Grundlagen der theoretischen Informatik

Kurt Sieber

Fakultät IV, Department ETI
Universität Siegen

SS 2013

Vorlesung vom 14.05.2013

Kontextfreie Sprachen

Analog zu Linksableitungen definiert man

Definition 2.45 *Ein Ableitungsschritt $u \Rightarrow v$ heißt* Rechtsableitungsschritt*, wenn es Wörter $x \in (N \cup \Sigma)^*$, $z \in \Sigma^*$ und eine Produktion $A \to y$ in G gibt mit $u = xAz$ und $v = xyz$. Eine* Rechtsableitung *ist eine Ableitung, die nur aus Rechtsableitungsschritten besteht.*

In einem Rechtsableitungsschritt wird also stets das letzte, d.h. das am weitesten rechts stehende Nichtterminalzeichen ersetzt.

Definition 2.46 *Sei $G = (\Sigma, N, S, P)$ eine kontextfreie Grammatik. Ein Wort $u \in (N \cup \Sigma)^*$ heißt* Satzform *(*Linkssatzform*,* Rechtssatzform*) von G, wenn es eine Ableitung (Linksableitung, Rechtsableitung) von u aus dem Startzeichen S gibt. Ein Wort $w \in L(G)$ wird manchmal auch als* Satz *der Grammatik G bezeichnet.*

Lemma 2.47 *Sei $G = (\Sigma, N, S, P)$ eine kontextfreie Grammatik, sei $A \in N$ und $w \in \Sigma^*$. Dann gilt:*

- *Zu jeder Ableitung $A \overset{*}{\Rightarrow} w$ existiert ein 'entsprechender' Ableitungsbaum mit Wurzel A und Blattwort w.*
- *Zu jedem Ableitungsbaum mit Wurzel A und Blattwort w existiert genau eine 'entsprechende' Linksableitung $A \overset{*}{\Rightarrow} w$.*

Formal:
Für jedes $A \in N$ und jedes $w \in \Sigma^*$ gilt: Es existiert eine bijektive Funktion zwischen der Menge aller Linksableitungen $A \overset{*}{\Rightarrow} w$ und der Menge aller Ableitungsbäume mit Wurzel A und Blattwort w. Also ist die Anzahl der unterschiedlichen Linksableitungen $A \overset{*}{\Rightarrow} w$ gleich der Anzahl der unterschiedlichen Ableitungsbäume mit Wurzel A und Blattwort w.

All dies gilt auch für Rechtsableitungen anstelle von Linksableitungen.

In der Praxis (Compilerbau) ist es wichtig, dass jedes Wort eine eindeutige Struktur besitzt, denn ein Compiler soll ja nicht nur testen, ob eine eingegebene Zeichenreihe ein arithmetischer Ausdruck ist, sondern er soll diesen Ausdruck auch weiterverarbeiten, d.h. letzten Endes in Maschinencode übersetzen.

Definition 2.48

- *Eine kontextfreie Grammatik heißt* eindeutig, *wenn für jedes Wort $w \in L(G)$ genau ein Ableitungsbaum mit Wurzel S und Blattwort w existiert (oder äquivalent dazu: genau eine Linksableitung $S \stackrel{*}{\Rightarrow} w$). Andernfalls heißt sie* mehrdeutig.
- *Eine kontextfreie Sprache L heißt* inhärent mehrdeutig, *wenn* jede *kontextfreie Grammatik G mit $L = L(G)$ mehrdeutig ist.*

Beispiel:
Unsere Grammatik G für vollständig geklammerte arithmetische Ausdrücke ist eindeutig.

Intuitive Begründung:
Durch die vollständige Klammerung ist die Struktur jedes Ausdrucks eindeutig festgelegt.

Beweisskizze:
Man beweist zunächst (durch eine einfache Induktion über die Länge der Ableitung von e), dass

- jedes Wort $e \in L(G)$ genauso viele öffnende wie schließende Klammern hat,
- jedes echte nichtleere Präfix eines Wortes $e \in L(G)$ *mehr* öffnende als schließende Klammern hat.

Es folgt sofort, dass kein Wort $e \in L(G)$ echtes Präfix eines anderen Wortes $e' \in L(G)$ sein kann.

