1 Automatentheorie und Formale Sprachen

1.1 Allgemeines

Beispiel: Arithmetische Ausdrücke: $\mathtt{EXPR}$

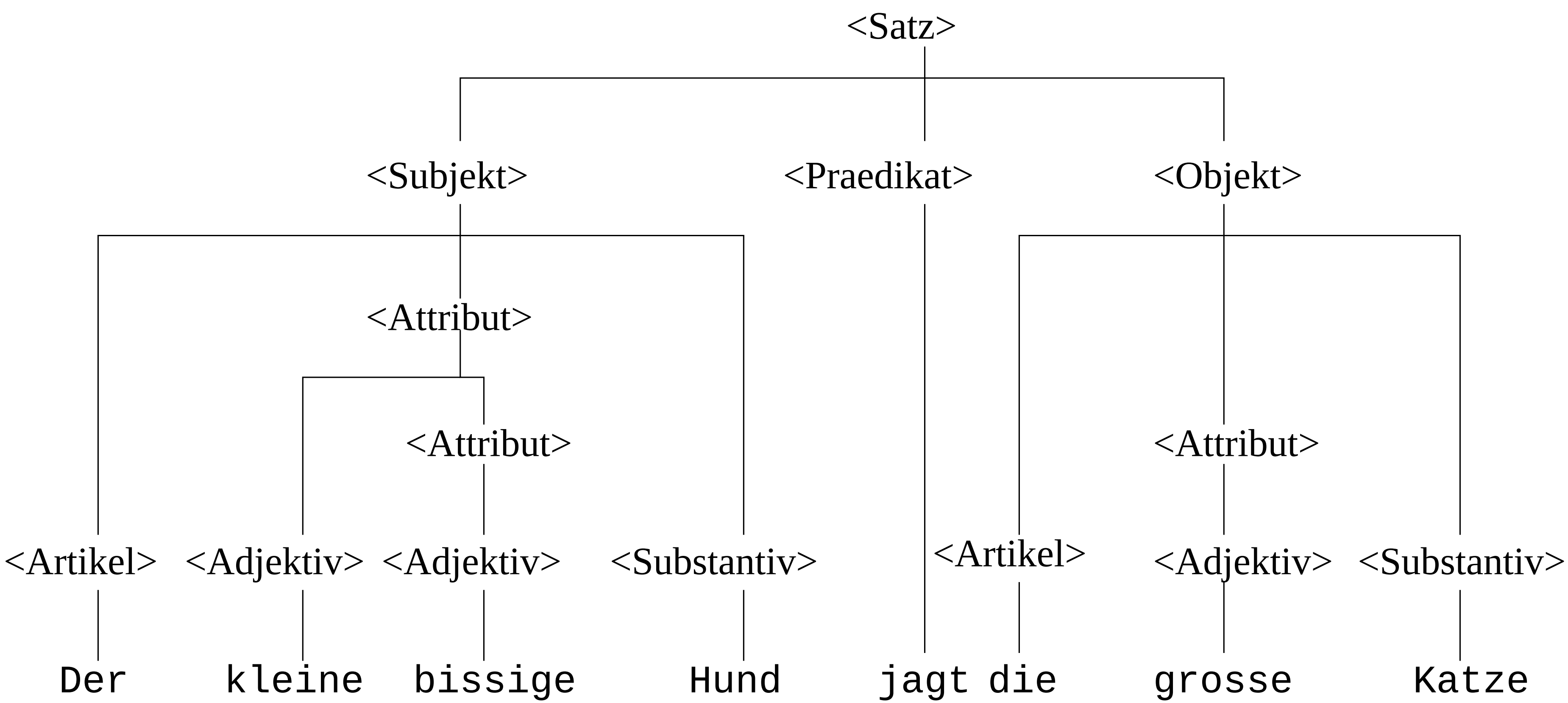
$$\Sigma = \{1, a, +, -, *, (,)\}$$

a ist Platzhalter für Konstanten oder Variablen

$$\begin{aligned} (a-a)*a+a/(a+a)-1 &\in \mathtt{EXPR} \\ (((a))) &\in \mathtt{EXPR} \\ ((a+)-a(&\notin \mathtt{EXPR} \end{aligned}$$

Wie wird sowas formalisiert?

Beispiel: Deutsche Grammatik



Zumindest ein Teil der Struktur lässt sich durch kontextfreie Grammatiken erfassen (stay tuned)

1.1.1 Grammatiken

Grammatik $G = (V, \Sigma, P, S)$

- V, Variablen
- Σ, Terminalalphabet $(V \cap \Sigma = \emptyset)$
- $P \subseteq (V \cup \Sigma)^+ \times (V \cup \Sigma)^*$, Produktionen, $|P| < \infty$

  Alle linken Seiten enthalten mindestens eine Variable
- S, Startvariable

Beispiel: Klammerausdrücke

$G = (\{E,T,F\},\{a,+,*,(,)\},P,E)$ mit

$$
\begin{aligned}
P = \{&E \to T,\\
&E \to E+T,\\
&T \to F,\\
&T \to T*F,\\
&F \to a,\\
&F \to (E)\}
\end{aligned}
$$

Übergangsrelation $\Rightarrow$

Gegeben, Grammatik $G = (V, \Sigma, P, S)$.

