# 7. Generative Grammatik

## 7.1 Sprache als Untermenge des freien Monoids

### 7.1.1 Definition von Sprache

Eine Sprache ist eine Menge von Wortfolgen.

### 7.1.2 Illustration des freien Monoids über LX = {a,b}

$\varepsilon$
a, b
aa, ab, ba, bb
aaa, aab, aba, abb, baa, bab, bba, bbb
aaaa, aaab, aaba, aabb, abaa, abab, abba, abbb, . . .
. . .

### 7.1.3 Informelle Beschreibung der künstlichen Sprache $a^k b^k$ (with $k \geq 1$)

Wohlgeformte Ausdrücke bestehen aus einer beliebigen Anzahl des Wortes a, gefolgt von der gleichen Anzahl des Wortes b.

### 7.1.4 Wohlgeformte Ausdrücke von $a^k b^k$

a b, a a b b, a a a b b b, a a a a b b b b, etc.,

### 7.1.5 Nichtwohlgeformte Ausdrücke von $a^k b^k$

a, b, b a, b b a a, a b a b, etc.,

### 7.1.6 PS-grammatische Definition von $a^k b^k$

$S \longrightarrow$ a S b
$S \longrightarrow$ a b

Eine formal Grammatik kann als ein Filter aufgefaßt werden, der aus dem freien Monoid über dem Lexikon der Sprache die wohlgeformten Ausdrücke herausfiltert.

### 7.1.7 Elementarformalismen der generativen Grammatik

1. Kategorial- oder C-Grammatik
2. Phrasenstruktur- oder PS-Grammatik
3. Linksassoziative oder LA-Grammatik

### 7.1.8 Algebraische Definition

Die algebraische Definition einer generativen Grammatik (i) zählt die formalen Bausteine des Systems explizit auf und (ii) definiert die strukturellen Bezüge zwischen den Bausteinen mit den Mitteln der Mengentheorie.

### 7.1.9 Abgeleitete Formalismen der PS-Grammatik

Syntactic Structures, Generative Semantics, Standard Theory (ST), Extended Standard Theory (EST), Revised Extended Standard Theory (REST), Government and Binding (GB), Barriers, Generalized Phrase Structure Grammar (GPSG), Lexical Functional Grammar (LFG), Head-driven Phrase Structure Grammar (HPSG)

### 7.1.10 Abgeleitete Formalismen der C-Grammatik

Montague grammar (MG), Functional Unification Grammar (FUG), Categorial Unification Grammar (CUG), Combinatory Categorial Grammar (CCG), Unification-based Categorial Grammar (UCG)

### 7.1.11 Beispiele semi-formaler Grammatiken

Dependenzgrammatik (Tesnière 1959), systemische Grammatik (Halliday 1985), Stratifikationsgrammatik (Lamb 1996)

# 7.2 Methodische Gründe für generative Grammatik

## 7.2.1 Grammatisch wohlgeformter Ausdruck

Die kleinen Hunde haben vorhin geschlafen

## 7.2.2 Grammatisch nicht wohlgeformter Ausdruck

* geschlafen vorhin haben Hunde kleinen die

### 7.2.3 Methodische Konsequenzen generativer Grammatik

- *Empirisch*: explizite Hypothesenbildung
  Die generative Analyse resultiert in formalen Regelsystemen, die explizit festlegen, welche Ausdrücke der Theorie nach grammatikalisch wohlgeformt sind und welche nicht. Diese explizite Hypothesenbildung verschafft Klarheit über die deskriptive Adäquatheit bzw. Inadäquatheit der Analyse und ist Voraussetzung für eine schrittweise Verbesserung der empirischen Beschreibung.
- *Mathematisch*: Bestimmung der formalen Eigenschaften
  Nur streng formalisierte Beschreibungen erlauben Aussagen über ihre mathematische Eigenschaften, z. B. Entscheidbarkeit, Komplexität und generative Kapazität. Die mathematischen Eigenschaften eines Grammatikformalismus bestimmen wiederum dessen Eignung für die empirische Arbeit und programmiertechnische Realisierung.
- *Programmiertechnisch*: Deklarative Spezifikation
  Nur ein formales Regelsystem ist als deklarative Spezifikation eines zugehörigen Parsers geeignet. Diese stellt seine notwendigen Eigenschaften dar, im Unterschied zu seinen akzidentiellen Eigenschaften, die sich aus der Wahl der Programmierumgebung etc. ergeben. Ein Parser ermöglicht die automatische Analyse von Sprachausdrücken, die wiederum für die Verikation von Einzelgrammatiken benötigt wird, die im Rahmen des generativen Grammatikformalismus geschrieben wurden.

