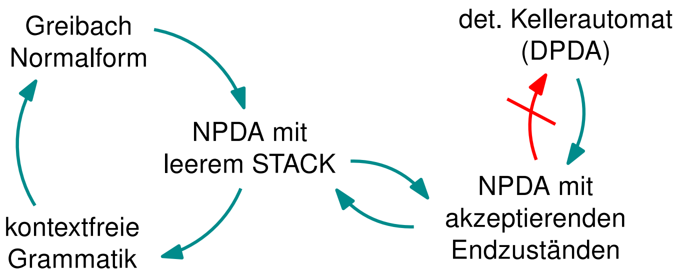
Theoretische Grundlagen der Informatik
Vorlesung am 23. Januar 2020

INSTITUT FÜR THEORETISCHE INFORMATIK

Letzte Vorlesung



Heute beweisen wir:

- Greibach-Normalform $\Longrightarrow$ NPDA mit leerem STACK
- NPDA mit leerem STACK $\Longrightarrow$ kontextfreie Grammatik

Wdh.: Greibach-Normalform, Kellerautomat

Eine kontextfreie Grammatik ist in **Greibach-Normalform**, wenn alle Regeln von der Form $A \to a\alpha$ mit $A \in V$, $a \in \Sigma$, $\alpha \in V^*$ sind.

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

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

Wdh.: Greibach-Normalform, Kellerautomat

Karlsruher Institut für Technologie

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

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

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.

Satz:
Für eine Grammatik G in Greibach-Normalform kann ein NPDA konstruiert werden, der $L(G)$ durch leeren STACK akzeptiert.

Beispiel – Greibach-Normalform

Zwei äquivalente Grammatiken über $\Sigma = \{0, 1\}$:

G_1 in Chomsky-Normalform

$$\begin{aligned} V &= \{S, A_1, A_0\} \\ R &= \{S \to A_1 A_0, \\ &\quad A_1 \to A_0 S | 1, \\ &\quad A_0 \to S A_1 | 0 \end{aligned}$$

G_2 in Greibach-Normalform

$$\begin{aligned} V &= \{S, A_1, A_0, B\} \\ R &= \{S \to \cdots | 1 A_0 A_1 S A_0 | 1 A_0 | \cdots, \\ &\quad A_1 \to \cdots | 1 | \cdots, \\ &\quad A_0 \to \cdots | 0 | \cdots, \\ &\quad B \to \cdots \end{aligned}$$

$$\begin{aligned} G_1\!: \; S &\to A_1 A_0 \to A_0 S A_0 \to S A_1 S A_0 \to A_1 A_0 A_1 S A_0 \\ &\to 1 A_0 A_1 S A_0 \to 10 A_1 S A_0 \to 101 S A_0 \to 101 A_1 A_0 A_0 \\ &\to 1011 A_0 A_0 \to 10110 A_0 \to 101100 \end{aligned}$$

$$\begin{aligned} G_2\!: \; S &\to 1 A_0 A_1 S A_0 \to 10 A_1 S A_0 \\ &\to 101 S A_0 \to 1011 A_0 A_0 \to 10110 A_0 \to 101100 \end{aligned}$$

Beispiel – Kellerautomat

G_2 in Greibach-Normalform

$$\begin{aligned}
V &= \{S, A_1, A_0, B\} \\
R &= \{S \to \cdots | 1A_0A_1SA_0, \\
&\qquad\qquad 1A_0 | \cdots, \\
&\quad A_1 \to \cdots |1| \cdots, \\
&\quad A_0 \to \cdots |0| \cdots, \\
&\quad B \to \cdots
\end{aligned}$$

$\mathcal{A}$ Kellerautomat

$$\begin{aligned}
Q &= \{q_0\} \\
\Gamma &= \{S, A_1, A_0, B\} \\
Z_0 &= S \\
\delta(q_0, 1, S) &= \{\ldots, (q_0, A_0A_1SA_0), \\
&\qquad (q_0, A_0), \ldots\} \\
\delta(q_0, 1, A_1) &= \{\ldots, (q_0, \varepsilon), \ldots\} \\
\delta(q_0, 0, A_0) &= \{\ldots, (q_0, \varepsilon), \ldots\}
\end{aligned}$$

G_2: $S \to 1A_0A_1SA_0 \to 10A_1SA_0$
$\to 101SA_0 \to 1011A_0A_0 \to 10110A_0 \to 101100$

$\mathcal{A}$: $(q_0, 101100, S) \vdash (q_0, 01100, A_0A_1SA_0) \vdash (q_0, 1100, A_1SA_0)$
$\vdash (q_0, 100, SA_0) \vdash (q_0, 00, A_0A_0) \vdash (q_0, 0, A_0) \vdash (q_0, \varepsilon, \varepsilon)$

