

Sprachverarbeitung II / 3 FS 2017

# Grammatik für natürliche Sprache

Buch: Kapitel 6.4 und 6.5

Beat Pfister



# Sprachverarbeitung II / 3

Vorlesung: **Sprachsynthese** (Teil II.3)

- Formale Grammatiken für natürliche Sprachen
- DCG-Formalismus (Definite Clause Grammar)

Übung: Grammatiken für mathematische Ausdrücke

# Formale Grammatiken für natürliche Sprachen

Beschreibung der Struktur von natürlichsprachlichen Sätzen und Wörtern

- **Satzgrammatik** (beschreibt Erzeugung von Sätzen)

Terminalsymbole = Wörter (“der”, “häuser”, “baumes”, ...)

Nicht-Terminalsymbole = *syntaktische Konstituenten*

- **Wortgrammatik** (beschreibt Erzeugung von Wörtern)

Terminalsymbole = Morphe (“häus”, “baum”, “e”, “er”, “ern”, “es”, ...)

Nicht-Terminalsymbole = *morphologische Konstituenten*

# Formale Grammatik für Sätze

Frage: Welche Sprachklasse liegt vor? (nach der Chomsky-Hierarchie)

Feststellung: Häufiges Phänomen: Einbettung / Verschachtelung

“Haben Sie den Mann, der gestern zum Haus,  
das auf dem Hügel steht, gerannt ist, gekannt?”

—→ ist analog zur Klammerung in mathematischen Ausdrücken

Folge: Grammatik für Sätze muss mindestens vom Typ 2 sein

Typ-1-Phänomene sehr selten, z.B. Schweizerdeutsch >>>

# Aufbauen einer Satzgrammatik

Einfachste Sätze: “Vögel fliegen”                      Subjekt + Prädikat

“der Hund bellt”

—→ bestehen aus einer Nominalgruppe und einer Verbalgruppe



# Beispiel einer einfachen Satzgrammatik

**Grammatik:** Produktionsregeln der formalen Grammatik

- $S \rightarrow NG\ VG$  (Satz := Nominalgruppe + Verbalgruppe)  
 $NG \rightarrow N$  (Nominalgruppe := Nomen; einfachster Fall)  
 $NG \rightarrow Art\ N$  (Nominalgruppe := Artikel + Nomen)  
 $VG \rightarrow V$  (Verbalgruppe := Verb; einfachster Fall)  
 $VG \rightarrow V\ NG$  (Verbalgruppe := Verb + Nominalgruppe)

**Lexikon:** Produktionsregeln für Terminalsymbole (Vollformen)

- $Art \rightarrow der \mid die \mid das \mid dem \mid ein \mid eines \mid \dots$   
 $N \rightarrow frau \mid mann \mid baum \mid hauses \mid häuser \mid \dots$   
 $V \rightarrow sehen \mid siehst \mid steht \mid blüht \mid \dots$

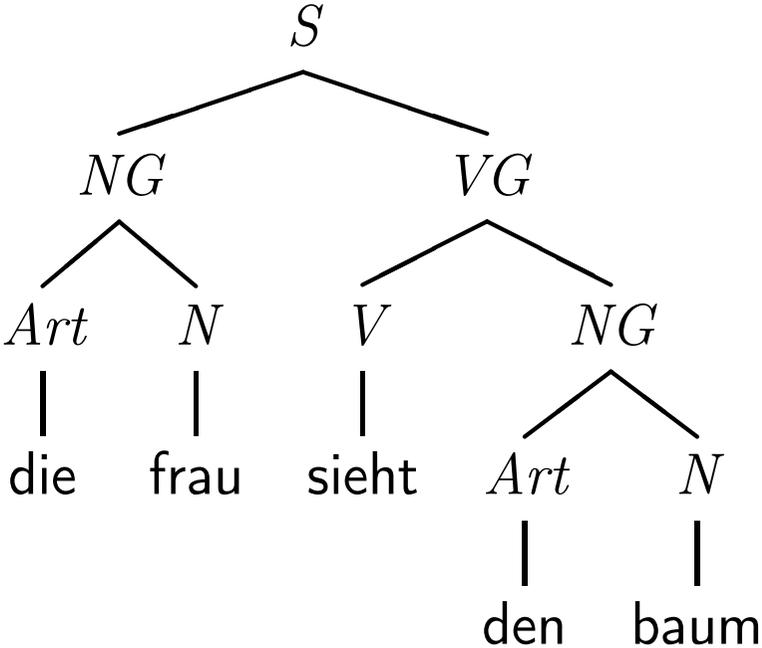
Gegeben: Satz (Folge von Wörtern)  
z.B. “die Frau sieht den Baum”

Gesucht: Ableitung anhand der Grammatik → Parsing

# Beispiel Ableitungsbaum (Syntaxbaum)

Satz: "die Frau sieht den Baum"

- $S \rightarrow NG\ VG$
- $NG \rightarrow N$
- $NG \rightarrow Art\ N$
- $VG \rightarrow V$
- $VG \rightarrow V\ NG$



## Durch Beispielgrammatik beschriebene korrekte Sätze

“die Frau sieht den Baum”

“der Baum blüht”

“Männer sehen Häuser”

## Durch Beispielgrammatik beschriebene unkorrekte Sätze

\* “das Mann sehen Baum” → **Einschränkung notwendig**

## Durch Beispielgrammatik nicht beschriebene Sätze

\* “Bäume Haus die gehen” (Ablehnung erwünscht, da Satz unkorrekt)

“der Mann ging gestern zum Haus” → **Erweiterung notwendig**

# Problem

Auch Sätze möglich wie: \* “das Hund bellt”, \* “des Kind spielen”

In natürlichen Sprachen existieren Restriktionsphänomene, z. B.

- Übereinstimmung in Fall / Zahl / Geschlecht (Kasus / Numerus / Genus) innerhalb von Nominalgruppen “der grosse Hund”
- Übereinstimmung von Numerus zwischen Subjekt und Verb  
“der grosse Hund bellt”, \* “der grosse Hund bellen”
- Wortformen abhängig von syntaktischer / lexikalischer Konstruktion  
“der grosse Hund”, “ein grosser Hund”

## Lösung mit kontextfreier Grammatik

*S* → *NG-nom-sg-m VG-sg*  
*S* → *NG-nom-pl-m VG-pl*  
*S* → *NG-nom-sg-f VG-sg*  
⋮  
*S* → *NG-nom-pl-n VG-pl*  
*NG-nom-sg-m* → *N-nom-sg-m*  
*NG-gen-sg-m* → *N-gen-sg-m*  
*NG-dat-sg-m* → *N-dat-sg-m*  
⋮  
*NG-akk-pl-f* → *N-akk-pl-f*  
*NG-nom-sg-m* → *Art-nom-sg-m N-nom-sg-m*  
*NG-gen-sg-m* → *Art-gen-sg-m N-gen-sg-m*  
⋮  
*NG-akk-pl-f* → *Art-akk-pl-f N-akk-pl-f*  
*VG-sg* → *V-sg*  
*VG-pl* → *V-pl*

## Lösung mit kontextfreier Grammatik

*S* → *NG-nom-sg-m VG-sg*  
*S* → *NG-nom-pl-m VG-pl*  
*S* → *NG-nom-sg-f VG-sg*  
⋮  
*S* → *NG-nom-pl-n VG-pl*  
*NG-nom-sg-m* → *N-nom-sg-m*  
*NG-gen-sg-m* → *N-gen-sg-m*  
*NG-dat-sg-m* → *N-dat-sg-m*  
⋮  
*NG-akk-pl-f* → *N-akk-pl-f*  
*NG-nom-sg-m* → *Art-nom-sg-m N-nom-sg-m*  
*NG-gen-sg-m* → *Art-gen-sg-m N-gen-sg-m*  
⋮  
*NG-akk-pl-f* → *Art-akk-pl-f N-akk-pl-f*  
*VG-sg* → *V-sg*  
*VG-pl* → *V-pl*

*Art-nom-sg-m* → der | ein  
*Art-gen-sg-m* → des | eines  
*Art-dat-sg-m* → dem | einem  
⋮  
*Art-nom-pl-m* → die  
*Art-nom-pl-f* → die  
*Art-nom-pl-n* → die  
⋮  
*Art-akk-pl-f* → die  
*N-nom-sg-m* → hund | ...  
*N-gen-sg-m* → hundes | ...  
*N-dat-sg-m* → hund | ...  
⋮  
*N-akk-pl-m* → hunde | ...  
*N-nom-sg-n* → kind | ...  
*N-nom-pl-n* → kinder | ...  
⋮  
*V-sg* → bellt | spielt | ...  
*V-pl* → bellen | spielen | ...  
⋮

>>>

# Bessere Lösung: Definite Clause Grammars

Erweiterung des Grammatik-Formalismus, um syntaktische Übereinstimmungsphänomene effizienter beschreiben zu können

DCG-Regeln:  $A(a_1, a_2, \dots) \rightarrow B(b_1, b_2, \dots) C(c_1, c_2, \dots) \dots$

- Kontextfreies “Skelett”: Regeln mit Kopf aus 1 NT
- *Attribute* zu NT zwecks Spezifikation von Einschränkungen  
mittels *Term-Unifikation*
- Ableitung als Baum darstellbar (Syntaxbaum)

>>>  
>>>

## Beispiel einer DCG für einfache, deutsche Sätze

$R_1: S \rightarrow NG(nom, ?num, ?gen) VG(?num)$   
 $R_2: NG(?kas, ?num, ?gen) \rightarrow N(?kas, ?num, ?gen)$   
 $R_3: NG(?kas, ?num, ?gen) \rightarrow Art(?kas, ?num, ?gen) N(?kas, ?num, ?gen)$   
 $R_4: VG(?num) \rightarrow V(?num)$

$Art(nom, sg, m)$	$\rightarrow$	der	$N(nom, sg, m)$	$\rightarrow$	hund
$Art(nom, sg, f)$	$\rightarrow$	die	$N(nom, pl, m)$	$\rightarrow$	hunde
$Art(nom, sg, n)$	$\rightarrow$	das	$N(nom, sg, n)$	$\rightarrow$	haus
$Art(nom, pl, n)$	$\rightarrow$	die	$N(nom, sg, n)$	$\rightarrow$	kind
$Art(nom, pl, m)$	$\rightarrow$	die	$N(nom, pl, n)$	$\rightarrow$	kinder
$Art(gen, sg, m)$	$\rightarrow$	des	$V(sg)$	$\rightarrow$	bellt
$Art(nom, sg, m)$	$\rightarrow$	ein	$V(sg)$	$\rightarrow$	spielt
			$V(pl)$	$\rightarrow$	bellen

# DCG-Ableitung

- Satz  $X$  gehört zur Sprache, falls eine Ableitung existiert:  $S \Rightarrow^* X$
- Ableitung analog zu Typ-2-Grammatiken, aber mit Unifikation
- Für jede Regelanwendung neuer Satz Variablen

# DCG-Ableitung

- Satz  $X$  gehört zur Sprache, falls eine Ableitung existiert:  $S \Rightarrow^* X$
- Ableitung analog zu Typ-2-Grammatiken, aber mit Unifikation
- Für jede Regelanwendung neuer Satz Variablen

**Beispiel:** Ableitung von “das Kind spielt”

# DCG-Ableitung

- Satz  $X$  gehört zur Sprache, falls eine Ableitung existiert:  $S \Rightarrow^* X$
- Ableitung analog zu Typ-2-Grammatiken, aber mit Unifikation
- Für jede Regelanwendung neuer Satz Variablen

