Automaten und Formale Sprachen
9. Vorlesung

Prof. Dr. Dietrich Kuske

FG Automaten und Logik, TU Ilmenau

Wintersemester 2022/23

Kontext-freie Sprachen

Wiederholung: Produktionen kontext-freier Grammatiken

Bei kontext-freien Grammatiken haben alle Produktionen die Form $A \to w$ mit $A \in V$ (d.h., A ist ein Nichtterminal) und $w \in (V \cup \Sigma)^*$.

Anwendungen kontext-freier Sprachen

Hauptanwendung: Beschreibung der Syntax von Programmiersprachen

Viele der hier besprochenen Techniken sind daher interessant für den Compilerbau.

Bemerkung: die natürliche Sprache hat viele kontext-freie Bestandteile, ist aber nicht wirklich kontext-frei, da viele subtile Kontextabhängigkeiten berücksichtigt werden müssen.

Ableitungsbäume

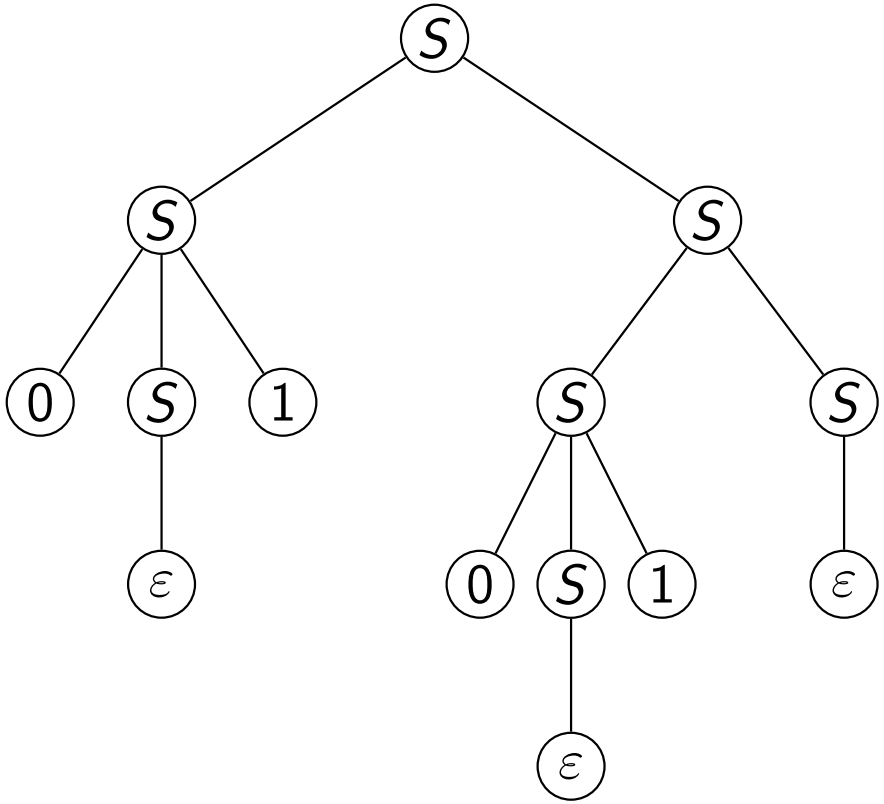
Beispiel:
Grammatik habe die Regeln

$$S \to \varepsilon,\ S \to SS \text{ und } S \to 0S1$$

(sie erzeugt die Menge `Kl` der korrekten Klammerausdrücke). Betrachte die Ableitung

$$\begin{aligned} \underline{S} &\Rightarrow S\underline{S} \Rightarrow S\underline{S}S \Rightarrow S0\underline{S}1S \Rightarrow \underline{S}01S \\ &\Rightarrow 0\underline{S}101S \Rightarrow 0101\underline{S} \Rightarrow 0101\,. \end{aligned}$$

Wir konstruieren daraus schrittweise einen Baum.



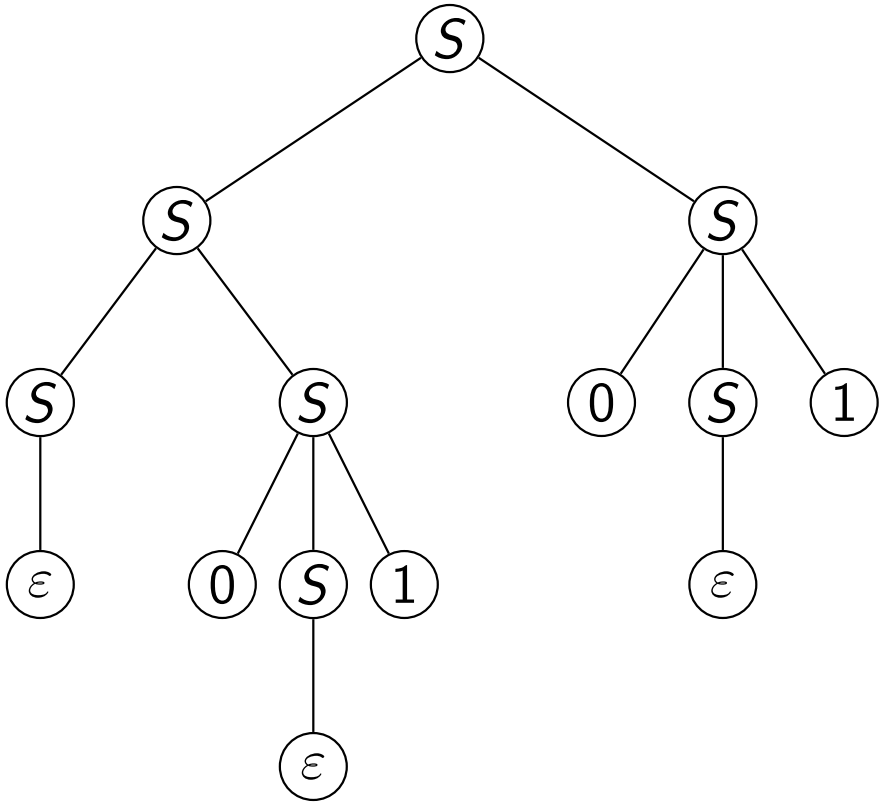
Auch die folgende Ableitung erzeugt diesen Baum (wirklich?):

$$\underline{S} \Rightarrow S\underline{S} \Rightarrow \underline{S}SS \Rightarrow 0S1S\underline{S} \Rightarrow 0\underline{S}1S \Rightarrow 01\underline{S} \Rightarrow 010\underline{S}1 \Rightarrow 0101$$

