

Transformationsgrammatik der 60er/70er
Perlmutter & Soames 1979
Dienstag, 11h15-12h45
Kl. HS, CLI, Liebigstr.

Fabian Heck
Institut für Linguistik
Universität Leipzig
Sommersemester 2006

1. Einführung

1.1. Grundsätzliches

Ziele des Seminars:

- (i) Es soll Einblick gewährt werden in die Syntaxtheorie der 60er-70er Jahre.
- (ii) Es soll herangeführt werden an die Technik des syntaktischen Argumentierens.

Hintergrund:

- (i) Viele Konzepte und Begriffe moderner Syntaxtheorien haben ihren Ursprung in früheren Theorien. Es kann daher nützlich sein, diese Theorien ein wenig zu kennen.
- (ii) Die Linguistik beschäftigt sich mit dem menschlichen Geist, ist also eine Geisteswissenschaft. Die Argumentation verläuft aber oft ähnlich wie in den Naturwissenschaften.

Vorgehensweise:

Grundlage des Kurses bildet das Buch "Syntactic Argumentation and the Structure of English" von David Perlmutter und Scott Soames (Perlmutter & Soames 1979).

1.2. Theoretischer Hintergrund

Vorweg:

- (i) Perlmutter & Soames (1979) bauen im wesentlichen auf Chomsky (1965) auf, was damals oft als "Standardtheorie" bezeichnet wurde.
- (ii) Im Folgenden sollen einige Eigenschaften der Theorie, wie sie in Chomsky (1965) gezeichnet wird, dargelegt werden.

Chomsky (1965):

- (i) Die Grammatik (das heißt, die Syntax) teilt sich auf in zwei Komponenten: die Basiskomponente und die Transformationskomponente.
- (ii) Zusammen sollen diese Komponenten in der Lage sein, alle grammatisch wohlgeformten Sätze (und nur diese!) einer Sprache zu erzeugen.
- (iii) Das liefert dann eine Theorie dieser Sprache.

1.2.1. Phrasenstrukturregeln

Basiskomponente:

- (i) Die Basiskomponente ist eine Menge von kontextfreien Phrasenstrukturregeln.
- (ii) Die Phrasenstrukturregeln generieren Tiefenstrukturen (TS). Tiefenstrukturen sind

hierarchisch gegliedert. Dies wird ausgedrückt, indem man sie als Bäume notiert (oder als Klammerstrukturen).

(iii) Jede Phrasenstrukturregel ist von der Form $X \rightarrow Y$. X ist ein Nonterminalsymbol, während Y eine Reihe aus Nonterminal- und Terminalsymbolen sein kann.

(iv) Terminalsymbole sind die Grundeinheiten der Syntax, also die Wörter.

(v) Nonterminalsymbole sind abstraktere Objekte, die keine phonologischen Merkmale haben wie Wörter. Man kann sie daher nicht hören. Sie sind aber notwendig, um hierarchische Strukturen aufzubauen.

(1) *Phrasenstrukturregeln für ein Fragment des Englischen*

- a. $S \rightarrow NP VP$
- b. $VP \rightarrow V$
- c. $VP \rightarrow V PP$
- d. $PP \rightarrow P NP$
- e. $P \rightarrow for$
- f. $NP \rightarrow Harry$
- g. $NP \rightarrow Luna$
- h. $V \rightarrow sang$

Erläuterungen:

(i) Terminalsymbole sind *for, Harry, Luna, sang*; Nonterminalsymbole sind S, NP, VP, PP, P und V . S, NP, VP , usw. sind Abkürzungen für Satz, Nominalphrase, Verbalphrase, usw.

(ii) Es wird hier nicht motiviert, wieso man überhaupt annimmt, dass Sätze hierarchisch gegliedert sind. Wir setzen eine solche Motivation hier voraus.

(iii) Es wird auch nicht motiviert, wieso man gerade diese Phrasenstrukturregel annimmt.

(iv) Mit den Regeln in (1) können die Tiefenstrukturen in (3) erzeugt werden, und zwar den Anweisungen in (3) folgend.