**Beispiel:** Ableitung von “das Kind spielt”

**1. Schritt:** Regel  $R_1$

$S$

⊙

$S \rightarrow NG(nom, ?num_1, ?gen_1) VG(?num_1)$

$\Rightarrow NG(nom, ?num_1, ?gen_1) VG(?num_1)$

# DCG-Ableitung

- Satz  $X$  gehört zur Sprache, falls eine Ableitung existiert:  $S \Rightarrow^* X$
- Ableitung analog zu Typ-2-Grammatiken, aber mit Unifikation
- Für jede Regelanwendung neuer Satz Variablen

**Beispiel:** Ableitung von “das Kind spielt”

**2. Schritt:** Regel  $R_3$

$NG(nom, ?num_1, ?gen_1) VG(?num_1)$

⊙

$NG(?kas_2, ?num_2, ?gen_2) \rightarrow Art(?kas_2, ?num_2, ?gen_2) N(?kas_2, ?num_2, ?gen_2)$

$\Rightarrow Art(nom, ?num_1, ?gen_1) N(nom, ?num_1, ?gen_1) VG(?num_1)$

# DCG-Ableitung

- Satz  $X$  gehört zur Sprache, falls eine Ableitung existiert:  $S \Rightarrow^* X$
- Ableitung analog zu Typ-2-Grammatiken, aber mit Unifikation
- Für jede Regelanwendung neuer Satz Variablen

**Beispiel:** Ableitung von “das Kind spielt”

**3. Schritt:** Regel  $R_4$

$Art(nom, ?num_1, ?gen_1) N(nom, ?num_1, ?gen_1) VG(?num_1)$

⊙

$VG(?num_3) \rightarrow V(?num_3)$

$\Rightarrow Art(nom, ?num_1, ?gen_1) N(nom, ?num_1, ?gen_1) V(?num_1)$

# DCG-Ableitung

- Satz  $X$  gehört zur Sprache, falls eine Ableitung existiert:  $S \Rightarrow^* X$
- Ableitung analog zu Typ-2-Grammatiken, aber mit Unifikation
- Für jede Regelanwendung neuer Satz Variablen

**Beispiel:** Ableitung von “das Kind spielt”

**4. Schritt:** (Lexikon)

$Art(nom, ?num_1, ?gen_1) N(nom, ?num_1, ?gen_1) V(?num_1)$

⊙

$Art(nom, sg, n) \rightarrow$  das

$\Rightarrow$  das  $N(nom, sg, n) V(sg)$

# DCG-Ableitung

- Satz  $X$  gehört zur Sprache, falls eine Ableitung existiert:  $S \Rightarrow^* X$
- Ableitung analog zu Typ-2-Grammatiken, aber mit Unifikation
- Für jede Regelanwendung neuer Satz Variablen

**Beispiel:** Ableitung von “das Kind spielt”

## 5. Schritt:

das  $N(nom,sg,n)$   $V(sg)$

⊙

$N(nom,sg,n) \rightarrow \text{kind}$

$\Rightarrow$  das kind  $V(sg)$

# DCG-Ableitung

- Satz  $X$  gehört zur Sprache, falls eine Ableitung existiert:  $S \Rightarrow^* X$
- Ableitung analog zu Typ-2-Grammatiken, aber mit Unifikation
- Für jede Regelanwendung neuer Satz Variablen

**Beispiel:** Ableitung von “das Kind spielt”

## 6. Schritt:

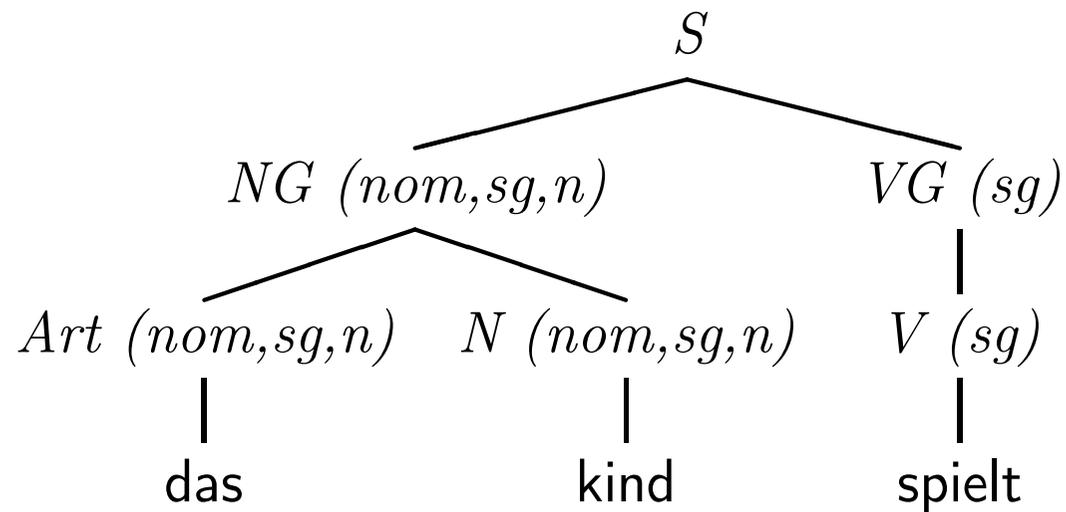
das kind  $V(sg)$

⊙

$V(sg) \rightarrow$  spielt

$\Rightarrow$  das kind spielt

## Darstellung als DCG-Ableitungsbaum



# DCG und Chart-Parsing

- Weil DCG kontextfreies Skelett  $\longrightarrow$  Parsing anwendbar  
(mit Berücksichtigung der Unifikation)
- Einfache Parsing-Algorithmen haben Aufwand  $\mathcal{O}(c^n)$  für Satz der Länge  $n$
- Effizienteres Verfahren: Chart-Parsing
  - Prinzip: Notieren von gefundenen Konstituenten und Hypothesen;  
Teilarbeiten nicht wiederholen
  - Aufwand  $\mathcal{O}(n^3)$
  - Verschiedene Strategien möglich (Tiefen-/Breitensuche,  
Top-down/Bottom-up)
  - Geeignet für kontextfreie Grammatiken und erweiterte Grammatiken  
mit kontextfreiem Skelett (z.B. DCG)

# Chart-Parsing

Datenstrukturen:	<b>Chart</b>	gefundene (Teil-)Lösungen	>>>
	<b>Agenda</b>	noch einzusetzende Kanten	>>>

Operationen:

<b>Fundamentalregel</b>	$(n_i, n_j, A \rightarrow \varphi_1 \bullet B \varphi_2)$	(aktive Kante)
	$(n_j, n_k, B)$	(passive Kante)
	$\Rightarrow (n_i, n_k, A \rightarrow \varphi_1 B \bullet \varphi_2)$	wenn $ \varphi_2  > 0$
	$(n_i, n_k, A)$	wenn $ \varphi_2  = 0$

<b>Top-down-Prädiktion</b>	$(n_i, n_j, A \rightarrow \varphi_1 \bullet B \varphi_2)$
	$\Rightarrow (n_j, n_j, B \rightarrow \bullet \psi)$

**Bottom-up-Prädiktion** siehe Buch Seite 164

# Grammatik für einfache deutsche Sätze

- (1)  $S \rightarrow NG(nom, ?num, ?gen) V(?num)$   
(2)  $S \rightarrow NG(nom, ?num, ?gen) MV(?num) VInf$   
(3)  $NG(?kas, ?num, ?gen) \rightarrow Art(?kas, ?num, ?gen) N(?kas, ?num, ?gen)$

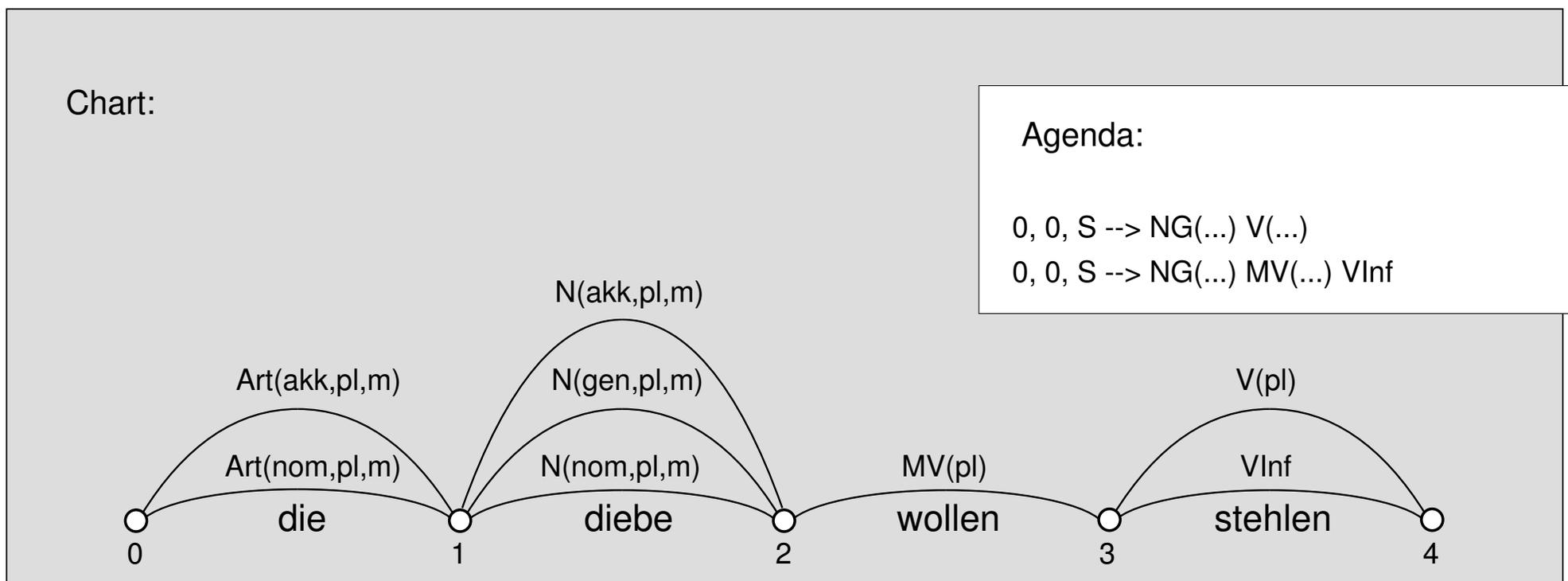
- |                        |               |        |                        |               |         |
|------------------------|---------------|--------|------------------------|---------------|---------|
| (4) $N(nom, sg, m)$    | $\rightarrow$ | dieb   | (15) $Art(akk, sg, m)$ | $\rightarrow$ | den     |
| (5) $N(gen, sg, m)$    | $\rightarrow$ | diebes | (16) $Art(nom, pl, m)$ | $\rightarrow$ | die     |
| (6) $N(dat, sg, m)$    | $\rightarrow$ | dieb   | (17) $Art(gen, pl, m)$ | $\rightarrow$ | der     |
| (7) $N(akk, sg, m)$    | $\rightarrow$ | dieb   | (18) $Art(dat, pl, m)$ | $\rightarrow$ | der     |
| (8) $N(nom, pl, m)$    | $\rightarrow$ | diebe  | (19) $Art(akk, pl, m)$ | $\rightarrow$ | die     |
| (9) $N(gen, pl, m)$    | $\rightarrow$ | diebe  | (20) $VInf$            | $\rightarrow$ | stehlen |
| (10) $N(dat, pl, m)$   | $\rightarrow$ | dieben | (21) $V(sg)$           | $\rightarrow$ | stiehlt |
| (11) $N(akk, pl, m)$   | $\rightarrow$ | diebe  | (22) $V(pl)$           | $\rightarrow$ | stehlen |
| (12) $Art(nom, sg, m)$ | $\rightarrow$ | der    | (23) $MV(sg)$          | $\rightarrow$ | will    |
| (13) $Art(gen, sg, m)$ | $\rightarrow$ | des    | (24) $MV(pl)$          | $\rightarrow$ | wollen  |
| (14) $Art(dat, sg, m)$ | $\rightarrow$ | dem    |                        |               |         |

