



1 Automatentheorie und Formale Sprachen

1.1 Allgemeines



Beispiel: Arithmetische Ausdrücke: EXPR

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

a ist Platzhalter für Konstanten oder Variablen

$$(a - a) * a + a / (a + a) - 1 \in \text{EXPR}$$

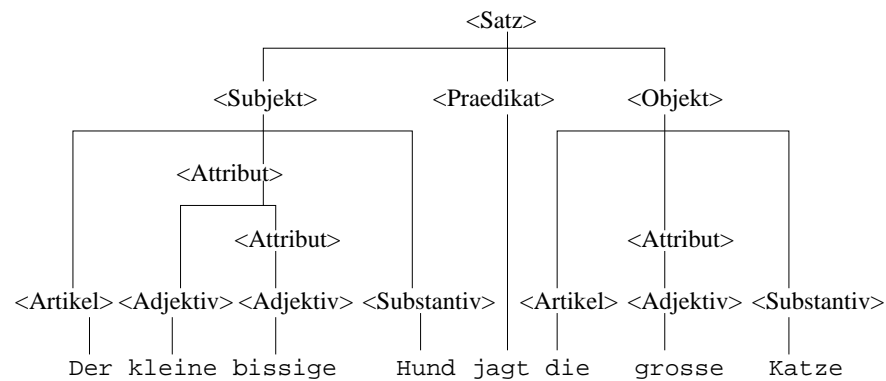
$$(((a))) \in \text{EXPR}$$

$$((a+) - a) \notin \text{EXPR}$$

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

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

**Übergangsrelation \Rightarrow**

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

$u \Rightarrow_G v$ falls

$$u = xyz \in (V \cup \Sigma)^*, \\ v = xy'z \in (V \cup \Sigma)^*, \\ y \rightarrow y' \in P.$$

„ u geht **unmittelbar** über in v .“

Subskript _{G} fehlt, wenn klar ist, welche Grammatik gemeint ist.

**Übergangsrelation $\xRightarrow{*}, \xRightarrow{n}$**

Ableitungen bestimmter Länge:

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

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

Ableitbarkeit:

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

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

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

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

$$L(G) := \{w \in \Sigma^* : S \xRightarrow{*} w\}$$



Ableitung

Folge von Satzformen,

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

heißt Ableitung von w_n falls

$$w_1 \Rightarrow w_2 \Rightarrow \dots \Rightarrow w_n.$$



$E \Rightarrow$	$E \rightarrow E + T$
$\Rightarrow E + T$	$E \rightarrow T$
$\Rightarrow T + T$	$T \rightarrow T * F$
$\Rightarrow T * F + T$	$T \rightarrow T * F$
$\Rightarrow T * F * F + T$	$T \rightarrow F$
$\Rightarrow F * F * F + T$	$F \rightarrow a$
$\Rightarrow a * F * F + T$	$F \rightarrow a$
$\Rightarrow a * a * F + T$	$F \rightarrow (E)$
Beispiel: $\Rightarrow a * a * (E) + T$	$E \rightarrow E + T$
$\Rightarrow a * a * (E + T) + T$	$E \rightarrow T$
$\Rightarrow a * a * (T + T) + T$	$T \rightarrow F$
$\Rightarrow a * a * (F + T) + T$	$F \rightarrow a$
$\Rightarrow a * a * (a + T) + T$	$T \rightarrow F$
$\Rightarrow a * a * (a + F) + T$	$F \rightarrow a$
$\Rightarrow a * a * (a + a) + T$	$T \rightarrow F$
$\Rightarrow a * a * (a + a) + F$	$F \rightarrow a$
$\Rightarrow a * a * (a + a) + a$	



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!