Wir betrachten nun die Ableitung

$$\underline{S} \Rightarrow \underline{S}S \Rightarrow \underline{S}SS \Rightarrow \underline{S}S \Rightarrow 0\underline{S}1S \Rightarrow 01\underline{S} \Rightarrow 010\underline{S}1 \Rightarrow 0101\,,$$

die dasselbe Wort erzeugt. Sie liefert allerdings einen anderen Baum:



Definition

Sei $G = (V, \Sigma, P, S)$ eine kontext-freie Grammatik und $X \in V \cup \Sigma$.
Ein X-Ableitungsbaum ist ein gerichteter, geordneter Baum T mit Wurzel, dessen Knoten mit Elementen von $V \cup \Sigma \cup \{\varepsilon\}$ beschriftet sind, wobei:

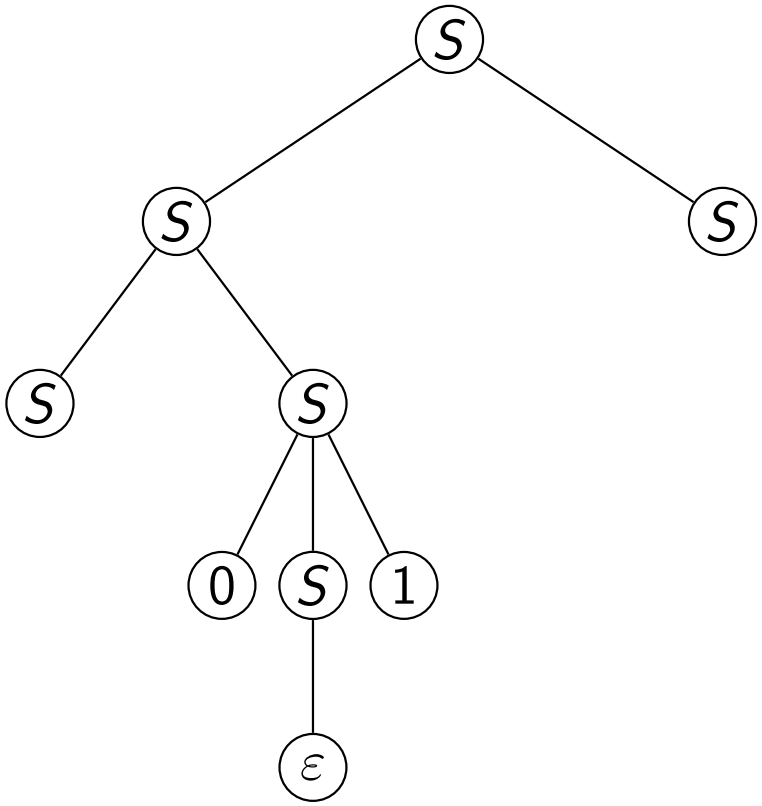
- die Wurzel mit X beschriftet ist,
- Knoten v mit $a \in \Sigma \cup \{\varepsilon\}$ beschriftet $\Longrightarrow$ v ist Blatt
- Knoten v mit $A \in V$ beschriftet und kein Blatt $\Longrightarrow$
  - es gibt eine Produktion $A \to X_1 \cdots X_r$ mit $X_1, \ldots, X_r \in \Sigma \cup V$ $(r \geq 1)$, so daß die Nachfolgerknoten von v mit X_1, X_2, $\ldots X_r$ beschriftet sind (in dieser Reihenfolge)
  - oder es gibt Produktion $A \to \varepsilon$ und v hat genau einen Nachfolger; dieser ist mit ε beschriftet.

Das Blattwort $\alpha(T)$ des X-Ableitungsbaums T erhält man, indem man die Beschriftungen der Blätter von links nach rechts betrachtet.
Ein Ableitungsbaum ist ein S-Ableitungsbaum.

Definition (Fortsetzung)

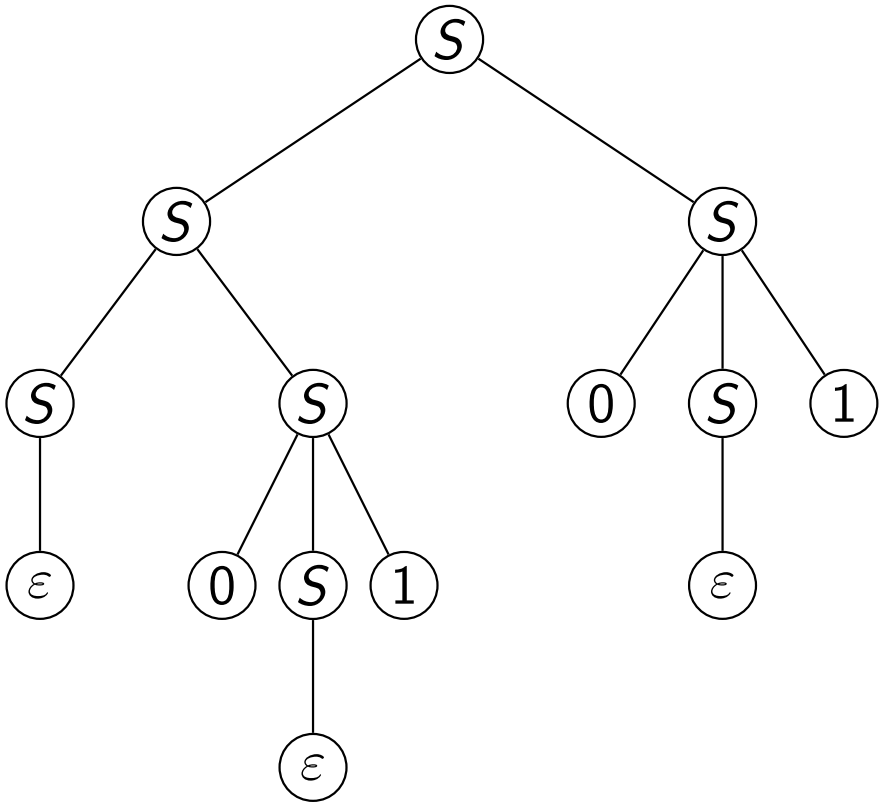
Ein X-Ableitungsbaum ist vollständig, wenn seine Blätter mit Elementen von $\Sigma \cup \{\varepsilon\}$ beschriftet sind.

