Theoretische Grundlagen der Informatik

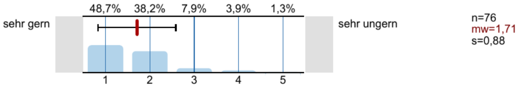
Vorlesung am 31.1.2023

Torsten Ueckerdt | 31. Januar 2023

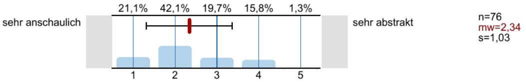
Karlsruher Institut für Technologie

Evaluationsergebnisse

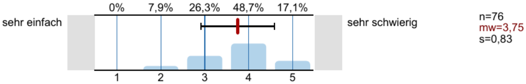
Wie gerne besuchen Sie diese Lehrveranstaltung?



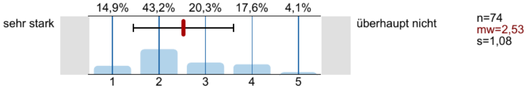
Anschaulichkeit (durch hilfreiche Beispiele)



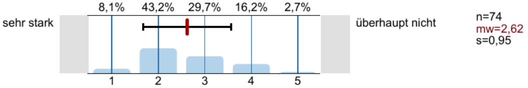
Inhalt



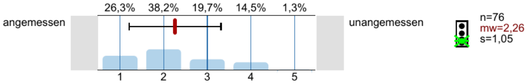
Verweist der Dozent auf aktuelle Forschung?



Verweist der Dozent auf Zusammenhänge zwischen Theorie und Praxis?



Der notwendige Arbeitsaufwand ist. . .



Evaluationsergebnisse

Gut gefallen hat mir insbesondere:

- verständliche Erklärungen (16×)
- Vortragsweise (12×)
- Zusammenfassungen (3×)
- Aufzeichnungen (3×)
- **Testen Sie Sich**-Aufgaben (2×)

Nicht gefallen hat mir insbesondere:

- Tempo zu hoch / zu niedrig (3×)
- lange, “trockene” Abschnitte (2×)
- Folien ohne Klicks $\rightsquigarrow$ Handout mit Definitionen (2×)
- übergangene Details
  z.B. Polynomialität von Reduktionen (2×)
- zu ausführliche GBI-Wiederholungen (2×)
- Dozent übersieht öfters Wortmeldungen
- keine Lösungen für **Testen Sie Sich**-Aufgaben

Letzte Vorlesung

- Pumping-Lemma für kontextfreie Sprachen
- Ogden’s Lemma für kontextfreie Sprachen
- $L = \{a^i b^i c^i \mid i \geq 1\}$ ist kontextsensitiv aber nicht kontextfrei
- Nutzlose Variablen effizient finden und eliminieren
- Leere und endliche kontextfreie Sprachen effizient erkennen
- weitere Eigenschaften kontextfreier Sprachen

Heute:

- Ein Maschinenmodell für kontextfreie Sprachen

Ein Maschinenmodell für Chomsky-2

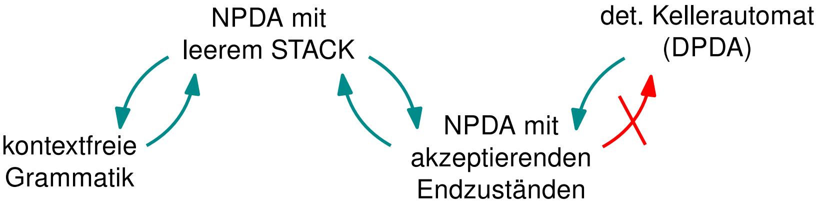
Typ-0	semi-entscheidbar	$\Longleftrightarrow$	DTM / NTM akzeptiert	
Typ-1	kontextsensitiv	$\Longleftrightarrow$	$\mathcal{NTAPE}(n)$	(offen, ob $\mathcal{NTAPE}(n) = \mathcal{DTAPE}(n)$)
Typ-2	kontextfrei	$\Longleftrightarrow$	**???**	
Typ-3	regulär	$\Longleftrightarrow$	DEA / NEA	

Ein Maschinenmodell für Chomsky-2

Typ-0	semi-entscheidbar	$\Longleftrightarrow$	DTM / NTM akzeptiert
Typ-1	kontextsensitiv	$\Longleftrightarrow$	$\mathcal{NTAPE}(n)$ (offen, ob $\mathcal{NTAPE}(n) = \mathcal{DTAPE}(n)$)
Typ-2	kontextfrei	$\Longleftrightarrow$	nichtdet. Kellerautomat (NPDA)
Typ-3	regulär	$\Longleftrightarrow$	DEA / NEA

Ein Maschinenmodell für Chomsky-2

Typ-0	semi-entscheidbar	$\Longleftrightarrow$	DTM / NTM akzeptiert	
Typ-1	kontextsensitiv	$\Longleftrightarrow$	$\mathcal{NTAPE}(n)$	(offen, ob $\mathcal{NTAPE}(n) = \mathcal{DTAPE}(n)$)
Typ-2	kontextfrei	$\Longleftrightarrow$	nichtdet. Kellerautomat (NPDA)	
Typ-3	regulär	$\Longleftrightarrow$	DEA / NEA	



Kellerautomaten

Definition.

