



# Formale Sprachen und Automaten

## TINF20B1



Markus Eble

# Chomsky-Normalform

- Die Chomsky-Normalform (CNF)
- Der Algorithmus von Cocke, Younger und Kasami

# Chomsky-Hierarchie

---

Typ	Name	Erlaubte Produktionen	Akzeptierende Maschine	Beispiel
3	Regulär	$N \rightarrow wM$ $w \in T^*$	Endlicher Automat	$a^n$
2	Kontextfrei	$N \rightarrow w$ $w \in (N \cup T)^*$	Kellerautomat	$a^n b^n$

Skript Worsch: Seite 76-80

# Ziel

---

Gegeben: eine Grammatik und ein Wort

Gesucht: der Ableitungsbaum

# Ansatz – Ausprobieren?

---

Die Zahl der Regeln ist endlich

Die Länge des Wortes ist endlich

Ableitungen der Form  $N \rightarrow \varepsilon$

Machen es schwer zu garantieren dass das Ausprobieren terminiert

# Ansatz

---

Sorge dafür, dass es keine beliebig langen Ableitungsfolgen für ein bestimmtes Wort geben kann.

- ▶ Monotone Grammatiken (nichtverkürzende Grammatiken).

# Ansatz: Monotone Grammatiken

---

- ▶ Gestalte die Grammatik so, dass das Wort mit einer weiteren Ableitung nur „länger“ werden kann.
- ▶ Dann kann eine „Suche durch Probieren“ „abbrechen“, wenn das abgeleitete Wort die Länge des gesuchten Wortes erreicht hat.
- ▶ Dafür nötig:
  - ▶ Keine Ableitungen der Form  $X \rightarrow \varepsilon$ .

# Definition 4.14: Chomsky-Normalform

---

Eine Grammatik  $G = (N, T, S, P)$  ist in Chomsky-Normalform (CNF) wenn gilt:

- ▶ Jede Produktion  $X \rightarrow w$  hat als rechte Seite
  - ▶ entweder ein Wort  $w \in N^2$  (genau zwei Nichtterminalsymbole)
  - ▶ oder ein Wort  $w \in T$  (genau ein Terminalsymbol).
- ▶ Zusätzlich ist die Produktion  $S \rightarrow \varepsilon$  zugelassen, dann kommt  $S$  bei keiner Produktion auf der rechten Seite vor.

2n-1 Ableitungen möglich => in endlichen Schritten möglich abzuleiten



## Satz 4.15: Transformation in Chomsky-NF

---

Zu jeder kontextfreien Grammatik  $G$  gibt es eine kontextfreie Grammatik  $G'$  in Chomsky-Normalform, die zu  $G$  äquivalent ist (also  $L(G) = L(G')$  gilt).

# Konstruktionsschritte für Chomsky-NF

---

- ▶ 1.  $S \rightarrow \varepsilon$  behandeln
- ▶ 2. Terminalsymbole durch Regeln der Form  $N \rightarrow t$  erzeugen
- ▶ 3. Regeln der Form  $N \rightarrow \varepsilon$  ersetzen
- ▶ 4. Regeln der Form  $N \rightarrow M$  ersetzen
- ▶ 5. Regeln mit mehr als 2 NonTerminals ersetzen

[Quelle: Chomsky-Normalform – Wikipedia](#)

# Beispiel

---

Grammatik mit Startsymbol  $S$  und den Regeln:

$$S \rightarrow SS \mid (S) \mid \varepsilon$$

# Konstruktionsschritte für Chomsky-NF

---

- ▶ 1.  $S \rightarrow \varepsilon$  behandeln

Falls die Grammatik  $S \rightarrow \varepsilon$  enthält:

- ▶ Füge neues Startsymbol  $S'$  ein
- ▶ Füge folgende Regeln hinzu
  - $S' \rightarrow \varepsilon$
  - $S' \rightarrow S$

# Beispiel: $S \rightarrow \varepsilon$ behandeln

---

Bisherige Grammatik:

$$S \rightarrow SS \mid (S) \mid \varepsilon$$

Füge hinzu:

$$S' \rightarrow S \mid \varepsilon$$

$S'$  wird neues Startsymbol

# Konstruktionsschritte für Chomsky-NF

---

- ▶ 2. Terminalsymbole durch Regeln der Form  $N \rightarrow t$  erzeugen

Für jedes Terminal  $t$  ein NonTerminal  $N_t$  mit der Regel  $N_t \rightarrow t$  erzeugen