$u \Rightarrow_G v$ falls

$u = xyz \in (V \cup \Sigma)^*$,

$v = xy'z \in (V \cup \Sigma)^*$,

$y \to y' \in P$.

„u geht unmittelbar über in v.“

Subskript$_G$ fehlt, wenn klar ist, welche Grammatik gemeint ist.

Übergangsrelation $\overset{*}{\Rightarrow}, \overset{n}{\Rightarrow}$

Ableitungen bestimmter Länge:

$\forall u \in (V \cup \Sigma)^* : u \overset{0}{\Rightarrow} u$

$\forall u, v, w \in (V \cup \Sigma)^* : u \Rightarrow v \wedge v \overset{n}{\Rightarrow} w \longrightarrow u \overset{n+1}{\Rightarrow} w$

Ableitbarkeit:

$\exists n \geq 0 : u \overset{n}{\Rightarrow} v \longrightarrow u \overset{*}{\Rightarrow} v$

Beobachtung: $\overset{*}{\Rightarrow}$ ist die reflexive und transitive Hülle von $\Rightarrow$.

$u \overset{*}{\Rightarrow}_G v$ bedeutet „v ist ableitbar aus u“

Von $G = (V, \Sigma, P, S)$ definierte Sprache

$$L(G) := \left\{ w \in \Sigma^* : S \overset{*}{\Rightarrow} w \right\}$$

Ableitung

Folge von Satzformen,

$$(\underbrace{w_1}_{=S}, \underbrace{w_2}_{\in(\Sigma\cup V)^*}, \ldots, \underbrace{w_{n-1}}_{\in(\Sigma\cup V)^*}, \underbrace{w_n}_{\in\Sigma^*})$$

heißt Ableitung von w_n falls

$$w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n.$$

Beispiel:

$E \Rightarrow$	$E \to E+T$
$\Rightarrow E+T$	$E \to T$
$\Rightarrow T+T$	$T \to T*F$
$\Rightarrow T*F+T$	$T \to T*F$
$\Rightarrow T*F*F+T$	$T \to F$
$\Rightarrow F*F*F+T$	$F \to a$
$\Rightarrow a*F*F+T$	$F \to a$
$\Rightarrow a*a*F+T$	$F \to (E)$
$\Rightarrow a*a*(E)+T$	$E \to E+T$
$\Rightarrow a*a*(E+T)+T$	$E \to T$
$\Rightarrow a*a*(T+T)+T$	$T \to F$
$\Rightarrow a*a*(F+T)+T$	$F \to a$
$\Rightarrow a*a*(a+T)+T$	$T \to F$
$\Rightarrow a*a*(a+F)+T$	$F \to a$
$\Rightarrow a*a*(a+a)+T$	$T \to F$
$\Rightarrow a*a*(a+a)+F$	$F \to a$
$\Rightarrow a*a*(a+a)+a$	

1.1.2 Chomsky-Hierarchie

- Elegante Spezifikation von Sprachen
- Klassifikation von Sprachen

Klassifikation von Grammatiken

$G = (V, \Sigma, P, S)$ **[Noam Chomsky, 1956]**

Sei $G = (V, \Sigma, P, S)$.

