Endlichkeitsproblem für CFG

gegeben:	kontextfreie Grammatik $G = (V, \Sigma, \mathcal{P}, S)$
gefragt:	gilt $\lvert\mathcal{L}(G)\rvert < \infty$?

für bereinigte CNF-Grammatiken lösbar durch Analyse des Variablengraphen $\mathcal{V}_G = (V, \hookrightarrow)$

Führe einen Zyklentest in $\mathcal{V}_G$ durch.

`IF` der Variablengraph hat keinen Zyklus

    `THEN` gib "ja, $\mathcal{L}(G)$ ist endlich" aus

    `ELSE` gib "nein, $\mathcal{L}(G)$ ist unendlich" aus.

`FI`

gegeben: kontextfreie Grammatik $G = (V, \Sigma, \mathcal{P}, S)$

gefragt: gilt $|\mathcal{L}(G)| < \infty$?

für bereinigte CNF-Grammatiken lösbar durch Analyse des Variablengraphen $\mathcal{V}_G = (V, \hookrightarrow)$

$\uparrow$

gerichteter Graph mit der Knotenmenge V und folgender Kantenrelation $\hookrightarrow$:

$$A \hookrightarrow B \quad \text{gdw} \quad \left\{ \begin{array}{l} \text{es gibt eine Regel der} \\ \text{Form } A \to uBv \text{ in } G \end{array} \right.$$

Abschlusseigenschaften für Typ 2

Die Klasse der kontextfreien Sprachen ist abgeschlossen unter

- Vereinigung
- Konkatenation
- Kleeneabschluss

Beweis:

durch Angabe entsprechender Operatoren auf **CFG**

Vereinigungsoperator für CFG

$$\left.\begin{array}{l} G_1 = (V_1, \Sigma, \mathcal{P}_1, S_1) \\ G_2 = (V_2, \Sigma, \mathcal{P}_2, S_2) \end{array}\right\} \text{ kontextfreie Grammatiken}$$

o.B.d.A.: $V_1 \cap V_2 = \varnothing$

Grammatik $G_1 \uplus G_2$:

- Terminalalphabet Σ
- Variablenmenge $V_1 \cup V_2 \cup \{S\}$, wobei $S \notin V_1 \cup V_2$
- Startsymbol S
- Produktionssystem besteht aus

Regeln in G_1 + Regeln in G_2 + Startregeln $S \to S_1 \mid S_2$

erzeugte Sprache: $\mathcal{L}(G_1 \uplus G_2) = \mathcal{L}(G_1) \cup \mathcal{L}(G_2)$

$$\left.\begin{array}{l} G_1 = (V_1, \Sigma, \mathcal{P}_1, S_1) \\ G_2 = (V_2, \Sigma, \mathcal{P}_2, S_2) \end{array}\right\} \text{Grammatiken vom Typ } \boldsymbol{i} \qquad \text{wobei } i \in \{0, 1, 2\}$$

Dann ist $G_1 \uplus G_2$ eine Grammatik vom Typ $\boldsymbol{i}$ für die Sprache $\mathcal{L}(G_1) \cup \mathcal{L}(G_2)$.

Die Klassen der

- Sprachen vom Typ **0**
- kontextsensitiven Sprachen (Typ **1**)
- kontextfreien Sprachen (Typ **2**)

sind abgeschlossen unter Vereinigung

$$\left.\begin{array}{l} G_1 = (V_1, \Sigma, \mathcal{P}_1, S_1) \\ G_2 = (V_2, \Sigma, \mathcal{P}_2, S_2) \end{array}\right\} \text{ kontextfreie Grammatiken}$$

o.B.d.A.: $V_1 \cap V_2 = \varnothing$

kontextfreie Grammatik $G_1 \circ G_2$:

- Terminalalphabet Σ
- Variablenmenge $V_1 \cup V_2 \cup \{S\}$, wobei $S \notin V_1 \cup V_2$
- Startsymbol S
- Produktionssystem besteht aus

