Theoretische Informatik I
Ausarbeitung der Vorlesung vom 07.07.2000

Daniel Marienfeld[1] Sven Friedrich[2]

21. Juli 2000

[1]dmarien@rz.uni-potsdam.de
[2]sfried@rz.uni-potsdam.de

5 Formale Sprachen und Grammatiken

Das nun folgende Kapitel enthält eine Typisierung von Grammatiken und Sprachen. Des weiteren werden allgemeine Aussagen über Grammatiken und ihre Darstellungen gemacht.

Eine Grammatik ist ein Kalkül zur Erzeugung einer Sprache. Formal ist eine Grammatik ein Quadtupel $G = \{N, T, R, \sigma\}$[1] aus einer Menge von Nonterminalen (Variabalen), von Terminalsymbolen, einer Regelmenge und einem Startsymbol. Jede Grammatikregel hat die Form $P \to Q$ für irgendeine linke Seite (Prämisse) und rechte Seite (Conclusio) Q. P muß mindestens ein Nonterminalsymbol enthalten, Q nicht. Eine solche Regel ist zu verstehen als Ersetzungsregel, die besagt, daß in einem Wort $\alpha P \beta$ das Teilwort P ersetzt werden darf durch Q. Eine Grammatik erzeugt eine Sprache, sie dient also der Synthese. Das Gegenstück dazu ist der Automat, der Sprachen erkennt, er dient der Analyse.

Für eine Grammatik im allgemeinen gibt es nur zwei Einschränkungen: Sie darf nur endlich viele Regeln haben, und jede Regelprämisse muß mindestens ein Nonterminal enthalten. Das Wort kann im Lauf der Ableitung beliebig wachsen und wieder schrumpfen. Wenn man die Form, die die Regeln einer Grammatik annehmen können, beschränkt, erhält man Grammatiktypen und damit auch Sprachtypen von verschiedenen Schwierigkeitsgraden. Diese Typen sollen Gegenstand des folgenden Abschnitts sein.

Definition A (Grammatiktypen). *Sei $G = \{N, T, R, \sigma\}$ eine Grammatik:*

a) *G heißt vom **Typ 3** oder rechtslinear oder regulär, genau dann wenn,*
   $\forall (P \to Q) \in R(P \in N \text{ und } Q \in T^* \cup T^+N))$.
   Es wird ein einzelnes Nonterminalsymbol ersetzt. Mit einer Regelanwendung wird jeweils höchstens ein nonterminales Symbol erzeugt, welches, wenn es auftritt, ganz rechts im Wort steht.

b) *G heißt vom **Typ 2** oder kontextfrei, genau dann wenn,*
   $\forall (P \to Q) \in R(P \in N \text{ und } Q \in (N \cup T)^*)$.
   Es wird ein einzelnes Nonterminalsymbol ersetzt. Das Wort in der Conclusio kann nonterminale und terminale Symbole in beliebiger Mischung enthalten.

c) *G heißt vom **Typ 1** oder kontextsensitiv, genau dann wenn,*
   $\forall (P \to Q) \in R(\exists u, v, \alpha \in (N \cup T)^* \ \exists A \in N(P = uAv \text{ und } Q = u\alpha v \text{ mit } |\alpha| \geq 1)))$, *oder die Regel hat die Form $S \to \varepsilon$. S kommt in keiner Regelconclusio vor. Es wird ein Nonterminalsymbol A in eine Zeichenkette α mit einer Länge von mindestens 1 überführt (d.h. das Wort wird durch die Regelanwendung nicht wieder kürzer). Diese Ersetzung von A durch α findet aber nur statt, wenn der in der Regel geforderte Kontext, links u und rechts v, im Wort vorhanden ist.*

d) *G heißt vom **Typ 0**, wenn G eine Grammatik ist.*

[1] In der Vorlesung wurde die Regelmenge R mit P bezeichnet

e) G heißt vom Erweiterungstyp, wenn für alle $(\alpha, \beta) \in R$ gilt: $|\alpha| \leq |\beta|$, d.h. das Wortproblem[2] ist lösbar für von kontextsensitiven Grammatiken erzeugte Sprachen (Typ 1, auch für Typ 2 und Typ 3).

f) G heißt lineare Grammatik, wenn G kontextfrei ist und für alle $(\alpha, \beta) \in R$ gilt: $\beta \in T^* \circ N \circ T^* \cup T^*$.