$\forall \ell \rightarrow r \in P:$

Typ 0: beliebig

Typ 1, kontextsensitiv: $|\ell| \leq |r|$

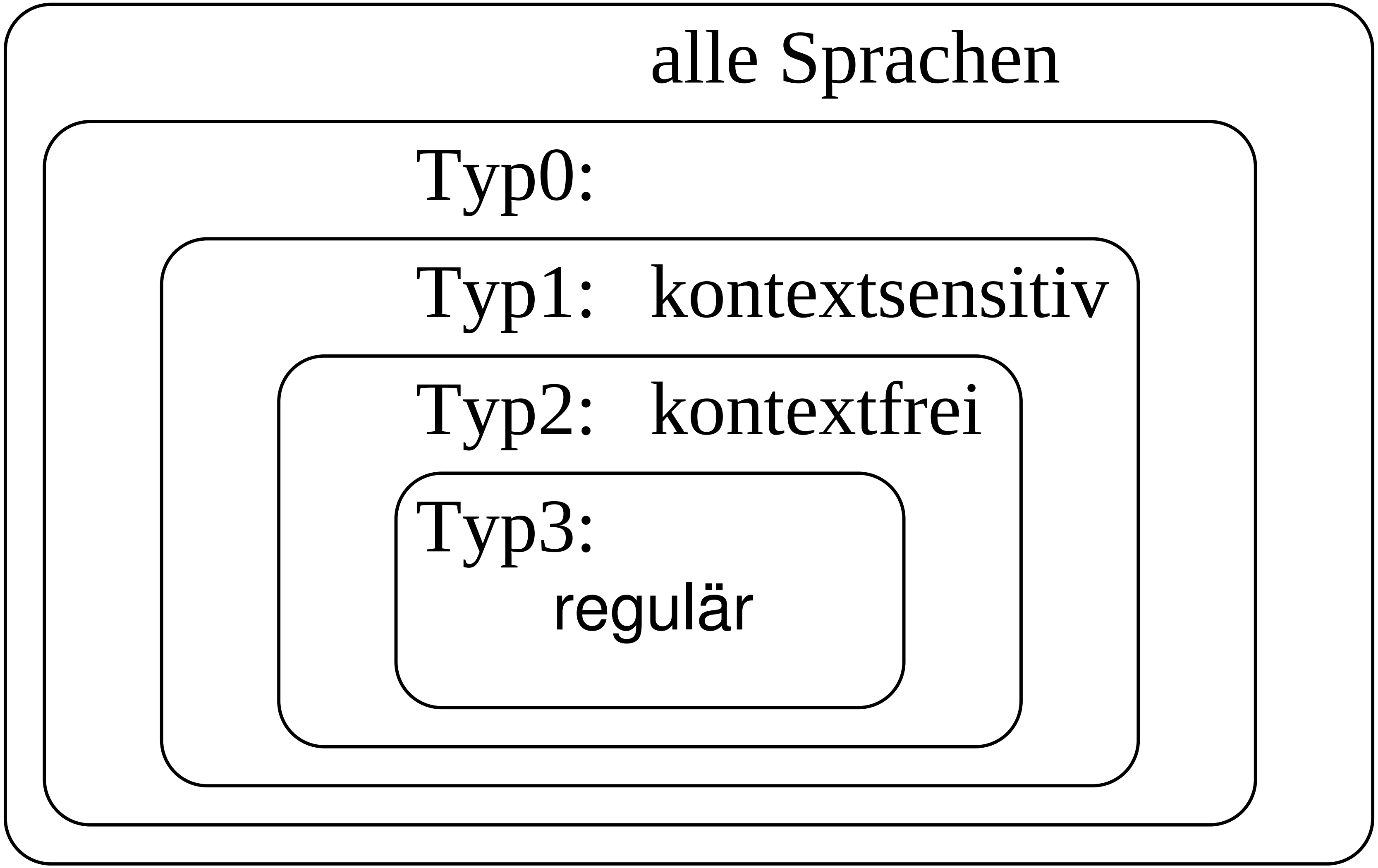
Sonderregel: $S \rightarrow \varepsilon$ erlaubt aber dann $S \notin r$,

Vorsicht: in der Lit. nicht einheitlich gehandhabt!

Typ 2, kontextfrei: Typ 1 und $\ell \in V$ $A \rightarrow \varepsilon$ z.T. erlaubt

Typ 3, regulär: Typ 2 und $r \in \{\varepsilon\} \cup \Sigma \cup \Sigma V$

Chomsky Hierarchie



Beispiel: Typ 3

$G = (\{A, B\}, \{a, b\}, P, A)$ mit

$$
\begin{aligned}
P = \{A &\to aA, \\
A &\to aB, \\
B &\to bB, \\
B &\to b\}
\end{aligned}
$$

Behauptung: $L(G) = \{a^n b^m : n \geq 1, m \geq 1\}$

Beweis, Ansatz:

1. $L(G) \supseteq \{a^n b^m : n \geq 1, m \geq 1\}$
2. $L(G) \subseteq \{a^n b^m : n \geq 1, m \geq 1\}$

Jeweils durch vollständige Induktion

$$\boxed{G = (\{A, B\}, \{a, b\}, \{A \to aA, A \to aB, B \to bB, B \to b\}, A)}$$

Beweis: $L(G) \supseteq \{a^n b^m : n \geq 1, m \geq 1\}$ ausführlich

Lemma 1: $\forall n \geq 1 : A \overset{*}{\Rightarrow} a^n B$

$n = 1$**:** $A \rightarrow aB \in P$

$n \rightsquigarrow n+1$**:** $A \rightarrow aA \underbrace{\overset{*}{\Rightarrow}}_{IV} aa^n B = a^{n+1} B$

$$\boxed{G = (\{A, B\}, \{a, b\}, \{A \rightarrow aA, A \rightarrow aB, B \rightarrow bB, B \rightarrow b\}, A)}$$

Beweis: $L(G) \supseteq \{a^n b^m : n \geq 1, m \geq 1\}$ **ausführlich**

Lemma 1: $\forall n \geq 1 : A \stackrel{*}{\Rightarrow} a^n B$

Lemma 2: $\forall m \geq 1 : B \stackrel{*}{\Rightarrow} b^m$

$m = 1$: $B \to b \in P$

$m \rightsquigarrow m+1$: $B \to bB \underbrace{\stackrel{*}{\Rightarrow}}_{IV} bb^m = b^{m+1}$

$$\boxed{G = (\{A, B\}, \{a, b\}, \{A \to aA, A \to aB, B \to bB, B \to b\}, A)}$$