# 7.3 Adäquatheit generativer Grammatiken

## 7.3.1 Desiderata einer generativen Grammatik für natürliche Sprachen

Die generative Analyse einer natürlichen Sprache sollte gleichzeitig

- *mathematisch* als eine formale Theorie mit niedriger Komplexität,
- *funktional* als linguistische Komponente der natürlichen Kommunikationsmechanik und
- *methodisch* als effizient implementiertes Computerprogramm definiert werden, das sowohl die Eigenschaften der formalsprachlichen Theorie als auch der empirischen Analyse natürlicher Sprachen in modularer und transparenter Form verkörpert.

# 7.4 Formalismus der C-Grammatik

## 7.4.1 Die historisch erste generative Grammatik

Die Kategorialgrammatik oder C-Grammatik wurde von den polnischen Logikern LEŚNIEWSKI 1929 und AJDUKIEWICZ 1935 mit dem Ziel entwickelt, das Russell-Paradox in der formalen Sprachanalyse zu vermeiden. Auf natürliche Sprachen wurde die C-Grammatik erstmals von BAR-HILLEL 1953 angewandt.

## 7.4.2 Struktur einer logischen Funktion

1. Funktionsname: 2. Argumentbereich $\Longrightarrow$ 3. Wertebereich

$\uparrow$

4. Zuordungsbestimmung

### 7.4.3 Algebraische Definition der C-Grammatik

Eine C-Grammatik ist ein Quintupel $<$ W, C, LX, R, CE $>$.

1. W ist eine endliche Menge von Wortformoberflächen.
2. C ist eine Menge von Kategorien der folgenden Form:
   (a) *Anfangsbedingung*
   u und v $\epsilon$ C,
   (b) *Induktion*
   wenn X und Y $\epsilon$ C, dann sind auch (X/Y) und (X\Y) $\epsilon$ C,
   (c) *Abschlußbedingung*
   Nichts ist in C außer dem, was (a) und (b) entspricht.
3. LX ist eine endliche Menge, wobei LX $\subset$ (W $\times$ C).
4. R ist eine Menge, die die folgenden zwei Regelschemata umfaßt:
   $\alpha_{(\mathrm{Y/X})} \circ \beta_{(\mathrm{Y})} \Rightarrow \alpha\beta_{(\mathrm{X})}$
   $\beta_{(\mathrm{Y})} \circ \alpha_{(\mathrm{Y\backslash X})} \Rightarrow \beta\alpha_{(\mathrm{X})}$
5. CE ist die Menge der Kategorien vollständiger Ausdrücke (*complete expressions*), mit CE $\subseteq$ C.

### 7.4.4 Rekursive Definition der unendlichen Menge C

Ausgehend von zwei Anfangselementen u und v in C gehören laut der Induktionsformel auch (u/v), (v/u), (u\v) und (v\u) zu C. Damit gehören aber auch ((u/v)/v), ((u/v)\v), ((u/v)/u), ((u/v)\u), (u/(u/v)), (v/(u/v)) etc. zu C.

### 7.4.5 Definition von LX als endliche Menge geordneter Paare

Jedes geordnete Paar besteht aus (i) einem Element von W und (ii) einem Element von C. Welche Oberflächen (d.h. Elemente von W) welche Elemente von C als ihre Kategorien nehmen wird in LX über explizite Auflistung der geordneten Paare bestimmt.

### 7.4.6 Definition der Menge von Regelschemata R

Die Regelschemata verwenden Variable $\alpha$ für die Oberfläche des Funktors, die Variable $\beta$ für die Oberfläche des Arguments, und die Variablen X und Y, um deren Kategoriemuster darzustellen.

### 7.4.7 Definition der Menge der vollständigen Ausdrücke CE

Die Menge CE beschreibt die Kategorien derjenigen Ausdrücke, die als *vollständig* betrachtet werden. Je nach spezifischer Grammatik und Sprache kann die Menge CE endlich sein und als Aufzählung spezifiziert werden oder unendlich sein und über Muster mit Variablen charakterisiert werden.