Beispiele zu den verschiedenen Grammatiktypen:

Typ 3 Die Sprache $L_a = \{a^n \mid n \in \mathbb{N}\}$ ist eine reguläre Sprache. Sie wird erzeugt von der folgenden Grammatik:

$$G_a = (\{S\}, \{a\}, P, S) \quad \text{mit}$$
$$P = \{S \to aS \mid \epsilon\}$$

Die Grammatik ist rechtslinear: In jeder Regel wird entweder mindestens ein Terminal erzeugt oder die Conclusio ist ϵ, und in jeder Conclusio kommt höchstens eine Variable vor, und zwar am Ende des Wortes.

Typ 2 Die Sprache $L_{ab} = \{a^n b^n \mid n \in \mathbb{N}\}$ ist eine kontextfreie Sprache. Sie wird erzeugt von der folgenden Grammatik:

$$G_{ab} = (\{S\}, \{a, b\}, \{P_1, P_2\}, S) \quad \text{mit}$$
$$P_1 = S \to aSb, \quad P_2 = S \to \epsilon$$

Die Grammatik ist kontextfrei: Auf der linken Seite jeder Regel, sowohl von P_1 als auch von P_2, steht nur eine einzelne Variable.

Typ 1 Die Sprache $L_{abc} = \{a^n b^n c^n \mid n \in \mathbb{N}\}$ ist kontextsensitiv. Sie wird erzeugt von der folgenden beschränkten Grammatik:

$$\begin{aligned}
G_{abc} &= (\{Z, A, B, H, C\}, \{a, b, c\}, \{P_1 \dots P_9\}, Z) \quad \text{mit} \\
P_1 &= Z \to \epsilon \\
P_2 &= Z \to Ac \\
P_3 &= A \to ab \\
P_4 &= A \to aACB \\
P_5 &= CB \to CH \\
P_6 &= CH \to BH \\
P_7 &= BH \to BC \\
P_8 &= B \to b \\
P_9 &= Cc \to cc
\end{aligned}$$

G_{abc} ist eine kontextsensitive Grammatik. Die kontextsensitiven Sprachen sind eine echte Obermenge der kontextfreien Sprachen.

[2] es gibt einen Algorithmus, der nach endlich vielen Schritten entscheidet, ob für ein Wort $w \in X^*$ $w \in L(G)$ gilt

Definition B (Sprachklassen). *Eine Sprache ist vom Typ i, $i = 0, 1, 2, 3$, wenn es eine Grammatik G gibt, die vom Typ i mit $L = L(G)$ ist.*

Sprachklasse	*definiert als*	*Eine Sprache aus dieser Sprachklasse heißt*
$\mathcal{L}_3$, *REG*	$\{L(G) \mid G$ *ist rechtslinear*$\}$	*regulär, Typ 3*
$\mathcal{L}_2$, *CFL*[3]	$\{L(G) \mid G$ *ist kontextfrei*$\}$	*kontextfrei, Typ 2*
$\mathcal{L}_1$, *CSL*[4]	$\{L(G) \mid G$ *ist kontextsensitiv*$\}$	*kontextsensitiv, Typ 1*
$\mathcal{L}_0$	$\{L(G) \mid G$ *ist eine Grammatik*$\}$	*rekursiv aufzählbar, Typ 0*
$\mathcal{L}$	$\{L \subseteq T^* \mid T$ *ist ein Alphabet*$\}$	*Sprache*