Beweis: $L(G) \supseteq \{a^n b^m : n \geq 1, m \geq 1\}$ ausführlich

Lemma 1: $\forall n \geq 1 : A \overset{*}{\Rightarrow} a^n B$

Lemma 2: $\forall m \geq 1 : B \overset{*}{\Rightarrow} b^m$

Beweis $\supseteq$: $\forall n \geq 1, m \geq 1 : A \underbrace{\overset{*}{\Rightarrow}}_{\text{Lemma 1}} a^n B \underbrace{\overset{*}{\Rightarrow}}_{\text{Lemma 2}} a^n b^m$

Also $a^n b^m \in L(G)$ (Def. $L(G)$)

$$\boxed{G = (\{A, B\}, \{a, b\}, \{A \to aA, A \to aB, A \to a, B \to bB, B \to b\}, A)}$$

Beweisskizze: $L(G) \supseteq \{a^n b^m : n \geq 1, m \geq 1\}$

$$\underbrace{A \overset{n-1}{\Rightarrow}}_{A \to aA} a^{n-1}A \Rightarrow a^n B \underbrace{\overset{m-1}{\Rightarrow}}_{B \to bB} a^n b^{m-1} B \Rightarrow a^n b^m$$

$$\boxed{G = (\{A, B\}, \{a, b\}, \{A \to aA, A \to aB, B \to bB, B \to b\}, A)}$$

Beweis: $L(G) \subseteq \{a^n b^m : n \geq 1, m \geq 1\}$ ausführlich

Induktion über die Ableitungslänge ℓ:

(Stärkere) Induktionsvoraussetzung: $\forall \alpha \in (V \cup \Sigma)^* : A \overset{\leq \ell}{\Rightarrow} \alpha \longrightarrow$
$\alpha \in \{a\}^* \cdot A \cup \{a\}^+ \cdot \{b\}^* \cdot B \cup \{a\}^+ \cdot \{b\}^+$

$\ell = 0$: $A \in \{a\}^* \cdot A$

$\ell \rightsquigarrow \ell + 1$:Betrachte Ableitung $S \overset{*}{\Rightarrow} \alpha' \overbrace{\Rightarrow}^{C \to \beta} \alpha$

α'	$C \to \beta$	α	$\longrightarrow \alpha \in$
$a^n A$	$A \to aA$	$a^{n+1} A$	$\{a\}^* \cdot A$
$a^n A$	$A \to aB$	$a^{n+1} B$	$\{a\}^+ \cdot \{b\}^* \cdot B$
$a^n b^m B$	$B \to bB$	$a^n b^{m+1} B$	$\{a\}^+ \cdot \{b\}^* \cdot B$
$a^n b^m B$	$B \to b$	$a^n b^{m+1}$	$\{a\}^+ \cdot \{b\}^+$

$\blacksquare$

Beweisskizze: $L(G) \subseteq \{a^n b^m : n \geq 1, m \geq 1\}$

Falls $A \overset{*}{\Rightarrow} \alpha$ so folgt $\alpha \in \{a\}^* \cdot A \cup \{a\}^+ \cdot \{b\}^* \cdot B \cup \{a\}^+ \cdot \{b\}^+$.

Die Produktionen erhalten diese

Invariante. ■

$$\boxed{G = (\{A, B\}, \{a, b\}, \{A \to aA, A \to aB, B \to bB, B \to b\}, A)}$$

Beispiel: Typ 2

$G = (\{S\}, \{a, b\}, \{S \to ab, S \to aSb\}, S)$.
$L(G) = \{a^n b^n : n \geq 1\}$.

Beweisskizze $L(G) \supseteq \{a^n b^n : n \geq 1\}$:
$S \overset{n-1}{\Rightarrow} a^{n-1} S b^{n-1} \Rightarrow a^n b^n$. ■

Beweisskizze $L(G) \subseteq \{a^n b^n : n \geq 1\}$:
$S \overset{*}{\Rightarrow} \alpha \longrightarrow \alpha \in \left\{a^k S b^k : k \geq 0\right\} \cup \left\{a^n b^n : n \geq 1\right\}$
(Invariante) ■

Beispiel: Typ 1

$$G = (\{S, B, C\}, \{a, b, c\}, P, S)$$

$$\begin{aligned} P = \{S &\to aSBC, \\ S &\to aBC, \\ CB &\to BC, \\ aB &\to ab, \\ bB &\to bb, \\ bC &\to bc, \\ cC &\to cc\} \end{aligned}$$