ein S-Ableitungsbaum



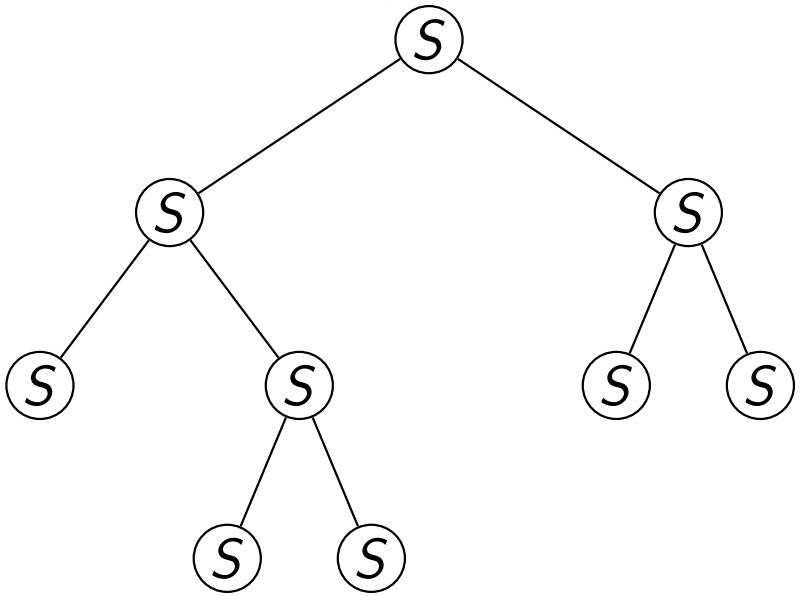
$\alpha(T) = S\,0\,\varepsilon\,1S = S01S$

ein vollständiger S-Ableitungsbaum



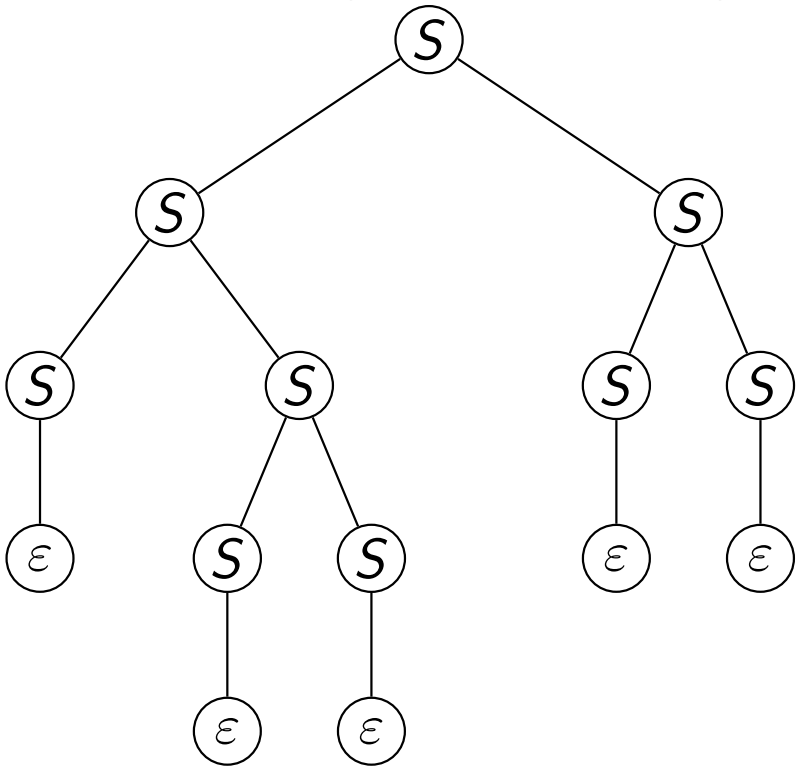
$\alpha(T) = \varepsilon\,0\,\varepsilon\,10\,\varepsilon\,1 = 0101$

ein S-Ableitungsbaum



$\alpha(T) = SSSSS$

ein vollständiger S-Ableitungsbaum



$\alpha(T) = \varepsilon\,\varepsilon\,\varepsilon\,\varepsilon\,\varepsilon = \varepsilon$

Lemma

$G = (V, \Sigma, P, S)$ kontext-freie Grammatik, $X \in V \cup \Sigma$, $w \in (V \cup \Sigma)^*$.
Dann sind äquivalent:

1. $X \Longrightarrow^* w$.
2. Es gibt einen X-Ableitungsbaum T mit $w = \alpha(T)$.

Insbesondere:

$$
\begin{aligned}
w \text{ ist Satzform von G} &\iff \text{es gibt einen } (S\text{-})\text{Ableitungsbaum } T \\
&\qquad \text{mit } \alpha(T) = w \\
w \in L(G) &\iff w \in \Sigma^* \text{ und es gibt einen} \\
&\qquad \text{Ableitungsbaum } T \text{ mit } \alpha(T) = w \\
&\iff \text{es gibt einen vollständigen} \\
&\qquad \text{Ableitungsbaum } T \text{ mit } \alpha(T) = w
\end{aligned}
$$

Beweis:

„(1) $\Rightarrow$ (2)“ per Induktion über die Länge n der Ableitung $X \Longrightarrow^* w$:

I.A. gilt $n = 0$, so hat T genau einen Knoten; dieser ist mit X beschriftet.

I.S. Gelte $X \Rightarrow^n v \Rightarrow w$. Dann existieren

1. X-Ableitungsbaum T_v mit Blattwort v (nach I.V.)
2. Regel $A \to y = X_1 X_2 \dots X_r$ und $x, z \in (V \cup \Sigma)^*$ mit $v = xAz$ und $w = xyz$.

Ergänze T_v um r Knoten, die mit X_1, X_2, $\dots X_r$ beschriftet sind und erkläre diese zu Kindern des entsprechenden A-Knotens im Blattwort xAz von T_v. Der neue Ableitungsbaum hat das Blattwort $xX_1 \dots X_r z = w$.

„(2) $\Rightarrow$ (1)“
Aus X-Ableitungsbaum T konstruiere eine Ableitung $X \Longrightarrow^* \alpha(T)$ durch Induktion über die Größe von T.

I.A.: Wenn T nur aus der Wurzel oder der Wurzel mit ε-Kind besteht: Behauptung klar.