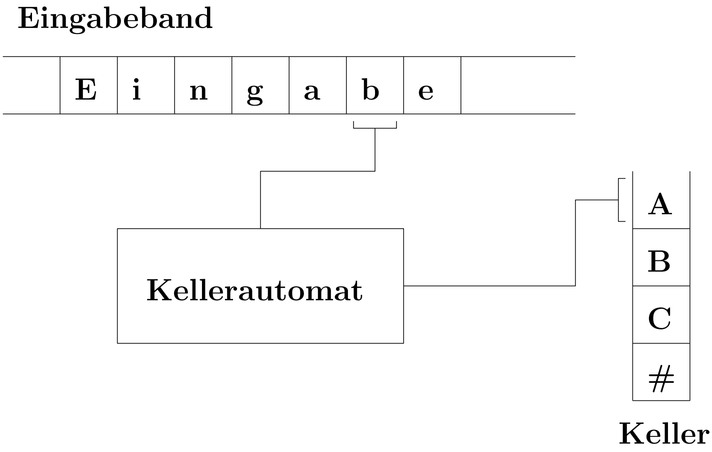
Ein nichtdeterministischer Kellerautomat (NPDA, Pushdown Automaton) besteht aus $(Q, \Sigma, \Gamma, q_0, Z_0, \delta, F)$, wobei

- Q endliche Zustandsmenge
- Σ endliches Eingabealphabet
- Γ endliches STACK-Alphabet
- $q_0 \in Q$ Anfangszustand
- $Z_0 \in \Gamma$ Initialisierung des STACK
- $\delta\colon\ Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \longrightarrow 2^{Q \times \Gamma^*}$ Übergangsrelation, d.h.
  - $\delta(q, a, Z) \subseteq \{(q, \gamma) \mid q \in Q, \gamma \in \Gamma^*\}$
  - $\delta(q, \varepsilon, Z) \subseteq \{(q, \gamma) \mid q \in Q, \gamma \in \Gamma^*\}$
- $F \subseteq Q$ Menge der akzeptierenden Endzustände.

Bemerkung:

$F = \emptyset$ ist möglich.

Kellerautomaten – Visualisierung



Kellerautomaten – Arbeitsweise

Definition.

Eine Konfiguration eines NPDA ist ein Tripel (q, w, α) mit

- $q \in Q$ aktueller Zustand,
- $w \in \Sigma^*$ der Teil der Eingabe, der noch nicht gelesen wurde,
- $\alpha \in \Gamma^*$ STACK-Inhalt.

Definition.

Zu Konfiguration $(q, w_1 \cdots w_k, Z_1 \cdots Z_m)$ gibt es die Nachfolgekonfigurationen:

$(q', w_2 \cdots w_k, Z_1' \cdots Z_r' Z_2 \cdots Z_m)$ für alle $(q', Z_1' \cdots Z_r') \in \delta(q, w_1, Z_1)$

und

$(q', w_1 \cdots w_k, Z_1' \cdots Z_r' Z_2 \cdots Z_m)$ für alle $(q', Z_1' \cdots Z_r') \in \delta(q, \varepsilon, Z_1)$.

Kellerautomaten – Arbeitsweise

Definition.

Ein NPDA akzeptiert ein $w \in \Sigma^*$ durch leeren STACK, wenn es eine zulässige Folge von Konfigurationen aus der Anfangskonfiguration (q_0, w, Z_0) in eine Konfiguration $(q, \varepsilon, \varepsilon)$, $q \in Q$, gibt.

Definition.

Ein NPDA akzeptiert ein $w \in \Sigma^*$ durch einen akzeptierenden Endzustand, wenn es eine zulässige Folge von Konfigurationen aus der Anfangskonfiguration (q_0, w, Z_0) in eine Konfiguration (q, ε, γ) mit $q \in F$ und $\gamma \in \Gamma^*$ gibt.

Definition.

Ein NPDA ist deterministisch (DPDA), falls

$$|\delta(q, a, Z)| + |\delta(q, \varepsilon, Z)| \leq 1$$

für alle $q \in Q$, $a \in \Sigma$, $Z \in \Gamma$.