Beweis: Greibach-Normalform $\rightarrow$ NPDA

- Sei $G = (\Sigma, V, S, R)$ eine Grammatik in Greibach-Normalform
- Konstruiere gewünschten Kellerautomaten $\mathcal{A} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$:

$$\begin{aligned} Q &:= \{q_0\} \\ \Gamma &:= V \\ Z_0 &:= S \\ \delta(q_0, a, A) &:= \{(q_0, \alpha) \mid (A \rightarrow a\alpha) \in R\} \end{aligned}$$

Per Induktion über die Länge i einer Ableitung beweisen wir:

- $S \xrightarrow{*} w_1 \cdots w_i A_1 \cdots A_m \quad \Leftrightarrow \quad \mathcal{A}$ kann beim Lesen von $w_1 \cdots w_i$ den STACK-Inhalt $A_1 \cdots A_m$ erzeugen. Möglicherweise ist $A_1 \cdots A_m = \varepsilon$.

Daraus folgt:

- $\mathcal{A}$ erkennt $w_1 \cdots w_n$ durch leeren STACK $\Leftrightarrow S \xrightarrow{*} w_1 \cdots w_n$ in G

Beweis: Greibach-Normalform $\to$ NPDA

$$Q := \{q_0\} \quad \Gamma := V \quad Z_0 := S$$
$$\delta(q_0, a, A) := \{(q_0, \alpha) \mid (A \to a\alpha) \in R\}$$

Induktionsanfang ist mit $i = 0$ trivialerweise erfüllt.
Induktionsschritt:
Sei $i \geq 1$ und "$\xrightarrow{j}$" stehe für eine Ableitung der Länge j. Dann gilt:

$$S \xrightarrow{i} w_1 \cdots w_i A_1 \cdots A_m \iff \begin{array}{l} \exists A' \in V, r \in \{1, \dots, m\} \text{ mit} \\ S \xrightarrow{i-1} w_1 \cdots w_{i-1} A' A_r \cdots A_m \\ \to w_1 \cdots w_i A_1 \cdots A_m. \end{array}$$

Mit Induktionsvoraussetzung ist dies äquivalent zu

$\exists A' \in V, r \in \{1, \dots, m\}$ sodass

- $\mathcal{A}$ das Wort $w_1 \cdots w_{i-1}$ lesen und dabei STACK-Inhalt $A' A_r \cdots A_m$ erzeugen kann, und
- $A' \to w_i A_1 \cdots A_{r-1}$ Regel von G ist.

Beweis: Greibach-Normalform $\rightarrow$ NPDA

$$Q := \{q_0\} \quad \Gamma := V \quad Z_0 := S$$
$$\delta(q_0, a, A) := \{(q_0, \alpha) \mid (A \rightarrow a\alpha) \in R\}$$

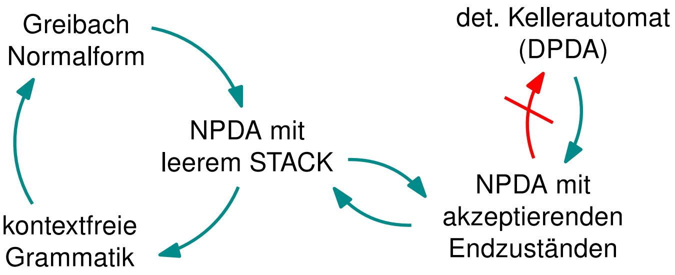
Induktionsanfang ist mit $i = 0$ trivialerweise erfüllt.
Induktionsschritt:
Sei $i \geq 1$ und "$\stackrel{j}{\rightarrow}$" stehe für eine Ableitung der Länge j.
Mit Induktionsvoraussetzung ist dies äquivalent zu

$\exists A' \in V, r \in \{1, \ldots, m\}$ sodass

- $\mathcal{A}$ das Wort $w_1 \cdots w_{i-1}$ lesen und dabei STACK-Inhalt $A' A_r \cdots A_m$ erzeugen kann, und
- $A' \rightarrow w_i A_1 \cdots A_{r-1}$ Regel von G ist.

Dies ist genau dann erfüllt, wenn $\mathcal{A}$ das Wort $w_1 \cdots w_i$ lesen und dabei den STACK-Inhalt $A_1 \cdots A_m$ erzeugen kann.

Übersicht



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

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

- Sei $\mathcal{A} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$ NPDA, der $L_\mathcal{A}$ durch leeren STACK akzeptiert.
- Wir geben eine kontextfreie Grammatik $G = (\Sigma, V, S, R)$ mit $L_\mathcal{A} = L(G)$ an.

Die Konstruktion von G heißt **Tripelkonstruktion**.

- Setze $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- Sei S Startsymbol.