I.S.: T hat X-Wurzel, $r > 0$ Kinder mit Beschriftung $X_1, \ldots X_r \in \Sigma \cup V$.

Dann ist $X \to X_1 \ldots X_r$ Produktion. Die Unterbäume $T_1, \ldots, T_r$ unter den r Kindern sind X_1-, …, X_r-Ableitungsbäume mit $\alpha(T) = \alpha(T_1) \ldots \alpha(T_r)$.

Nach I.V. gibt es Ableitungsfolgen $X_i \Longrightarrow^* \alpha(T_i)$, $i = 1, \ldots, r$.

Baue diese zusammen: $X \Rightarrow X_1 \ldots X_r \Longrightarrow^* \alpha(T_1)X_2 \ldots X_r \Longrightarrow^* \alpha(T_1)\alpha(T_2)X_3 \ldots X_r \Longrightarrow^* \ldots \Longrightarrow^* \alpha(T_1) \ldots \alpha(T_r) = \alpha(T)$. □

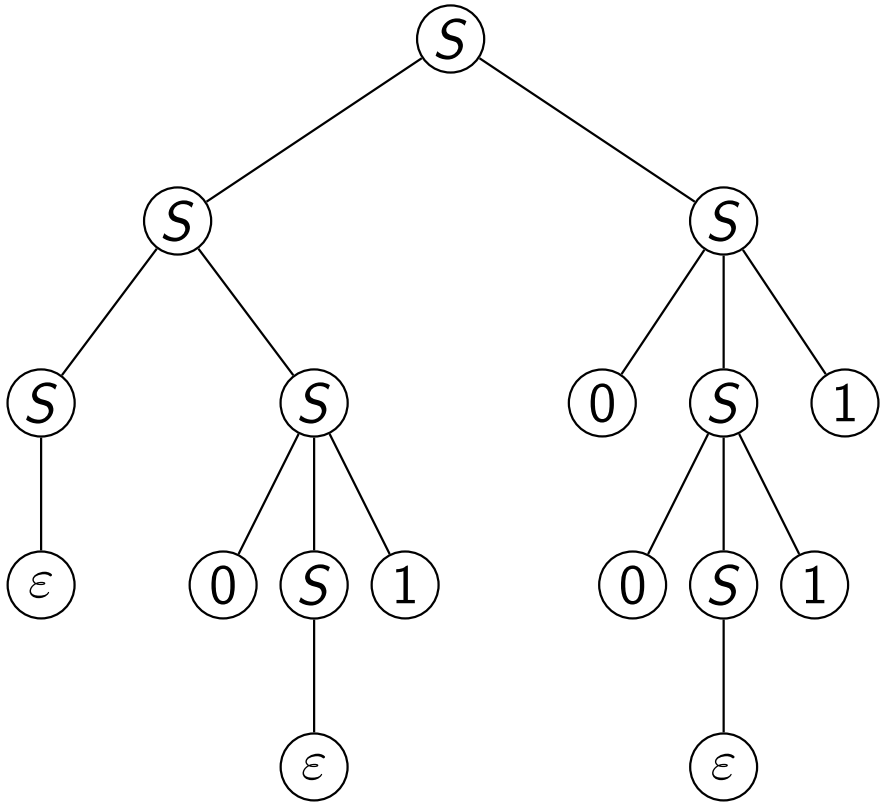
Beispiel: Grammatik mit Produktionen $S \to SS \mid 0S1 \mid \varepsilon$.

Betrachte die folgenden Ableitungen:

$$\begin{aligned}
\underline{S} &\Rightarrow \underline{S}S \Rightarrow \underline{S}SS \Rightarrow \underline{S}S \Rightarrow 0\underline{S}1S \Rightarrow 01\underline{S} \Rightarrow 010\underline{S}1 \Rightarrow 0100\underline{S}11 \\
&\Rightarrow 010011 \\
\underline{S} &\Rightarrow S\underline{S} \Rightarrow S0\underline{S}1 \Rightarrow S00\underline{S}11 \Rightarrow \underline{S}0011 \Rightarrow S\underline{S}0011 \Rightarrow S0\underline{S}10011 \\
&\Rightarrow \underline{S}010011 \Rightarrow 010011 \\
\underline{S} &\Rightarrow \underline{S}S \Rightarrow SS\underline{S} \Rightarrow \underline{S}S0S1 \Rightarrow \underline{S}0S1 \Rightarrow 0S10\underline{S}1 \Rightarrow 0\underline{S}100S11 \\
&\Rightarrow 0100\underline{S}11 \Rightarrow 010011
\end{aligned}$$

Dies sind verschiedene Ableitungen des Worts 010011 ...

. . . aber all diese Ableitungen erzeugen den Ableitungsbaum



Linksableitungen

Wir haben gesehen, daß es zu jedem Ableitungsbaum eine Ableitung gibt. Es kann aber auch mehrere geben (siehe Beispiele). Wir werden sehen, daß es immer genau eine „Linksableitung“ gibt.

Definition

Eine Ableitung $S = w_0 \Rightarrow w_1 \Rightarrow \cdots \Rightarrow w_r = w \in \Sigma^*$ heißt Linksableitung, wenn in jedem Schritt das am weitesten links stehende Nichtterminal ersetzt wird: in

$$w_{s-1} = u_{s-1} A v_{s-1} \Rightarrow u_{s-1} x v_{s-1} = w_s$$

gilt $u_{s-1} \in \Sigma^*$.

Analog werden Rechtsableitungen definiert: man verlangt $v_{s-1} \in \Sigma^*$.

Beispiel: Grammatik mit Produktionen $S \to SS \mid 0S1 \mid \varepsilon$.

