

1 Chomsky-Hierarchie

Chomsky Typ	Wortproblem	Definition	Bsp	Maschinenmodell
Typ 0	semi-entscheidbar	$G = (\Sigma, V, S, R)$ R beliebig	universelle Sprache	NTM/DTM akzeptiert L
Typ 1	NP-Schwer	$u \rightarrow v, u \leq v $ $u \in V^+, S \notin V$ $S \rightarrow \epsilon$	$L = \{a^i b^i c^i \mid i \leq 1\}$	NTM mit Platzbedarf n erkennt Wörter der Länge n in L $\Rightarrow NTAPE(n)$
Typ 2	polynomiell	$A \rightarrow v, A \in V$ v beliebig	$L = \{a^i b^i \mid i \leq 1\}$	CYK-Alg. erkennt L in polynom. Zeit, Chomsky
Typ 3	linear	$A \rightarrow v, A \in V$ $V \in \epsilon \cup \Sigma \cdot V$	$L = \{a^i \mid i \leq 1\}$	NEA/DEA erkennt L

1.1 Automaten

DEA $A = (Q, \Sigma, \delta : Q \times \Sigma \rightarrow Q, s \in Q, F \subseteq Q)$

NEA $A = (Q, \Sigma, \delta : Q \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^Q, s \in Q, F \subseteq Q)$

NPDA $M = (Q, \Sigma, \Gamma, q_0 \in Q, \delta : Q \times (\Sigma \cup \{\epsilon\}) \times \Gamma \rightarrow 2^{Q \times \Gamma^*}, F \subseteq Q)$

DPDA

DTM $M = (Q, \Sigma, \sqcup \notin \Sigma, \Gamma \supseteq \Sigma \cup \{\sqcup\}, s \in Q, \delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R, N\}, F \subseteq Q)$

NTM $M = (Q, \Sigma, \sqcup \notin \Sigma, \Gamma \supseteq \Sigma \cup \{\sqcup\}, s \in Q, \delta : Q \times (\Gamma \cup \{\epsilon\}) \rightarrow 2^{Q \times \Gamma \times \{L, R, N\}}, F \subseteq Q)$

1.2 Pumping-Lemma

Erfüllt:

" \exists " Wähle $n = 2$

" \forall " Betrachte beliebiges $w \in L$ mit $|w| > 2$

" \exists " Wähle zerlegung $w = uvx$ mit $u = \epsilon, v = aa, x = a^{2(j-1)}$

" \forall " Für alle $i \in \mathbb{N}_0 : uv^i x = a^{2i} a^{2(j-1)} = a^{2(i+j-1)} \in L$

Widerlegen:

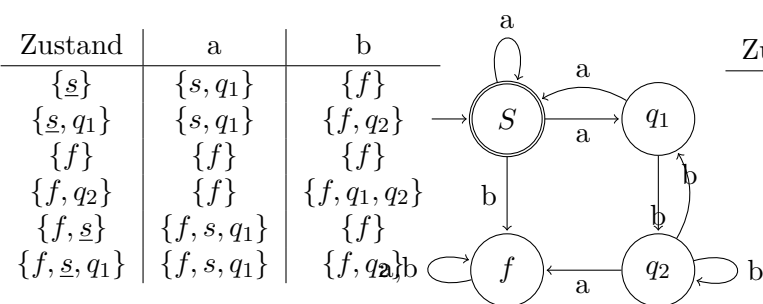
" \exists " Wähle $n = 2$

" \forall " Betrachte beliebiges $w \in L$ mit $|w| > 2$

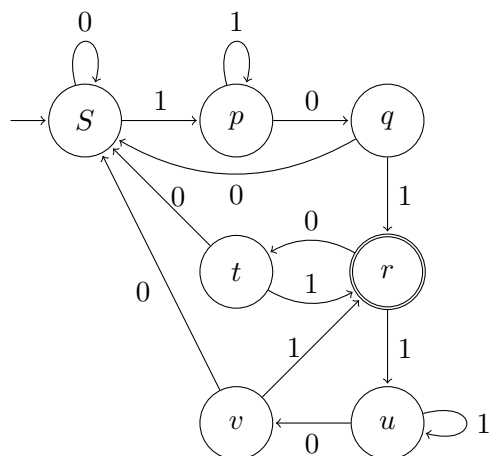
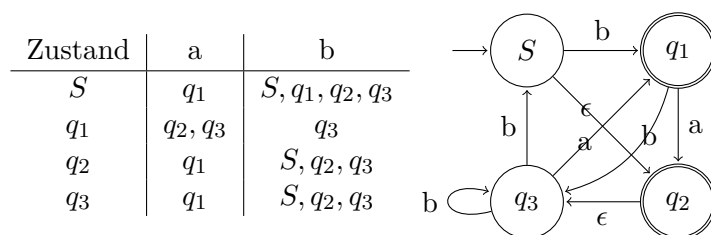
" \exists " Wähle zerlegung $w = uvx$ mit $u = \epsilon, v = aa, x = a^{2(j-1)}$