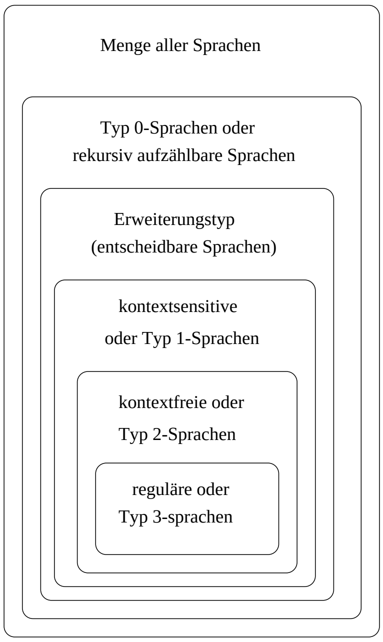


Abbildung 1:

Für die in Definition B angegebenen Sprachklassen gilt die **Chomsky-Hierarchie**:

$$\mathcal{L}_3 \subset \mathcal{L}_2 \subset \mathcal{L}_1 \subset \mathcal{L}_0 \subset \mathcal{L}$$

Der Erweiterungstyp würde in dieser Hierarchie die Teilmenge von $\mathcal{L}_1$ enthalten. Sprachlich würde diese Hierarchie folgendermaßen formuliert werden: Jede Grammatik ist automatisch vom *Typ 0*. Eine Grammatik ist vom *Typ 1*, wenn sie die entsprechenden Eigenschaften besitzt. Eine *Typ 1*–Grammatik ist vom *Typ 2*, wenn sie alle hierzu gehörenden Regeln einhält. Eine *Typ 2*–Grammatik ist vom *Typ 3*, wenn die erforderlichen Bedingungen erfüllt sind.

[3] context free language

[4] context sensitive language

5.1 Kontextfreie Sprachen

Beispiel Sei $G = \{N, T, R, \sigma\}$ mit $T = \{v, c, +, \cdot\}$[5], $N = \{E, T, F\}$[6], $R = \{R_1 \dots R_7\}$ mit $\sigma \to$ E und für R:

$$\begin{array}{lcl} E & \to & E + E \\ E & \to & T \\ T & \to & T \cdot T \\ T & \to & F \\ F & \to & (E) \\ F & \to & v \\ F & \to & c \end{array}$$

$L(G) = \{$arithmetische Ausdrücke mit Operationen $+, \cdot$ und mit den Operanten v, c mit korrekter Klammerung$\}$

Beispiel für eine Ableitung nach obigen Regeln:

$$E \to E + E \to E + E + E \to E + T \cdot T + E \to E + T \cdot T \cdot T + E \to \dots$$
$$\to E + F \cdot F \cdot F + E \to \dots \to^* v + v \cdot c \cdot v + v$$

Definition C (Ableitungsbaum). *Sei $G = \{N, T, R, \sigma\}$ eine kontextfreie Grammatik. Ein Ableitungsbaum*[7] *zu G ist ein angeordneter, knotengewichteter Baum $B = (V, E, v_0)$ für den gilt:*

- *Jeder Knoten $v \in V$ ist mit einem Symbol aus $N \cup T \cup \{\epsilon\}$ markiert.*
- *Die Wurzel v_0 ist mit σ markiert.*
- *Jeder innere Knoten ist mit einem Nonterminalsymbol markiert.*
- *Jedes Blatt ist mit einem Symbol aus $T \cup \{\sigma\}$ markiert*
- *Ist $v \in V$ ein innerer Knoten mit den Söhnen $v_1, \dots, v_R \in V$ in dieser Anordnung, ist A die Markierung von v und A_i die Markierung von v_i, so ist $A \to A_1 \dots A_R \in R$.*
- *Ein mit ϵ markiertes Blatt hat keinen Bruder (denn das entspräche einer Ableitung wie $A \to ab\epsilon Bc$)*

Satz D. *Sei $G = \{N, T, R, \sigma\}$ eine kontextfreie Grammtik. Dann gilt für jedes $w \in T^*$ mit $w = x_1 \dots x_n$, $\sigma \to_G^* w$ genau dann wenn:*
Es gibt einen Ableitungsbaum zu G, dessen Blätter von links nach rechts gelesen mit $x_1, \dots, x_n$ markiert sind.