Betrachte wieder die folgenden Ableitungen von Folie 9.13, die alle denselben Ableitungsbaum erzeugen:

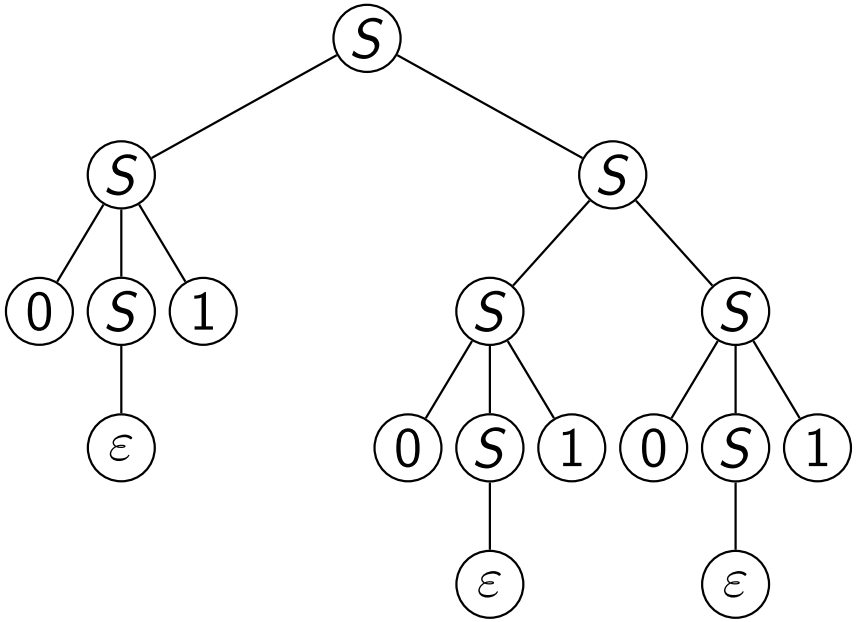
$$\begin{aligned}
\underline{S} &\Rightarrow \underline{S}S \Rightarrow \underline{S}SS \Rightarrow \underline{S}S \Rightarrow 0\underline{S}1S \Rightarrow 01\underline{S} \Rightarrow 010\underline{S}1 \Rightarrow 0100\underline{S}11 \\
&\Rightarrow 010011 \\
\underline{S} &\Rightarrow S\underline{S} \Rightarrow S0\underline{S}1 \Rightarrow S00\underline{S}11 \Rightarrow \underline{S}0011 \Rightarrow S\underline{S}0011 \Rightarrow S0\underline{S}10011 \\
&\Rightarrow \underline{S}010011 \Rightarrow 010011 \\
\underline{S} &\Rightarrow \underline{S}S \Rightarrow SS\underline{S} \Rightarrow \underline{S}S0S1 \Rightarrow \underline{S}0S1 \Rightarrow 0S10\underline{S}1 \Rightarrow 0\underline{S}100S11 \\
&\Rightarrow 0100\underline{S}11 \Rightarrow 010011
\end{aligned}$$

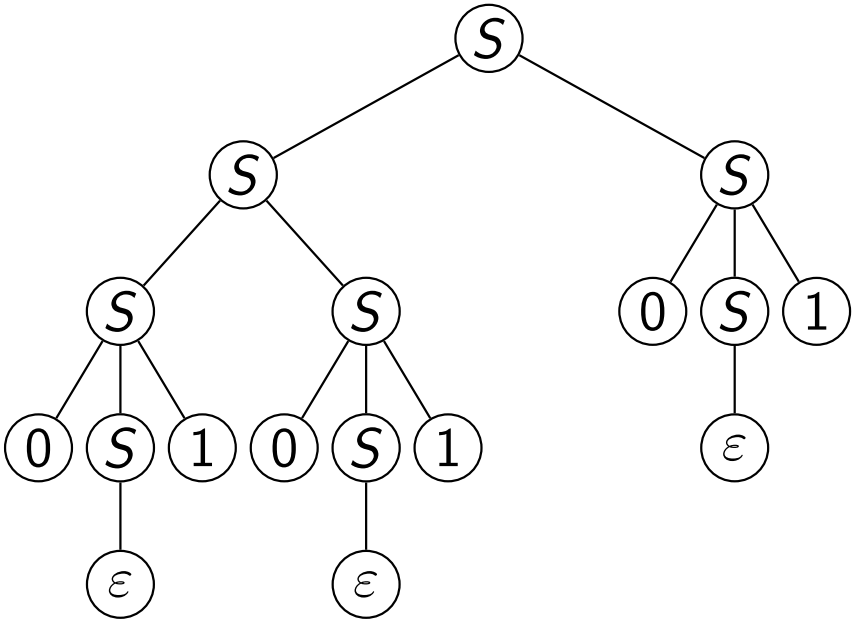
Die erste ist eine Linksableitung, die zweite eine Rechtsableitung und die dritte weder Links- noch Rechtsableitung.

Beispiel:

$$\begin{aligned}
\underline{S} &\Rightarrow \underline{S}S \Rightarrow 0\underline{S}1S \Rightarrow 01\underline{S} \Rightarrow 01\underline{S}S \Rightarrow 010\underline{S}1S \Rightarrow 0101\underline{S} \Rightarrow 01010\underline{S}1 \\
&\Rightarrow 010101 \\
\underline{S} &\Rightarrow \underline{S}S \Rightarrow \underline{S}SS \Rightarrow 0\underline{S}1SS \Rightarrow 01\underline{S}S \Rightarrow 010\underline{S}1S \Rightarrow 0101\underline{S} \Rightarrow 01010\underline{S}1 \\
&\Rightarrow 010101
\end{aligned}$$

sind verschiedene Linskableitungen des Wortes 010101, die unterschiedliche Bäume erzeugen:





Ableitungsbäume und Linksableitungen für w entsprechen einander eineindeutig, genauer:

Satz

Die Konstruktion auf Folie 9.11 ist eine Bijektion der Menge der Linksableitungen von Wörtern aus Σ^* auf die Menge der vollständigen Ableitungsbäume.

Beweis: Für eine Linksableitung LA sei T_{LA} der auf Folie 9.11 konstruierte Ableitungsbaum.

Wir zeigen also, daß $LA \mapsto T_{LA}$ eine Bijektion ist:

1. z.z. ist Injektivität:

   Betrachte zwei verschiedene Linksableitungen LA und LA' der Wörter w bzw. w' aus $L(G)$. Wir betrachten den ersten Unterschied in den Ableitungen LA und LA':

   $$LA = (w_0 \Rightarrow w_1 \Rightarrow \cdots \Rightarrow w_{t-1} \Rightarrow w_t \Rightarrow^* w)$$

   und

   $$LA' = (w_0 \Rightarrow w_1 \Rightarrow \cdots \Rightarrow w_{t-1} \Rightarrow w'_t \Rightarrow^* w')$$

   mit $w_t \neq w'_t$
   Dann wird im Übergang von w_{t-1} nach w_t bzw. w'_t dasselbe Nichtterminal A in w_{t-1} (nämlich die am weitesten links stehende) durch verschiedene rechte Seiten von Produktionen $A \to r$ bzw. $A \to r'$ ersetzt. In T_{LA} bzw. $T_{LA'}$ hat derselbe A-Knoten also unterschiedliche Nachfolger-Strukturen.