  Regeln in G_1 + Regeln in G_2 + Startregel $S \to S_1 S_2$

zeige:

$$\mathcal{L}(G_1 \circ G_2) = \{ w_1 w_2 : w_1 \in \mathcal{L}(G_1), w_2 \in \mathcal{L}(G_2) \}$$

Beispiel: Vereinigung, Konkatenation für CFG

CFG G_1:	CFG G_2:	CFG G_3:
$S_1 \to a$	$S_2 \to b$	$S_3 \to \varepsilon$

$$\mathcal{L}(G_1) = \{a\} \qquad \mathcal{L}(G_2) = \{b\} \qquad \mathcal{L}(G_3) = \{\varepsilon\}$$

Konstruiere CFG für $\mathcal{L}(G_1)\mathcal{L}(G_2) \cup \mathcal{L}(G_3) = \{ab, \varepsilon\}$

$$S_1 \to a \qquad S \to S_1 S_2 \qquad S' \to S \mid S_3$$
$$S_2 \to b$$
$$S_3 \to \varepsilon$$

CFG $(G_1 \circ G_2) \uplus G_3$

neues Startsymbol S'

Kleeneabschluss für CFG

gegeben: CFG $G = (V, \Sigma, \mathcal{P}, S)$

Grammatik G^*:

- Terminalalphabet Σ
- Variablenmenge $V \cup \{S_*\}$, wobei $S_* \notin V$
- Startsymbol S_*
- Regeln in G + Startregeln $S_* \to \varepsilon \mid SS_*$

Korrektheit: $\mathcal{L}(G^*) = \mathcal{L}(G)^*$

CFG G_1:

$S_1 \to a \mid bS_1$

CFG G_2:

$S_2 \to c$

$\mathcal{L}(G_1) = \mathcal{L}(b^*a)$

$\mathcal{L}(G_2) = \{c\}$

Konstruiere CFG für $\mathcal{L}(G_1)^* \, \mathcal{L}(G_2) = \mathcal{L}((b^*a)^*c)$

$$
\begin{array}{ll}
S_1 \to a \mid bS_1 & S_* \to \varepsilon \mid S_1 S_* \\
S_2 \to c & S \to S_* S_2
\end{array}
$$

CFG $G_1^* \circ G_2$

neues Startsymbol S

Vereinigung $G_1 \uplus G_2$

- Regeln in G_1 und G_2
- neues Startsymbol S mit $S \rightarrow S_1 \mid S_2$

Konkatenation $G_1 \circ G_2$

- Regeln in G_1 und G_2
- neues Startsymbol S mit $S \rightarrow S_1 S_2$

Kleeneabschluss G^*

- Regeln in G
- neues Startsymbol S_* mit $S_* \rightarrow \varepsilon \mid S S_*$

Die Klasse der kontextfreien Sprachen ist nicht abgeschlossen unter Durchschnitt.

Beweis: Betrachte die Sprachen

$$L_1 = \{ a^n b^n c^m : n, m \geqslant 1 \}$$
$$L_2 = \{ a^n b^m c^m : n, m \geqslant 1 \}$$

L_1, L_2 sind kontextfrei, aber deren Durchschnitt

$$L_1 \cap L_2 = \{ a^n b^n c^n : n \geqslant 1 \}$$

ist nicht kontextfrei.

Die Klasse der kontextfreien Sprachen ist nicht abgeschlossen unter Komplement.

Beweis:

Angenommen sie ist abgeschlossen unter Komplement.

Dann wäre sie auch unter Durchschnitt abgeschlossen:

- die Klasse der kontextfreien Sprachen ist unter Vereinigung abgeschlossen
- für alle Sprachen L_1, L_2 gilt:

$$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$$

Widerspruch

Eine CFG ist in Greibach Normalform, falls alle Regeln von folgender Form sind:

$$A \to a \quad \text{oder} \quad A \to aB_1 \dots B_k,$$

wobei a ein Terminal, A, $B_1, \dots, B_k$ Variablen, $k \geqslant 1$

Beispiel für eine CFG in Greibach Normalform:

$$S \to aSB \mid aB \qquad B \to b$$

erzeugte Sprache: $\{a^n b^n : n \geqslant 1\}$, z.B.