[5] v - Variable, c - Konstante
[6] E - Expresion, T-Turn, F-Factor
[7] in der Literatur auch Syntaxbaum

Beweis: Die Beweisidee soll hier nur kurz angedeutet werden, da beide Beweisrichtigungen völlig offensichtlich sind. Man beweist hier noch etwas mehr, als der Satz aussagt, indem man w von T^* auf den Bereich $(N \cup T)^*$ und σ auf eine beliebige Variable A verallgemeinert: Zu G und w existiert ein Ableitungsbaum B, so daß

$$(A \Rightarrow_G^* w) \Leftrightarrow \quad \text{Es gibt einen A-Baum in } B \text{ mit Front } w$$

„$\Rightarrow$“ durch Induktion über die Länge der Ableitung

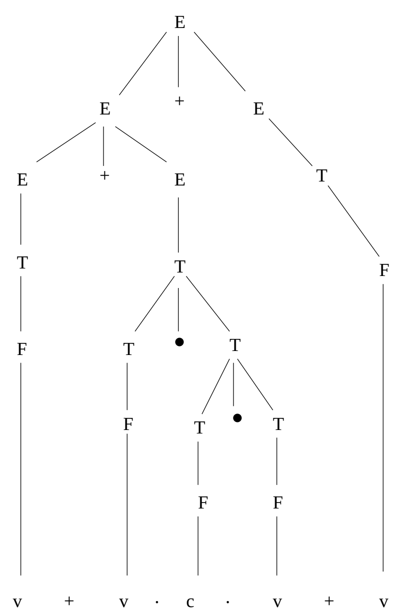
„$\Leftarrow$“ durch Induktion über die Höhe (Tiefe) des Baumes

Verschiedene Ableitungen können durch denselben Ableitungsbaum dargestellt werden.

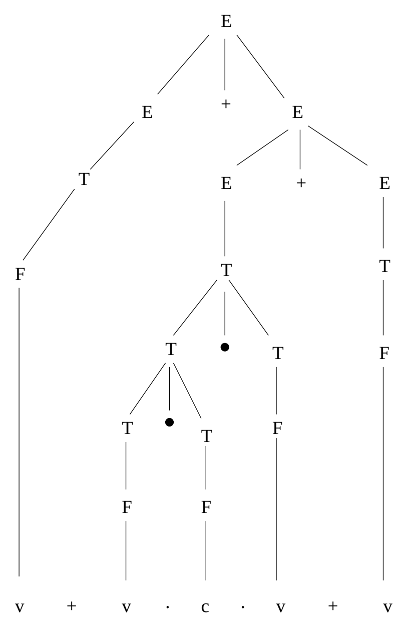
$$\begin{array}{ll} (1) & E \to E + E \to E + T \to T + T \to \ldots \\ (2) & E \to E + E \to T + E \to F + E \to v + E \to \ldots \\ & \quad \to v + T \to v + F \to v + v \end{array}$$

Definition 1 (Linksableitung). [8] *Eine Ableitung $w_1 \Rightarrow_G w_2 \Rightarrow_G \ldots \Rightarrow_G w_n$ heißt Linksableitung, falls $w_i + 1$ durch Ersetzen der linkesten Variable in w_i entsteht für alle $i < n$.*

Definition 2 (Rechtsableitung). *Die Rechtsableitung ist analog zur Linksableitung definiert.*