Kellerautomaten – Beispiel

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$. Informelle Beschreibung:

- Betrachte beliebiges Wort $w_1 \cdots w_n\#w_n \cdots w_1 \in L$.

Phase 1

- Lies $w_1 \cdots w_n$ und schreibe jeweils w_i auf den STACK bis $\#$ gelesen.

Phase 2

- Lies $w_n \cdots w_1$ und vergleiche den jeweils gelesenen Buchstaben mit dem jeweils obersten Buchstaben auf dem STACK.
  - Gleichheit: Nimm obersten Buchstaben vom STACK
  - Sonst: Stoppe in nichtakzeptierenden Zustand

Phase 3

- Ist nur noch Z_0 auf dem STACK
  - Entferne Z_0
  - Akzeptiere die Eingabe “mit leerem STACK”

Kellerautomaten – Beispiel

Karlsruher Institut für Technologie

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$\mathcal{A} = (Q, \Sigma, \Gamma, q_0, Z_0, \delta, F)$$

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\} && \text{Phase 1}\\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\} &&\\
\delta(q_1, 0, 0) &= \{(q_1, 00)\} &&\\
\delta(q_1, 0, 1) &= \{(q_1, 01)\} &&\\
\delta(q_1, 1, 0) &= \{(q_1, 10)\} &&\\
\delta(q_1, 1, 1) &= \{(q_1, 11)\} &&\\
\delta(q_1, \#, 0) &= \{(q_2, 0)\} && \text{Trennzeichen gelesen} \Rightarrow \text{Zu Phase 2}\\
\delta(q_1, \#, 1) &= \{(q_2, 1)\} &&\\
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\} && \text{Phase 2}\\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\} &&\\
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\} && \text{Phase 3}
\end{aligned}
$$

$$Q = \{q_0, q_1, q_2\}$$

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

$$\Gamma = \Sigma \cup \{Z_0\}$$

$$F = \emptyset$$

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\} \\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\} \\[0.5em]
\delta(q_1, 0, 0) &= \{(q_1, 00)\} \\
\delta(q_1, 0, 1) &= \{(q_1, 01)\} \\
\delta(q_1, 1, 0) &= \{(q_1, 10)\} \\
\delta(q_1, 1, 1) &= \{(q_1, 11)\} \\[0.5em]
\delta(q_1, \#, 0) &= \{(q_2, 0)\} \\
\delta(q_1, \#, 1) &= \{(q_2, 1)\} \\[0.5em]
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\} \\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\} \\[0.5em]
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\}\\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\}\\
\delta(q_1, 0, 0) &= \{(q_1, 00)\}\\
\delta(q_1, 0, 1) &= \{(q_1, 01)\}\\
\delta(q_1, 1, 0) &= \{(q_1, 10)\}\\
\delta(q_1, 1, 1) &= \{(q_1, 11)\}\\
\delta(q_1, \#, 0) &= \{(q_2, 0)\}\\
\delta(q_1, \#, 1) &= \{(q_2, 1)\}\\
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\}\\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\}\\
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\} \\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\} \\[6pt]
\delta(q_1, 0, 0) &= \{(q_1, 00)\} \\
\delta(q_1, 0, 1) &= \{(q_1, 01)\} \\
\delta(q_1, 1, 0) &= \{(q_1, 10)\} \\
\delta(q_1, 1, 1) &= \{(q_1, 11)\} \\[6pt]
\delta(q_1, \#, 0) &= \{(q_2, 0)\} \\
\delta(q_1, \#, 1) &= \{(q_2, 1)\} \\[6pt]
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\} \\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\} \\[6pt]
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0
q_1	$01\#100$	$0Z_0$

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\}\\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\}\\[4pt]
\delta(q_1, 0, 0) &= \{(q_1, 00)\}\\
\delta(q_1, 0, 1) &= \{(q_1, 01)\}\\
\delta(q_1, 1, 0) &= \{(q_1, 10)\}\\
\delta(q_1, 1, 1) &= \{(q_1, 11)\}\\[4pt]
\delta(q_1, \#, 0) &= \{(q_2, 0)\}\\
\delta(q_1, \#, 1) &= \{(q_2, 1)\}\\[4pt]
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\}\\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\}\\[4pt]
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0
q_1	$01\#100$	$0Z_0$
q_1	$1\#100$	$00Z_0$