## 7.4.8 Implizite Musterabgleichung C-grammatischer Kompositionen

Funktorwort Argumentwort Resultat der Komposition

a ○ b $\Longrightarrow$ ab
(u/v) (u) (v)

Resultatkategorie

Argumentkategorie

Argumentwort Funktorwort Resultat der Komposition

b ○ a $\Longrightarrow$ ba
(u) (u\v) (v)

Resultatkategorie

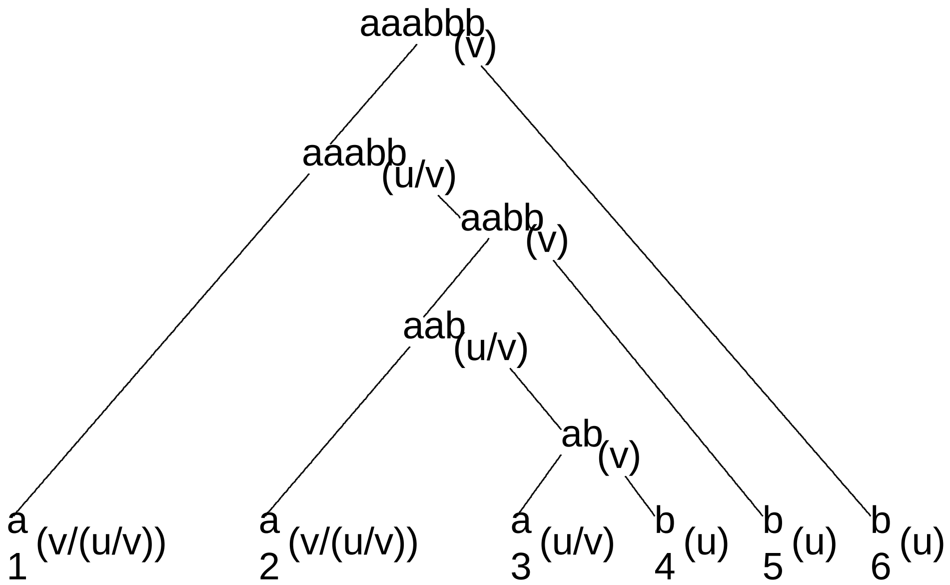
Argumentkategorie

### 7.4.9 Kategorialgrammatische Definition von $a^k b^k$

LX $=_{def}$ {$a_{(u/v)}$, $b_{(u)}$, $a_{(v/(u/v))}$}
CE $=_{def}$ {(v)}

Das Wort a ist in 7.4.9 mit zwei verschiedenen Kategorien definiert, nämlich (u/v) und (v/(u/v)) – aus Gründen, die in der folgenden Ableitung sichtbar werden.

### 7.4.10 Example of $a^k b^k$ derivation, for k = 3



# 7.5 C-Grammatik für natürliche Sprachen

## 7.5.1 C-Grammatik für ein winziges Deutsch-Fragment

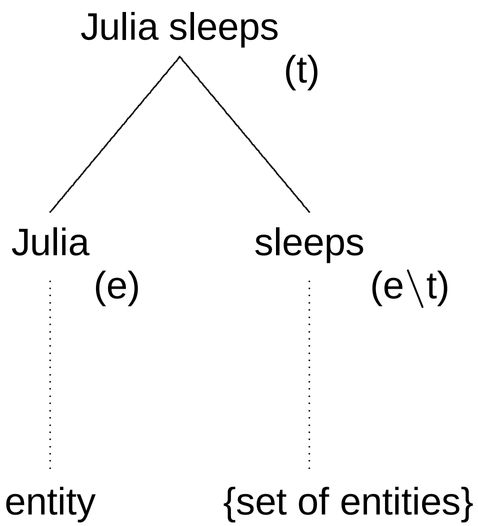
LX $=_{def}$ $\{W_{(e)} \cup W_{(e\backslash t)}\}$, wobei