Linksableitung



Rechtsableitung

[8]Von den Autoren dieses Stückes wurden zusätzliche Definitionen, Sätze oder Hilfssätze eingefügt, da nach deren Meinung diese für das Verständnis des Textes notwending sind. Sie sind im Gegensatz zu den Sätzen aus der Vorlesung mit arabischen Ziffern bezeichnet

Fazit: Jedem Ableitungsbaum läßt sich eine Linksableitung zuordnen. Das heißt, für jede kontextfreie Grammatik G gilt:

$$w \in L(G) \Leftrightarrow \text{Es gibt eine Linksableitung von } w \text{ bezüglich } G$$

Andererseits: Es kann zu einer Grammatik G und einem Wort $w \in L(G)$ mehrere strukturell unterschiedliche Ableitungen geben.

5.1.1 Eindeutige Grammatiken

Definition 3. *Eine Sprache ist genau dann eindeutig, wenn es eine eindeutige Grammatik gibt, d.h., es gibt genau eine Linksableitung zu einem Wort $w \in L(G)$.*

Beispiel: Eindeutige Grammatik $G = \{N, T, R, \sigma\}$ für arithmetische Ausdrücke mit mit $T = \{v, c, +, \cdot\}$, $N = \{E, T, F\}$ und $R = \{R_1 \dots R_7\}$ mit $\sigma \rightarrow$ E und für R:

$$\begin{aligned}
E &\rightarrow E + T \\
E &\rightarrow T \\
T &\rightarrow T \cdot F \\
T &\rightarrow F \\
F &\rightarrow (E) \\
F &\rightarrow v \\
F &\rightarrow c
\end{aligned}$$

5.1.2 Inhärent mehrdeutige Grammatiken

Definition 4. *Eine cf-Grammatik G heißt mehrdeutig - genau dann wenn - es ein Wort $w \in L(G)$ gibt, so daß G zwei verschiedene Linksableitungen zu w besitzt. Eine Sprache $L \in \mathcal{L}_2$ (Chomsky Typ-2) heißt inhärent mehrdeutig gdw. alle kontextfreien Grammatiken für L mehrdeutig sind. Eine Gramatik G ist somit mehrdeutig, wenn es zwei verschiedene Ableitungsbäume in G in gleicher Form gibt.*

Beispiel Ein Beispiel einer inhärent mehrdeutigen Sprache ist die kontextfreie Sprache: $L = \{a^i b^j c^k | i = j \, oder \, j = k\}$. Um den Beweis über die Mehrdeutigkeit bestimmter Sprachen führen zu können, wird zunächst der Begriff des Ogden´s Lemma eingeführt.

Lemma 1 (Ogden's Lemma). *Für jede kontextfreie Sprache L gibt es eine Konstante $n \in N$, so daß für jedes Wort $z \in L$ mit $|z| \geq n$ folgende Aussage gilt:*
Wenn wir in z mindestens n Buchstaben markieren, läßt sich z als $z = uvwxy$ schreiben, daß mindestens ein Buchstabe in vx markiert, in vwx höchstens n Buchstaben markiert sind und für alle $i \geq 0$ das Wort $uv^iwx^iy \in L$ ist.

Wenn wir uns in Ogden's Lemma dafür entscheiden, alle Buchstaben zu markieren, erhalten wir das Pumping-Lemma[9]. Es genügt also, Ogden's Lemma zu beweisen.

Beweis: Für eine kontextfreie Sprache L betrachen wir eine kontextfreie Grammatik G in Chomsky-Normalform[10]. Wir wählen $n = 2^{|v|+1}$. Diese Zahl wird später auch als die Konstante aus Ogden's Lemma bezeichnet. Sei $z \in L$ mit $|z| \geq n$. Im Wort seien mindestens n Buchstaben markiert.