$$S \Longrightarrow_L aSB \Longrightarrow_L aaSBB \Longrightarrow_L^* a^{n-1}SB^{n-1}$$
$$\Longrightarrow_L^* a^n B^n \Longrightarrow_L^* a^n b^n$$

Die Greibach Normalform

Eine CFG ist in Greibach Normalform, falls alle Regeln von folgender Form sind:

$$A \to a \quad \text{oder} \quad A \to aB_1 \dots B_k,$$

wobei a ein Terminal, A, $B_1, \dots, B_k$ Variablen, $k \geqslant 1$

Beispiel für eine CFG in Greibach Normalform:

$$S \to aSB \mid aB \qquad B \to b$$

Zu jeder CFG G mit $\varepsilon \notin \mathcal{L}(G)$ gibt es eine äquivalente CFG G' in Greibach Normalform.

(ohne Beweis)

Linksableitungen in Greibach Normalformen

Jede Linksableitung hat die Form:

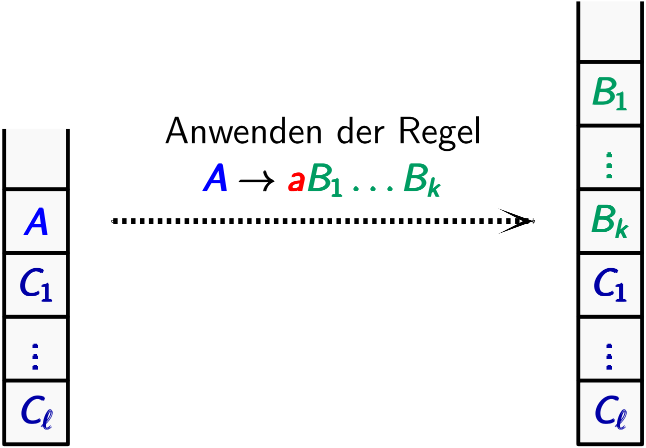
$$
\begin{aligned}
S &\Longrightarrow_L a_1\, B_1 B_2 \ldots B_k \\
&\Longrightarrow_L a_1\, a_2\, C_1 C_2 \ldots C_\ell\, B_2 \ldots B_k \\
&\Longrightarrow_L a_1\, a_2\, a_3\, D_1 D_2 \ldots D_m\, C_2 \ldots C_\ell\, B_2 \ldots B_k \\
&\Longrightarrow_L a_1\, a_2\, a_3\, a_4\, D_2 \ldots D_m\, C_2 \ldots C_\ell\, B_2 \ldots B_k \\
&\ \vdots \\
&\Longrightarrow_L a_1\, a_2\, a_3\, a_4\, a_5\, a_6\, \ldots\, a_n
\end{aligned}
$$

im i-ten Ableitungsschritt wird das i-te Zeichen a_i generiert

Linksableitungen und Keller

Linksableitungsschritt

$$a_1 \dots a_i A C_1 \dots C_\ell \implies_L a_1 \dots a_i a B_1 \dots B_k C_1 \dots C_\ell$$



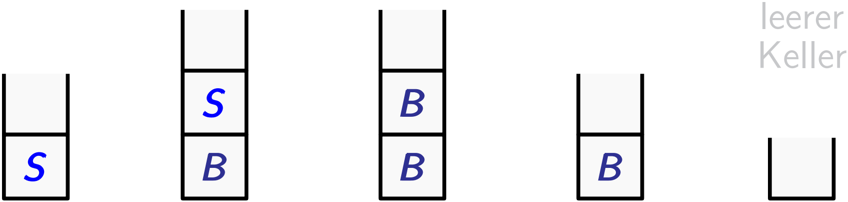
Beispiel: Greibach NF und Keller

CFG in Greibach NF: $S \to aSB \mid aB \qquad B \to b$

Linksableitung:

$$S \Longrightarrow_L aSB \Longrightarrow_L aaBB \Longrightarrow_L aabB \Longrightarrow_L aabb$$

Keller zur Verwaltung der Variablen:



pop; *push*(S) *push*(B);

pop; *push*(B);

Anwenden der Regel $B \to b$ entspricht *pop*