Ziel: Aus $[q, X, p]$ sollen genau die $w \in \Sigma^*$ ableitbar sein, für die es eine Abarbeitung von $\mathcal{A}$ gibt,

- die im Zustand q mit STACK-Inhalt X beginnt und
- nach Lesen von w im Zustand p mit leerem STACK endet.

Beweis: NPDA $\to$ kontextfreie Grammatik

Karlsruher Institut für Technologie

- Sei $\mathcal{A} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0)$ NPDA, der $L_{\mathcal{A}}$ durch leeren STACK akzeptiert.
- Wir geben eine kontextfreie Grammatik $G = (\Sigma, V, S, R)$ mit $L_{\mathcal{A}} = L(G)$ an.

Die Konstruktion von G heißt **Tripelkonstruktion**.

- Setze $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- Sei S Startsymbol.

Die Regelmenge R ist gegeben durch

- $S \to [q_0, Z_0, q]$ für alle $q \in Q$,
- $[q, X, q_{m+1}] \to a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$ für alle Möglichkeiten $q_2, q_3, \ldots, q_{m+1} \in Q$, falls $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$,
- insbes. $[q, X, p] \to a$, falls $(p, \varepsilon) \in \delta(q, a, X)$.

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Für eine Folge von Konfigurationen (q, w, α) nach (p, w', β) schreiben wir auch

$$(q, w, \alpha) \overset{*}{\vdash} (p, w', \beta)$$

beziehungsweise

$$(q, w, \alpha) \overset{k}{\vdash} (p, w', \beta)$$

für eine Folge von genau k Konfigurationen.

Beweis: NPDA $\to$ kontextfreie Grammatik

Wir werden per Induktion beweisen, dass für alle $p, q \in Q$, $X \in \Gamma$ und $w \in L$ gilt:

$$[q, X, p] \overset{*}{\to} w \text{ in } G \iff (q, w, X) \overset{*}{\vdash} (p, \varepsilon, \varepsilon)$$

Aus dieser Behauptung folgt dann:

$$\begin{aligned}
w \in L_{\mathcal{A}} &\iff \exists p \in Q \text{ mit } (q_0, w, Z_0) \overset{*}{\vdash} (p, \varepsilon, \varepsilon), \text{ wobei} \\
& (q_0, w, Z_0) \text{ Anfangskonfiguration von } \mathcal{A} \text{ ist} \\
&\iff \exists p \in Q \text{ mit } [q_0, Z_0, p] \overset{*}{\to} w \\
&\iff \exists p \in Q \text{ mit } S \to [q_0, Z_0, p] \overset{*}{\to} w \\
&\iff w \in L(G)
\end{aligned}$$

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Richtung

$$[q, X, p] \xrightarrow{*} w \text{ in } G \quad \Longrightarrow \quad (q, w, X) \overset{*}{\vdash} (p, \varepsilon, \varepsilon)$$

Beschreibung:

- Induktion über die Länge k einer Ableitung $[q, X, p] \xrightarrow{k} w$ in G

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Wir zeigen: $[q, X, p] \overset{k}{\rightarrow} w$ in $G \Longrightarrow (q, w, X) \overset{*}{\vdash} (p, \varepsilon, \varepsilon)$

- $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- $[q, X, q_{m+1}] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$
  für alle $q_2, q_3, \ldots, q_{m+1} \in Q$, mit $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$.

Induktionsanfang:

- Für $k = 1$ gilt, dass $[q, X, p] \rightarrow w$ eine Regel in G ist.
- Also ist $(p, \varepsilon) \in \delta(q, w, X)$ und $|w| = 1$.
- Also gibt es die Abarbeitung $(q, w, X) \overset{1}{\vdash} (p, \varepsilon, \varepsilon)$ in $\mathcal{A}$.

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Wir zeigen: $[q, X, p] \overset{k}{\rightarrow} w$ in $G \Longrightarrow (q, w, X) \overset{*}{\vdash} (p, \varepsilon, \varepsilon)$

- $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- $[q, X, q_{m+1}] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$ für alle $q_2, q_3, \ldots, q_{m+1} \in Q$, mit $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$.

Induktionsschritt:

- Betrachte eine Ableitung $[q, X, p] \overset{k}{\rightarrow} w$.
- Schreibe diese als

$$[q, X, p] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}] \overset{k-1}{\rightarrow} w,$$

wobei $q_{m+1} = p$ und $w = aw_1 \cdots w_m$, mit $w_i \in \Sigma^*$, $a \in \Sigma$ und $[q_j, Y_j, q_{j+1}] \overset{k'}{\rightarrow} w_j$ mit $k' \leq k - 1$ für alle $1 \leq j \leq m$.

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Wir zeigen: $[q, X, p] \overset{k}{\rightarrow} w$ in $G \Longrightarrow (q, w, X) \overset{*}{\vdash} (p, \varepsilon, \varepsilon)$

- $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- $[q, X, q_{m+1}] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$
  für alle $q_2, q_3, \ldots, q_{m+1} \in Q$, mit $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$.