Wir betrachten einen Ableitungsbaum B für z. Da die Grammatik in Chomsky-Normalform ist, ist der Baum binär und hat $|z|$ Blätter, die von links nach rechts das Wort z ergeben. Die einzigen Knoten mit dem Grad 1 sind die Väter der Blätter. Wir wählen einen Weg von der Wurzel von B zu einem Blatt aus. Wir wählen dabei stets die Richtung, in der mehr markierte Blätter liegen. Bei Gleichheit können wir uns beliebig entscheiden. Knoten auf diesem Weg, für die im linken und im rechten Teilbaum markierte Blätter liegen, heißen Verzweigungsknoten. Da $n > 2^{|v|}$ ist, folgt aus unserer Wahl des Weges, daß auf ihm mindestens $|V| + 1$ Verweigungsknoten liegen. Von diesen betrachten wir die letzten $|V| + 1$. Von ihnen müssen nach dem Schubfachprinzip mindestens zwei mit der gleichen Variablen A belegt sein, diese Verzweigungsknoten nennen wir v_1 und v_2 (Abb. 2, Seite 8).

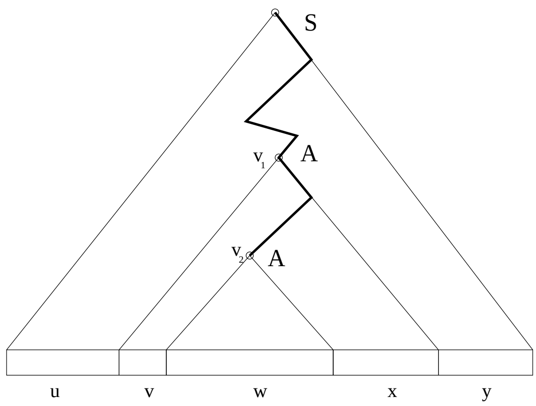


Abbildung 2: Syntaxbaum zum Beweis Ogdens Lemma

In dem Teilbaum mit Wurzel v_2 werde das Teilwort w und z erzeugt. Damit sind auch u und y als passendes Präfix und Suffix für z eindeutig definiert.

Da v_1 Verzweigungsknoten ist, enthalten der linke und der rechte Teilbaum markierte Blätter, und damit enthält vx mindestens ein markiertes Blatt bzw. einen markierten Buchstaben. Da wir einschließlich v_1 auf den in v_1 startenden Teil des ausgewählten Weges nur $|V| + 1$ Verzweigungsknoten haben, und wegen der speziellen Wahl unserer Weges enthält vwx höchstens $2^{|V|+1} = n$ markierte Buchstaben.

In der Grammatik sind, da v_1 und v_2 mit A belegt sind, folgende Ableitungen möglich:

$$S \to^* uAy, A \to^* vAx, A \to^* w.$$

[9]siehe Vorlesung vom 14.07.2000

[10]siehe Vorlesung vom 14.07.2000

Aus diesen Ableitungen erhalten wir die Ableitung

$$S \to^* uAy \to^* uwy = uv^0wx^0y,$$

sowie für $i \geq 1$ die Ableitung

$$S \to^* uAy \to^* uvAxy \to \cdots \to uv^iAx^iy \to uv^iwx^iy.$$

Also ist uv^iwx^iy für $i \geq 0$ in L enthalten.

$\square$

Beweis inhärent mehrdeutige Sprachen:
Beispiel ist die kontextfreie Sprache: $L = \{a^ib^jc^k | i = j \text{ } oder \text{ } j = k\}$.

L läßt sich als Vereinigung zweier Sprachen L' und L'' darstellen. Dabei sind $L' = \{a^ib^i | i \geq 0\}\{c\}^*$und $L'' = \{a\}^*\{b^ic^i | i \geq 0\}$ als Konkatenation kontextfreier Sprachen selbst wieder kontextfrei.

Schneidet man L' und L'' so ist die Sprache $L' \cap L'' = \{a^ib^ic^i | i \geq 0\}$ nicht kontextfrei. Es gibt Wörter aus diesem Teil der Sprache L mit mehreren Ableitungsbäumen. Ziel ist es eine beliebige kontextfreie Grammatik G für L ein m finden, so daß es für $a^mb^mc^m$ zwei Ableitungsbäume bzgl. G gibt. Die Regeln sind definiert, daß der ein Ableitungsbaum die Gleichheit der Anzahl der Buchstaben a und b testet und der andere die Anzahl der Buchstaben b und c.