Behauptung: $L(G) = \{a^n b^n c^n : n \in \mathbb{N}\}$

Beispiel

$$\underline{S} \Rightarrow a\underline{S}BC \Rightarrow aa\underline{S}BCBC \Rightarrow aaaB\underline{CB}CBC$$

$$\Rightarrow aaaBBC\underline{CB}C \Rightarrow aaaBB\underline{CB}CC \Rightarrow aa\underline{aB}BBCCC$$

$$\Rightarrow aaa\underline{bB}BCCC \Rightarrow aaab\underline{bB}CCC \Rightarrow aaabb\underline{bC}CC$$

$$\Rightarrow aaabbb\underline{cC}C \Rightarrow aaabbbc\underline{cC} \Rightarrow aaabbbccc$$

$$\boxed{S \rightarrow aSBC, S \rightarrow aBC, CB \rightarrow BC, aB \rightarrow ab, bB \rightarrow bb, bC \rightarrow bc, cC \rightarrow cc}$$

Beweisskizze $a^n b^n c^n \subseteq L(G)$

$S \overset{n-1}{\Rightarrow} a^{n-1} S (BC)^{n-1}$ $(S \to aSBC)$

$\Rightarrow a^n (BC)^n$ $(S \to aBC)$

$\underbrace{\overset{*}{\Rightarrow}}_{\text{Lemma S}} a^n B^n C^n$ $(CB \to BC)$

$\Rightarrow a^n b B^{n-1} C^n$ $(aB \to ab)$

$\overset{n-1}{\Rightarrow} a^n b^n C^n$ $(bB \to bb)$

$\Rightarrow a^n b^n c C^{n-1}$ $(bC \to bc)$

$\overset{n-1}{\Rightarrow} a^n b^n c^n$ $(cC \to cc)$

Einschub: lexikographische Reihenfolge

Betrache $\alpha, \beta \in \Sigma^*$

$\forall \alpha \in \Sigma^* : \varepsilon \leq \alpha$

$a\alpha \leq b\beta$ gdw $a < b$ oder $a = b$ und $\alpha \leq \beta$ $\quad (a, b \in \Sigma; \alpha, \beta \in \Sigma^*)$

Beobachtung: $\leq$ definiert eine totale Ordnung

Beispiel: $\varepsilon < \mathtt{a} < \mathtt{aa} < \mathtt{ab} < \mathtt{b} < \mathtt{ba} < \mathtt{bb}$

- Analog für Tupel
- Wir können über endliche Teilfolgen einer totalen Ordnung Induktionsbeweise machen

Lemma S: $(BC)^n \overset{*}{\Rightarrow} B^n C^n$ mittels $CB \to BC$

Beweis durch Induktion über die lexikographische Reihenfolge von

$$\left\{ w \in \{B,C\}^{2n} : w \text{ enthält gleich viele } B \text{ und } C \right\}$$

IA: α minimal $\longrightarrow \alpha = B^n C^n$

IS: α nicht minimal $\longrightarrow$

$$\begin{aligned} \alpha &= \gamma CB\beta \\ &\Rightarrow \gamma BC\beta && \text{wird kleiner!} \\ &\overset{*}{\Rightarrow} B^n C^n && \text{IV} \end{aligned}$$

■

Beweisskizze $L(G) \subseteq a^n b^n c^n$

Invariante: $\#a = \#(b,B) = \#(c,C)$

Insbesondere $\forall w \in L(G) : \#a = \#b = \#c$.

Es bleibt zu zeigen, dass $L(G) \subseteq a^* b^* c^*$.