Induktionsschritt:

- Induktionsvoraussetzung: $(q_j, w_j, Y_j) \overset{*}{\vdash} (q_{j+1}, \varepsilon, \varepsilon)$ für alle $1 \leq j \leq m$.
- Also $(q_j, w_j, Y_j \cdots Y_m) \overset{*}{\vdash} (q_{j+1}, \varepsilon, Y_{j+1} \cdots Y_m)$ für alle $1 \leq j \leq m$.
- Damit

$$
\begin{aligned}
(q, w, X) &\vdash (q_1, w_1 \cdots w_m, Y_1 \cdots Y_m) \\
&\overset{*}{\vdash} (q_2, w_2 \cdots w_m, Y_2 \cdots Y_m) \\
&\overset{*}{\vdash} (q_3, w_3 \cdots w_m, Y_3 \cdots Y_m) \\
&\overset{*}{\vdash} \cdots \overset{*}{\vdash} (q_m, w_m, Y_m) \overset{*}{\vdash} (q_{m+1}, \varepsilon, \varepsilon) = (p, \varepsilon, \varepsilon)
\end{aligned}
$$

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Richtung

$$[q, X, p] \xrightarrow{*} w \text{ in } G \quad \Longleftarrow \quad (q, w, X) \overset{*}{\vdash} (p, \varepsilon, \varepsilon)$$

Beschreibung:

- Induktion über die Länge k einer Abarbeitung $(q, w, X) \overset{k}{\vdash} (p, \varepsilon, \varepsilon)$

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Wir zeigen: $[q, X, p] \xrightarrow{*} w$ in $G \Longleftarrow (q, w, X) \overset{k}{\vdash} (p, \varepsilon, \varepsilon)$

- $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- $[q, X, q_{m+1}] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$
  für alle $q_2, q_3, \ldots, q_{m+1} \in Q$, mit $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$.

Induktionsanfang:

- Für $k = 1$ folgt aus $(q, w, X) \vdash (p, \varepsilon, \varepsilon)$, dass
  - $|w| = 1$ und
  - $(p, \varepsilon) \in \delta(q, w, X)$.
- Dann ist $[q, X, p] \rightarrow w$ eine Regel von G.

Beweis: NPDA $\to$ kontextfreie Grammatik

Wir zeigen: $[q, X, p] \xrightarrow{*} w$ in $G \Longleftarrow (q, w, X) \overset{k}{\vdash} (p, \varepsilon, \varepsilon)$

- $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- $[q, X, q_{m+1}] \to a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$
  für alle $q_2, q_3, \ldots, q_{m+1} \in Q$, mit $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$.

Induktionsschritt:

- Betrachte eine Abarbeitung $(q, w, X) \overset{k}{\vdash} (p, \varepsilon, \varepsilon)$.
- Zerlege $w = aw'$, wobei
  - $a = \varepsilon$, falls der erste Schritt von $\mathcal{A}$ ein ε-Übergang ist,
  - $a \in \Sigma$, also der erste Buchstabe von w, sonst.
- Sei $(q_1, w', Y_1 \cdots Y_m)$ die Konfiguration von $\mathcal{A}$ nach dem 1. Schritt.
- Dann gilt

$$(q, aw', X) \vdash (q_1, w', Y_1 \cdots Y_m) \overset{k'}{\vdash} (p, \varepsilon, \varepsilon)$$

  mit $k' = k - 1$.

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Wir zeigen: $[q, X, p] \xrightarrow{*} w$ in $G \Longleftarrow (q, w, X) \overset{k}{\vdash} (p, \varepsilon, \varepsilon)$

- $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- $[q, X, q_{m+1}] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$
  für alle $q_2, q_3, \ldots, q_{m+1} \in Q$, mit $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$.

Sei

$$w' = w_1 \cdots w_m \text{ Zerlegung von } w \text{ mit } w_j \in \Sigma^*,$$

sodass $\mathcal{A}$ startend mit der Konfiguration

$$(q_1, w', Y_1 \cdots Y_m)$$

bei der betrachteten Abarbeitung gerade nach dem Lesen von $w_1 \cdots w_j$ zum ersten Mal den STACK-Inhalt $Y_{j+1} \cdots Y_m$ erzeugt. Sei q_{j+1} der zu diesem Zeitpunkt erreichte Zustand.

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Wir zeigen: $[q, X, p] \xrightarrow{*} w$ in $G \Longleftarrow (q, w, X) \overset{k}{\vdash} (p, \varepsilon, \varepsilon)$

- $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- $[q, X, q_{m+1}] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$
  für alle $q_2, q_3, \ldots, q_{m+1} \in Q$, mit $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$.