Nach Satz F[11] existiert zu jeder kontextfreien Grammatik eine äquivalente Grammatik in Chomsky-Normalform. Es wird angenommen, daß G in Chomsky-Normalform vorliegt.

Sei $n \geq 4$ die Konstante aus Ogden's Lemma für die Sprache L. wir betrachten zunächst das Wort $z = a^nb^nc^{n+n!} \in L$ und markieren alle Buchstaben a. Sei $z = uvwxy$ die nach Ogden's Lemma existierende Zerlegung. Da $uv^iwx^iy \in L$ für alle $i \geq 0$ ist, gilt $v, x \in a^* \cup b^* \cup c^*$. Wegen unserer Wahl der Markierung besteht mindestens eines der Wörter v oder x nur aus Buchstaben a.

Fall 1. $x \in a^*$. Dann ist $vx = a^p$ mit $1 \leq p \leq n$. Aber dann ist $uv^2wx^2y = a^{n+p}b^nc^{n+n!} \notin L$ im Widerspruch zu Ogden's Lemma.

Fall 2. $x \in c^*$. Dann ist $v \in a^+$, also $v = a^p$ mit $1 \leq p \leq n$. Wieder ist $uv^2wx^2y \notin L$ im Widerspruch zu Ogden's Lemma.

Fall 3. $x \in b^*$, d.h. $x = b^j$ mit $0 \leq j \leq n$. Dann ist $v = a^p$ mit $1 \leq p \leq n$. Damit $uv^2wx^2y = a^{n+p}b^{n+j}c^{n+n!} \in L$ ist, muß $j = p$ sein.

Damit muß es innerhalb von G (siehe Ogden's Lemma) die folgende Ableitung für eine Variable A geben:

$$S \to^* uAy \to^* uv^kAx^ky \to^* uv^kwx^ky \in L.$$

Für den oben konstruierten Wert p gilt $1 \leq p \leq n$. Sei $k = \frac{n!}{p} + 1$. Dann ist

$$uv^kwx^ky = a^{n+n!}b^{n+n!}c^{n+n!}.$$

[11] siehe Vorlesung vom 14.07.2000

Wir haben also einen Syntaxbaum für $a^{n+n!}b^{n+n!}c^{n+n!}$ gefunden, in dem $A \to^* v^k w x^k$ gilt und wobei in diesem Teil der Ableitung aus A nur die Buchstaben a und b erzeugt werden.

Wir führen nun die gleichen Überlegungen durch, indem wir mit dem Wort $z' = a^{n+n!}b^n c^n \in L$ starten. Wir erhalten auf die gleiche Weise einen Ableitungsbaum für $a^{n+n!}b^{n+n!}c^{n+n!}$, in dem aus der Variablen B, die in diesem Überlegungen die Rolle von A übernimmt, nur die Buchstaben b und c abgeleitet werden. Wenn die gegebene Grammatik G eindeutig ist, sind die beiden Ableitungsbäume isomorph. Da der Teilbaum mit Wurzel A, der $v^k w x^k$ erzeugt, a-Blätter und keine c-Blätter enthält, während der entsprechende Teilbaum mit Wurzel B c-Blätter und keine a-Blätter enthält, liegen diese beiden Knoten in dem Ableitungsbaum nicht auf demselben Weg. Es gibt also eine Ableitung gemäß diesem Ableitungsbaum, die folgendermaßen aussieht:

$$S \to^* t_1 A t_2 B t_3 \quad \text{mit} \quad t_1, t_2, t_3 \in T^*$$

Außerdem gilt

$$A \to^* v^k w x^k \quad \text{für} \quad k \geq 0, v = a^p \quad \text{und} \quad x = b^p \text{ für ein } p > 0.$$

