Syntax und Semantik

Syntax	Semantik
endliche Beschreibungen datenverarbeitender Systeme (von Mensch und Computer bearbeitbar)	Systemverhalten, mögliche Abläufe und Berechnungen (das eigentlich Interessierende, oft unendlich)
Programme	berechnete Funktionen
endliche Automaten	erkannte Sprachen

Syntax und Semantik

- Was verrät Syntax über Semantik? (Z.B. Wortproblem für von endlichen Automaten erkannte Sprachen ist lösbar.)
- Wie lässt sich syntaktisch semantischer Effekt festlegen? (Z.B. Produktautomat erkennt Schnitt.)

Modularisierung und Kompositionalität

- große DV-Systeme werden aus kleineren Komponenten aufgebaut, die selbst aus Unterkomponenten bestehen können usw.
  - welche Bausteine eignen sich?
  - wie können sie zusammengesetzt werden?
  - wie wirkt sich eine Komponente auf das Ganze aus?
- endliche Automaten als Bausteine haben exellente Kompositionalitätseigenschaften.
  - Vereinigung, Konkatenation, Kleene-Hülle, Schnitt, Komplement

Reguläre Sprachen

Sprachoperationen

Gegeben: I: Alphabet, $L, L_1, L_2 \subseteq I^*$

Vereinigung: $L_1 \cup L_2 = \{w \in I^* \mid w \in L_1 \text{ oder } w \in L_2\}$

Konkatenation: $L_1 L_2 = \{w_1 w_2 \mid w_1 \in L_1, w_2 \in L_2\}$

Kleene-Hülle: $L^* = \bigcup_{i \in \mathbb{N}} L^i$ mit $L^0 = \{\lambda\}$ und $L^{i+1} = L^i L$

Reguläre Sprachen

Definition

Die Menge $\mathcal{L}_{REG(I)}$ der regulären Sprachen ist rekursiv wie folgt definiert:

- $\emptyset, \{\lambda\}, \{x\} \in \mathcal{L}_{REG(I)}$ für $x \in I$,
- $L, L_1, L_2 \in \mathcal{L}_{REG(I)}$ impliziert $L_1 \cup L_2$, L_1L_2, $L^* \in \mathcal{L}_{REG(I)}$.

Beispiel

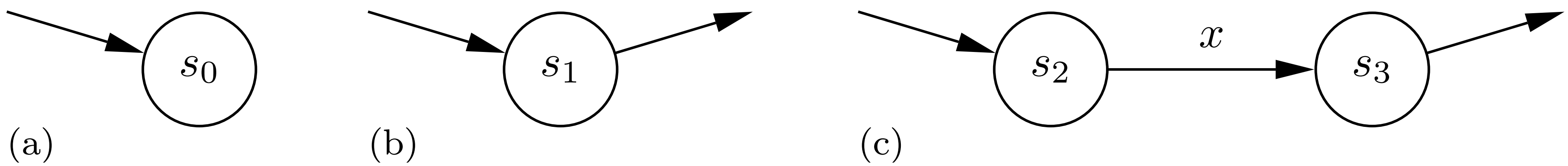
$$(\{\lambda\} \cup (\{a\}\{b\}\{b\}))^* = (\{\lambda\} \cup \{abb\})^* = \{(abb)^n \mid n \in \mathbb{N}\}.$$

Satz

Jede reguläre Sprache wird von einem endlichen Automaten erkannt.

Beweisskizze

Endliche Automaten für $\emptyset$, $\{\lambda\}$ und $\{x\}$:



Vereinigungsautomat

Gegeben: $A_1 = (Z_1, I, d_1, s_{0_1}, F_1)$, $A_2 = (Z_2, I, d_2, s_{0_2}, F_2)$ mit $Z_1 \cap Z_2 = \emptyset$.

$A_1 \cup A_2 = (Z_1 \cup Z_2 \cup \{s_0\}, I, d, s_0, F)$ mit

- $s_0 \notin Z_1 \cup Z_2$;
- $F = \begin{cases} F_1 \cup F_2 \cup \{s_0\}, & \text{falls } s_{0_i} \in F_i, i \in \{1,2\} \\ F_1 \cup F_2 & \text{sonst;} \end{cases}$
- $d = d_1 \cup d_2 \cup \{(s_0, x, s) \mid (s_{0_i}, x, s) \in d_i, i = 1, 2\}$

Satz

$$L(A_1 \cup A_2) = L(A_1) \cup L(A_2).$$

Konkatenationsautomat

Gegeben: $A_1 = (Z_1, I, d_1, s_{0_1}, F_1)$, $A_2 = (Z_2, I, d_2, s_{0_2}, F_2)$
mit $Z_1 \cap Z_2 = \emptyset$.

$A_1 \circ A_2 = (Z_1 \cup Z_2, I, d, s_{0_1}, F)$ mit

- $F = \begin{cases} F_1 \cup F_2, & \text{falls } s_{0_2} \in F_2 \\ F_2 & \text{sonst.} \end{cases}$
- $d = d_1 \cup d_2 \cup \{(s', x, s) \mid (s_{0_2}, x, s) \in d_2, s' \in F_1\}$

Satz

$$L(A_1 \circ A_2) = L(A_1)L(A_2).$$

Sternautomat

Gegeben: $A = (Z, I, d, s_0, F)$.

$A_* = (Z \cup \{s_*\}, I, d_*, s_*, F \cup \{s_*\})$ mit $s_* \notin Z$ und

$$d_* = d \cup \{(s', x, s) \mid (s_0, x, s) \in d, s' \in F \cup \{s_*\}\}$$

Satz

$$L(A_*) = L(A)^*.$$