# Beispiel: Regeln für Terminalsymbole

---

Bisherige Grammatik:

$$S' \rightarrow S \mid \varepsilon$$

$$S \rightarrow SS \mid (S) \mid \varepsilon$$

Füge hinzu:

$$Z_{(} \rightarrow ($$

$$Z_{)} \rightarrow )$$

Ersetze:

$$S \rightarrow (S) \text{ durch } S \rightarrow Z_{(}SZ_{)}$$

# Konstruktionsschritte für Chomsky-NF

---

## ► 3. Regeln der Form $N \rightarrow \varepsilon$ ersetzen

Für jede Regel der Form  $N \rightarrow \varepsilon$

    Für jede Regel  $M \rightarrow uNv$

        Regel  $M \rightarrow uv$  zufügen

    Danach  $N \rightarrow \varepsilon$  streichen

Füge dabei keine Regel hinzu die bereits ersetzt wurde

Dadurch wird der  $\varepsilon$ -Übergang von  $N$  auf die Verwendungsstellen von  $N$  übertragen



# Beispiel: Regeln der Form $N \rightarrow \varepsilon$ ersetzen

---

Bisherige Grammatik:

$$S' \rightarrow S \mid \varepsilon$$

$$S \rightarrow SS \mid Z(SZ) \mid \varepsilon$$

Suche nach Vorkommen von  $S$  und füge  $\varepsilon$  statt  $S$  ein

Füge hinzu:

$$S \rightarrow Z(Z) \quad (\varepsilon \text{ eingefügt in } S \rightarrow Z(SZ))$$

Andere Vorkommen von  $S$ :

$$S' \rightarrow S \quad (S' \rightarrow \varepsilon \text{ gibt es bereits})$$

$$S \rightarrow SS \quad (S \rightarrow \varepsilon \text{ streichen wir gerade})$$

# Konstruktionsschritte für Chomsky-NF

---

## ► 4. Regeln der Form $N \rightarrow M$ ersetzen

Für jede Regel der Form  $N \rightarrow M$

    Für jede Regel der Form  $M \rightarrow w$

        Regel  $N \rightarrow w$  hinzufügen

    Danach  $N \rightarrow M$  streichen

Füge dabei keine Regel hinzu die bereits ersetzt wurde

Dadurch werden die Regeln von  $N$  nach  $M$  übertragen

# Beispiel: Regeln der Form $N \rightarrow M$ ersetzen

---

Bisherige Grammatik:

$$S' \rightarrow S \mid \varepsilon$$

$$S \rightarrow SS \mid Z(SZ) \mid Z(Z)$$

Ersetze  $S' \rightarrow S$

Füge hinzu:

$$S' \rightarrow SS \mid Z(SZ) \mid Z(Z)$$

# Konstruktionsschritte für Chomsky-NF

---

- ▶ 5. Regeln mit mehr als 2 NonTerminals ersetzen

Jede Regel der Form  $N \rightarrow M_1 M_2 \dots M_k$  mit  $k \geq 3$  entfernt und ersetzt durch die k-1 Regel

$$N \rightarrow M_1 L_1, L_1 \rightarrow M_2 L_2 \dots L_{k-1} \rightarrow M_{k-1} M_k$$

Dabei sind  $L_i$  jeweils neue Nichtterminalsymbole

# Beispiel: Regeln mit mehr als 2 NonTerminals ersetzen

---

Bisherige Grammatik:

$$S' \rightarrow SS \mid Z(SZ) \mid Z(Z) \mid \varepsilon$$

$$S \rightarrow SS \mid Z(SZ) \mid Z(Z)$$

Ersetze  $S' \rightarrow Z(SZ)$  und  $S \rightarrow Z(SZ)$

Füge hinzu:

$$S_1 \rightarrow SZ_1$$

$$S' \rightarrow Z_1(S_1)$$

$$S \rightarrow Z_1(S_1)$$

# Beispiel: Grammatik in Chomsky NF

---

Original Grammatik:

$$S \rightarrow SS \mid (S) \mid \varepsilon$$

Grammatik in Chomsky Normalform:

$$S' \rightarrow SS \mid Z(S) \mid Z(Z) \mid \varepsilon$$

$$S \rightarrow SS \mid Z(S) \mid Z(Z)$$

$$S_j \rightarrow SZ_j$$

# Lernziele

---

- ▶ Die Idee der monotonen Grammatik um eine Obergrenze für die Zahl der Ableitungsschritte zu bestimmen
- ▶ Grammatiken ähnlich wie Gleichungssysteme durch Äquivalenzumformungen in eine gewünschte Form bringen

# Mögliche Klausuraufgaben

---

- ▶ Prüfen ob eine Grammatik in Chomsky-Normalform ist
- ▶ Eine Grammatik in Chomsky-Normalform bringen