Analog gilt

$$B \to^* (v')^k w' (x')^k \quad \text{für} \quad k \geq 0, v' = b^q \quad \text{und} \quad x' = c^q \text{ für ein } q > 0.$$

Es folgt

$$S \to^* t_1 a^{pk} w b^{pk} t_2 b^{qk} w' c^{qk} t_3 \in L.$$

Jeder Erhöhung von k erhöht die Zahl der a-Buchstaben um p, die Zahl der b-Buchstaben um $p+q$ und die Zahl der c-Buchstaben um q. Für ein genügend großes k ist die Zahl der b-Buchstaben sowohl größer als die Zahl der a-Buchstaben als auch größer als die Zahl der c-Buchstaben. Dies steht im Widerspruch dazu, daß die so erzeugten Wörter alle in L liegen. Wir haben also folgendes Resultat bewiesen.

Die Sprache $L = \{a^i b^j c^k | i = j \text{ } oder \text{ } j = k\}$ ist inhärent mehrdeutig.

□

5.1.3 Backus – Naur – Form

Unter der Backus-Naur-Form (folgend BNF) versteht man eine kompakte Notation zur Niederschrift von kontextfreien (Typ 2) Grammatiken. Die BNF wurde im Zusammenhang mit der Konstruktion der Programmiersprache ALGOL 60 eingeführt. Bisher wurde bei mehreren Regeln, die alle die gleiche linke Seite besitzten geschrieben:

$$\begin{array}{lcl} A & \to & \beta_1 \\ A & \to & \beta_2 \\ & \vdots & \\ A & \to & \beta_n \end{array}$$

Diese Regeln lassen sich nun nach der BNF mit Hilfe eines Metazeichens (|) kürzer notieren:

$$A \rightarrow \beta_1 \,|\, \beta_2 \,|\, \ldots \,|\, \beta_n$$

Eine weitere, kürzere Schreibweise für die folgenden zwei Regeln

$$\begin{aligned} A &\rightarrow \alpha\gamma \\ A &\rightarrow \alpha\beta\gamma \end{aligned}$$

stellt dies dar (hierbei kann das Wort β auch weggelassen werden):

$$A \rightarrow \alpha[\beta]\gamma$$

Für die nachstehenden Regeln

$$\begin{aligned} A &\rightarrow \alpha\gamma \\ A &\rightarrow \alpha B\gamma \\ B &\rightarrow \beta \\ B &\rightarrow \beta B \end{aligned}$$

läßt sich nach Backus und Naur folgendes schreiben:

$$A \rightarrow \alpha\{\beta\}\gamma$$

Das bedeutet, daß das Wort β zwischen α und γ beliebig oft wiederholt (auch 0–mal) werden kann.

In der Literatur [2] spricht man auch von der *erweiterten* Backus-Naur-Form. Da die BNF und die kontextfreien Grammatiken gleichwertig sind, ist es möglich, daß durch die BNF genau die kontextfreien (Typ 2) Sprachen beschrieben werden können. Zusätzlich sei auf die Veranstaltung Grundlagen der Programmierung I bei Professor Budach von 14.12.1999 verwiesen.

Index

Literatur

[1] Katrin Erk and Lutz Priese. *Theoretische Informatik.* Springer, Berlin, 2000.

[2] Uwe Schoening. *Theoretische Informatik - kurzgefaßt.* Spektrum, Heidelberg, Berlin, Oxford, 1995.

[3] Volker Sperschneider and Babara Hammer. *Theoretische Informatik.* Springer, Berlin, 1996.

[4] Ingo Wegener. *Theoretische Informatik.* B.G. Teubner Stuttgart, Stuttgart, 1993.

[5] Siegfried Wendt. *Nichtphysikalische Grundlagen der Informationstechnik.* unveränderter Nachdruck, Potsdam, 1999.