Dann gilt: $q_{m+1} = p$ und

$$(q_j, w_j \cdots w_m, Y_j \cdots Y_m) \overset{k'}{\vdash} (q_{j+1}, w_{j+1} \cdots w_m, Y_{j+1} \cdots Y_m),$$

$k' \leq k - 1$, und während der gesamten Abarbeitung liegt $Y_{j+1} \cdots Y_m$ ungelesen auf dem STACK.
Also gilt auch

$$(q_j, w_j, Y_j) \overset{k'}{\vdash} (q_{j+1}, \varepsilon, \varepsilon).$$

Beweis: NPDA $\rightarrow$ kontextfreie Grammatik

Wir zeigen: $[q, X, p] \xrightarrow{*} w$ in $G \Longleftarrow (q, w, X) \overset{k}{\vdash} (p, \varepsilon, \varepsilon)$

- $V := \{[q, X, p] \mid p, q \in Q,\ X \in \Gamma\} \cup \{S\}$.
- $[q, X, q_{m+1}] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}]$
  für alle $q_2, q_3, \ldots, q_{m+1} \in Q$, mit $(q_1, Y_1 Y_2 \cdots Y_m) \in \delta(q, a, X)$.

Also gilt auch

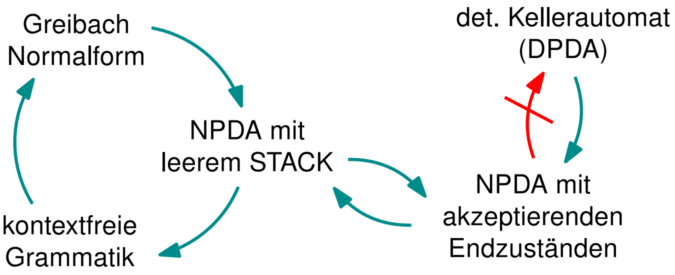
$$(q_j, w_j, Y_j) \overset{k'}{\vdash} (q_{j+1}, \varepsilon, \varepsilon).$$

Nach Induktionsvoraussetzung folgt daraus, dass $[q_j, Y_j, q_{j+1}] \xrightarrow{*} w_j$ in G existiert. Damit erhalten wir, dass auch

$$[q, X, p] \rightarrow a[q_1, Y_1, q_2][q_2, Y_2, q_3] \cdots [q_m, Y_m, q_{m+1}] \xrightarrow{*} aw_1 \cdots w_m = w$$

in G existiert.

Übersicht



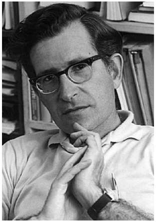
Korollar
Die Klasse der von nichtdeterministischen Kellerautomaten akzeptierten Sprachen ist gleich der Klasse der kontextfreien Sprachen.

Exkurs

Karlsruher Institut für Technologie

Wofür braucht man eigentlich Grammatiken und Berechnungsmodelle wie endliche Automaten oder Turingmaschinen?

- Die Chomsky-Hierarchie wurde 1956 von dem Linguisten Noam Chomsky entworfen. Ursprünglich war sie als Mittel zur Beschreibung natürlicher Sprachen gedacht (hat sich nicht erfüllt).
- Grammatiken und Automaten sind fundamental für die Beschreibung von Programmiersprachen.
- XML basiert auf sogenannten Dokumenttypdefinitionen (DTD). Diese sind kontextfreie Grammatiken.



Noam Chomsky (geb. 1928, hier um 1960)

Zwischenfazit zu kontextfreien Grammatiken

- Es kann in polynomialer Laufzeit entschieden werden, ob zu einer kontextfreien Grammatik G die Sprache $L(G)$ leer bzw. endlich ist.

  $\Rightarrow$ Entfernung nutzloser Variablen (vgl. VL 15, Folie 22)

- Das Wortproblem für kontextfreie Grammatiken ist in polynomialer Laufzeit entscheidbar.

  $\Rightarrow$ Chomsky-Normalform und CYK-Algorithmus

- Für kontextfreie Grammatiken G, G_1 und G_2 sind auch die Sprachen $L(G)^*$, $L(G_1) \cup L(G_2)$ und $L(G_1) \cdot L(G_2)$ kontextfrei.

  $\Rightarrow$ (vgl. VL 16, Folie 2)

- Die kontextfreien Sprachen sind genau die Sprachen, die von nichtdeterministischen Kellerautomaten (NPDAs) akzeptiert werden.

  $\Rightarrow$ Greibach-Normalform (vgl. heute, Folie 16)

Unentscheidbare Probleme für kontextfreie Grammatiken

Satz:
Das Problem, für kontextfreie Grammatiken G_1 und G_2 zu entscheiden, ob $L(G_1) \cap L(G_2) = \emptyset$ ist, ist nicht entscheidbar.