   $\Longrightarrow$ Die Abbildung $LA \mapsto T_{LA}$ ist injektiv.

2. z.z. ist Surjektivität: Für einen vollständigen Ableitungsbaum T sei $LA(T)$ die auf Folien 9.12 konstruierte Ableitung.

   Dann ist $LA(T)$ Linksableitung (überprüfen, dies stimmt i.a. nicht, wenn T nicht vollständig ist!) und es gilt $T_{LA(T)} = T$.

   $\Longrightarrow$ Die Abbildung $LA \mapsto T_{LA}$ ist surjektiv. □

(für Rechtsableitungen kann man analog argumentieren)

Wir haben gesehen, daß es Wörter gibt, die verschiedene Linksableitungen (und damit verschiedene Ableitungsbäume) haben. Da die Ableitungsbäume Stukturinformationen über das Wort wiedergeben, ist dies nicht erwünscht.

Definition

Eine kontext-freie Grammatik G heißt mehrdeutig, wenn es zwei verschiedene vollständige Ableitungsbäume T und T' gibt mit $\alpha(T) = \alpha(T')$.
Sonst heißt G eindeutig, d.h. G ist eindeutig wenn jedes Wort $w \in L(G)$ genau einen Ableitungsbaum besitzt.

Eine kontext-freie Sprache L heißt inhärent mehrdeutig, wenn jede kontext-freie Grammatik G mit $L = L(G)$ mehrdeutig ist.

Bemerkung: In dieser Definition hätten wir auch „Linksableitung“ an Stelle von „Ableitungsbaum“ schreiben können.

Beispiel

1. Die Grammatiken $S \to \varepsilon \mid 0S1$ bzw. $S \to 0S1S \mid \varepsilon$ sind eindeutig.
2. Die Grammatik $S \to \varepsilon \mid SS \mid 0S1$ ist mehrdeutig.
3. Die kontext-freie Sprache $L = \{a^k b^\ell c^m \mid k = \ell \text{ oder } \ell = m\}$ ist inhärent mehrdeutig.

Bemerkungen:

1. ist trivial bzw. moderate schwierig.
2. siehe Folie 9.17.
3. Die Grammatik mit den folgenden Regeln erzeugt L:

$$\begin{array}{lllllll} S & \to & AB \mid CD & A & \to & \mathtt{a}A\mathtt{b} \mid \varepsilon & C \to \mathtt{a}C \mid \varepsilon \\ & & & B & \to & \mathtt{c}B \mid \varepsilon & D \to \mathtt{b}D\mathtt{c} \mid \varepsilon \end{array}$$

Dann hat $a^n b^n c^n \in L(G)$ zwei Linksableitungen: die erste beginnt mit $S \to AB$, die zweite mit $S \to CD$.
Dies beweist nicht die inheränte Mehrdeutigkeit, vermittelt aber zumindest die Intuition (wir werden den Beweis hier nicht führen).

kontext-freie Sprachen sind kontext-sensitiv

Lemma

Aus einer kontext-freien Grammatik $G = (V, \Sigma, P, S)$ kann eine kontext-sensitive und gleichzeitig kontext-freie Grammatik G' berechnet werden mit $L(G) = L(G')$.

Beweis:
zunächst berechnen wir Mengen P_i von Produktionen wie folgt:

$$\begin{aligned} P_0 &= P \\ P_{i+1} &= P_i \cup \left\{ (A, \alpha\beta) \;\middle|\; (A, \alpha B \beta), (B, \varepsilon) \in P_i \right\} \end{aligned}$$

und definieren $G_i = (V, \Sigma, P_i, S)$.

Behauptung 1 Für alle $i \in \mathbb{N}$ gilt $L(G_i) = L(G)$.

IA $i = 0$: trivial wegen $G = G_0$

IS $i \geq 0$: wir zeigen $L(G_{i+1}) = L(G)$.

„$\supseteq$“: trivial wegen $P_{i+1} \supseteq P_i \supseteq P_{i-1} \supseteq \cdots \supseteq P_0 = P$.

„$\subseteq$“: Sei $S = \gamma_0 \Rightarrow_{G_{i+1}} \gamma_1 \Rightarrow_{G_{i+1}} \gamma_2 \cdots \Rightarrow_{G_{i+1}} \gamma_n = w \in L(G_{i+1})$.
Wir zeigen $\gamma_k \Rightarrow^*_{G_i} \gamma_{k+1}$ für alle $0 \leq k < n$: Sei also $0 \leq k < n$.

- 1. Fall: $\gamma_k \Rightarrow_{G_i} \gamma_{k+1}$ - fertig
- 2. Fall: es gilt nicht $\gamma_k \Rightarrow_{G_i} \gamma_{k+1}$
  $\Longrightarrow$ Schritt $\gamma_k \Rightarrow_{G_{i+1}} \gamma_{k+1}$ verwendet eine Regel aus $P_{i+1} \setminus P_i$
  Diese Regel hat die Form $(A, \alpha\beta)$ mit $(A, \alpha B \beta), (B, \varepsilon) \in P_i$.
  Wendet man diese beiden Regeln nacheinander auf γ_k an, so erhält man $\gamma_k \Rightarrow^2_{G_i} \gamma_{k+1}$.

Also gilt $w \in L(G_i) \stackrel{IV}{=} L(G)$. **q.e.d.**

Behauptung 2 Es gibt $j \in \mathbb{N}$ mit $P_j = P_{j+1}$.

Beweis Es gilt $P \subseteq P_1 \subseteq P_2 \subseteq \cdots$.

Sei $n \in \mathbb{N}$ mit $|\gamma| \leq n$ für alle $(A, \gamma) \in P$.

Nach Konstruktion gilt dann auch $|\gamma| \leq n$ für alle $(A, \gamma) \in P_i$, d.h. $P_i \subseteq V \times (V \cup \Sigma)^{\leq n}$.