Alle a-s kommen vor allen b-s und c-s. $(S \to aSBC, S \to aBC)$

Das erste b folgt dem letzten a. $(aB \to ab)$

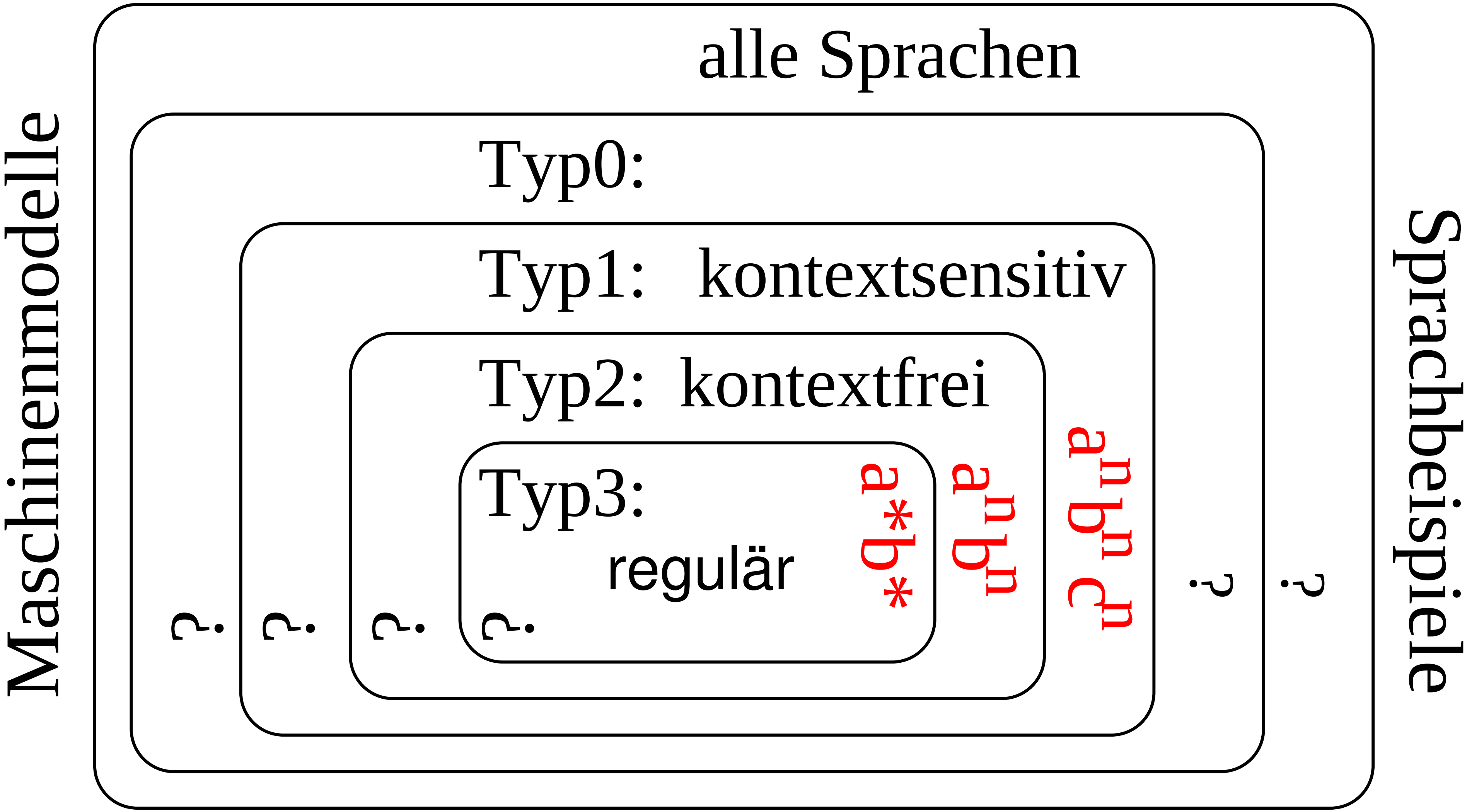
Weitere b-s folgen existierenden b-s. $(bB \to bb)$

Das erste c folgt dem letzen b. $(bC \to bc)$

Weitere c-s folgen existierenden c-s. $(cC \to cc)$

$$\boxed{S \to aSBC, S \to aBC, CB \to BC, aB \to ab, bB \to bb, bC \to bc, cC \to cc}$$

Chomsky Hierarchie



Todo

- ☐ Zuordnung Grammatiktypen $\leftrightarrow$ Maschinenmodelle
- ☐ Zeigen, dass die Sprachbeispiele nicht mit einfacheren Typen erzeugbar sind
- ☐ Sprachbeispiel für Typ 0
- ☐ Algorithmen und Beweistechniken für Standardprobleme

1.1.3 Wortproblem

Das wichtigste Standardproblem für formale Sprachen.

Gegeben: $G = (V, \Sigma, P, S)$, $w \in \Sigma^*$

Frage: $w \in L(G)$?

($\Leftrightarrow$ $S \overset{*}{\Rightarrow} w$?)

Das Wortproblem für Typ 1 Sprachen

Gegeben: $G = (V, \Sigma, P, S)$, $w \in \Sigma^*$

Frage: $w \in L(G)$?

Betrachte den endlichen Graphen $H = (U, E)$ mit

$U = \{x \in (\Sigma \cup V)^* : |x| \leq |w|\}$ und

$E = \{(x, y) : x \Rightarrow_G y\}$.

$w \in L(G)$ gdw. w ist in H von S aus erreichbar.

Korollar:

Das Typ 1-Wortproblem ist in endlicher Zeit algorithmisch lösbar.

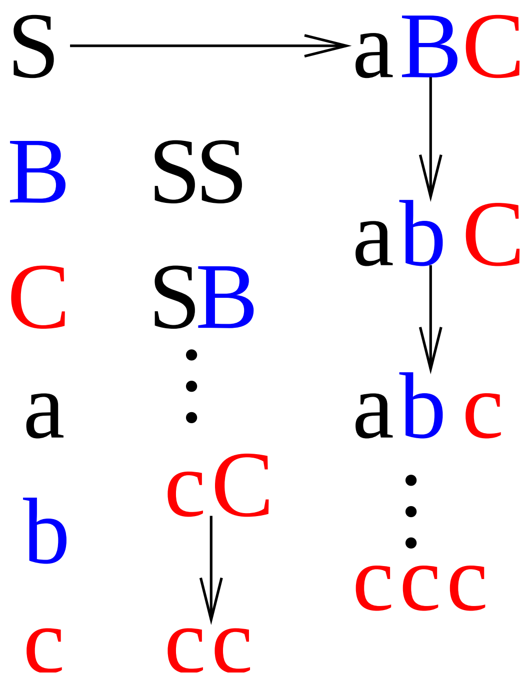
Frage: Warum klappt das nicht bei Typ 0?