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

Initialisierung von Chart und Agenda (nach sep. Wortanalyse)



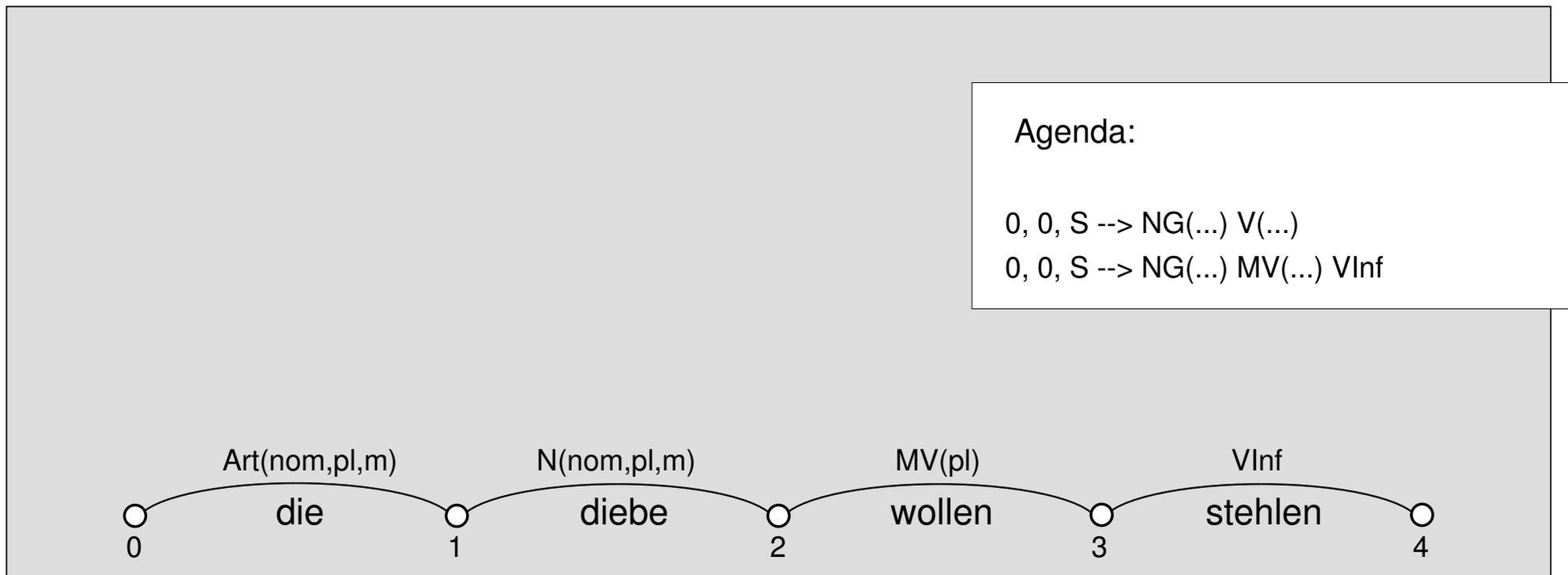
>>>

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

Initialisierung von Chart und Agenda (ohne unbenötigte Wortkanten)



Agenda:  
  
0, 0, S  $\rightarrow$  NG(...) V(...)  
0, 0, S  $\rightarrow$  NG(...) MV(...) VInf