Da diese Menge endlich ist, existiert ein $j \in \mathbb{N}$ mit $P_j = P_{j+1}$. **q.e.d.**

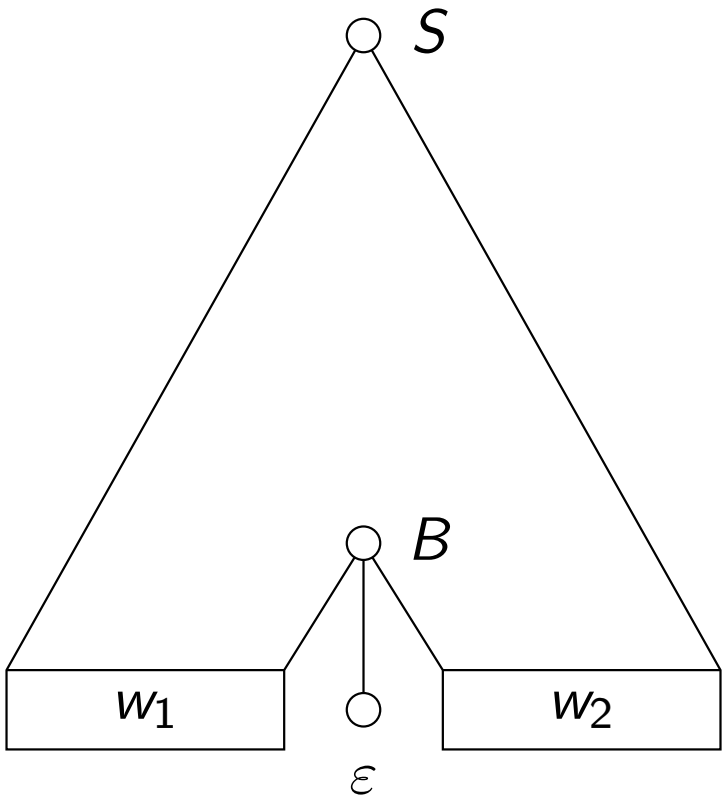
Wir setzen $\widehat{P} = P_j$ und $\widehat{G} = G_j$.

Es gilt

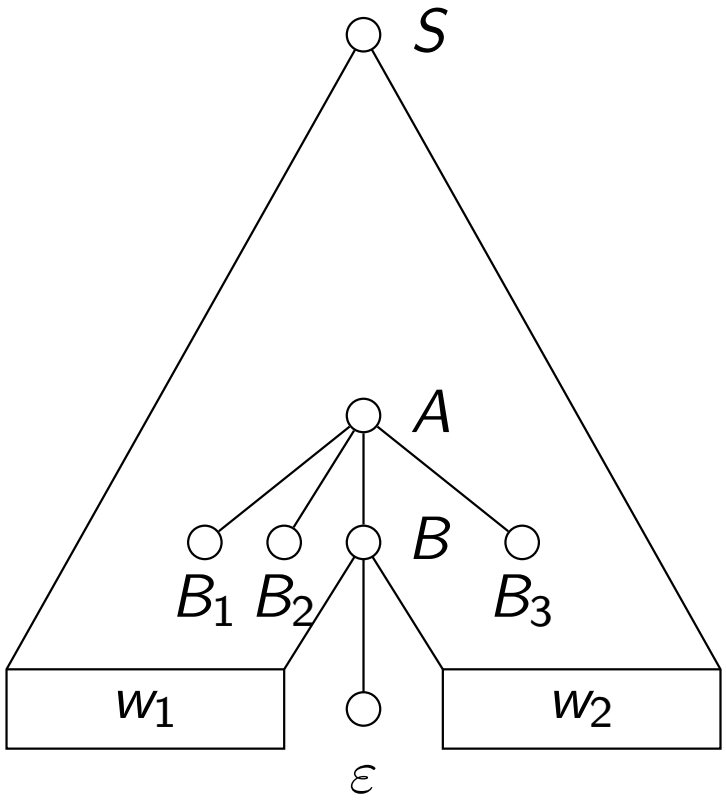
$$(A, \alpha B \beta), (B, \varepsilon) \in \widehat{P} \implies (A, \alpha\beta) \in P_{j+1} = \widehat{P} \qquad (1)$$

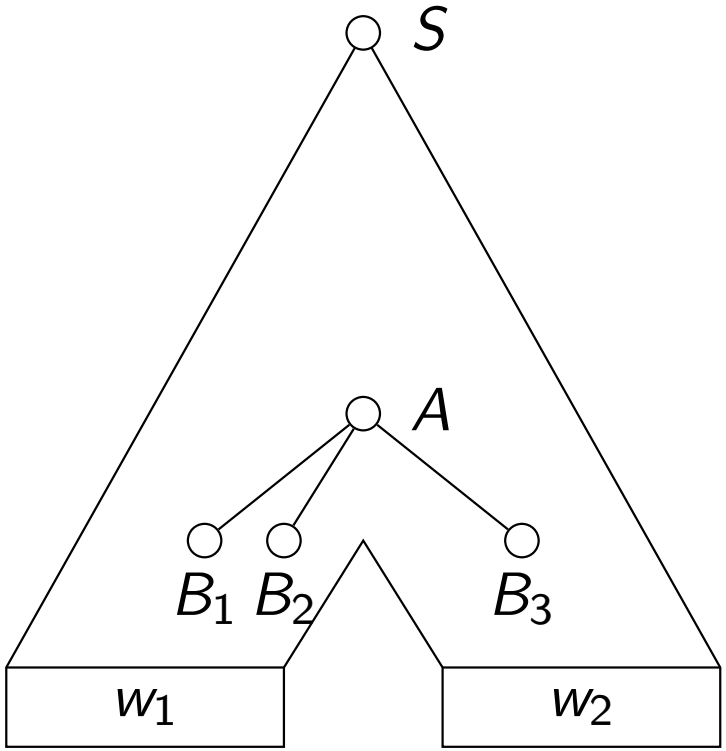
Behauptung 3 Für jedes Wort $w \in L(\widehat{G}) \cap \Sigma^+$ existiert ein Ableitungsbaum T mit Blattwort w, der keine Regel $(B, \varepsilon) \in \widehat{P}$ verwendet.

Beweis Sei $w \in L(\widehat{G}) \cap \Sigma^+$. Sei T ein kleinster Ableitungsbaum mit Blattwort w. Angenommen, dieser verwendet eine Regel (B, ε):



Wegen $w = w_1 w_2 \neq \varepsilon$ existiert der Vater des B-beschrifteten Knotens; sei A seine Beschriftung. Also ist $(A, B_1 B_2 B B_3) \in \widehat{P}$. Nach (1) ist auch $(A, B_1 B_2 B_3) \in \widehat{P}$, wir erhalten also den rechten echt kleineren Ableitungsbaum mit Blattwort w. Da dies im Widerspruch zur Wahl von T steht, verwendet T tatsächlich keine Regel (B, ε). **q.e.d.**





Behauptung 4 $\varepsilon \in L(\widehat{G}) \iff (S, \varepsilon) \in \widehat{P}$.

