Zweites Beispiel

Wir gehen von der folgenden Grammatik aus:

$$S \to aSb \mid cD \mid B$$

$$B \to Sb$$

$$D \to cD \mid cBB \mid c$$

Die einzige Regel der Form (u, v) mit $v \in V$ ist die Regel $S \to B$.

1) Ringableitungen entfernen

   Es sind keine Ringableitungen vorhanden, also nichts zu tun in diesem Punkt.

2) Variablen anordnen

   Die Anordnung $A_1 = S, A_2 = B, A_3 = D$ erfüllt den Zweck.

3) Abkürzungen verwenden

   Die Regel $S \to B$ wird gestrichen, $S \to Sb$ dafür dazu genommen.

Beispiel 2, Fortsetzung

4) Pseudoterminale einführen

Ergänze Regeln $V_a \to a$, $V_b \to b$, $V_c \to c$ und erhalte:

$$S \to V_a S V_b \mid V_c D \mid S V_b \qquad V_a \to a$$

$$B \to S V_b \qquad V_b \to b$$

$$D \to V_c D \mid V_c B B \mid c \qquad V_c \to c$$

5) $S \to V_a S V_b$ und $D \to V_c BB$ jeweils durch 2 Regeln ersetzen.

Wir verwenden neue Variablen X und Y und erhalten:

$$S \to V_a X \mid V_c D \mid S V_b \qquad X \to S V_b \qquad V_a \to a$$

$$B \to S V_b \qquad V_b \to b$$

$$D \to V_c D \mid V_c Y \mid c \qquad Y \to BB \qquad V_c \to c$$

Drittes Beispiel

Jetzt betrachten wir die folgende Grammatik:

$S \to A \mid aB \mid aC$ $\qquad$ $A \to B \mid C \mid cAd$

$B \to S \mid Ba$ $\qquad$ $C \to D \mid c$

$D \to d \mid dDD$

$S, A, B \in V$ bilden einen Ring, also ersetzen wir sie durch X:

$X \to aX \mid aC \mid C \mid cXd \mid Xa$ $\qquad$ $C \to D \mid c$

$D \to d \mid dDD$

Die Anordnung muss der Reihenfolge X vor C vor D folgen:

$D \to d \mid dDD$ $\qquad$ $C \to d \mid dDD \mid c$

$X \to d \mid dDD \mid c \mid aX \mid aC \mid cXd \mid Xa$

Beispiel 3, Fortsetzung

Einführung von Pseudoterminalen V_a, V_c und V_d:

$$D \to d \mid V_d DD \qquad C \to d \mid V_d DD \mid c$$

$$X \to d \mid V_d DD \mid c \mid V_a X \mid V_a C \mid V_c X V_d \mid X V_a$$

$$V_a \to a \qquad V_c \to c \qquad V_d \to d$$

Jetzt nutzen wir aus, dass $V_d DD$ dreimal vorkommt, daher verwenden wir dreimal $Y \to DD$:

$$D \to d \mid V_d Y \qquad C \to d \mid V_d Y \mid c$$

$$X \to d \mid V_d Y \mid c \mid V_a X \mid V_a C \mid V_c Z \mid X V_a$$

$$V_a \to a \qquad V_c \to c \qquad V_d \to d$$

$$Y \to DD \qquad Z \to X V_d$$

Satz (Greibach-Normalform)

Zur Erinnerung: Eine Typ-2 Grammatik ist in GNF, wenn für alle $(u, v) \in P$ gilt: $v \in \Sigma V^*$.

Satz: Zu jeder kontextfreien Grammatik G mit $\varepsilon \notin L(G)$ gibt es eine Grammatik G' in Greibach-Normalform, so dass

$$L(G) = L(G')$$

gilt.

Auch hier besteht der Beweis wieder darin, dass wir eine Strategie angeben, wie eine beliebige Typ-2 Grammatik in eine äquivalente Typ-2 Grammatik in GNF umgewandelt werden kann.

Beseitigung von Linksrekursion

Wir können jederzeit für jede Variable A die Regeln der Form (A, v) aus P in zwei Gruppen aufteilen, nämlich die, bei denen $v = Ax$ für ein $x \neq \varepsilon$ gilt, und die, bei denen v nicht mit A beginnt:

$$A \to A\alpha_1 \mid A\alpha_2 \mid \ldots \mid A\alpha_k \mid \beta_1 \mid \beta_2 \mid \ldots \mid \beta_\ell$$

Diese $k + \ell$ Regeln können durch die folgenden $2k + 2\ell$ Regeln äquivalent ersetzt werden (mit neuer Variable B):

$$A \to \beta_1 \mid \beta_2 \mid \ldots \mid \beta_\ell$$

$$A \to \beta_1 B \mid \beta_2 B \mid \ldots \mid \beta_\ell B$$

$$B \to \alpha_1 \mid \alpha_2 \mid \ldots \mid \alpha_k$$

$$B \to \alpha_1 B \mid \alpha_2 B \mid \ldots \mid \alpha_k B$$

Hier sind keine linksrekursiven Regeln mehr vorhanden!

Der erste Algorithmus

Der erste von zwei Algorithmen, die zusammen die GNF erzeugen, hat zum Ziel, dass Regeln $A_i \to A_j\beta$ nur mit $i < j$ vorkommen. Dabei sei $V = \{A_1, \ldots, A_m\}$, d.h. Variablen sind nummeriert.

```
FOR i := 1 TO m DO FOR j := 1 TO i − 1 DO
    FORALL A_i → A_j α ∈ P DO
        IF A_j → β_1 | ... | β_r alle A_j-Regeln in P THEN
            Nimm A_i → β_1 α | ... | β_r α zu P hinzu;
            Streiche A_i → A_j α aus P heraus;
    ENDIF ENDFORALL ENDFOR
    Entferne Linksrekursion bzgl. A_i;
ENDFOR
```

Damit ist das Ziel erreicht, dass $A_i \to A_j\beta$ nur mit $i < j$ vorkommt.