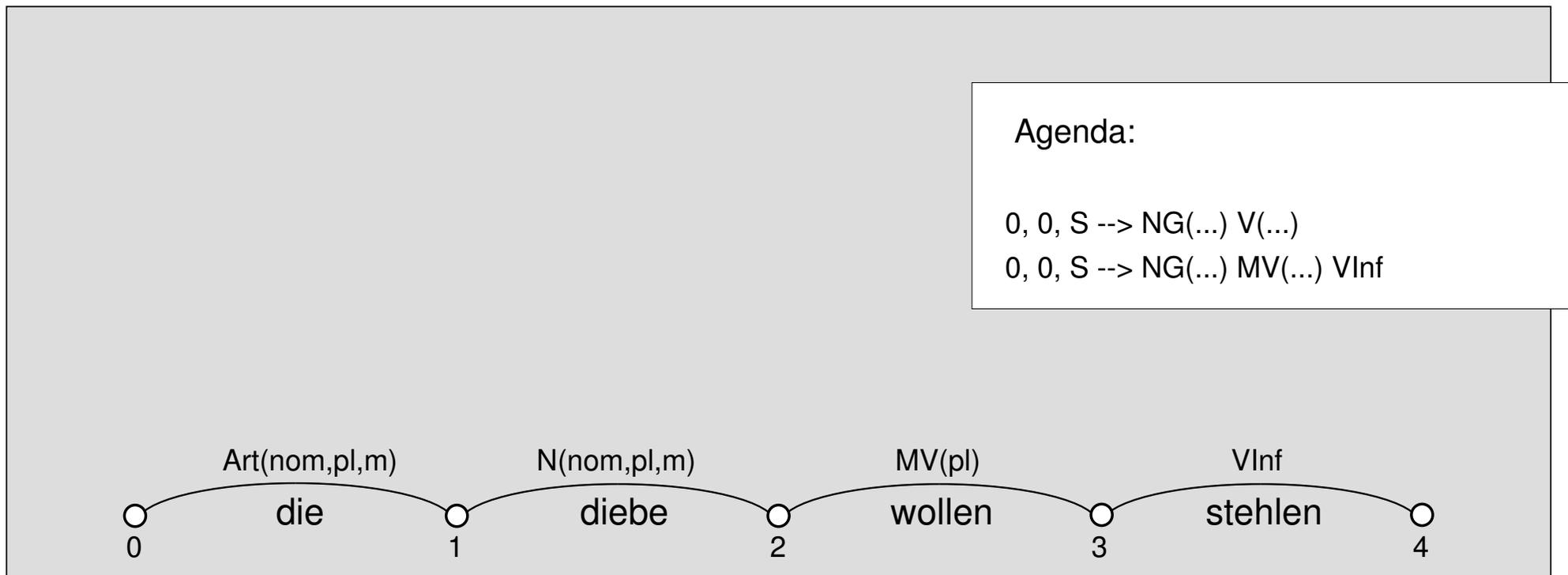
>>>

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

Fundamentalregel und Top-down-Prädiktion: **nicht anwendbar!**



Agenda:  
0, 0, S --> NG(...) V(...)  
0, 0, S --> NG(...) MV(...) VInf

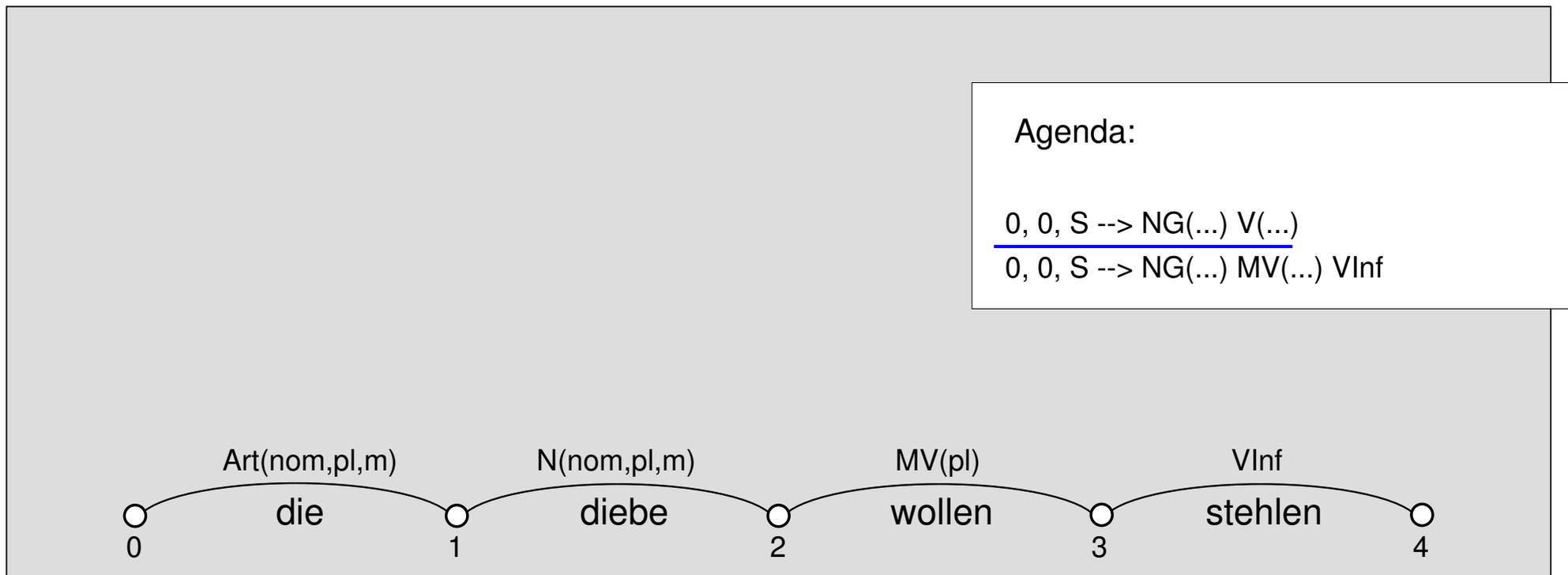
>>>

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

oberste Kante aus Agenda entfernen

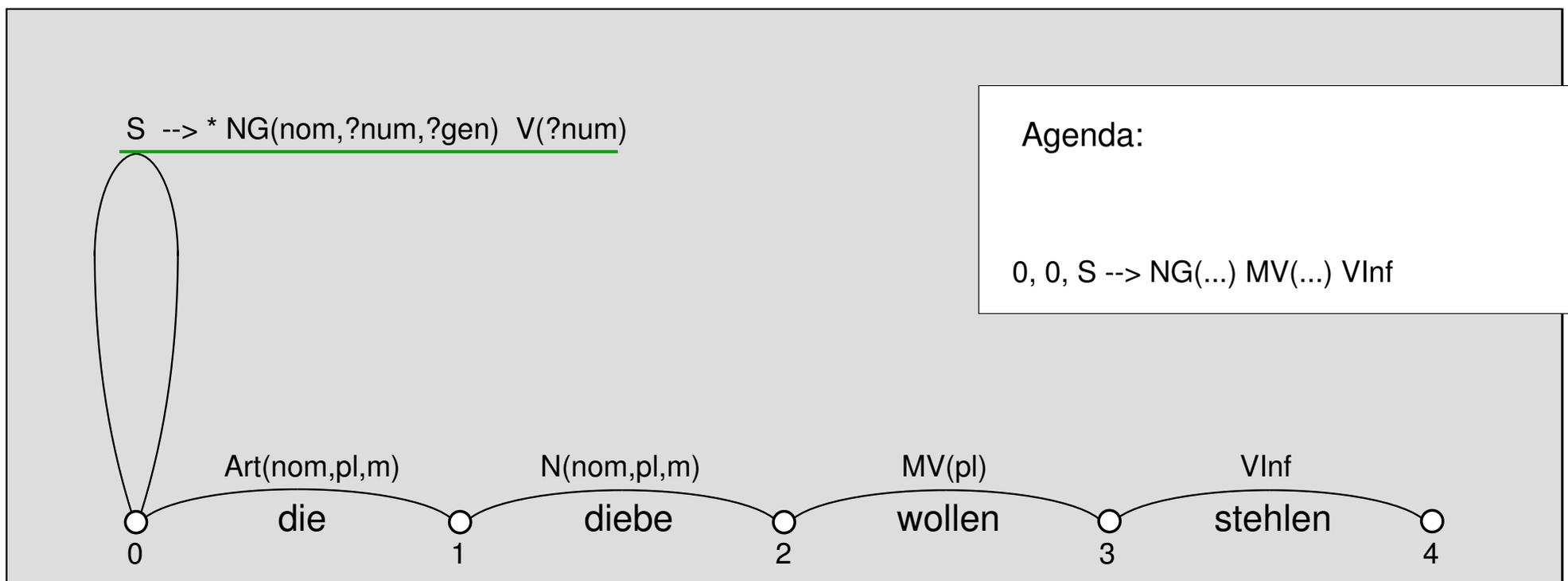


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

neue aktive S-Kante in Chart

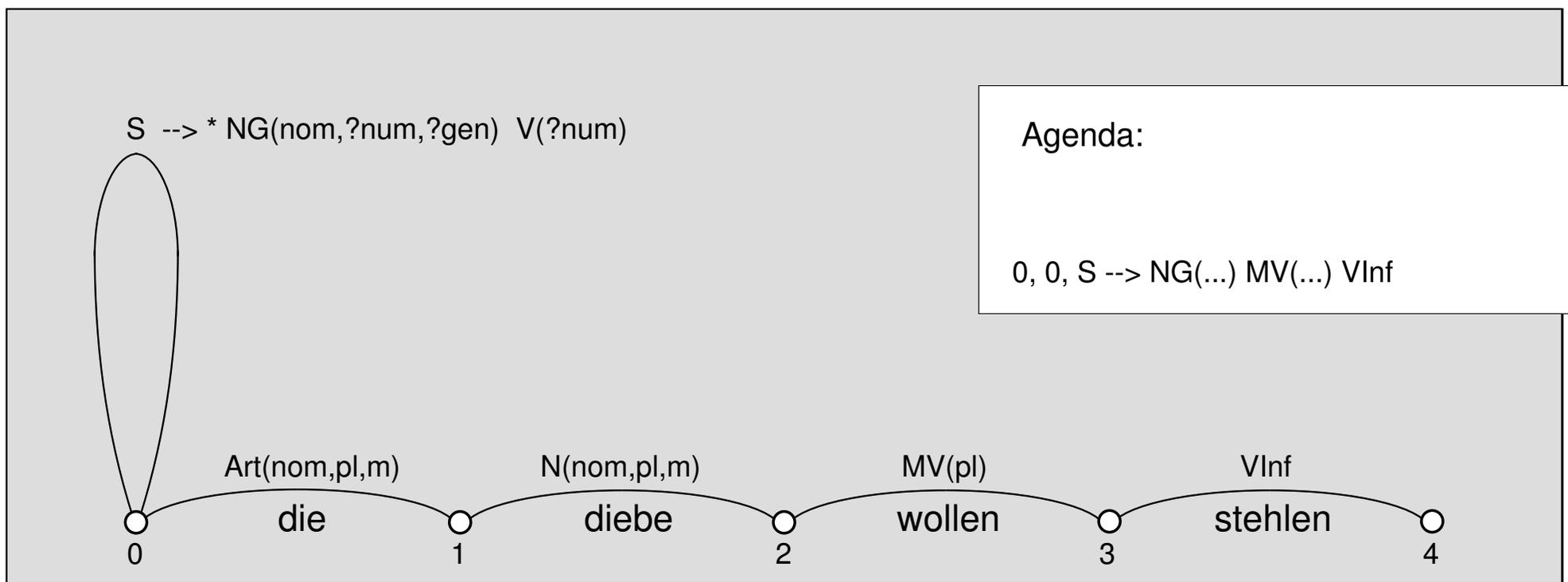


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

Fundamentalregel: **nicht anwendbar!**

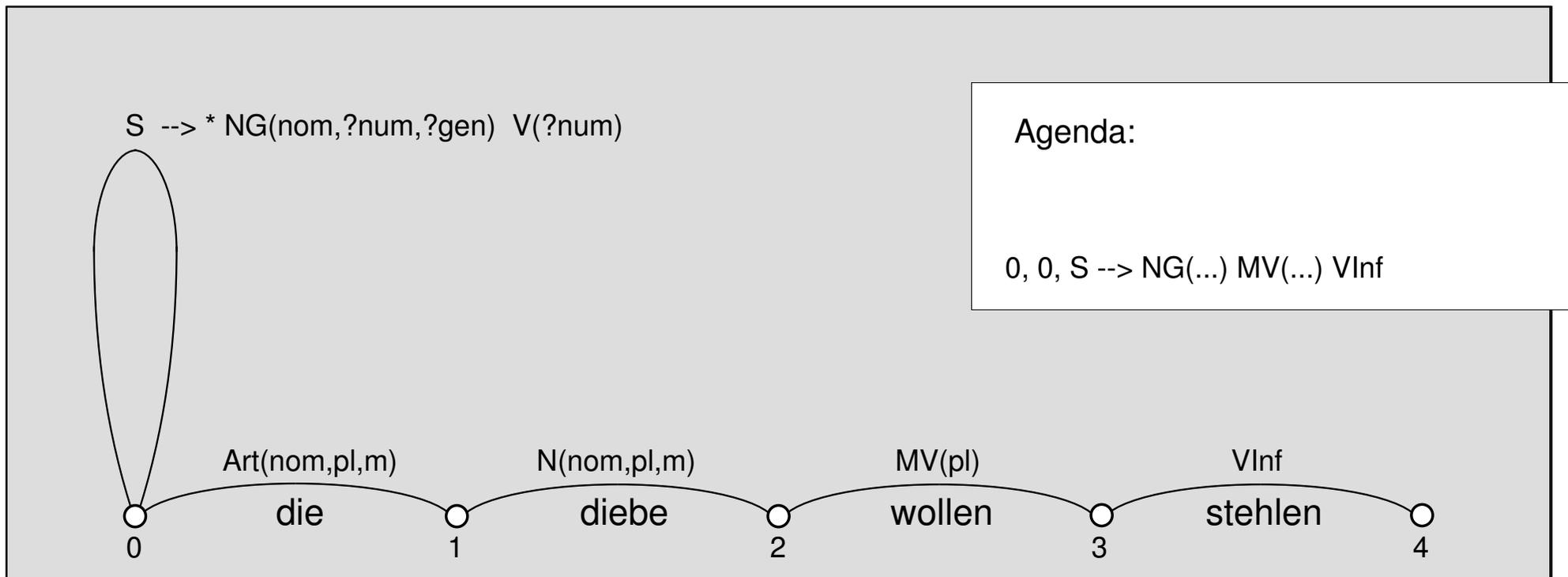


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

## Top-Down-Prädiktion

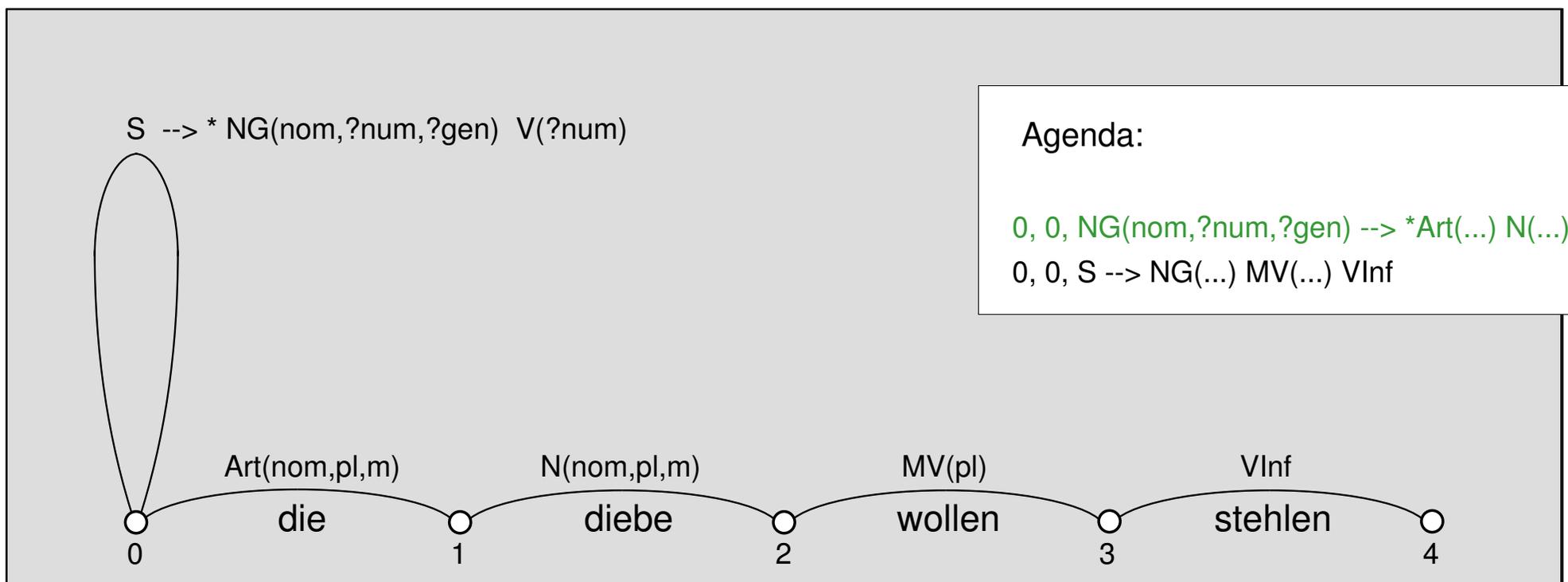


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

neue NG-Kante in Agenda



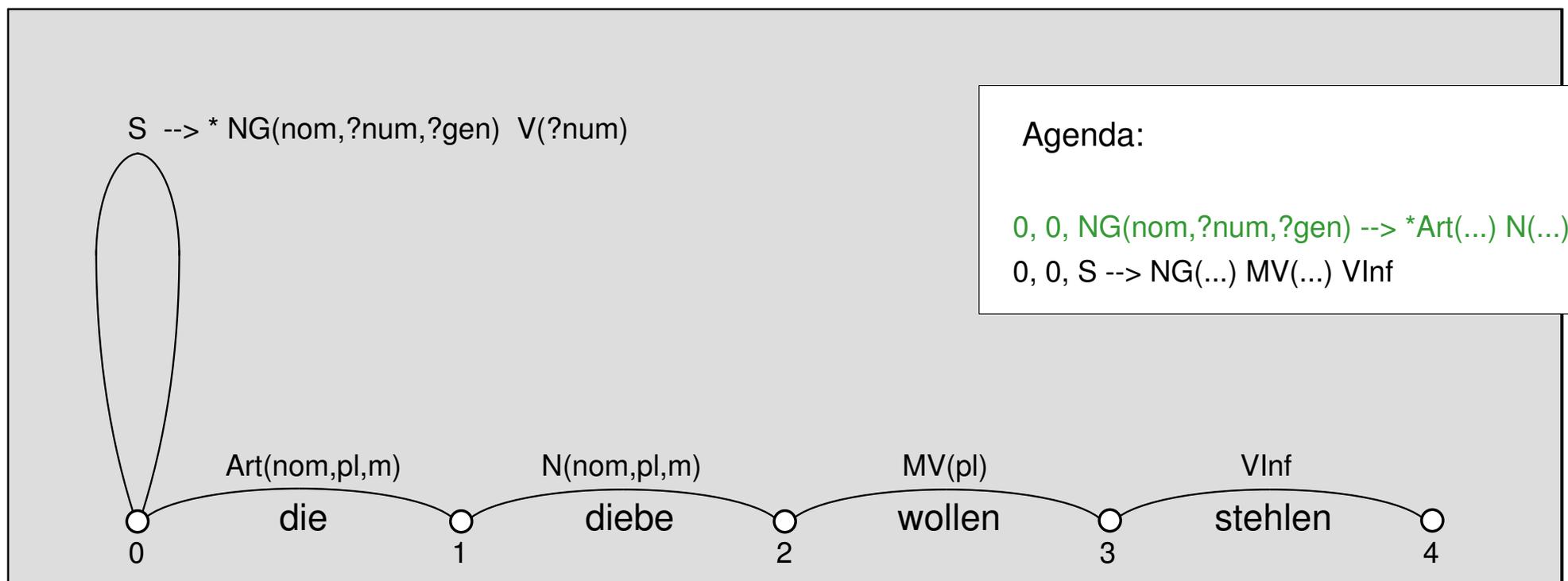
>>>

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

oberste Kante aus Agenda entfernen

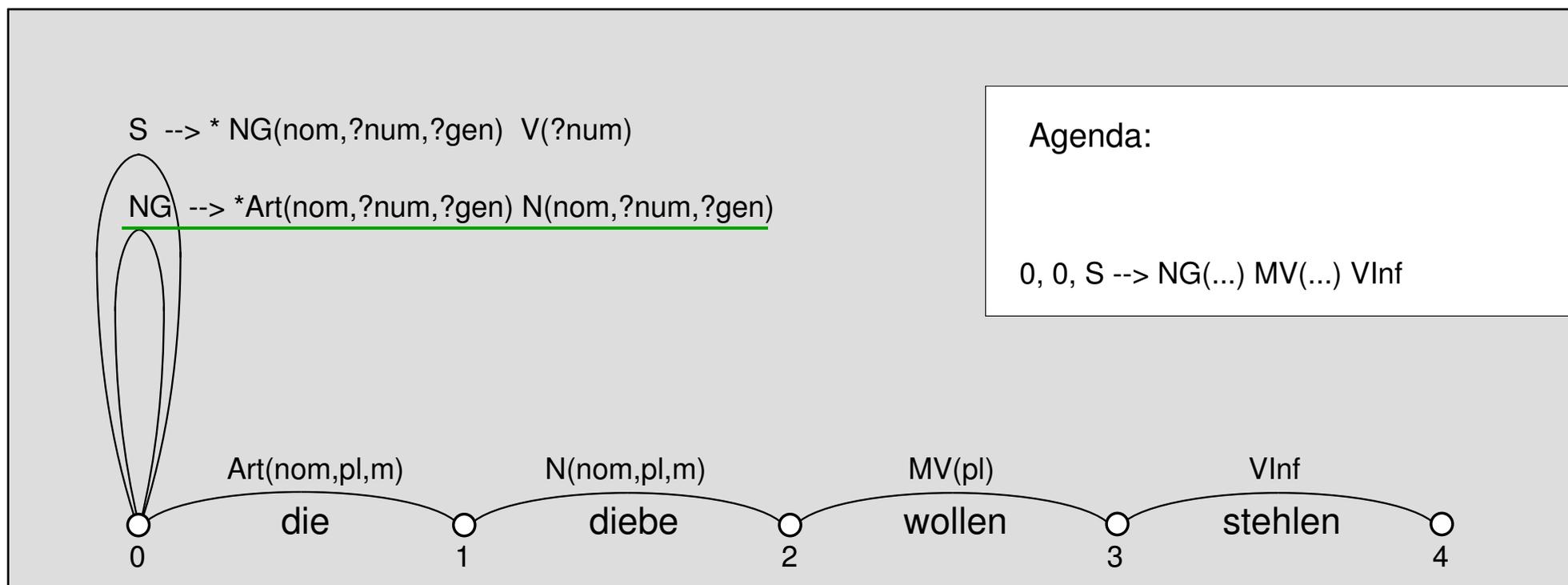


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

neue aktive NG-Kante in Chart

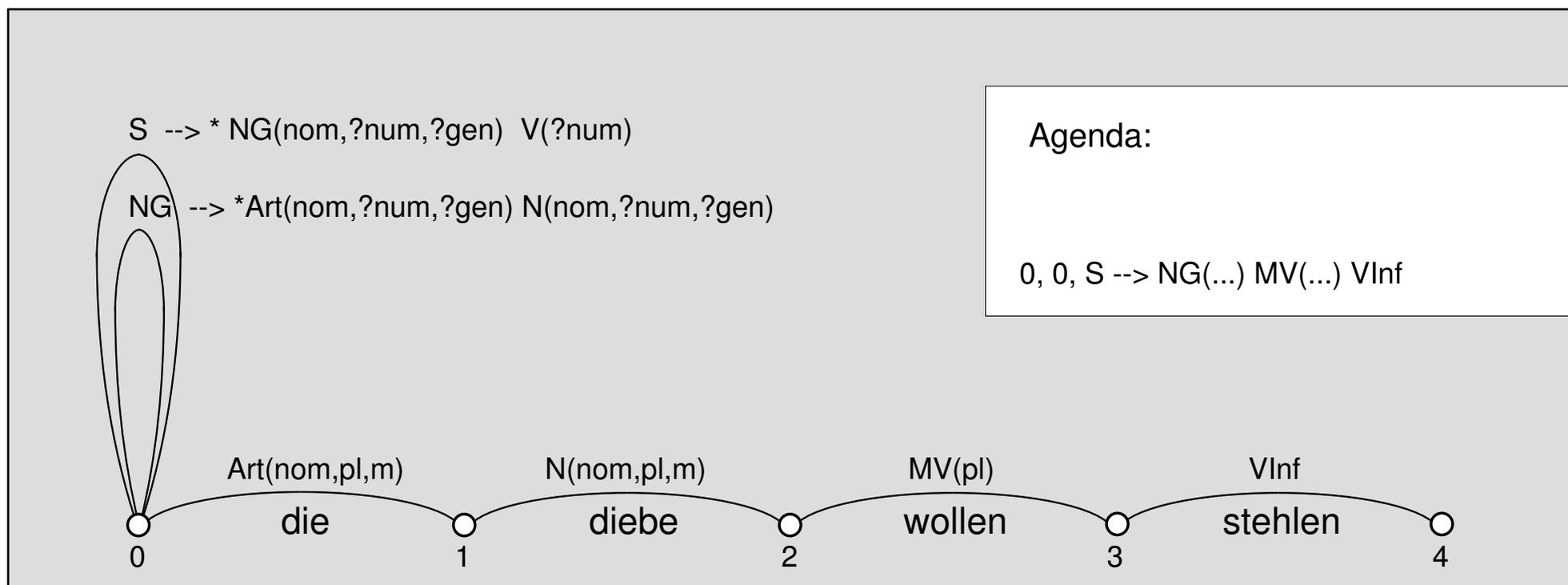


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

## Fundamentalregel

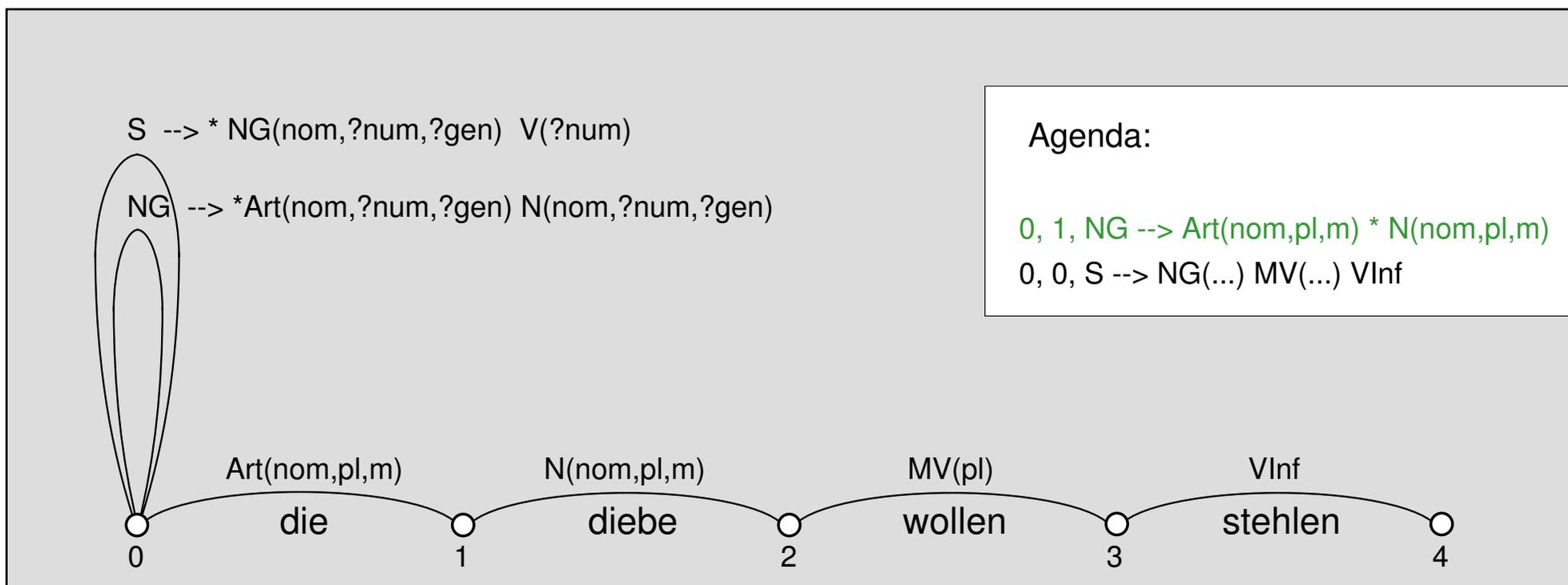


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

neue NG-Kante in Agenda



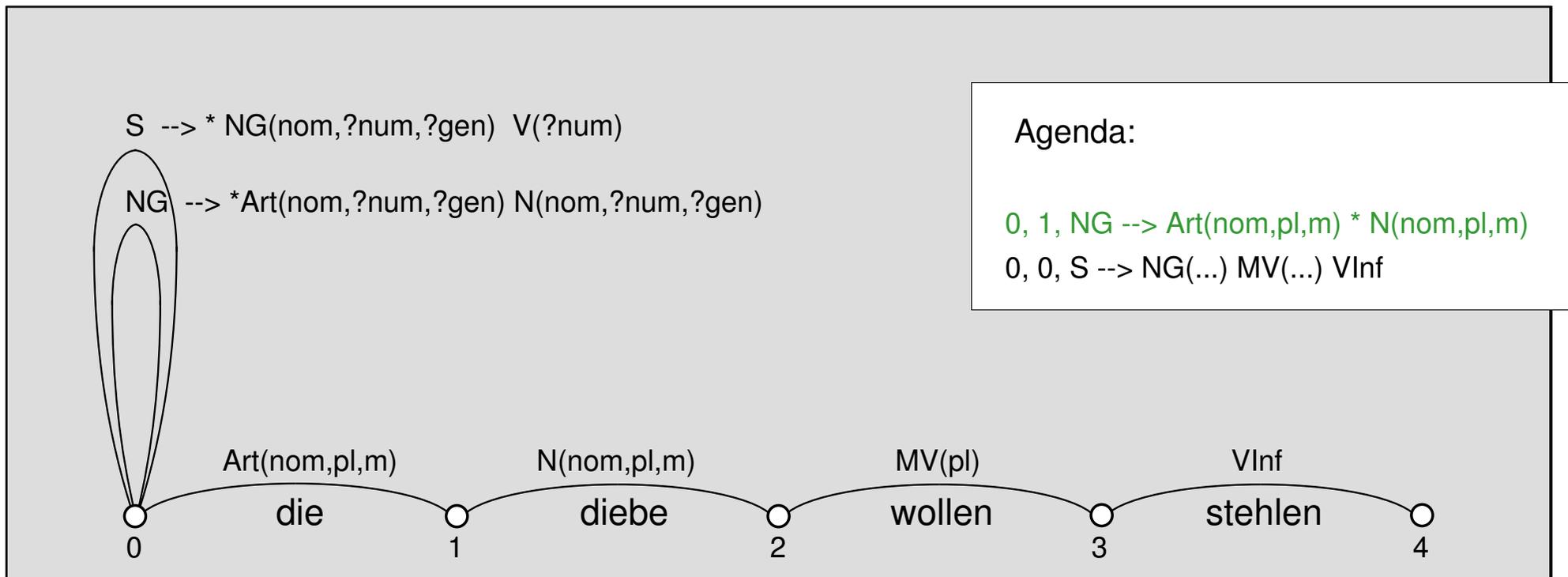
>>>