Beweis

„$\Leftarrow$“: klar

„$\Rightarrow$“: Wie in Behauptung 3 wird gezeigt, daß der kleinste Ableitungsbaum mit Blattwort ε nur zwei Knoten hat (mit Beschriftung S bzw. ε), woraus $(S, \varepsilon) \in \widehat{P}$ folgt. **q.e.d.**

Sei $S' \notin V \cup \Sigma$ eine neues Startsymbol. Setze

$$P' = \begin{cases} \left\{(A, \gamma) \in \widehat{P} \colon \gamma \neq \varepsilon\right\} \cup \left\{(S', S)\right\} & \text{falls } (S, \varepsilon) \notin \widehat{P} \\ \left\{(A, \gamma) \in \widehat{P} \colon \gamma \neq \varepsilon\right\} \cup \left\{(S', S), (S', \varepsilon)\right\} & \text{sonst} \end{cases}$$

$$\text{und } G' = \left(V \cup \{S'\}, \Sigma, P', S'\right).$$

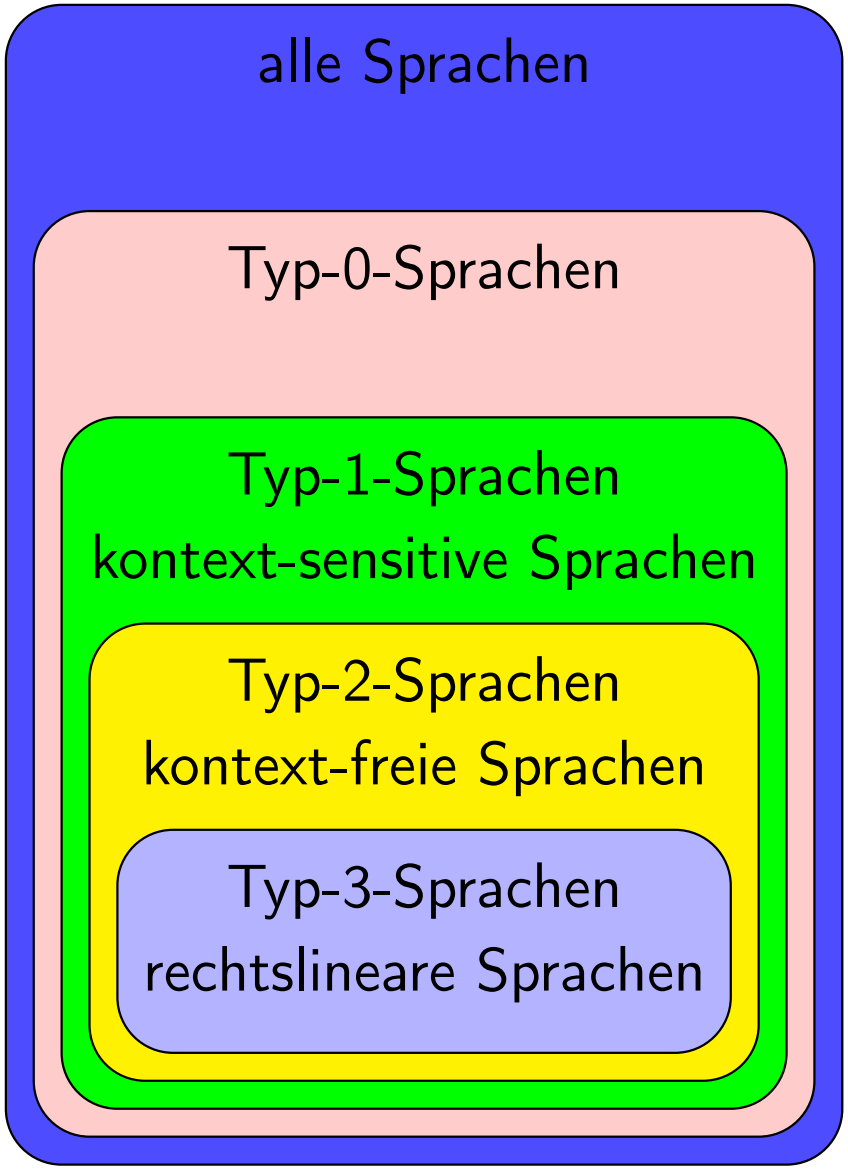
Dann ist G' kontext-frei und kontext-sensitiv. Nach Behauptungen 3 und 4 gilt $L(G') = L(G)$. □

Satz

Jede kontext-freie Sprache ist kontext-sensitiv.

Auf Folie 2.23 wurden $\mathcal{L}_1 \subseteq \mathcal{L}_0$ und $\mathcal{L}_3 \subseteq \mathcal{L}_2$ gezeigt. Jetzt haben wir auch $\mathcal{L}_2 \subseteq \mathcal{L}_1$, es ergibt sich also nebenstehendes Bild.

Zusätzlich wissen wir $\mathcal{L}_3 \subsetneq \mathcal{L}_2$ und $\mathcal{L}_0 \subsetneq$ Klasse aller Sprachen, d.h. es gibt eine kontext-freie Sprache, die nicht rechtslinear ist, und eine Sprache, die nicht rekursiv aufzählbar ist.



Folgerung

Es gibt einen Algorithmus, der als Eingabe eine Typ-2-Grammatik $G = (V, \Sigma, P, S)$ und ein Wort $w \in \Sigma^*$ bekommt und nach endlicher Zeit entscheidet, ob $w \in L(G)$ gilt.

Beweis:

Zunächst wird die Typ-2-Grammatik G in eine äquivalente Typ-1-Grammatik umgewandelt (siehe Lemma auf Folie 9.23). Dann kann Satz von Folie 2.24 angewandt werden. □

Zusammenfassung 9. Vorlesung

in dieser Vorlesung neu

- Ableitungsbäume, Linksableitungen
- jede kontext-freie Sprache ist kontext-sensitiv, es kann also (in exponentieller Zeit) bestimmt werden, ob ein geg. Wort von einer geg. kontext-freien Grammatik erzeugt wird.

kommende Vorlesung

- Chomsky-Normalform („schöne kontext-freie Grammatiken“)
- darauf aufbauend schnellerer Algorithmus für obige Frage (CYK-Algorithmus)