Unentscheidbare Probleme für kontextfreie Grammatiken

Satz:
Das Problem, für kontextfreie Grammatiken G_1 und G_2 zu entscheiden, ob $L(G_1) \cap L(G_2) = \emptyset$ ist, ist nicht entscheidbar.

Beweisskizze:

- Wir beweisen, dass aus der Entscheidbarkeit von $L(G_1) \cap L(G_2) = \emptyset$ die Entscheidbarkeit des Post'schen Korrespondenzproblems (PKP) folgt.
- Dies ist ein Widerspruch zur Nichtentscheidbarkeit des PKP.
- Wir geben für jede PKP-Instanz K kontextfreie Grammatiken G_1 und G_2 an, sodass es ein Wort $w \in L(G_1) \cap L(G_2)$ genau dann gibt, wenn es eine Lösung für K gibt.

Das Post'sche Korrespondenzproblem

Post'sches Korrespondenzproblem
Gegeben ist eine endliche Folge von Wortpaaren

$$K = ((x_1, y_1), \ldots, (x_k, y_k))$$

über einem endlichen Alphabet Σ. Es gilt $x_i \in \Sigma^+$ und $y_i \in \Sigma^+$. Gefragt ist, ob es eine endliche Folge von Indizes $i_1, \ldots, i_\ell \in \{1, \ldots, k\}$ gibt, sodass $x_{i_1} \cdots x_{i_\ell} = y_{i_1} \cdots y_{i_\ell}$ gilt.

Beweis

- Gegeben sei PKP-Instanz $K = ((x_1, y_1), \ldots, (x_k, y_k))$ über Alphabet Σ.
- Es sei $\Sigma' = \Sigma \cup \{a_1, \ldots, a_k\}$ für neue Symbole $a_1, \ldots, a_k$.
- Es sei $G_1 = (\Sigma', V_1 = \{S_1\}, S_1, R_1)$ mit Regeln

$$S_1 \to a_i x_i \text{ und } S_1 \to a_i S_1 x_i \text{ für alle } 1 \leq i \leq k.$$

- Es sei $G_2 = (\Sigma', V_2 = \{S_2\}, S_2, R_2)$ mit Regeln

$$S_2 \to a_i y_i \text{ und } S_2 \to a_i S_2 y_i \text{ für alle } 1 \leq i \leq k.$$

Dann gilt:

$$\begin{aligned} L(G_1) &= \{a_{i_n} \cdots a_{i_1} x_{i_1} \cdots x_{i_n} \mid n \in \mathbb{N},\ 1 \leq i_j \leq k\} \\ L(G_2) &= \{a_{i_n} \cdots a_{i_1} y_{i_1} \cdots y_{i_n} \mid n \in \mathbb{N},\ 1 \leq i_j \leq k\} \end{aligned}$$

Beweis

$$
\begin{aligned}
K &= ((x_1, y_1), \ldots, (x_k, y_k)) \\
L(G_1) &= \{a_{i_n} \cdots a_{i_1} x_{i_1} \cdots x_{i_n} \mid n \in \mathbb{N},\ 1 \leq i_j \leq k\} \\
L(G_2) &= \{a_{i_n} \cdots a_{i_1} y_{i_1} \cdots y_{i_n} \mid n \in \mathbb{N},\ 1 \leq i_j \leq k\}
\end{aligned}
$$

Es folgt:

$$
\begin{aligned}
K \text{ hat Lösung} \quad &\Leftrightarrow \quad \exists i_1, \ldots, i_n \text{ mit } x_{i_1} \cdots x_{i_n} = y_{i_1} \cdots y_{i_n} \\
&\Leftrightarrow \quad \exists i_1, \ldots, i_n \text{ mit } a_{i_n} \cdots a_{i_1} x_{i_1} \cdots x_{i_n} = a_{i_n} \cdots a_{i_1} y_{i_1} \cdots y_{i_n} \\
&\Leftrightarrow \quad \exists w \in L(G_1) \cap L(G_2) \\
&\Leftrightarrow \quad L(G_1) \cap L(G_2) \neq \emptyset
\end{aligned}
$$

Eindeutigkeit von kontextfreien Grammatiken

Karlsruher Institut für Technologie

Eine Grammatik G ist eindeutig, wenn es für jedes $w \in L(G)$ genau einen Syntaxbaum gibt.

Satz:
Das Problem, für eine kontextfreie Grammatik G zu entscheiden, ob sie eindeutig ist, ist nicht entscheidbar.

Eindeutigkeit von kontextfreien Grammatiken

Eine Grammatik G ist eindeutig, wenn es für jedes $w \in L(G)$ genau einen Syntaxbaum gibt.

Satz:
Das Problem, für eine kontextfreie Grammatik G zu entscheiden, ob sie eindeutig ist, ist nicht entscheidbar.

Beweisskizze.