Kontextfreie Sprachen

Damit kann man nun zeigen, dass jeder Ausdruck $e \in L(G)$ eine eindeutige Struktur hat: Nehmen wir z.B. an, dass ein Ausdruck $e \in L(G)$ sich sowohl in der Form $(e_1 + e_2)$ als auch in der Form $(e'_1 * e'_2)$ darstellen lässt (mit $e_1, e_2, e'_1, e'_2 \in L(G)$).

Dann ist einer der Ausdrücke e_1, e'_1 kürzer als der andere, und müsste deshalb echtes nichtleeres Präfix des anderen sein (weil die beiden Zeichenreihen $(e_1$ und $(e'_1$ Präfixe von e sind). Das ist aber nicht möglich. $\square$

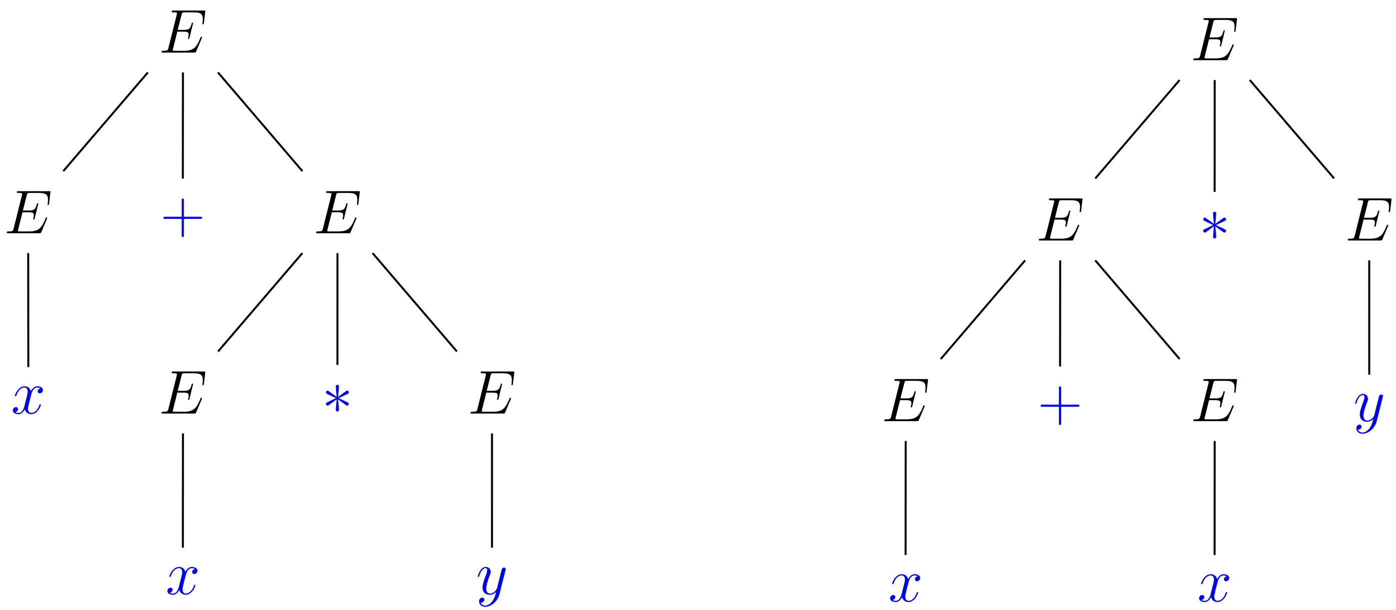
Beispiel für eine mehrdeutige Grammatik:

Sei $G_1 = (\Sigma, N, E, P)$ mit

- $\Sigma = \{0, 1, x, y, -, +, *, (,)\}$
- $N = \{E\}$
- $P = \{E \to 0,\ E \to 1,\ E \to x,\ E \to y,\ E \to -E,$

  $E \to E + E,\ E \to E - E,\ E \to E * E,\ E \to (E)\}$

G_1 erzeugt die Sprache der unvollständig geklammerten arithmetischen Ausdrücke und ist (in hohem Maße) mehrdeutig,
z.B. hat das Wort $x + x * y$ die beiden folgenden Ableitungsbäume.



