Übersicht

3 Kontextfreie Sprachen

- 3.1 Kontextfreie Sprachen und Grammatiken
- 3.2 Ableitungsbäume
- 3.3 Die *pre**-Operation
- 3.4 Entscheidungsprobleme für CFGs
- 3.5 Normalformen für CFGs
- 3.6 Chomsky-Normalform
- 3.7 Greibach-Normalform
- 3.8 Das Pumping-Lemma für CFLs
- 3.9 Kellerautomaten
- 3.10 Deterministische Kellerautomaten
- 3.11 Abschlusseigenschaften kontextfreier Sprachen

Linksrekursion

Definition 3.7.1

Jede Regel $A \to A\,\alpha$ ist linksrekursiv.

Linksrekursion lässt sich eliminieren:
Für $A \in N$ seien

- $A \to A\,\alpha_1 \mid \cdots \mid A\,\alpha_k$ die linksrekursiven und
- $A \to \beta_1 \mid \cdots \mid \beta_m$ die anderen Regeln.

Ersetze $A \to A\,\alpha_1 \mid \cdots \mid A\,\alpha_k$ durch

1. $A \to \beta_1\,Z \mid \cdots \mid \beta_m\,Z$ und
2. $Z \to \alpha_1\,Z \mid \cdots \mid \alpha_k\,Z \mid \alpha_1 \mid \cdots \mid \alpha_k$,

wobei Z ein neues Nonterminal ist.

Beispiel

$$S \to SS \mid AS \mid SB$$

$$S \to AS \mid ASZ$$
$$Z \to SZ \mid BZ \mid S \mid B$$

Greibach-Normalform

Definition 3.7.2

Eine CFG ist in Greibach-Normalform (GNF), falls jede Regel von der Form $A \rightarrow aBCD \cdots$ *ist, d.h.* $A \rightarrow \beta$ *mit* $\beta \in T\,N^*$.

Satz 3.7.3

Es sei G *eine CFG mit* $\epsilon \notin L(G)$. *Dann gibt es eine CFG* G' *in GNF mit* $L(G') = L(G)$.

Starte mit G in CNF ohne ϵ-Produktionen und ohne unproduktive oder unerreichbare Symbole. O.b.d.A. sei $N = \{A_1, \cdots, A_n\}$.

1. Schritt: Eliminiere *verkleinernde* Regeln ($A_i \to A_j\,\alpha$ mit $i \geq j$).
Falls $i > j$, dann ist die Regel *echt verkleinernd*.

- Sei i die kleinste Zahl mit verkleinernden Regeln $A_i \to \gamma$.
  - Sei j die kleinste Zahl, für die es eine Regel $A_i \to A_j\,\alpha$ mit $i > j$ gibt.
  - Seien $A_j \to \beta_1 \mid \cdots \mid \beta_m$ alle Regeln für A_j.
    Ersetze $A_i \to A_j\,\alpha$ durch $A_i \to \beta_1\alpha \mid \cdots \mid \beta_m\alpha$.
  - Wiederhole dies, bis es keine echt verkleinernde Regel $A_i \to \gamma$ mehr gibt.
- Falls es noch linksrekursive Regeln $A_i \to A_i\alpha$ gibt, dann eliminiere sie mit neuen Symbolen, die kleiner als alle bisherigen sind.

Alle Regeln haben die Form $A_i \to a\,\alpha$ oder $A_i \to A_j\,\alpha$ mit $i < j$.

2. Schritt: Eliminiere Regeln $A_i \to A_j\,\alpha$, so dass rechte Seiten immer mit Terminalsymbol beginnen.

- Beginne mit größtem Nicht-Terminal A_i.
- Ersetze $A_i \to A_j\alpha$ durch $A_i \to \beta_1\alpha | \cdots | \beta_k\alpha$, falls $A_j \to \beta_1 | \cdots | \beta_k$.

Beispiel

$$
\begin{aligned}
S &\to AB \mid a\\
A &\to AA \mid b\\
B &\to AB \mid a
\end{aligned}
$$

$$
\begin{aligned}
Z &\to AZ \mid A\\
S &\to AB \mid a\\
A &\to b \mid bZ\\
B &\to AB \mid a
\end{aligned}
$$

$$
\begin{aligned}
Z &\to AZ \mid A\\
S &\to AB \mid a\\
A &\to b \mid bZ\\
B &\to bB \mid bZB \mid a
\end{aligned}
$$

$$
\begin{aligned}
Z &\to AZ \mid A\\
S &\to bB \mid bZB \mid a\\
A &\to b \mid bZ\\
B &\to bB \mid bZB \mid a
\end{aligned}
$$

$$
\begin{aligned}
Z &\to bZ \mid bZZ \mid b\\
S &\to bB \mid bZB \mid a\\
A &\to b \mid bZ\\
B &\to bB \mid bZB \mid a
\end{aligned}
$$