- Annahme: Es sei entscheidbar, ob eine kontextfreie Grammatik eindeutig ist.
- Dann könnten wir das PKP entscheiden.
- Dies ist ein Widerspruch.

Beweis

- Gegeben sei PKP-Instanz $K = ((x_1, y_1), \dots, (x_k, y_k))$ über Alphabet Σ.
- Seien $G_1 = (\Sigma', V_1, S_1, R_1)$ und $G_2 = (\Sigma', V_2, S_2, R_2)$ wie im letzten Beweis.
- Wir konstruieren eine neue Grammatik $G = (\Sigma', V, S, R)$, die genau dann mehrdeutig ist, wenn $L(G_1) \cap L(G_2) \neq \emptyset$:

$$\begin{aligned} V &= V_1 \cup V_2 \cup \{S\}, \text{ wobei } S \text{ neues Startsymbol} \\ R &= R_1 \cup R_2 \cup \{S \to S_1 | S_2\} \end{aligned}$$

- Da G_1 und G_2 eindeutig sind, existiert $w \in L(G_1) \cap L(G_2)$ genau dann, wenn es in G Ableitungen $S \to S_1 \xrightarrow{*} w$ und $S \to S_2 \xrightarrow{*} w$ gibt, also G mehrdeutig ist.

Sprache der korrekten Rechenwege

- Sei $\mathcal{M} = (Q, \Sigma, \Gamma, \sqcup, q_0, \delta, F)$ eine TM.
- Eine Berechnung von $\mathcal{M}$ kann durch die Folge der durchlaufenen

  Konfigurationen $\alpha(q)\beta$ mit $\alpha, \beta \in \Gamma^*$ und $q \in Q$

  beschrieben werden.
- $\alpha(q)\beta$ bedeutet, dass
  - auf dem Band das Wort $\alpha\beta$, umgeben von Blanksymbolen, steht,
  - die Turingmaschine im Zustand q ist
  - und der Lese-/Schreibkopf auf die Stelle des Bandes, an der das erste Symbol von β steht, zeigt.
- Wenn $w_1, w_2, \ldots, w_n$ die Abfolge der Konfigurationen einer Berechnung von $\mathcal{M}$ ist, so kann dieser Rechenweg durch das Wort $w_1 \# w_2 \# \cdots \# w_n \#$, mit $\# \notin \Gamma$ Trennsymbol, kodiert werden.
- Hierbei sei (q) für $q \in Q$ ein einziges Zeichen.

Sprache der korrekten Rechenwege

- Allerdings lässt sich die Sprache aller Wörter, die in dieser Weise die korrekten Rechenwege einer TM kodieren, nicht unbedingt durch kontextfreie Grammatiken beschreiben.
- Daher wird ein „Trick“ angewendet und jede zweite Konfiguration gespiegelt kodiert.

Sprache der korrekten Rechenwege

Die **Sprache $\mathbf{B}_{\mathcal{M}}$ der korrekten Rechenwege einer TM** $\mathcal{M}$ besteht aus allen Worten

$$w_1\#w_2^R\#w_3\#w_4^R\cdots w_n^R\#,\ \text{falls } n \text{ gerade, und}$$

$$w_1\#w_2^R\#w_3\#w_4^R\cdots w_n\#,\ \text{falls } n \text{ ungerade,}$$

wobei

- die w_i, $1 \leq i \leq n$, Konfigurationen von $\mathcal{M}$ sind,
- w_1 eine Anfangskonfiguration,
- w_n eine akzeptierende Konfiguration und
- für alle $1 \leq i \leq n-1$ die Konfiguration w_{i+1} die direkte Nachfolgekonfiguration von w_i bei einer korrekten Berechnung von $\mathcal{M}$

ist.

Sprache der korrekten Rechenwege

Lemma:
Für alle Turingmaschinen $\mathcal{M}$ ist $B_{\mathcal{M}}$ der Durchschnitt zweier Sprachen

- $L_1 = L(G_1)$
- $L_2 = L(G_2)$,

wobei G_1 und G_2 kontextfreie Grammatiken sind.

Beweis

Wir konstruieren L_1 und L_2 aus den Sprachen

$$\begin{aligned} L &:= \{u\#v^R \mid v \text{ ist direkte Nachfolgekonfiguration von } u \text{ für } \mathcal{M}\} \\ L' &:= \{v^R\#u \mid u \text{ ist direkte Nachfolgekonfiguration von } v \text{ für } \mathcal{M}\} \\ E &:= \{\varepsilon\} \cup (\Gamma^* \cdot \{(q) \mid q \in F\} \cdot \Gamma^* \cdot \{\#\}) \end{aligned}$$

Falls L und L' kontextfrei sind, so sind auch

$$\begin{aligned} L_1 &:= (L \cdot \{\#\})^* \cdot E \\ L_2 &:= \{(q_0)\} \cdot \Sigma^* \cdot \{\#\} \cdot (L' \cdot \{\#\})^* \cdot E \end{aligned}$$

kontextfrei, wobei

- Γ Bandalphabet,
- Σ Eingabealphabet,
- q_0 Anfangszustand und
- F Endzustandsmenge

von $\mathcal{M}$.