Beispiel

$abc \in$

$G = (\{S, B, C\}, \{a, b, c\}, P, S)$?

$$P = \{S \to aSBC,$$
$$S \to aBC,$$
$$CB \to BC,$$
$$aB \to ab,$$
$$bB \to bb,$$
$$bC \to bc,$$
$$cC \to cc\}$$



Laufzeitabschätzung

Gegeben: $G = (V, \Sigma, P, S)$, $w \in \Sigma^*$

Frage: $w \in L(G)$?

Betrachte den endlichen Graphen $H = (U, E)$ mit

$U = \{x \in (\Sigma \cup V)^* : |x| \leq |w|\}$ und

$E = \{(x, y) : x \Rightarrow_G y\}$.

Erreichbarkeit geht in Zeit $\mathcal{O}(|U| + |V|)$.

Dominierend ist Zeit für Aufstellung des Graphen

$(|V| + |\Sigma|)^{|w|}$ Knoten (!)

$\times |w|$ mögliche Ersetzungspunke

$\times |P|$ mögliche Produktionen

$\times \mathcal{O}(|w|)$ Zeit für Prüfung und Ersetzung

1.1.4 Syntaxbäume (nützlich für Typ 2)

Geordneter Baum beschreibt Ableitung (Typ2) $S \overset{*}{\Rightarrow} w$ unabhängig von der Reihenfolge der Ersetzung.

Konstruktion aus Ableitung

$S = x_1 \Rightarrow x_2 \Rightarrow \cdots \Rightarrow x_n = x \in \Sigma^*$:

Wurzel S.

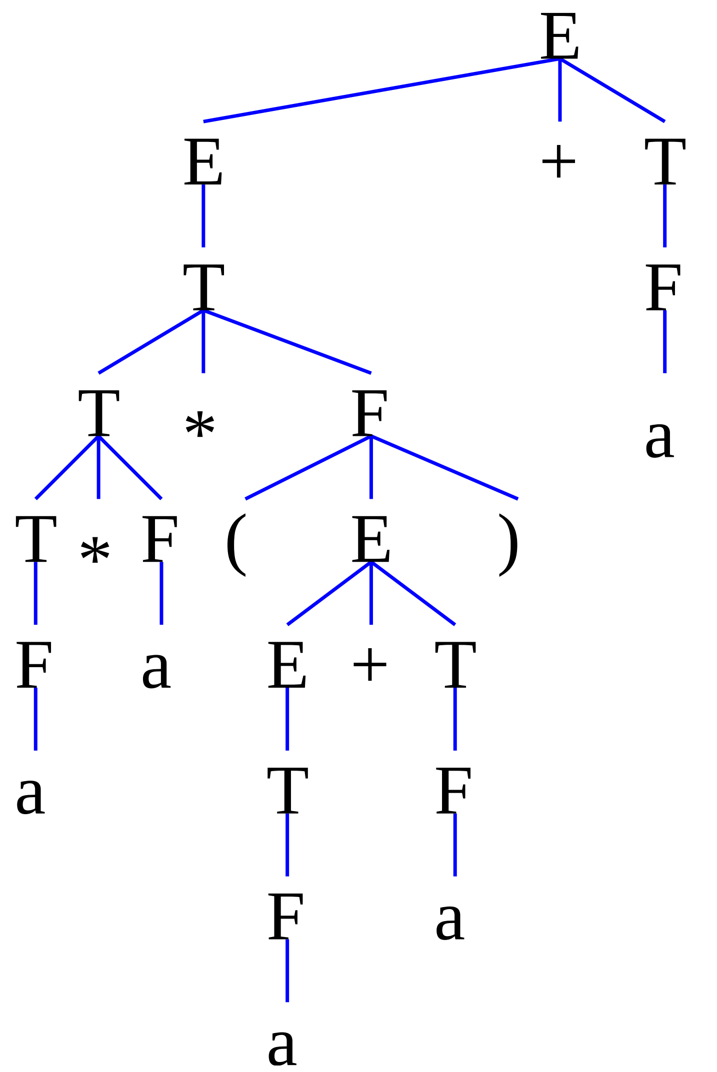
In Schritt i wird $A \to z = z_1, \ldots, z_k$ ersetzt.

$\to$ Knoten für A bekommt Nachfolger $z_1, \ldots, z_k$.

Beobachtung: Blätter sind die Zeichen von x.