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\}\\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\}\\[1ex]
\delta(q_1, 0, 0) &= \{(q_1, 00)\}\\
\delta(q_1, 0, 1) &= \{(q_1, 01)\}\\
\delta(q_1, 1, 0) &= \{(q_1, 10)\}\\
\delta(q_1, 1, 1) &= \{(q_1, 11)\}\\[1ex]
\delta(q_1, \#, 0) &= \{(q_2, 0)\}\\
\delta(q_1, \#, 1) &= \{(q_2, 1)\}\\[1ex]
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\}\\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\}\\[1ex]
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0
q_1	$01\#100$	$0Z_0$
q_1	$1\#100$	$00Z_0$
q_1	$\#100$	$100Z_0$

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\} \\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\} \\[1ex]
\delta(q_1, 0, 0) &= \{(q_1, 00)\} \\
\delta(q_1, 0, 1) &= \{(q_1, 01)\} \\
\delta(q_1, 1, 0) &= \{(q_1, 10)\} \\
\delta(q_1, 1, 1) &= \{(q_1, 11)\} \\[1ex]
\delta(q_1, \#, 0) &= \{(q_2, 0)\} \\
\delta(q_1, \#, 1) &= \{(q_2, 1)\} \\[1ex]
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\} \\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\} \\[1ex]
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0
q_1	$01\#100$	$0Z_0$
q_1	$1\#100$	$00Z_0$
q_1	$\#100$	$100Z_0$
q_2	100	$100Z_0$

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\}\\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\}\\
\\
\delta(q_1, 0, 0) &= \{(q_1, 00)\}\\
\delta(q_1, 0, 1) &= \{(q_1, 01)\}\\
\delta(q_1, 1, 0) &= \{(q_1, 10)\}\\
\delta(q_1, 1, 1) &= \{(q_1, 11)\}\\
\\
\delta(q_1, \#, 0) &= \{(q_2, 0)\}\\
\delta(q_1, \#, 1) &= \{(q_2, 1)\}\\
\\
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\}\\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\}\\
\\
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0
q_1	$01\#100$	$0Z_0$
q_1	$1\#100$	$00Z_0$
q_1	$\#100$	$100Z_0$
q_2	100	$100Z_0$
q_2	00	$00Z_0$

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\}\\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\}\\[1ex]
\delta(q_1, 0, 0) &= \{(q_1, 00)\}\\
\delta(q_1, 0, 1) &= \{(q_1, 01)\}\\
\delta(q_1, 1, 0) &= \{(q_1, 10)\}\\
\delta(q_1, 1, 1) &= \{(q_1, 11)\}\\[1ex]
\delta(q_1, \#, 0) &= \{(q_2, 0)\}\\
\delta(q_1, \#, 1) &= \{(q_2, 1)\}\\[1ex]
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\}\\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\}\\[1ex]
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0
q_1	$01\#100$	$0Z_0$
q_1	$1\#100$	$00Z_0$
q_1	$\#100$	$100Z_0$
q_2	100	$100Z_0$
q_2	00	$00Z_0$
q_2	0	$0Z_0$

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\}\\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\}\\[4pt]
\delta(q_1, 0, 0) &= \{(q_1, 00)\}\\
\delta(q_1, 0, 1) &= \{(q_1, 01)\}\\
\delta(q_1, 1, 0) &= \{(q_1, 10)\}\\
\delta(q_1, 1, 1) &= \{(q_1, 11)\}\\[4pt]
\delta(q_1, \#, 0) &= \{(q_2, 0)\}\\
\delta(q_1, \#, 1) &= \{(q_2, 1)\}\\[4pt]
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\}\\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\}\\[4pt]
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0
q_1	$01\#100$	$0Z_0$
q_1	$1\#100$	$00Z_0$
q_1	$\#100$	$100Z_0$
q_2	100	$100Z_0$
q_2	00	$00Z_0$
q_2	0	$0Z_0$
q_2		Z_0

Beispiel – Berechnung für Eingabe $001\#100$

Ein DPDA für $L = \{w\#w^R \mid w \in \{0,1\}^*\}$.

$$
\begin{aligned}
\delta(q_0, 0, Z_0) &= \{(q_1, 0Z_0)\}\\
\delta(q_0, 1, Z_0) &= \{(q_1, 1Z_0)\}\\[1ex]
\delta(q_1, 0, 0) &= \{(q_1, 00)\}\\
\delta(q_1, 0, 1) &= \{(q_1, 01)\}\\
\delta(q_1, 1, 0) &= \{(q_1, 10)\}\\
\delta(q_1, 1, 1) &= \{(q_1, 11)\}\\[1ex]
\delta(q_1, \#, 0) &= \{(q_2, 0)\}\\
\delta(q_1, \#, 1) &= \{(q_2, 1)\}\\[1ex]
\delta(q_2, 0, 0) &= \{(q_2, \varepsilon)\}\\
\delta(q_2, 1, 1) &= \{(q_2, \varepsilon)\}\\[1ex]
\delta(q_2, \varepsilon, Z_0) &= \{(q_2, \varepsilon)\}
\end{aligned}
$$