Beweis

Offensichtlich haben alle Wörter aus L_1 die Form

$$w_1 \# w_2^R \# \cdots w_{2i-1} \# w_{2i}^R \# \text{ oder}$$

$$w_1 \# w_2^R \# \cdots w_{2i-1} \# w_{2i}^R \# w_{2i+1} \#$$

mit

- w_j Konfiguration von $\mathcal{M}$
- w_{2j} direkte Nachfolgekonfiguration von w_{2j-1}

für alle $1 \leq j \leq i$ und w_{2i+1} akzeptierende Konfiguration, falls vorhanden.

Beweis

Analog haben alle Wörter aus L_2 die Form

$$w_1 \# w_2^R \# \cdots w_{2i-1} \# w_{2i}^R \# \text{ oder}$$

$$w_1 \# w_2^R \# \cdots w_{2i-2}^R \# w_{2i-1} \#$$

mit

- w_j Konfiguration von $\mathcal{M}$
- w_1 Anfangskonfiguration
- w_{2j+1} direkte Nachfolgekonfiguration von w_{2j}

für alle $1 \leq j \leq i - 1$ und w_{2i} akzeptierende Konfiguration, falls vorhanden.

Dann ist $B_{\mathcal{M}} = L_1 \cap L_2$.

Beweis

$$L := \{u\#v^R \mid v \text{ ist direkte Nachfolgekonfiguration von } u \text{ für } \mathcal{M}\}$$

Wir geben nun eine kontextfreie Grammatik G für L an mit Startvariable S und zusätzlicher Variable A.

G enthalte folgende Regeln:

(i) alle Regeln $S \to aSa$, $a \in \Gamma \backslash \{\sqcup\}$;

(ii) für alle Übergänge $\delta(q, a) = (q', b, R)$ von $\mathcal{M}$ die Regeln $S \to (q)aA(q')b$;

(iii) für alle Übergänge $\delta(q, a) = (q', b, L)$ von $\mathcal{M}$ und alle $x \in \Gamma \cup \{\varepsilon\}$ die Regeln $S \to x(q)aAbx(q')$;

(iv) für alle Übergänge $\delta(q, a) = (q', b, N)$ von $\mathcal{M}$ die Regeln $S \to (q)aAb(q')$;

(v) für alle $a \in \Gamma$ die Regeln $A \to aAa$;

(vi) die Regel $A \to \#$.

Beweis

- Analog kann eine kontextfreie Grammatik G' für L' angegeben werden.
- Es ist leicht zu zeigen, dass $L(G) = L$ und $L(G') = L'$ ist.
- Damit ist die Behauptung bewiesen.

Beweis

- Analog kann eine kontextfreie Grammatik G' für L' angegeben werden.
- Es ist leicht zu zeigen, dass $L(G) = L$ und $L(G') = L'$ ist.
- Damit ist die Behauptung bewiesen.

Lemma:
Für alle Turingmaschinen $\mathcal{M}$ ist $B_{\mathcal{M}}$ der Durchschnitt zweier kontextfreier Sprachen.

Bemerkung:
Falls $\mathcal{M}$ in jeder Berechnung nur höchstens einen Rechenschritt ausführt, ist $B_{\mathcal{M}}$ sogar selbst kontextfrei.

Weiterhin gilt ...

Lemma:
Für alle Turingmaschinen $\mathcal{M}$ ist $B_{\mathcal{M}}$ der Durchschnitt zweier kontextfreier Sprachen.

Bemerkung:
Falls $\mathcal{M}$ in jeder Berechnung nur höchstens einen Rechenschritt ausführt, ist $B_{\mathcal{M}}$ sogar selbst kontextfrei.

Lemma:
Sei $\mathcal{M}$ eine TM, die auf jeder Eingabe mindestens zwei Rechenschritte ausführt. Dann ist die Sprache $B_{\mathcal{M}}$ genau dann kontextfrei, wenn $L(\mathcal{M})$ endlich ist.

Ohne Beweis.

Zusammenfassung Chomsky-Hierarchie

Testen Sie sich:

- ▷ Können Sie folgende Tabelle ausfüllen?
- ▷ Welche Ergebnisse sind aus der heutigen Vorlesung?

Chomsky-Typ	Regeln in der Grammatik	Komplexität des Wort-problems	zugehöriges Maschinen-modell	Beispiel-sprache
Typ 0				
Typ 1				
Typ 2				
Typ 3				