(2) *Tiefenstrukturen, durch (1) erzeugt*

- a. Harry sang
- b. Luna sang
- c. Harry sang for Luna
- d. Luna sang for Harry
- e. Harry sang for Harry
- f. Luna sang for Luna

(3) *Erzeugung einer Tiefenstruktur:*

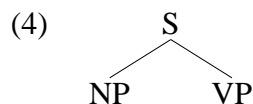
- a. Nimm eine Regel R und notiere das Nonterminalsymbol X , welches auf der linken Seite steht.
- b. Notiere unter X alle Symbole Y_1, Y_2, \dots, Y_n der rechten Seite von R in ihrer Reihenfolge von links nach rechts.
- c. Verbinde X jeweils durch eine Kante mit den Symbolen Y_1, Y_2, \dots, Y_n .
- d. Für jedes Symbol Y_i aus Y_1, Y_2, \dots, Y_n , welches ein Nonterminalsymbol ist,

nimm eine Regel R' , auf deren linken Seite Y_i steht.

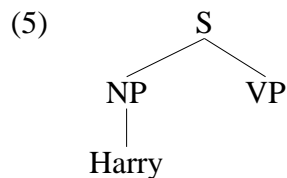
- e. Springe wieder zu Punkt (3-b), wobei X in (3-b) als Y_i und R in (3-b) als R' verstanden wird.

Beispiel:

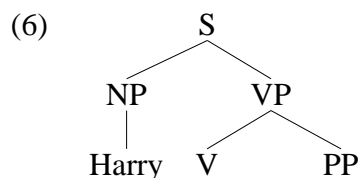
- (i) Man wählt z.B. die Regel $S \rightarrow NP VP$ und notiert S (siehe (3-a)).
(ii) Darunter werden die Symbole der rechten Regelseite (NP und VP) von links nach rechts notiert (siehe (3-b)).
(iii) S wird über Kanten mit NP und VP verbunden (siehe (3-c)).



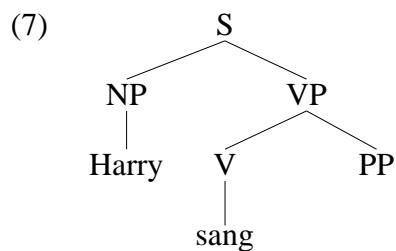
- (iv) Es gibt zwei Nonterminale, die entsprechend (3-d) zu betrachten sind: NP und VP . Man beginnt mit NP und wählt z.B. die Regel $NP \rightarrow Harry$.
(v) Man springt zu (3-b), wobei $X = NP$ und $R = NP \rightarrow Harry$. Man notiert unter NP alle Symbole der rechten Seite: es gibt nur ein solches, nämlich $Harry$.
(vi) Die entsprechende Kante wird eingezogen (siehe (3-c)):



- (vii) $Harry$ ist kein Nonterminal, (3-d) irrelevant. Man springt zurück zu dem Punkt, an dem man die Wahl zwischen NP und VP hatte, (3-d) (siehe (iv) oben).
(viii) Hier ist noch das Nonterminal VP zu bearbeiten. Man wählt die Regel $VP \rightarrow V PP$ und springt zu (3-b), wobei $X = VP$ und $R = VP \rightarrow V PP$.
(ix) Man notiert V und PP unter VP und zieht Kanten ein (siehe (3-c)).



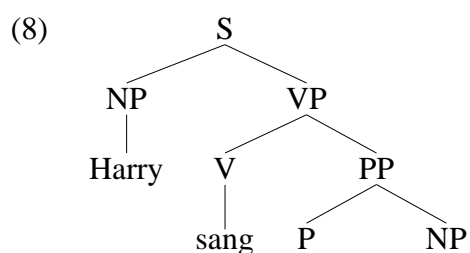
- (x) Entsprechend (3-d) wählt man die Regel $V \rightarrow sang$ und springt zu (3-b) mit $X = V$ und $R = V \rightarrow sang$.
(xi) Das Terminalsymbol $sang$ wird notiert.
(xii) Die Kante zwischen V und $sang$ wird eingefügt (siehe (3-c)).



(xiii) Man springt zurück, um die PP abzuhandeln. Dafür wählt man die Regel $PP \rightarrow P$ NP (siehe (3-d)).

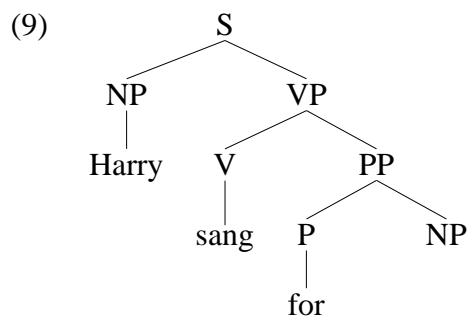
(xiv) Zurück zu (3-b), mit $X = PP$ und $R = PP \rightarrow P$ NP. P und NP werden notiert.