Zustand	Eingabe	STACK
q_0	$001\#100$	Z_0
q_1	$01\#100$	$0Z_0$
q_1	$1\#100$	$00Z_0$
q_1	$\#100$	$100Z_0$
q_2	100	$100Z_0$
q_2	00	$00Z_0$
q_2	0	$0Z_0$
q_2		Z_0

akzeptiert durch leeren STACK

Kellerautomaten – Beispiel 2

Ein NPDA für $L = \{ww^R \mid w \in \{0,1\}^*\}$. Informelle Beschreibung:

- In diesem Fall fehlt das Trennzeichen.
- Der NPDA funktioniert wie der DPDA aus dem letzten Beispiel.
- Der Übergang in Phase 2 funktioniert allerdings nichtdeterministisch.

Bemerkung:

- Für die Sprache $L = \{ww^R \mid w \in \{0,1\}^*\}$ gibt es keinen DPDA.
- NPDAs können also mehr als DPDAs.

Kellerautomaten – Beispiel 2

Ein NPDA für $L = \{ww^R \mid w \in \{0,1\}^*\}$**.**

$\mathcal{A} = (Q, \Sigma, \Gamma, q_0, Z_0, \delta, F)$

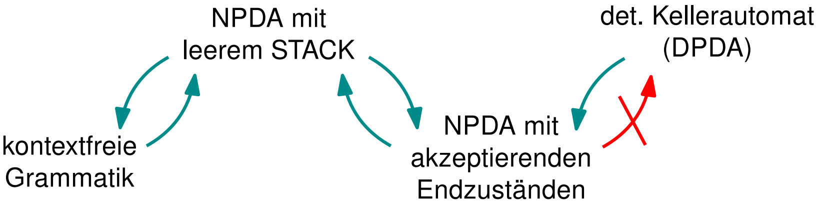
$$\begin{array}{lcll}
\delta(q_0, 0, Z_0) & = & \{(q_1, 0Z_0)\} & \text{Phase 1} \\
\delta(q_0, 1, Z_0) & = & \{(q_1, 1Z_0)\} & \\
\delta(q_1, 0, 0) & = & \{(q_1, 00), (q_2, \epsilon)\} & \\
\delta(q_1, 0, 1) & = & \{(q_1, 01)\} & \\
\delta(q_1, 1, 0) & = & \{(q_1, 10)\} & \\
\delta(q_1, 1, 1) & = & \{(q_1, 11), (q_2, \epsilon)\} & \\
\delta(q_2, 0, 0) & = & \{(q_2, \varepsilon)\} & \text{Phase 2} \\
\delta(q_2, 1, 1) & = & \{(q_2, \varepsilon)\} & \\
\delta(q_2, \varepsilon, Z_0) & = & \{(q_2, \varepsilon)\} & \text{Phase 3}
\end{array}$$

$$Q = \{q_0, q_1, q_2\}$$
$$\Sigma = \{0, 1, \#\}$$
$$\Gamma = \Sigma \cup \{Z_0\}$$
$$F = \emptyset$$

Ein Maschinenmodell für Chomsky-2

Satz.

Zu einem NPDA, der eine Sprache L durch einen akzeptierenden Endzustand akzeptiert, kann ein NPDA konstruiert werden, der L mit leerem STACK akzeptiert.



Beweis – Beschreibung

- Sei $\mathcal{A}_1 = (Q_1, \Sigma, \Gamma_1, \delta_1, q_0^1, Z_0^1, F_1)$ NPDA, der L durch Übergang in einen Zustand aus F_1 akzeptiert.
- Wir konstruieren dazu einen NPDA $\mathcal{A}_2 = (Q_2, \Sigma, \Gamma_2, \delta_2, q_0^2, Z_0^2)$, der L durch leeren STACK akzeptiert.
- Seien q_0^2, q_E neue Zustände.
- Sei Z_0^2 ein neues STACK-Symbol.

Idee der Konstruktion von $\mathcal{A}_2$.