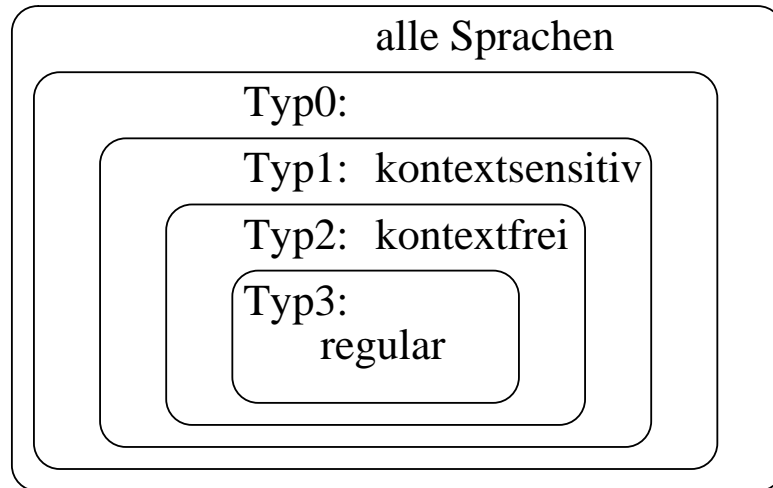
Typ 2, kontextfrei: Typ 1 und $\ell \in V$

$A \rightarrow \varepsilon$ z.T. erlaubt

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



Chomsky Hierarchie



Beispiel: Typ 3

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

$$P = \{A \rightarrow aA, \\ A \rightarrow aB, \\ B \rightarrow bB, \\ B \rightarrow b\}$$

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**

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



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

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

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

$n \rightsquigarrow n + 1 : A \rightarrow aA \xRightarrow{*} aa^n B = a^{n+1} B$

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



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

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

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

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

$m \rightsquigarrow m + 1 : B \rightarrow \underbrace{bB \xRightarrow{*} bb^m}_{IV} = b^{m+1}$

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



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

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

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

Beweis $\supseteq : \forall n \geq 1, m \geq 1 : A \xRightarrow[\text{Lemma 1}]{*} a^n B \xRightarrow[\text{Lemma 2}]{*} a^n b^m$

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

(Def. $L(G)$)

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



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

$$A \xRightarrow[\underbrace{A \rightarrow aA}]{n-1} a^{n-1} A \Rightarrow a^n B \xRightarrow[\underbrace{B \rightarrow bB}]{m-1} a^n b^{m-1} B \Rightarrow a^n b^m$$

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



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

Induktion über die Ableitungslänge ℓ :

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

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

$\ell \rightsquigarrow \ell + 1$: Betrachte Ableitung $S \xRightarrow{*} \alpha' \xRightarrow[\underbrace{C \rightarrow \beta}]{c} \alpha$

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

■



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

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

Die Produktionen erhalten diese

Invariante. ■

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



Beispiel: Typ 2 (Klammerausdrücke)

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

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



Beispiel: Typ 2

$G = (\{S\}, \{a, b\}, \{S \rightarrow ab, S \rightarrow aSb\}, S)$.

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

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

$S \xRightarrow{n-1} a^{n-1} S b^{n-1} \Rightarrow a^n b^n$. ■

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

$S \xRightarrow{*} \alpha \rightarrow \alpha \in \{a^k S b^k : k \geq 0\} \cup \{a^n b^n : n \geq 1\}$

(Invariante) ■



Beispiel: Typ 1

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

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

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



Beispiel

$S \Rightarrow aSBC \Rightarrow aaSBCBC \Rightarrow aaaBCBCBC$
 $\Rightarrow aaaBBCCBC \Rightarrow aaaBBCBCC \Rightarrow aaaBBBCCC$
 $\Rightarrow aaabBBCCC \Rightarrow aaabbBCCC \Rightarrow aaabbbCCC$
 $\Rightarrow aaabbbCC \Rightarrow aaabbbcc \Rightarrow aaabbbccc$

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



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

$S \xRightarrow{n-1} a^{n-1} S(BC)^{n-1}$	$(S \rightarrow aSBC)$
$\Rightarrow a^n (BC)^n$	$(S \rightarrow aBC)$
$\xRightarrow{*} a^n B^n C^n$	$(CB \rightarrow BC)$
Lemma S	
$\Rightarrow a^n b B^{n-1} C^n$	$(aB \rightarrow ab)$
$\xRightarrow{n-1} a^n b^n C^n$	$(bB \rightarrow bb)$
$\Rightarrow a^n b^n c C^{n-1}$	$(bC \rightarrow bc)$
$\xRightarrow{n-1} a^n b^n c^n$	$(cC \rightarrow cc)$