Gibt es eine eindeutige KFG für $L(G_1)$?

Kontextfreie Sprachen

Sei $G_2 = (\Sigma, N, E, P)$ mit:

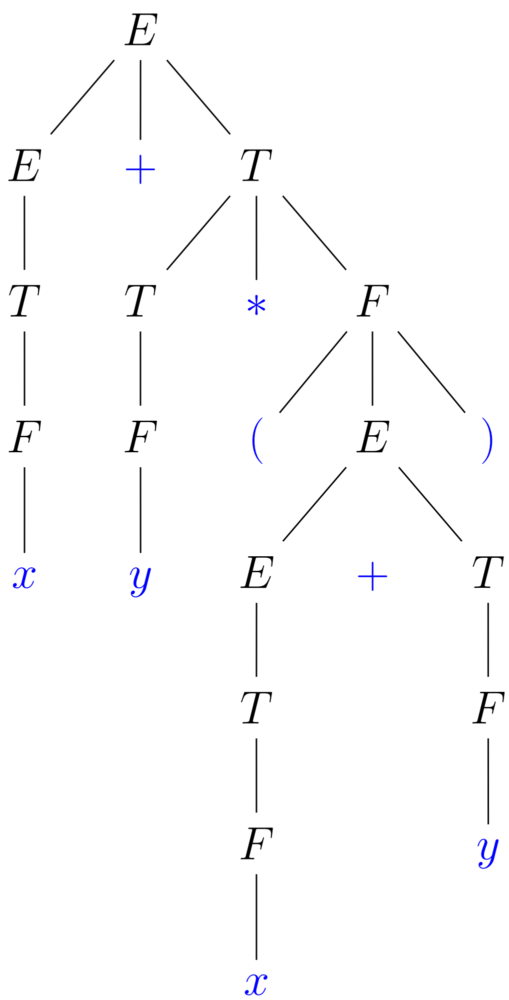
- $\Sigma = \{0, 1, x, y, -, +, *, (,)\}$
- $N = \{E, T, F\}$
- $$\begin{array}{lllll} P = \{ & E & \to & E+T, & (1) \\ & E & \to & E-T, & (2) \\ & E & \to & T, & (3) \\ & T & \to & T*F, & (4) \\ & T & \to & F, & (5) \\ & F & \to & (E), & (6) \\ & F & \to & -F, & (7) \\ & F & \to & 0 \mid 1 \mid x \mid y\} & (8) \end{array}$$

Die eindeutige Linksableitung für $E \stackrel{*}{\Rightarrow} x + y * (x + y)$:

$$\begin{array}{llll} E & \Rightarrow & E+T & (1) \\ & \Rightarrow & T+T & (3) \\ & \Rightarrow & F+T & (5) \\ & \Rightarrow & x+T & (8) \\ & \Rightarrow & x+T*F & (4) \\ & \Rightarrow & x+F*F & (3) \\ & \Rightarrow & x+y*F & (8) \\ & \Rightarrow & x+y*(E) & (6) \\ & \Rightarrow & y+y*(E+T) & (1) \\ & \vdots & & \\ & \Rightarrow & x+y*(x+y) & \end{array}$$

Kontextfreie Sprachen

und der zugehörige Ableitungsbaum



Jetzt wäre noch zu zeigen, dass G_2 eindeutig ist und dass $L(G_2) = L(G_1)$. Das beweisen wir nicht.

Man beachte, dass im Ableitungsbaum die üblichen Prioritäten der Operatoren zum Ausdruck kommen: $*$ bindet stärker als $+$, deshalb ist der Gesamtausdruck von der Form $E + T$. Etwas anderes lässt die Grammatik G_2 nicht zu, weil links von $*$ niemals ein $+$ stehen kann, das nicht durch Klammern ‘geschützt’ ist.

Kontextfreie Sprachen

Wie beweist man, dass eine KFG die gewünschte Sprache erzeugt?

Beispiel:

Sei $L = \{a^n b^n \mid n \geq 0\}$
und $G = (\{a,b\},\{S\},S,P)$ mit $P = \{S \to aSb, S \to \varepsilon\}$

Behauptung: $L(G) = L$.