- Lege zu Beginn Z_0^2 vor Z_0^1 auf den STACK, so dass der STACK nicht "versehentlich" geleert werden kann.
- Dann Verfahre wie in $\mathcal{A}_1$.
- Wenn Zustand in F_1 erreicht wird: Gehe zu q_E und leere den STACK

Beweis – Konstruktion

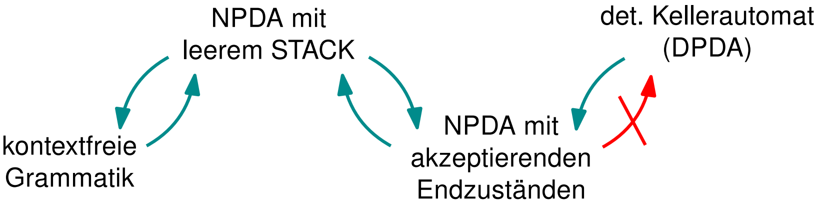
- Gegeben $\mathcal{A}_1 = (Q_1, \Sigma, \Gamma_1, \delta_1, q_0^1, Z_0^1, F_1)$; akzeptiert durch Endzustand
- Wir konstruieren $\mathcal{A}_2 = (Q_2, \Sigma, \Gamma_2, \delta_2, q_0^2, Z_0^2)$; akzeptiert durch leeren STACK
- Seien q_0^2, q_E neue Zustände. Sei Z_0^2 ein neues STACK-Symbol.

$$
\begin{aligned}
Q_2 &= Q_1 \cup \{q_0^2, q_E\} \\
\Gamma_2 &= \Gamma_1 \cup \{Z_0^2\} \\
\delta_2(q_0^2, \varepsilon, Z_0^2) &= \{(q_0^1, Z_0^1 Z_0^2)\} \\
\delta_2(q, a, Z) &= \delta_1(q, a, Z) \text{ für } q \in Q_1,\ a \neq \varepsilon,\ Z \in \Gamma_1 \\
&\qquad\qquad\qquad\quad q \in Q_1 \setminus F_1,\ a = \varepsilon,\ Z \in \Gamma_1 \\
\delta_2(q, \varepsilon, Z) &= \delta_1(q, \varepsilon, Z) \cup \{(q_E, \varepsilon)\} \text{ für } q \in F_1,\ Z \in \Gamma_2 \\
\delta_2(q_E, \varepsilon, Z) &= \{(q_E, \varepsilon)\} \text{ für } Z \in \Gamma_2 \\
\delta(\cdot) &= \emptyset \text{ sonst}
\end{aligned}
$$

Ein Maschinenmodell für Chomsky-2

Satz.

Zu einem NPDA, der eine Sprache L mit leerem STACK akzeptiert, kann ein NPDA konstruiert werden, der L durch einen akzeptierenden Endzustand akzeptiert.



Beweis – Beschreibung

- Sei $\mathcal{A}_1 = (Q_1, \Sigma, \Gamma_1, \delta_1, q_0^1, Z_0^1)$, ein NPDA der $w \in L$ mit leerem STACK akzeptiert.
- Wir konstruieren dazu einen NPDA $\mathcal{A}_2 = (Q_2, \Sigma, \Gamma_2, \delta_2, q_0^2, Z_0^2, F_2)$, der genau die $w \in L$ durch Übergang in einen Zustand $q \in F_2$ akzeptiert.
- Sei Z_0^2 ein neues STACK-Symbol.
- Sei q_F ein neuer (End-)Zustand.
- Sei q_0^2 ein neuer (Anfangs-)Zustand.

Idee der Konstruktion von $\mathcal{A}_2$.

- Lege zu Beginn Z_0^2 vor Z_0^1 auf den STACK, und lösche Z_0^2 nur, wenn die Abarbeitung von $\mathcal{A}_1$ durch leeren STACK akzeptiert hätte.
- Gehe in Endzustand q_F, wenn $\mathcal{A}_1$ durch leeren STACK akzeptiert hätte.

Beweis – Konstruktion

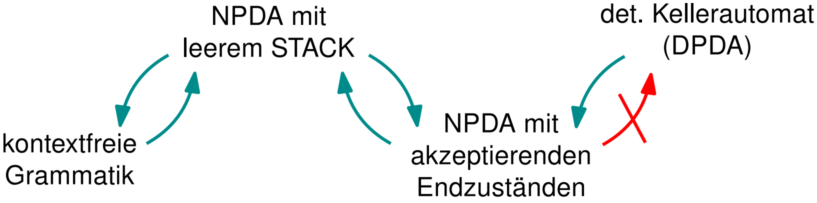
- $\mathcal{A}_1 = (Q_1, \Sigma, \Gamma_1, \delta_1, q_0^1, Z_0^1, F_1)$ akzeptiert durch leeren STACK.
- $\mathcal{A}_2 = (Q_2, \Sigma, \Gamma_2, \delta_2, q_0^2, Z_0^2)$, akzeptiert durch Endzustand.
- q_0^2 neuer Anfangszustand.
- q_F neuer (End-)Zustand.
- Z_0^2 ein neues STACK-Symbol.