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

oberste Kante aus Agenda entfernen



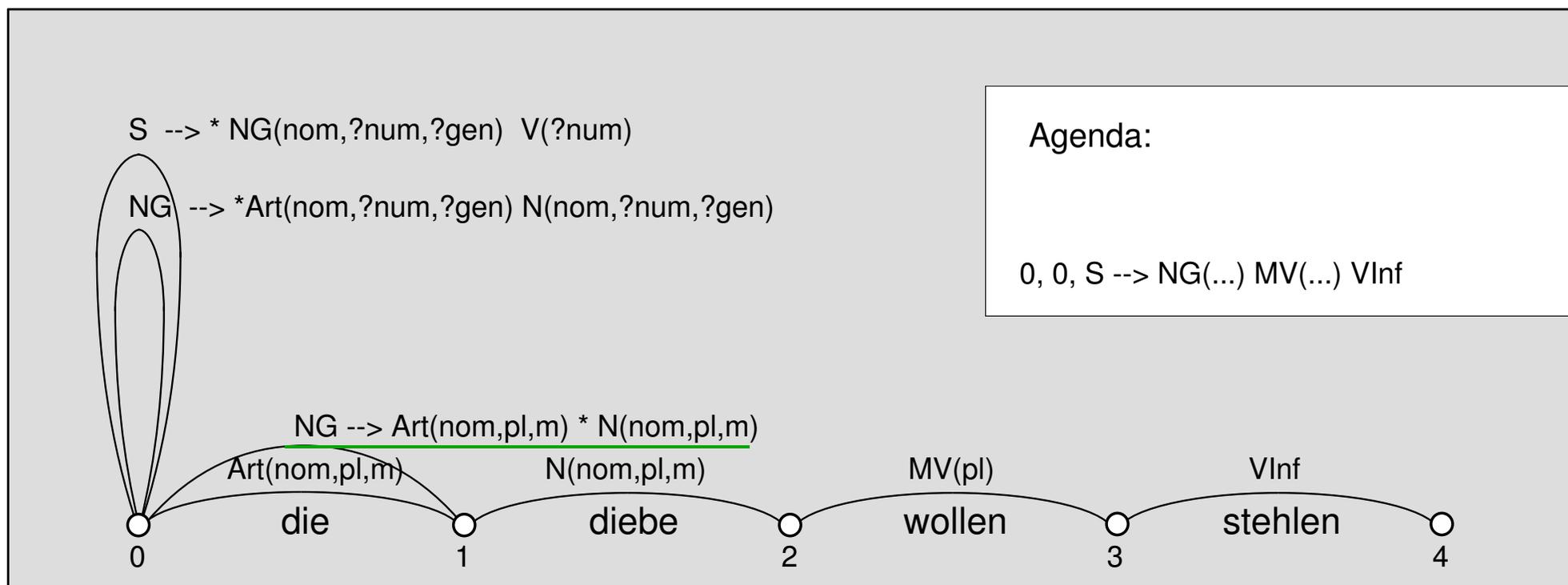
>>>

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

neue aktive NG-Kante in Chart

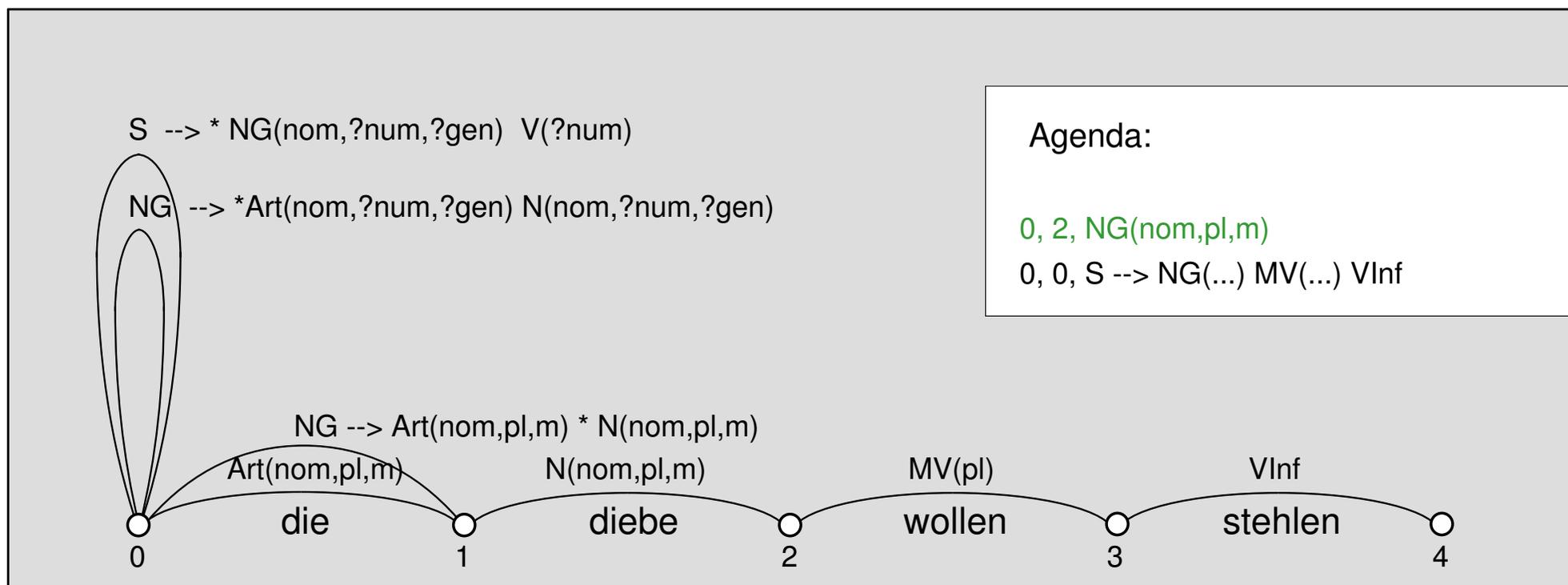


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

Fundamentalregel → passive NG-Kante in Agenda



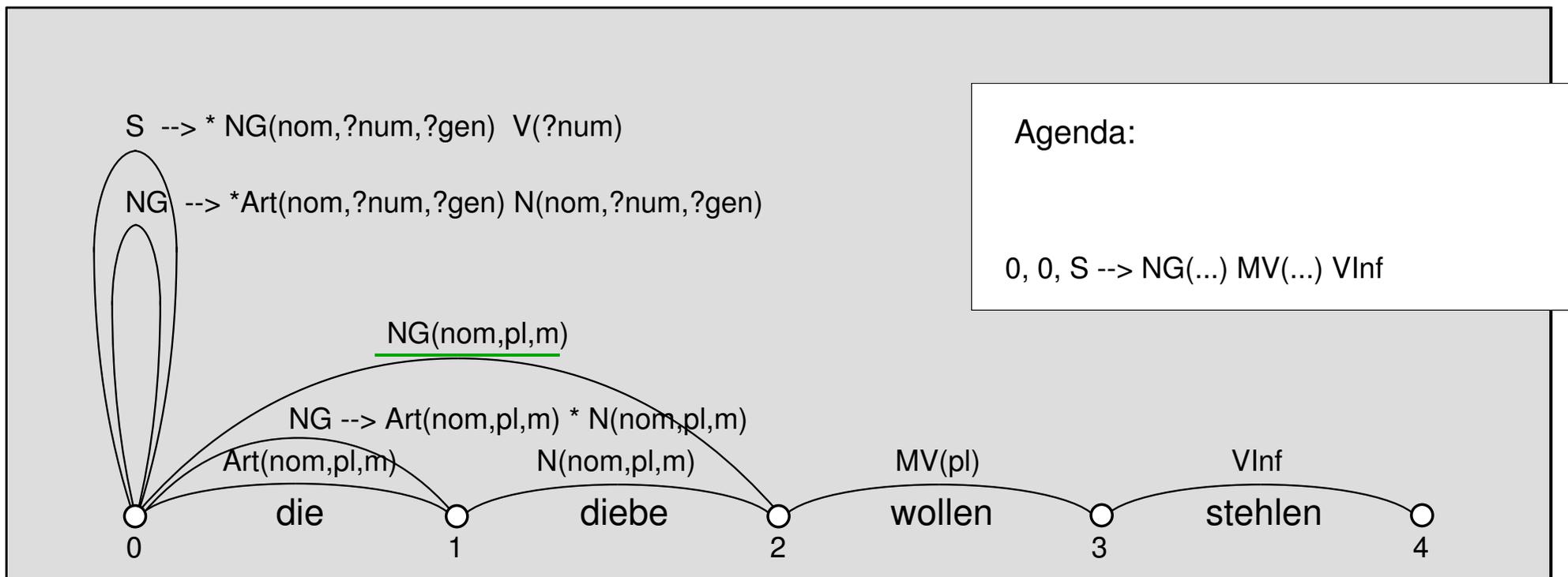
>>>

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

oberste Kante aus Agenda → neue passive NG-Kante in Chart

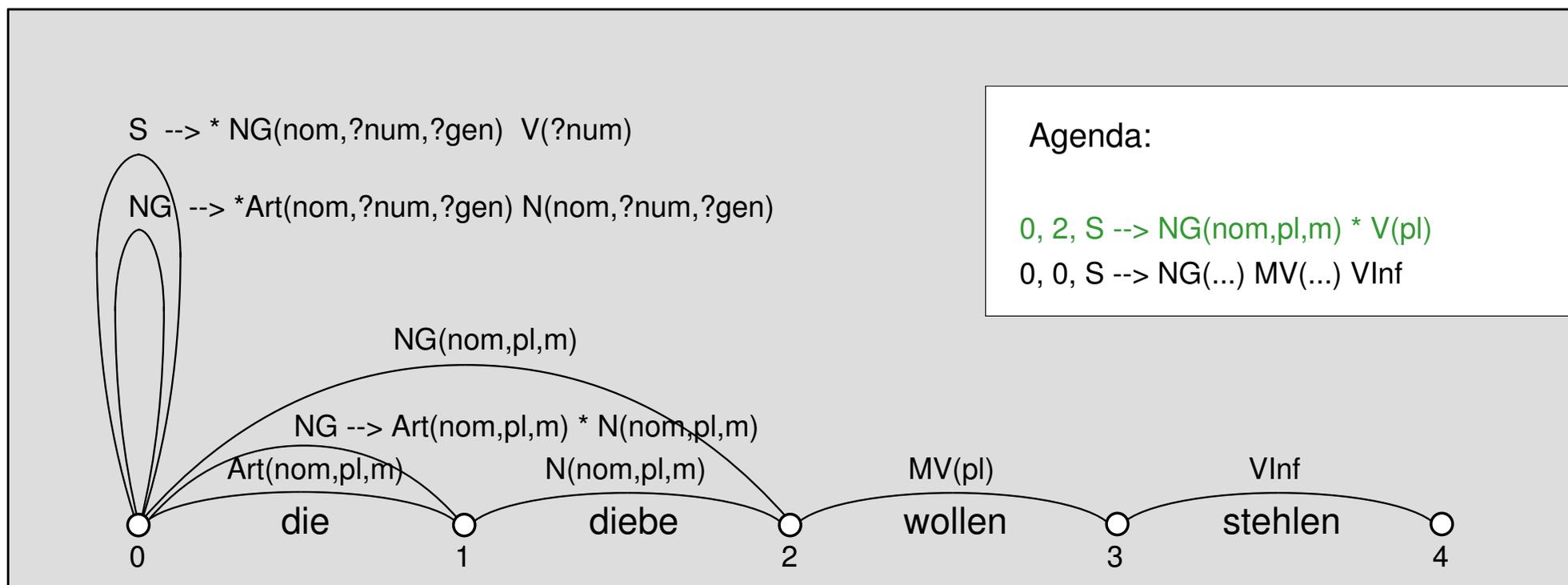


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

Fundamentalregel → aktive S-Kante in Agenda

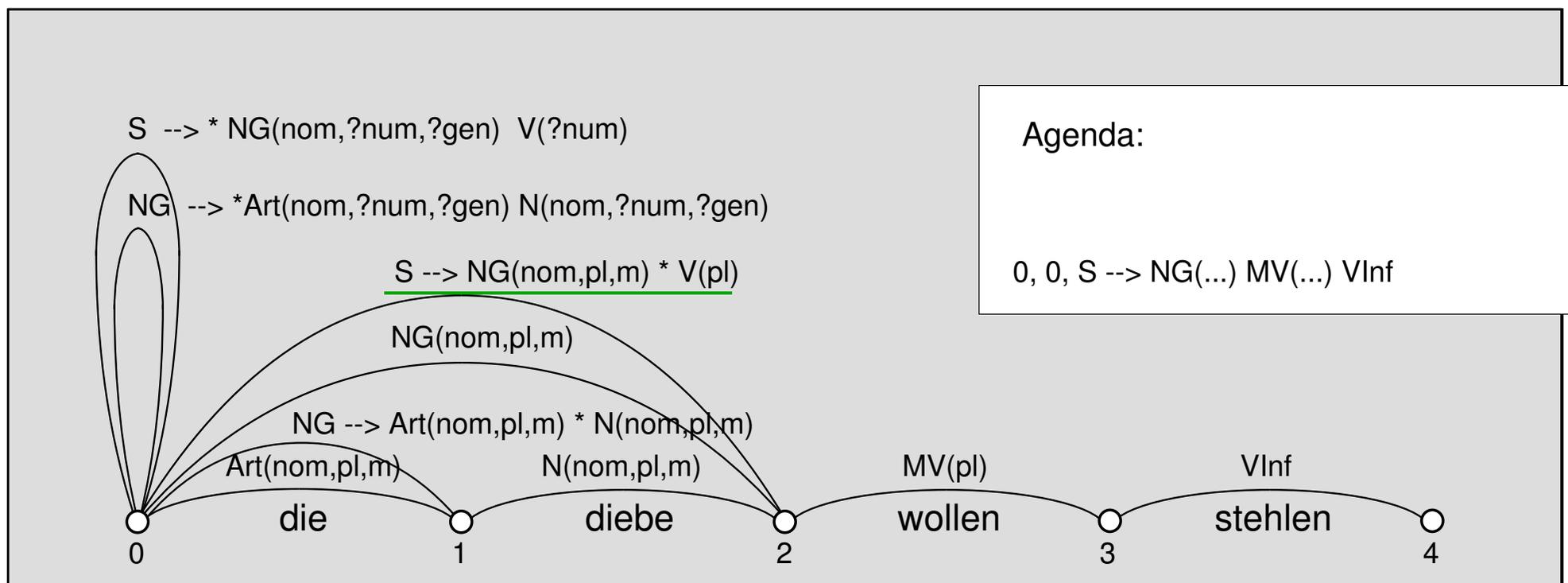


>>>

# Chart-Parsing mit DCG    Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

oberste Kante aus Agenda → neue aktive S-Kante in Chart

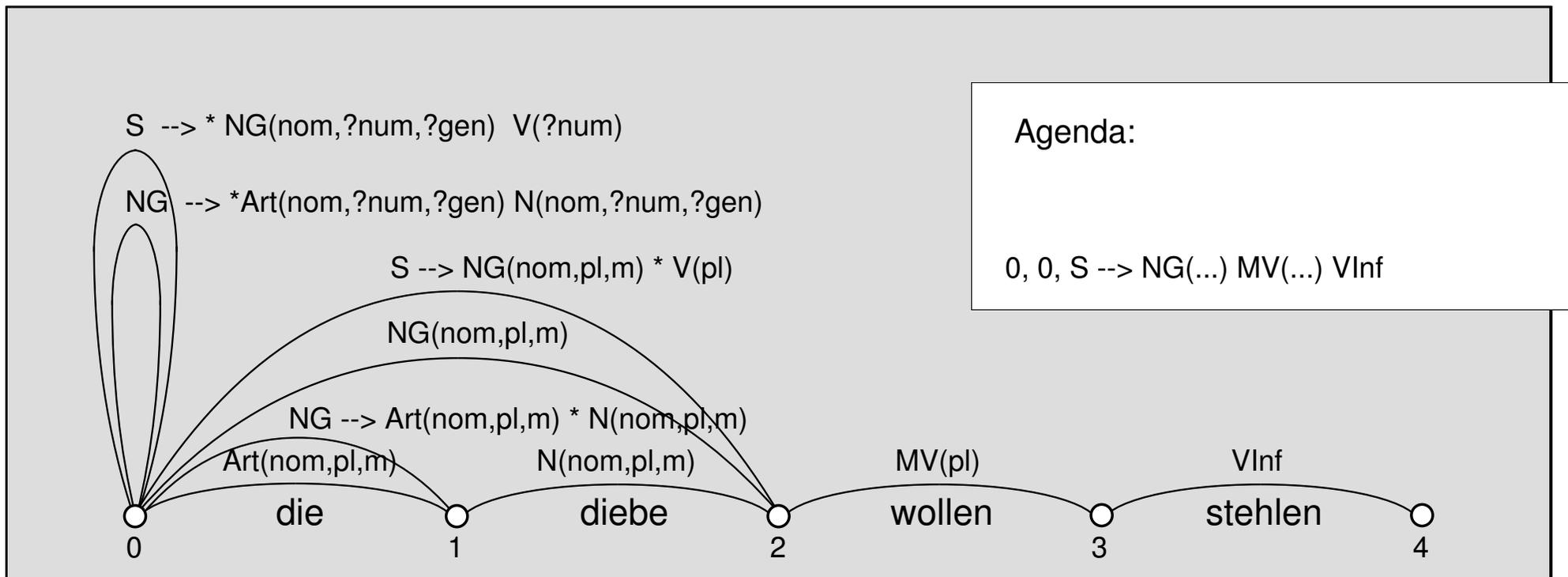


>>>

# Chart-Parsing mit DCG Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

Fundamentalregel und Top-down-Prädiktion: **nicht anwendbar!**<sup>1)</sup>



1) Die Top-down-Prädiktion produziert sinnvollerweise keine Wortkanten

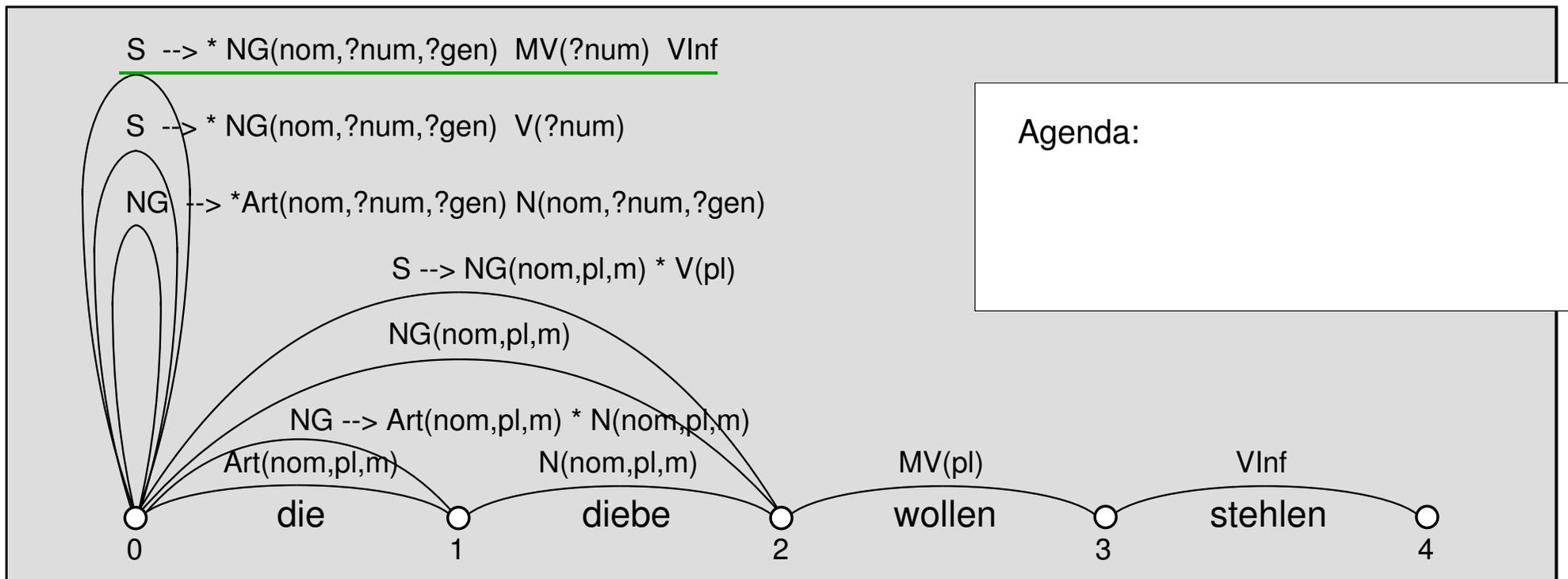
>>>

# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

oberste Kante aus Agenda → neue aktive S-Kante in Chart

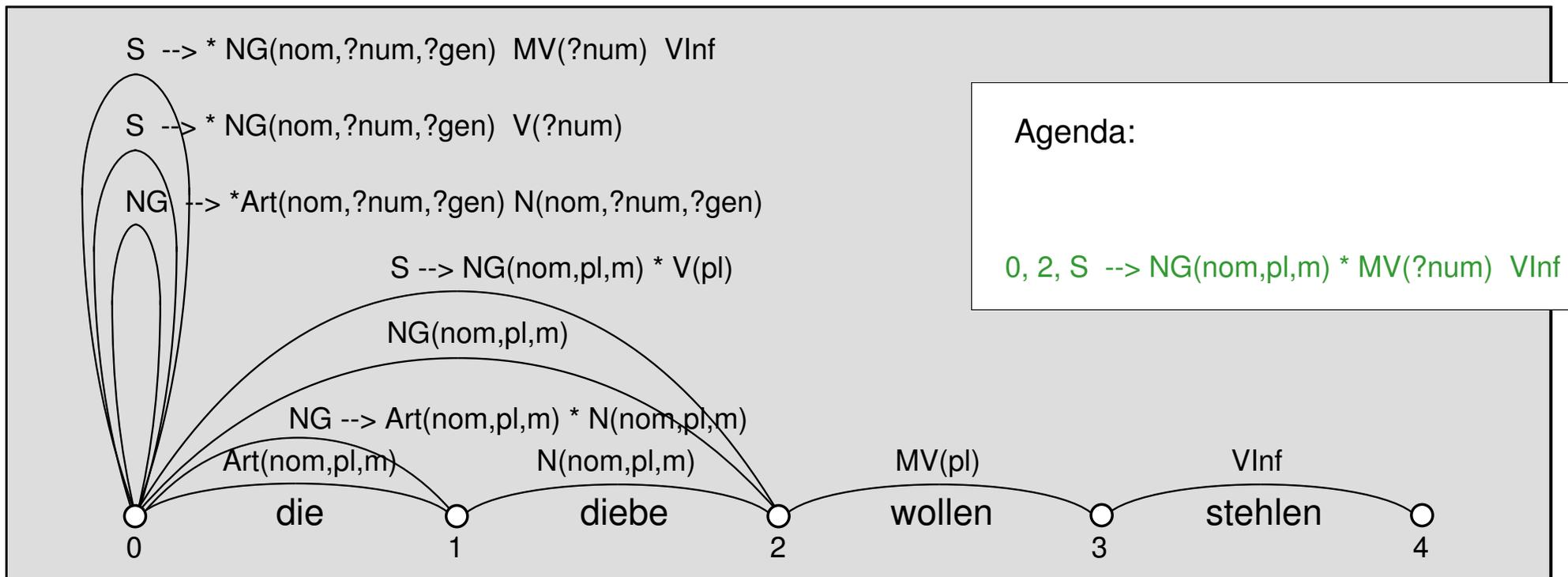


# Chart-Parsing mit DCG

Top-down-Analyse mit Tiefensuche

Analyse des Satzes: "Die Diebe wollen stehlen."

