quad \underbrace{(RS)(RS) \dots (RS)}_R R \quad R(SR) \dots (SR) \end{aligned}$$

Beweis: $L((RS + R)^* R) = L(R(SR + R)^*)$
für jeden beliebigen Ausdruck R, S

sei $w \in L((RS + R)^* R)$ $L(ac)^* a$

dann

$$w = \overbrace{w_1 \dots w_k}^{w_1 \dots w_k} a$$

$$w = w_1 \dots w_k r_{k+1}$$

$$w = ac \, ac \dots a$$

also

$$w = w_1 \dots w_k r_{k+1}$$

mit $w_i = ac$
 $w_i \in L(RS + R)$
 $r_{k+1} \in L(R)$

$w_1 = a^3$
 $w_i = a$

mit $w_i = r_i s_i$ oder $w_i = r_i$
für $r_i \in L(R)$
und $s_i \in L(S)$

also

$$w = r_1 v_2 \dots v_{k+1}$$

mit $v_i = s_{i-1} r_i$

oder $v_i = r_i$

also

$$w = r_1 v_2 \dots v_{k+1}$$

mit $r_1 \in L(R)$, $v_i \in L(SR + R)$

also

$$w \in L(R(SR + R)^*)$$

Umkehrung genauso

müssen

stattdessen zeige

$$(ab + a)^* a \approx a(b + a)^*$$

$$(a(\underline{b + \epsilon}))^* a \approx a(\underline{(b + \epsilon)} a)^*$$

$$(a \underline{c})^* a \approx a(\underline{c} a)^*$$

$$L(a(ca)^*)$$

AP-NBA's

sind vom theoretischen (Übungs) Konzept

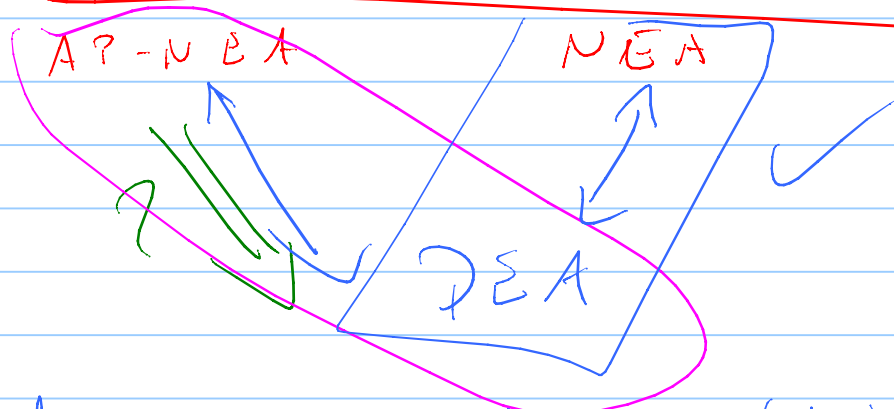
Zeige

AP-NBA

äquivalent zu NBA

jeder Pfad akzeptiert?

$$\emptyset \neq \hat{\delta}(q_0, w) \in F$$



Bedingung

$$DSA = L(A_D) = \{w \mid \hat{\delta}(q_0, w) \in F\}$$

$$NBA = L(A_N) = \{w \mid \hat{\delta}_N(q_0, w) \cap F \neq \emptyset\}$$

$$AP-NFA = L(A_A) = \{ w \mid \emptyset \neq \hat{\delta}_A(q_0, w) \subseteq F \}$$

$$AP-NFA \rightarrow DFA$$

wie bei NFA
aber mit anderer Bedingung an F_D

wie war das bei NFA $\rightarrow$ DFA?

$$A_N \rightarrow A_D = (P(Q_N), \Sigma, \delta_D, \varepsilon\text{-Hülle}(q_0), F_D)$$

$\uparrow$
 (optimiert) $\delta_D(q_D, a) = \{p \mid \exists q \in q_D. p \in \hat{\delta}_N(q, a)\}$

$$F_D = \{ S \in P(Q_N) \mid \boxed{S} \cap F \neq \emptyset \}$$

was ändert sich?

wie oben!

$$A_A \rightarrow A_D = (P(Q_A), \Sigma, \delta_D, \varepsilon\text{-Hülle}(q_0), F_D')$$

$$F_D' = \{ S \in P(Q_A) \mid \boxed{\emptyset \neq S \subseteq F_A} \}$$

$$L(A_D) = \{ w \mid \hat{\delta}_D(\varepsilon\text{-Hülle}(q_0), w) \in F_D' \}$$

$$= \{ w \mid \emptyset \neq \hat{\delta}_D(\varepsilon\text{-Hülle}(q_0), w) \subseteq F_A \}$$

$$= \{ w \mid \emptyset \neq \delta_A(q_0, w) \subseteq F_A \} = L(A_A)$$

$$\rightarrow \delta_D(\varepsilon\text{-Hülle}(q_0), w) = \delta_A(q_0, w)$$

Folie 18 Bin hat 2.2 für NFA

• Der Vollständigkeitshalt habe zeige

jede DFA kann zu äquivalenter (18) NFA
umgewandelt werden

$$\delta_A(q, a) = \{ \delta(q, a) \}$$

4.5 DFA hat endlich viele Zustände

Gesucht

Sprache bei der man beliebig hoch zählen muß!

einfachste Idee "gleich viele 0 und 1
aber beliebig große Wörter"

$$L = \{ 0^n 1^n \mid n \in \mathbb{N} \} = \{ \epsilon, 01, 0011, 000111, \dots \}$$

Annahme $L = L(A)$ und A hat 100 Zustände

betrachte $w = \underbrace{00 \dots 0}_{101} \mid \underbrace{1 \dots 1}_{101}$ ein Zustand i
wurde doppelt
benutzt

dann gibt es eine Zahl $i < 100$ mit
beisatz
von w

$$p = \hat{\delta}(q_0, 0^i) = \hat{\delta}(q_0, \underbrace{0 \dots 0}_{101})$$

$$\Rightarrow \hat{\delta}(q_0, w) = \hat{\delta}(q_0, 0^i 1^{101})$$

A erkennt Wörter $\notin L$