$$
\begin{aligned}
Q_2 &= Q_1 \cup \{q_0^2, q_F\}, \\
F_2 &= \{q_F\} \\
\Gamma_2 &= \Gamma_1 \cup \{Z_0^2\} \\
\delta_2(q_0^2, a, X) &= \begin{cases} \{q_0^1, Z_0^1 Z_0^2\} & \text{falls } a = \varepsilon \text{ und } X = Z_0^2 \\ \emptyset & \text{sonst} \end{cases} \\
\delta_2(q, a, Z) &= \delta_1(q, a, Z), \text{ falls } q \in Q_1, a \in \Sigma \cup \{\varepsilon\} \text{ und } Z \in \Gamma_1 \\
\delta_2(q, \varepsilon, Z_0^2) &= \{(q_F, \varepsilon)\} \text{ für } q \in Q_1.
\end{aligned}
$$

Ein Maschinenmodell für Chomsky-2

Satz.

Für eine kontextfreie Grammatik G kann ein NPDA konstruiert werden, der $L(G)$ mit leerem STACK akzeptiert.



Beweis – Konstruktion des NPDA

- Sei $G = (\Sigma, V, S, R)$ eine kontextfreie Grammatik.
  - Jede Regel in R ist von der Form $A \to r$ mit $A \in V$ und $r \in (\Sigma \cup V)^*$.
- Für jedes $a \in \Sigma$, ersetze jedes Vorkommen von a in R durch neue Variable Y_a.
  Füge neue Regel $Y_a \to a$ hinzu.
- $\rightsquigarrow$ Erhalte äquivalente kontextfreie Grammatik $G' = (\Sigma, V', S, R')$.
  - Jede Regel in R' ist von der Form $A \to r$ mit $A \in V'$ und $r \in \Sigma \cup V'^*$.

Beweis – Konstruktion des NPDA

- Sei $G = (\Sigma, V, S, R)$ eine kontextfreie Grammatik.
  - Jede Regel in R ist von der Form $A \to r$ mit $A \in V$ und $r \in (\Sigma \cup V)^*$.
- Für jedes $a \in \Sigma$, ersetze jedes Vorkommen von a in R durch neue Variable Y_a.
  Füge neue Regel $Y_a \to a$ hinzu.
- $\rightsquigarrow$ Erhalte äquivalente kontextfreie Grammatik $G' = (\Sigma, V', S, R')$.
  - Jede Regel in R' ist von der Form $A \to r$ mit $A \in V'$ und $r \in \Sigma \cup V'^*$.

Konstruiere gewünschten Automaten $\mathcal{A} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$.

$$Q := \{q_0\} \quad \Gamma := V' \quad Z_0 := S$$

$$\delta(q_0, a, A) := \{(q_0, \varepsilon) \mid (A \to a) \in R', a \in \Sigma\}$$

$$\delta(q_0, \varepsilon, A) := \{(q_0, r) \mid (A \to r) \in R', r \in V'^*\}$$

Beweis – Konstruktion des NPDA

- Sei $G = (\Sigma, V, S, R)$ eine kontextfreie Grammatik.
  - Jede Regel in R ist von der Form $A \to r$ mit $A \in V$ und $r \in (\Sigma \cup V)^*$.
- Für jedes $a \in \Sigma$, ersetze jedes Vorkommen von a in R durch neue Variable Y_a.
  Füge neue Regel $Y_a \to a$ hinzu.
- $\rightsquigarrow$ Erhalte äquivalente kontextfreie Grammatik $G' = (\Sigma, V', S, R')$.
  - Jede Regel in R' ist von der Form $A \to r$ mit $A \in V'$ und $r \in \Sigma \cup V'^*$.

Konstruiere gewünschten Automaten $\mathcal{A} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$.

$$Q := \{q_0\} \qquad \Gamma := V' \qquad Z_0 := S$$

$$\delta(q_0, a, A) := \{(q_0, \varepsilon) \mid (A \to a) \in R', a \in \Sigma\}$$

$$\delta(q_0, \varepsilon, A) := \{(q_0, r) \mid (A \to r) \in R', r \in V'^*\}$$

Wir werden beweisen:

- $S \xrightarrow{*} w_1 \cdots w_n$ in $L(G') \Leftrightarrow \mathcal{A}$ erkennt $w_1 \cdots w_n$ mit leerem STACK

Beweis – Der Plan

Konstruktion: $Q := \{q_0\} \quad \Gamma := V' \quad Z_0 := S$

$$\delta(q_0, a, A) := \{(q_0, \varepsilon) \mid (A \longrightarrow a) \in R', a \in \Sigma\}$$
$$\delta(q_0, \varepsilon, A) := \{(q_0, r) \mid (A \longrightarrow r) \in R', r \in V'^*\}$$