Fundamentalregel: früher analysierte NG wird wieder verwendet!



# Zusammenfassung

## DCG-Formalismus (*definite clause grammar*)

- Eignet sich zur Beschreibung natürlicher Sprachen  
(Verschachtelung erfordert mindestens Typ-2-Grammatik)
- Dank Attributen effizient → geringe Anzahl Regeln nötig
- Unifikation der Attribute
- kontextfreies Skelett → erlaubt Chart-Parsing  
Ableitungsbaum (Syntaxbaum)

Thema der nächsten Lektion:

# Grammatik für natürlichsprachliche Wörter

## Two-Level-Regeln und Transduktoren

Zur Übersicht der Vorlesung *Sprachverarbeitung II* >>>



## Beispiel eines Satzes einer kontextsensitiven Sprache

“D’Mueter hät d’Chind im Heiri sis Huus lo hälfe striiche”

<<<

# Subjekt

Vögel

Nomen

der Hund

Artikel Nomen

er

Pronomen

Hans

Nomen (Eigename)

der grosse Hund

Artikel Adjektiv Nomen

der Hund beim Haus

Artikel Nomen Präposition Nomen

# Subjekt

Vögel	Nomen
der Hund	Artikel Nomen
er	Pronomen
Hans	Nomen (Eigename)
der grosse Hund	Artikel Adjektiv Nomen
der Hund beim Haus	Artikel Nomen Präposition Nomen
	→ Nominalgruppe (NG)

<<<



# Prädikat

fliegt	Verb
fliegt immer	Verb Adverb
ist geflogen	Hilfsverb Verb
ist weit geflogen	Hilfsverb Adverb Verb
ist auf den Baum geflogen	Hilfsverb Präposition Artikel Nomen Verb
flog auf den Baum	Verb Präposition Artikel Nomen

# Prädikat

fliegt	Verb
fliegt immer	Verb Adverb
ist geflogen	Hilfsverb Verb
ist weit geflogen	Hilfsverb Adverb Verb