Beweis:

'$\subseteq$': Sei $w \in \Sigma^*$ mit $S \overset{*}{\Rightarrow} w$.

Es ist zu zeigen, dass $w \in L$.

Dazu beweist man eine 'passende Behauptung' durch *Induktion über die Länge der Ableitung*.

Für den Induktionsschritt hat man zwei Möglichkeiten: Entweder man spaltet den *ersten* Ableitungsschritt ab oder den *letzten*.

Will man den *letzten* Ableitungsschritt abspalten, so muss man eine Behauptung für alle $u \in (N \cup \Sigma)^*$ mit $S \overset{*}{\Rightarrow} u$ (also für alle Satzformen) aufstellen, z.B.:

$$\text{Wenn } u \notin \Sigma^*, \text{ dann ist } u = a^n S b^n \text{ für ein } n \geq 0. \qquad (*)$$

$S \overset{0}{\Rightarrow} u$:

Dann ist $u = S = a^0 S b^0$.

$S \overset{+}{\Rightarrow} u$, d.h. $S \overset{*}{\Rightarrow} v \Rightarrow u$ für ein $v \notin \Sigma^*$:

Dann ist $v = a^n S b^n$ nach Induktionsannahme. Wegen $u \notin \Sigma^*$ kann im Ableitungsschritt $v \Rightarrow u$ nur die Produktion $S \to aSb$ angewandt worden sein, also ist $u = a^{n+1} S b^{n+1}$.

Man kann die Argumentation noch verkürzen: Es genügt offensichtlich zu zeigen, dass (*) für S gilt und bei jedem Ableitungsschritt erhalten bleibt. (*) ist also eine *Invariante* für die Ableitungen aus S.

Kontextfreie Sprachen

Aus (*) folgt schließlich $L(G) \subseteq L$:

Wenn $w \in L(G)$, dann gilt $S \overset{*}{\Rightarrow} u \Rightarrow w$ für ein $u \notin \Sigma^*$, also $u = a^n S b^n$ für ein $n \geq 0$. Wegen $w \in \Sigma^*$ kann $u \Rightarrow w$ nur mit der Produktion $S \to \varepsilon$ erfolgen, also ist $w = a^n b^n \in L$.

Die Abspaltung des *ersten* Ableitungsschrittes ist in unserem Falle wesentlich einfacher.

Man stellt eine Behauptung für alle $w \in \Sigma^*$ mit $S \overset{*}{\Rightarrow} w$ auf, nämlich:

$$w \in L \qquad (**)$$

$S \overset{1}{\Rightarrow} w$:

Dann ist $w = \varepsilon \in L$.

$S \Rightarrow aSb \overset{*}{\Rightarrow} w$:

Dann ist $w = avb$ mit $S \overset{*}{\Rightarrow} v$, also gilt $v \in L$, d.h. $v = a^n b^n$ für ein $n \geq 0$ und damit $w = a^{n+1} b^{n+1}$.

Kontextfreie Sprachen

Geht es immer so einfach?

Im allgemeinen muss man für *jedes* $A \in N$ eine Behauptung über alle $w \in \Sigma^*$ mit $A \stackrel{*}{\Rightarrow} w$ aufstellen, und diese Behauptungen durch *simultane Induktion* beweisen.

Mit anderen Worten: Wenn $|N| = k$, so muss man eine Behauptung über die k Sprachen $L_G(A) = \{w \in \Sigma^* \mid A \stackrel{*}{\Rightarrow} w\}$ aufstellen und durch simultane Induktion beweisen.

'$\supseteq$': Sei $w \in L = \{a^n b^n \mid n \geq 0\}$.

Durch Induktion über $|w|$ beweist man $w \in L(G)$.

$|w| = 0$, d.h. $w = \varepsilon$:

Dann gilt $S \Rightarrow w$ mit Produktion $S \to \varepsilon$.

$|w| > 0$, d.h. $w = a^n b^n$ mit $n > 0$:

Dann ist $w = aa^{n-1}b^{n-1}b$, also $w = avb$ mit $v \in L$. Nach Induktionsannahme gilt $S \stackrel{*}{\Rightarrow} v$, also $S \Rightarrow aSb \stackrel{*}{\Rightarrow} avb = w$.