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 $\alpha \leq \beta$ ($a, b \in \Sigma; \alpha, \beta \in \Sigma^*$)

Beobachtung: \leq definiert eine **totale Ordnung**

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

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



Lemma S: $(BC)^n \xRightarrow{*} B^n C^n$ mittels $CB \rightarrow BC$

Beweis durch Induktion über die **lexikographische Reihenfolge** von $\{w \in \{B, C\}^{2n} : w \text{ enthält gleich viele } B \text{ und } C\}$

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

IS: α nicht minimal \longrightarrow

$\alpha = \gamma CB \beta$
 $\Rightarrow \gamma BC \beta$ wird kleiner!
 $\xRightarrow{*} B^n C^n$ IV

■



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 for allen b -s und c -s. $(S \rightarrow aSBC, S \rightarrow aBC)$

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

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

Das erste c folgt dem letzten b . $(bC \rightarrow bc)$

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

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

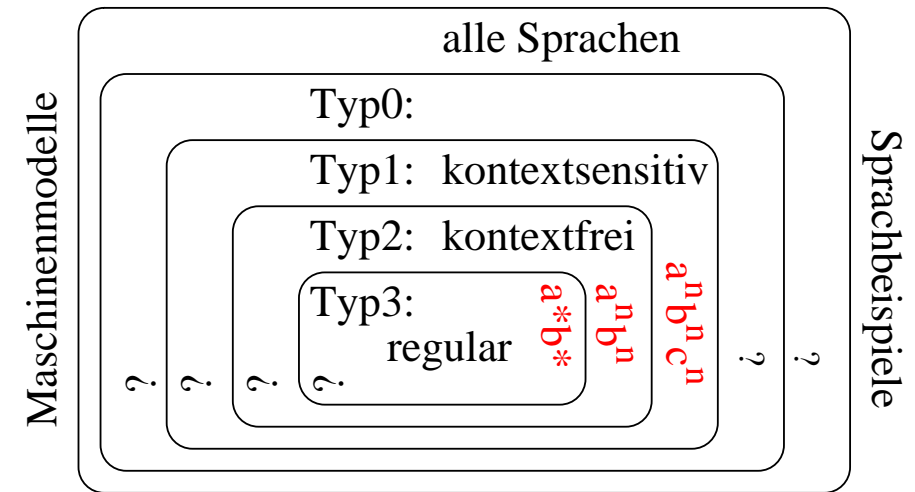


Todo

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



Chomsky Hierarchie



1.1.3 Wortproblem

Das wichtigste Standardproblem für formale Sprachen.

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

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

$(\Leftrightarrow S \xrightarrow{*} w?)$



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?



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 Ersetzungspunkte

$\times |P|$ mögliche Produktionen

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



Beispiel

$abc \in$

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

$P = \{S \rightarrow aSBC,$

$S \rightarrow aBC,$

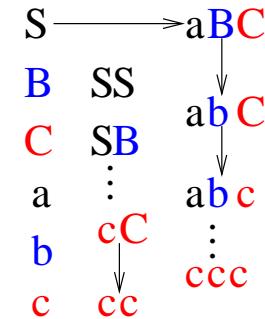
$CB \rightarrow BC,$

$aB \rightarrow ab,$

$bB \rightarrow bb,$

$bC \rightarrow bc,$

$cC \rightarrow cc\}$



1.1.4 Syntaxbäume (nützlich für Typ 2)

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

Kontstruktion aus Ableitung

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

Wurzel S .

In Schritt i wird $A \rightarrow z = z_1, \dots, z_k$ ersetzt.

\rightarrow Knoten für A bekommt Nachfolger z_1, \dots, z_k .

Beobachtung: Blätter sind die Zeichen von x .



1.1.5 Backus-Naur-Form

nicht hier