(xv) P und NP werden durch Kanten mit PP verbunden (siehe (3-c)).



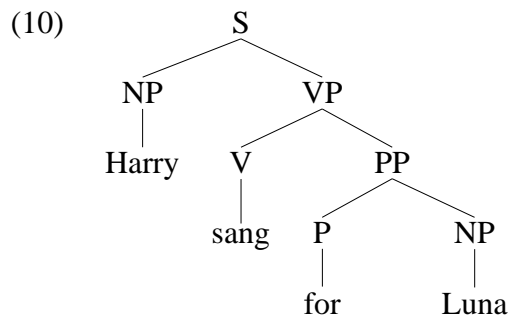
(xvi) P ist ein Nonterminal, also wählt man eine Regel mit P auf der linken Seite: $P \rightarrow for$ (siehe (3-d)).

(xvii) Man springt zurück zu (3-b) mit $X = P$ und $R = P \rightarrow for$ und notiert die rechte Seite (siehe (3-b)). Anschließend wird wieder eine Kante gezogen.



(xviii) Schließlich wird für NP die Regel $NP \rightarrow Luna$ gewählt (siehe (3-d)).

(ixx) Man springt zu (3-b), wobei $X = NP$ und $R = NP \rightarrow Luna$. Das Terminalsymbol der rechten Seite von R wird notiert, die letzte Kante wird eingezogen (siehe (3-c)).



Beachte:

- (i) **Harry sang for Harry* wird ebenfalls von (1) erzeugt, ist aber nicht grammatisch (wenn das *Harry* beides mal für dieselbe Person steht).
- (ii) Daran sieht man: Tiefenstrukturen müssen nicht grammatisch sein; wir kommen darauf zurück, wenn wir über obligatorische Transformationen sprechen.

Rekursion:

Rekursion nennt man das Auftauchen desselben Nonterminals auf der linken und rechten Seite einer Regel (oder einer Folge von hintereinander anwendbaren Regeln).

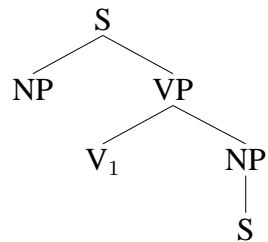
(11) *Beispiel für Rekursion:*

- a. $S \rightarrow NP VP$
- b. $VP \rightarrow V_1 NP$
- c. $VP \rightarrow V_2$
- d. $V_1 \rightarrow \textit{believed}$
- e. $V_2 \rightarrow \textit{sang}$
- f. $NP \rightarrow S$
- g. $NP \rightarrow \textit{Luna, Harry, Ron}$

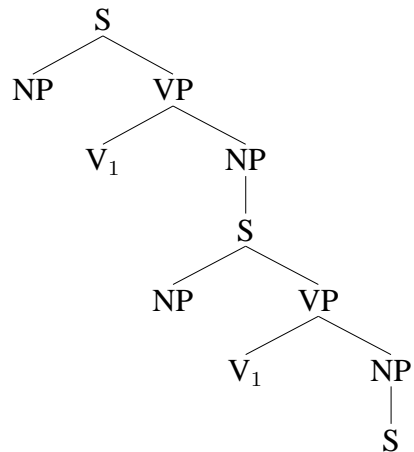
Erläuterung:

- (i) Es gibt eine Sequenz von hintereinander angewandten Regeln, nämlich (11-a), (11-b) und (11-f), so dass das Symbol S auf der linken Seite der ersten und der rechten Seite der letzten Regel auftaucht.
- (ii) Damit kann aber nach Anwendung von (11-f) wieder die Regel (11-a) angewandt werden, dann wieder (11-b) und (11-f), und so weiter.
- (iii) Wenn die Ableitung gestoppt werden soll, muss irgendwann die Rekursion durchbrochen werden, durch Anwendung der Regel (11-c).