Beispiel

$E \Rightarrow$	$E \to E+T$
$\Rightarrow E+T$	$E \to T$
$\Rightarrow T+T$	$T \to T*F$
$\Rightarrow T*F+T$	$T \to T*F$
$\Rightarrow T*F*F+T$	$T \to F$
$\Rightarrow F*F*F+T$	$F \to a$
$\Rightarrow a*F*F+T$	$F \to a$
$\Rightarrow a*a*F+T$	$F \to (E)$
$\Rightarrow a*a*(E)+T$	$E \to E+T$
$\Rightarrow a*a*(E+T)+T$	$E \to T$
$\Rightarrow a*a*(T+T)+T$	$T \to F$
$\Rightarrow a*a*(F+T)+T$	$F \to a$
$\Rightarrow a*a*(a+T)+T$	$T \to F$
$\Rightarrow a*a*(a+F)+T$	$F \to a$
$\Rightarrow a*a*(a+a)+T$	$T \to F$
$\Rightarrow a*a*(a+a)+F$	$F \to a$
$\Rightarrow a*a*(a+a)+a$	



Linksableitung

In jedem Schritt der Ableitung:

ersetze die linkeste Variable

Beispiel: vorherige Seite

1-1 Beziehung Linksableitung$\leftrightarrow$ Syntaxbaum

Beobachtung (Satz) (für Typ 2)

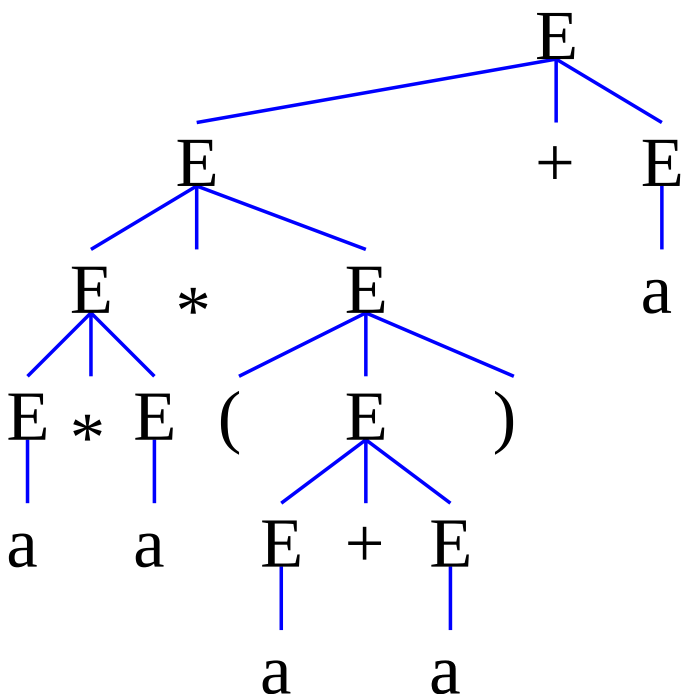
$$x \in L(G) \Leftrightarrow \exists \text{ Ableitung von } x$$
$$\Leftrightarrow \exists \text{ Syntaxbaum mit } x \text{ an den Blättern}$$
$$\Leftrightarrow \exists \text{ Linksableitung von } x$$

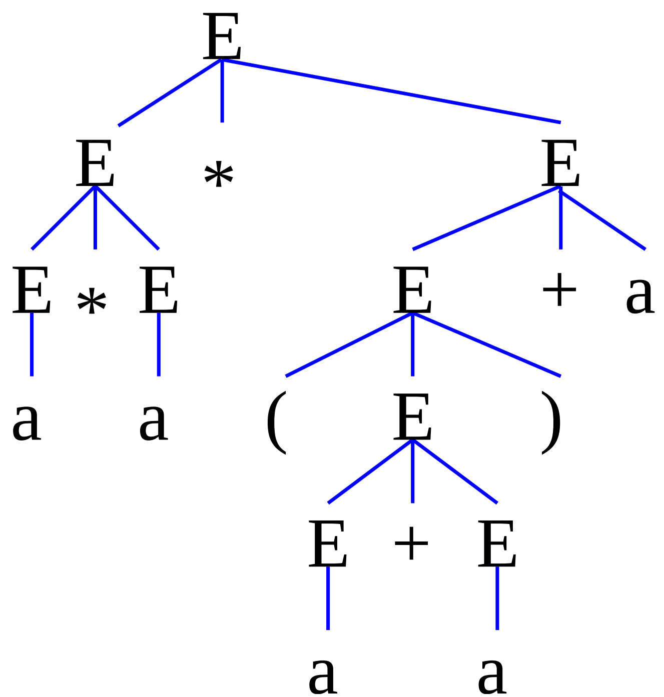
Aufgabe: Definiere Rechtsableitung mit Eigenschaften

Beispiel für nichteindeutigen Syntaxbaum

$G = (\{E\}, \{a, +, *, (,)\}, P, E)$ mit

$$
\begin{aligned}
P = \{E &\to E + E, \\
E &\to E * E, \\
E &\to a, \\
E &\to (E)\}
\end{aligned}
$$





1.1.5 Backus-Naur-Form

nicht hier