ist auf den Baum geflogen	Hilfsverb Präposition Artikel Nomen Verb
flog auf den Baum	Verb Präposition Artikel Nomen
	→ Verbalgruppe (VG)





# Beispiel einer einfachen Satzgrammatik

**Grammatik:** Produktionsregeln der formalen Grammatik

- $S \rightarrow NG\ VG$  (Satz := Nominalgruppe + Verbalgruppe)
- $NG \rightarrow N$  (Nominalgruppe := Nomen; einfachster Fall)
- $NG \rightarrow Art\ N$  (Nominalgruppe := Artikel + Nomen)
- $VG \rightarrow V$  (Verbalgruppe := Verb; einfachster Fall)
- $VG \rightarrow V\ NG$  (Verbalgruppe := Verb + Nominalgruppe)

**Lexikon:** Produktionsregeln für Terminalsymbole (Vollformen)

$Art \rightarrow der \mid die \mid das \mid dem \mid ein \mid eines \mid \dots$

$N \rightarrow frau \mid mann \mid baum \mid hauses \mid häuser \mid \dots$

$V \rightarrow sehen \mid siehst \mid steht \mid blüht \mid \dots$

<<<



## Format von DCG-Regeln

$S \rightarrow NG(nom, ?num, ?gen) \quad VG(?num)$

$NG(?kas, ?num, ?gen) \rightarrow Art(?num, ?kas, ?gen) \quad N(?kas, ?gen, ?num)$

- Beliebige Anzahl von Attributen
- Jedes Attribut ist entweder eine Konstante (*nom*) oder eine Variable (*?kas*)
- Reihenfolge und Anzahl der Attribute eines bestimmten NT sind fix
- Reihenfolge und Anzahl der Attribute verschiedener NT sind frei

<<<



# Grundlegende Operation: Term-Unifikation

Unifikation zweier Terme

⇒ Finden von Variablensubstitutionen so, dass Terme identisch werden  
(Substitutionen = Variablenbindungen)