" \forall " Für alle $i \in \mathbb{N}_0 : uv^i x = a^{2i} a^{2(j-1)} = a^{2(i+j-1)} \in L$

1.2.1 Potenzmengenkonstruktion NEA \rightarrow DEA

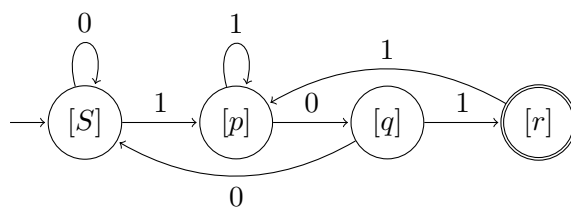


1.2.2 Entfernen von ϵ -Übergängen



s,p,q,r,t,a,v
 ϵ trennt
 r s,p,q,t,u,v
 1 trennt
 s,p,u q,t,v
 0 trennt
 s p,u

1.3 Minimierung von Automaten



1.4 Nerode-Relation

1.5 Chomsky-NF

1. ersetze alle $a \in \Sigma$ in regeln durch neue Variable Y_a und füge $Y_a \rightarrow a$ hinzu.
2. ersetze $A \rightarrow B_1 \dots B_m$ mit $m > 2$ durch $A \rightarrow B_1 C_1, C_i \rightarrow B_{i+1} C_{i+1}, C_{m-2} \rightarrow B_{m-1} B_m$ für $1 \leq i < m - 2$
3. (a) Finde die Menge V' aller Variablen A für die $A \rightarrow^* \epsilon$ existiert
(b) Streiche alle Regeln $A \rightarrow \epsilon$; Für $A \rightarrow BC$ füge ein $A \rightarrow B$ falls $C \in V'$, $A \rightarrow B$ falls $B \in V'$
4. (a) Finde alle Kreise $A_1 \rightarrow A_2 \rightarrow \dots \rightarrow A_n \rightarrow A_1$. Ersetze alle A_i durch A_1 (rechts und links)
(b) Für jede regel $A \rightarrow B$ und jede Regel $B \rightarrow C$ füge Regel $A \rightarrow C$ hinzu, lösche Regel $A \rightarrow B$

Falls $S \rightarrow^* \epsilon$ existierte, füge Startsymbol S' mit Regel $S' \rightarrow S|\epsilon$ hinzu.

2 Kellerautomaten

3 NP-Vollständigkeit

Falls $L_1, L_2 \in NP, L_1 \leq L_2$ und L_1 NP-Schwer, dann ist auch L_2 NP-Schwer. "reduziere polynomiell von L_1 auf L_2 "

3.1 4COLOR $\in NP$

Einen Lösungsvorschlag können wir in $O(|E|)$ verifizieren, indem wir für jede Kante $\{u, v\}$ überprüfen, ob $c(u) \neq c(v)$.

3.2 3COLOR \leq 4COLOR

3.2.1 Transformation

Sei $G = (V, E)$ eine 3COLOR Instanz. Erstelle dann eine 4COLOR Instanz $G' = (V', E')$ mit $V' = V \cup \{u\}, E' = E \cup \{\{w, u\} | u \in V\}$. Die Transformation ist polynomial.

3.2.2 Äquivalenz/Korrektheit

Sei $G = (V, E)$ eine 3COLOR Instanz mit Lösung c . Dann ist $c'(u) = c(u), u \in V$ mit $c'(w) = 3$ eine Lösung für die 4COLOR Instanz G' , da nach Voraussetzung kein Nachbar von w die farbe 3 haben kann.

Sei $G' = (V', E')$ eine 4COLOR Instanz mit Lösung c . Dann ist $c = c'|_V$ eine Lösung für die 3COLOR Instanz G da per Konstruktion kein Knoten die leiche Farbe wie w haben kann.

4 Approximationsalgorithmen

Approximationsalgorithmen

Absolute Approxomation (additiver Fehler)

Relative Approxomation (multiplikativer Fehler)

$R_A(I)$

Minimierungsproblem

$A(I) \leq OPT(I) + K$

$A(I) \leq OPT(I) \cdot K$

$\frac{A(I)}{OPT(I)}$

Maximierungsproblem

$A(I) \geq OPT(I) - K$

$A(I) \geq \frac{OPT(I)}{K}$

$\frac{OPT(I)}{A(I)}$

5 Huffman-Kodierung