$W_{(e)}$ = {Julia, Peter, Maria, Fritz, Susi ...}

$W_{(e\backslash t)}$ = {schläft, lacht, singt ...}

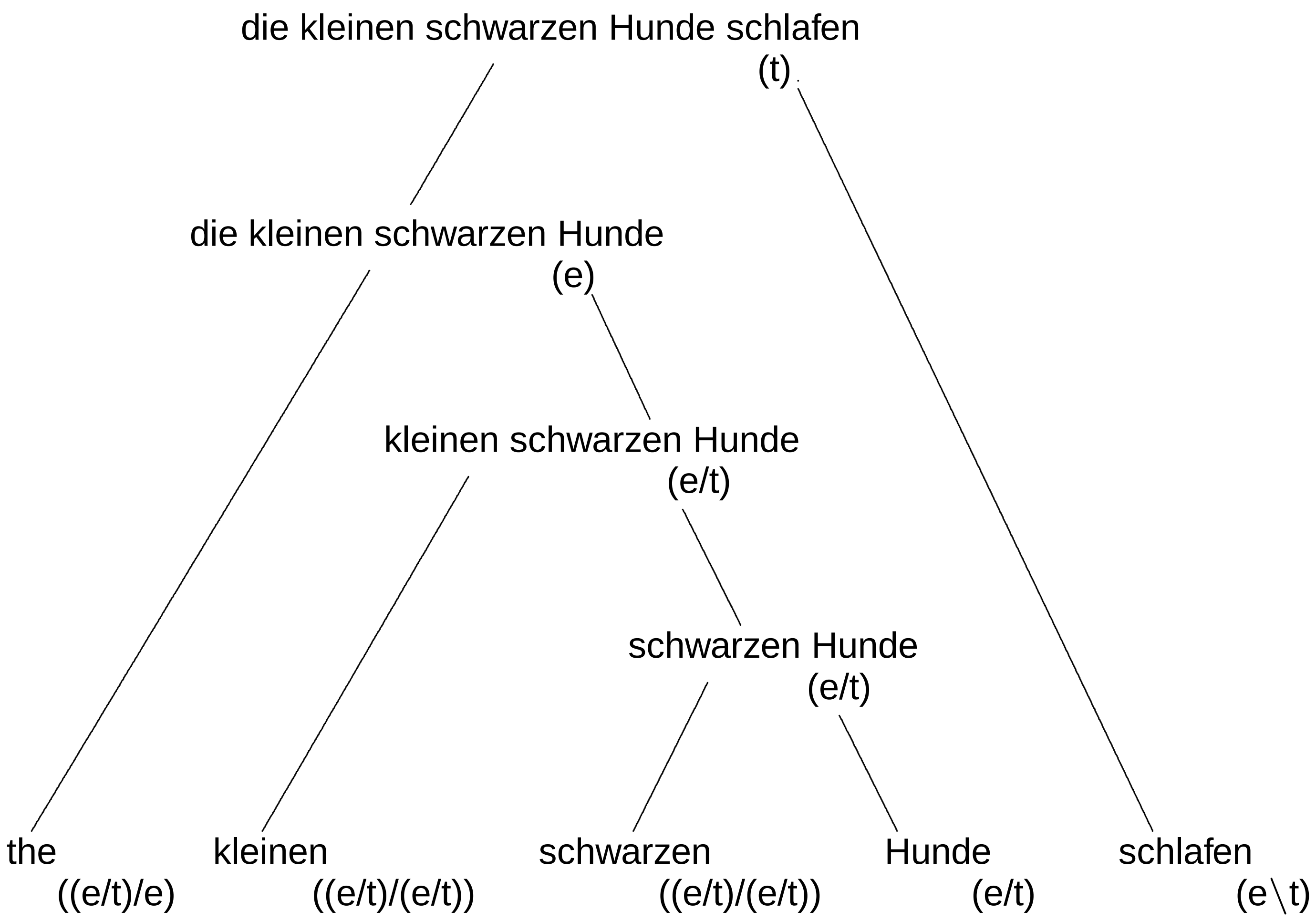
CE $=_{def}$ {(t)}

## 7.5.2 Simultane syntaktische und semantische Analyse




Denotations (in the model $\mathcal{M}$):

## 7.5.3 Kategoriale Analyse eines Satzes



### 7.5.4 C-Grammatik für Satz 7.5.3

$LX =_{def} \{W_{(e)} \cup W_{(e\backslash t)} \cup W_{(e/t)} \cup W_{((e/t)/(e/t))} \cup W_{((e/t)/t)}\}$, wobei

$W_{(e)}$ = {Julia, Peter, Maria, Fritz, Susi ...}

$W_{(e\backslash t)}$ = {schläft, lacht, singt ... schlafen, lachen, singen ...}

$W_{(e/t)}$ = {Hund, Hunde, Katze, Katzen, Tisch, Tische ...}

$W_{((e/t)/(e/t))}$ = {kleine, kleinen, schwarze, schwarzen ...}

$W_{((e/t)/t)}$ = {der, die, den ...}

$CE =_{def} \{(t)\}$

### 7.5.5 Empirische Nachteile der C-Grammatik für natürliche Sprache

- Die C-grammatische Ableitung von Ausdrücken hat den Charakter von *problem solving*.
- Die Behandlung alternativer Wortstellungen und Kongruenzbeschränkungen erfordert einen extrem hohen Grad lexikalischer Ambiguitäten.