# Grundlegende Operation: Term-Unifikation

Unifikation zweier Terme

⇒ Finden von Variablensubstitutionen so, dass Terme identisch werden  
(Substitutionen = Variablenbindungen)

Beispiele:

$$?x \odot ?y \Rightarrow ?x = ?y$$

$$?z \odot g \Rightarrow ?z = g$$

# Regeln der Unifikation von Termen

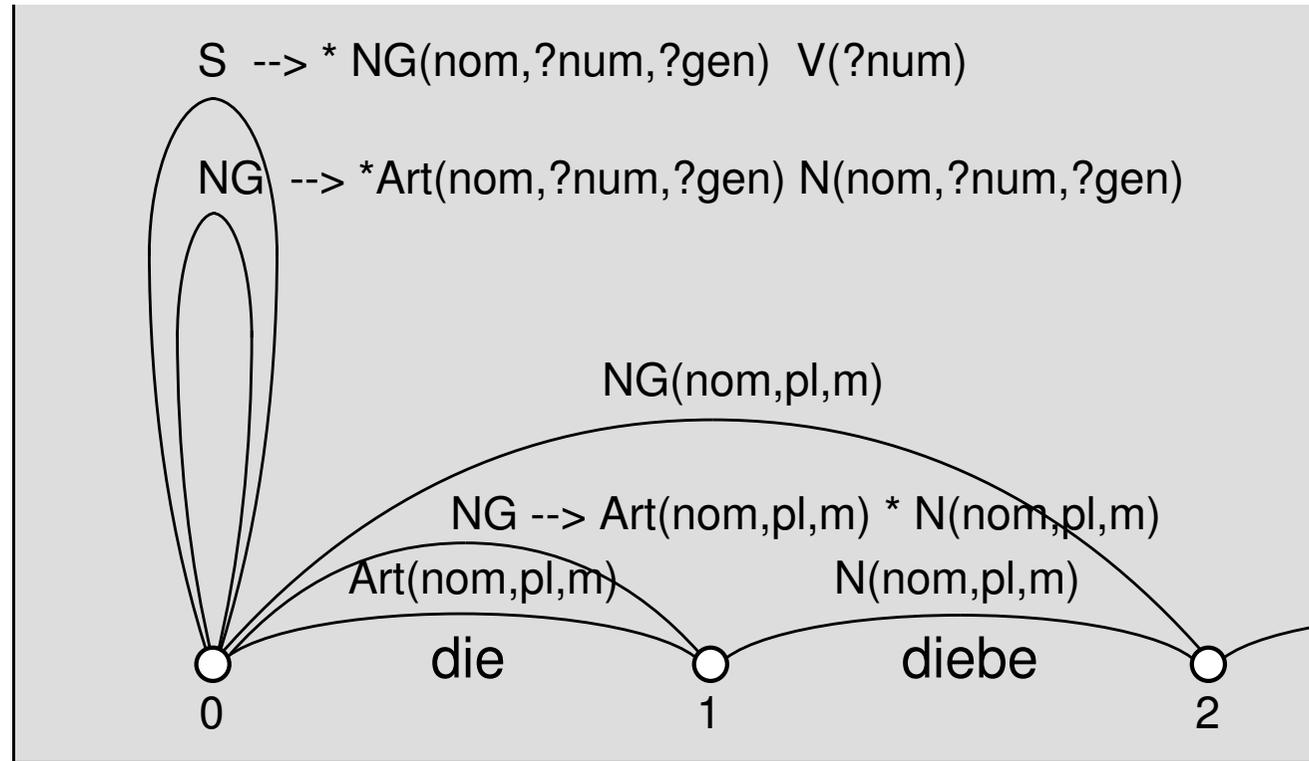
- Identische Konstanten sind unifizierbar  $a \odot a$
- Ungleiche Konstanten sind *nicht* unifizierbar  $a \oslash b$
- Variablen sind mit Konstanten unifizierbar  $?v \odot h \Rightarrow ?v = h$
- Variablen sind mit Variablen unifizierbar  $?v \odot ?u \Rightarrow ?v = ?u$   
(Variablen werden identisch)

<<<



# Chart

Graphische Darst.:



Symbolische Darst.:

Liste von Kanten

- 0, 1,  $\text{Art}(\text{nom}, \text{pl}, \text{m})$  ← passiv
- 1, 2,  $N(\text{nom}, \text{pl}, \text{m})$  ←
- ⋮
- 0, 0,  $S \rightarrow * NG(\text{nom}, ?\text{num}, ?\text{gen}) V(?num)$  ← aktiv
- 0, 0,  $NG \rightarrow * \text{Art}(\text{nom}, \text{pl}, \text{m}) N(\text{nom}, \text{pl}, \text{m})$  ←
- 0, 1,  $NG \rightarrow \text{Art}(\text{nom}, \text{pl}, \text{m}) * N(\text{nom}, \text{pl}, \text{m})$  ←
- 0, 2,  $NG(\text{nom}, \text{pl}, \text{m})$  ←

<<<



# Agenda

Stack:

```
⋮  
0, 1, NG --> Art(nom,pl,f) * N(nom,pl,f)  
⋮  
0, 1, NG --> Art(akk,pl,m) * N(akk,pl,m)  
0, 1, NG --> Art(nom,pl,m) * N(nom,pl,m)  
0, 0, S --> * NG(nom,?num,?gen) MV(?num) Vinf
```

**Merke:** LIFO: → Tiefensuche

FIFO: → Breitensuche