Auch diese Richtung wird schwieriger, wenn man mehrere Nichtterminalzeichen hat.

Wenn $N = \{A_1, \ldots, A_k\}$, so definiert man k Sprachen $L_1, \ldots, L_k \subseteq \Sigma^*$ und zeigt, dass $L_i \subseteq L_G(A_i)$.

Wir haben im Beispiel die folgenden Eigenschaften von Ableitungen benutzt:

Lemma 2.49

1. *Für alle $A \in N$ und $u, v, w \in (N \cup \Sigma)^*$ gilt:*

   Wenn $A \overset{*}{\Rightarrow} v$, dann $uAw \overset{*}{\Rightarrow} uvw$.

2. *Für alle $A \in N$ und $u, w, x \in \Sigma^*$ gilt:*

   Wenn $uAw \overset{*}{\Rightarrow} x$, dann existiert ein $v \in \Sigma^*$ mit $A \overset{*}{\Rightarrow} v$ und $x = uvw$.

Weiteres Beispiel:

Sei $L = \{w \in \{a,b\}^* \mid \#_a(w) = \#_b(w)\}$.

Es gibt eine einfache Grammatik, die L erzeugt, nämlich:

$G = (\{a,b\}, \{S\}, S, P)$ mit $P = \{S \to SS, S \to aSb, S \to bSa, S \to \varepsilon\}$

Behauptung: $L(G) = L$.

Beweis:

'$\subseteq$': Für jedes $u \in (N \cup \Sigma)^*$ mit $S \overset{*}{\Rightarrow} u$ gilt:

$$\#_a(u) = \#_b(u) \qquad (*)$$

denn (*) gilt für S und bleibt bei jedem Ableitungsschritt erhalten (weil in jeder Produktion gleich viele as und bs hinzukommen).

Also gilt (*) insbesondere für jedes $w \in L(G)$, d.h. $L(G) \subseteq L$.

'$\supseteq$': Sei $w \in L$, $w = a_1 \dots a_n$.

Wenn $w = \varepsilon$, so gilt $S \Rightarrow w$ mit Produktion $S \to \varepsilon$.

Es bleibt der Fall $|w| > 0$ zu betrachten.

Sei $f : \{a, b\}^* \to \mathbb{Z}$ mit $f(v) = \#_a(v) - \#_b(v)$.

Dann gilt $L = \{v \in \{a, b\}^* \mid f(v) = 0\}$.

Wir betrachten den 'Verlauf' der Funktion f für die Präfixe von w, d.h. für die Wörter $w_i = a_1 \dots a_i$ mit $0 \leq i \leq n$.

Es gilt $f(w_0) = f(\varepsilon) = 0$ und $f(w_n) = f(w) = 0$, weil $w \in L$.

1. Fall: $f(w_i) \geq 0$ für $i = 0, \dots, n$

Dann ist $f(w_1) = 1$ und $f(w_{n-1}) = 1$, also $w = avb$ für ein $v \in L$.

Nach Induktionsannahme gilt $S \overset{*}{\Rightarrow} v$, also $S \Rightarrow aSb \overset{*}{\Rightarrow} avb = w$.

2. Fall: $f(w_i) \leq 0$ für $i = 0, \ldots, n$

   Analog zum 1. Fall folgt $w = bva$ für ein $v \in L$,

   also gilt $S \Rightarrow bSa \overset{*}{\Rightarrow} bva = w$.

3. Fall: Es existieren $i, j \in \{0, \ldots, n\}$ mit $f(w_i) > 0$ und $f(w_j) < 0$.

   Dann existiert eine Zahl k zwischen i und j mit $f(w_k) = 0$, also $w_k = a_1 \ldots a_k \in L$.

   Dann ist auch $w'_k = a_{k+1} \ldots a_n \in L$ und es gilt $w = w_k w'_k$.

   Nach Induktionsannahme gilt $S \overset{*}{\Rightarrow} w_k$ und $S \overset{*}{\Rightarrow} w'_k$, also $S \Rightarrow SS \overset{*}{\Rightarrow} w_k S \overset{*}{\Rightarrow} w_k w'_k = w$. □