Wir werden beweisen:

- $S \xrightarrow{*} w_1 \cdots w_n$ in $L(G') \Leftrightarrow \mathcal{A}$ erkennt $w_1 \cdots w_n$ mit leerem STACK

Beweis – Der Plan

Konstruktion: $Q := \{q_0\} \quad \Gamma := V' \quad Z_0 := S$

$$\delta(q_0, a, A) := \{(q_0, \varepsilon) \mid (A \to a) \in R', a \in \Sigma\}$$

$$\delta(q_0, \varepsilon, A) := \{(q_0, r) \mid (A \to r) \in R', r \in V'^*\}$$

Wir werden beweisen:

- $S \xrightarrow{*} w_1 \cdots w_n$ in $L(G') \Leftrightarrow \mathcal{A}$ erkennt $w_1 \cdots w_n$ mit leerem STACK

Wir beweisen mittels Induktion die stärkere Aussage für jedes t:

$S \xrightarrow{t} w_1 \cdots w_i A_1 \cdots A_m$ mittels Linksableitung $\Leftrightarrow$ $\mathcal{A}$ kann beim Lesen von $w_1 \cdots w_i$ in t Schritten den STACK-Inhalt $A_1 \cdots A_m$ erzeugen.

Induktionsanfang: $t = 0$

- $S \xrightarrow{0} S \quad \Leftrightarrow \quad \mathcal{A}$ initialisiert STACK $Z_0 = S$

Beweis – Korrektheit

Induktionsschritt: $t \geq 1$

$S \xrightarrow{t} w_1 \cdots w_i A_1 \cdots A_m$

- Betrachte als $S \xrightarrow{t-1} w_1 \cdots w_{i'} B_1 \cdots B_{m'}$ und einen weiteren Schritt $B_1 \to r$.
- Nach Induktionsvoraussetzung gilt

  $S \xrightarrow{t-1} w_1 \cdots w_{i'} B_1 \cdots B_{m'} \Leftrightarrow \mathcal{A}$ kann beim Lesen von $w_1 \cdots w_{i'}$ in $t-1$ Schritten den STACK-Inhalt $B_1 \cdots B_{m'}$ erzeugen

1. Fall: $r = a$ mit $a \in \Sigma$

- D.h. $i' = i - 1$, $w_i = a$ und $A_1 \cdots A_m = B_2 \cdots B_{m'}$.
- Der t-te Schritt von $\mathcal{A}$ ist $(q_0, \varepsilon) \in \delta(q_0, a, B_1)$.

2. Fall: $r \in V'^*$

- D.h. $i' = i$ und $A_1 \cdots A_m = r\, B_2 \cdots B_{m'}$.
- Der t-te Schritt von $\mathcal{A}$ ist $(q_0, r) \in \delta(q_0, \varepsilon, B_1)$.

$\Rightarrow \mathcal{A}$ kann in t Schritten STACK-Inhalt $A_1 \cdots A_m$ erzeugen.

Beweis – Korrektheit

Induktionsschritt: $t \geq 1$

$S \xrightarrow{t} w_1 \cdots w_i A_1 \cdots A_m$

- Betrachte als $S \xrightarrow{t-1} w_1 \cdots w_{i'} B_1 \cdots B_{m'}$ und einen weiteren Schritt $B_1 \rightarrow r$.
- Nach Induktionsvoraussetzung gilt
  $S \xrightarrow{t-1} w_1 \cdots w_{i'} B_1 \cdots B_{m'} \Leftrightarrow \mathcal{A}$ kann beim Lesen von $w_1 \cdots w_{i'}$ in $t-1$ Schritten den STACK-Inhalt $B_1 \cdots B_{m'}$ erzeugen

1. Fall: $r = a$ mit $a \in \Sigma$

- D.h. $i' = i - 1$, $w_i = a$ und $A_1 \cdots A_m = B_2 \cdots B_{m'}$.
- Der t-te Schritt von $\mathcal{A}$ ist $(q_0, \varepsilon) \in \delta(q_0, a, B_1)$.

2. Fall: $r \in V'^*$

- D.h. $i' = i$ und $A_1 \cdots A_m = r\, B_2 \cdots B_{m'}$.
- Der t-te Schritt von $\mathcal{A}$ ist $(q_0, r) \in \delta(q_0, \varepsilon, B_1)$.

$\Leftrightarrow \mathcal{A}$ kann in t Schritten STACK-Inhalt $A_1 \cdots A_m$ erzeugen.

Ein Maschinenmodell für Chomsky-2

Satz.

Jede durch einen NPDA durch leeren STACK akzeptierte Sprache ist kontextfrei.