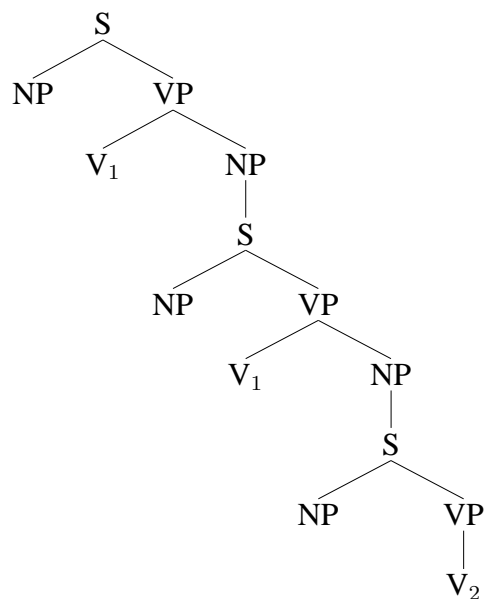
(12) a. 1. Rekursion durch (11-a), (11-b), (11-f)



b. 2. Rekursion durch (11-a), (11-b), (11-f)



c. Ende der Rekursion durch (11-c)



Seitenbemerkung:

Es wurde hier angenommen, dass eingebettete Sätze immer unter einem NP-Knoten hängen. Das soll uns nicht weiter stören.

Beachte:

(i) Rekursion ist ein wichtiges Merkmal natürlicher Sprache. Dadurch können mit einer

endlichen Anzahl von Regeln unendlich viele Strukturen erzeugt werden.

(ii) Das Beispiel in (11) kann beliebige große Strukturen wie in (13) erzeugen.

- (13) a. Harry sang
b. Luna believed Harry sang
c. Ron believed Luna believed Harry sang
d. ...

Seitenbemerkungen:

(i) Phrasenstrukturregeln heißen kontextfrei, wenn jeweils auf der linken Seite der Regeln nur ein Nonterminalsymbol auftaucht.

(ii) Es gibt auch andere Typen von Phrasenstrukturregeln, welche dann eine andere Klasse von Sprache definieren (siehe z.B. Partee, ter Meulen & Wall 1990).

1.2.2. Transformationen

Transformationskomponente:

(i) Die Transformationskomponente besteht aus Transformationsregeln. Jede Transformationsregel nimmt als Input einen Phrasenstrukturbaum und bildet ihn auf einen anderen Phrasenstrukturbaum (den Output) ab. Der Output der Transformation kann Input einer anderen Transformation sein.

(ii) Am Beginn einer (möglicherweise leeren) Folge von Transformationen steht immer eine Tiefenstruktur. Der finale Output einer Transformationssequenz ist eine sogenannte Oberflächenstruktur (OS).

(iii) Eine Tiefenstruktur kann auch gleichzeitig Oberflächenstruktur sein, wenn keine Transformation darauf angewandt wird.

Beachte:

Den Begriff der Ungrammatikalität kann man sich jetzt folgendermaßen vorstellen.

(14) *Ungrammatikalität:*

Eine Struktur S ist ungrammatisch, wenn S nicht ausgehend von einer Tiefenstruktur durch Anwendung einer (möglicherweise leeren) Abfolge von Transformationen abgeleitet werden kann.

Seitenbemerkung:

Oft wird ein Argument darum gehen, ob zwei Oberflächenstrukturen von derselben Tiefenstruktur abgeleitet werden, oder nicht.

Drei Typen von Transformationen:

Es gibt wenigstens drei Typen von Transformationen;

- (i) Tilgung
(ii) Einfügung
(iii) Permutation

Tilgung:

Angenommen, es gäbe eine Transformation, die α tilgt, wenn α auf ein Verb folgt. Dies würde ausgedrückt wie in (15).

(15) α -Tilgung:

| | | | | | |
|-----|---|---|----------|---|---|
| | X | V | α | Y | |
| SB: | 1 | 2 | 3 | 4 | → |
| SW: | 1 | 2 | ∅ | 4 | |

Einfügung:

Angenommen, es gäbe eine Transformation, die α zwischen einem Verb und einer NP einfügt, dann könnte man dies folgendermassen ausdrücken:

(16) α -Einfügung:

| | | | | | |
|-----|---|---|----------|---|---|
| | X | V | NP | Y | |
| SB: | 1 | 2 | 3 | 4 | → |
| SW: | 1 | 2 | α | 3 | 4 |

Permutation:

Eine Transformation, die ein α , das einer NP direkt vorangeht, direkt hinter diese NP verschiebt, zeigt (17).

(17) α -Bewegung:

| | | | | | |
|-----|---|----------|----|---|---|
| | X | α | NP | Y | |
| SB: | 1 | 2 | 3 | 4 | → |
| SW: | 1 | ∅ | 3 | 2 | 4 |

Konvention:

Die Strukturbeschreibung einer Transformation muss den gesamten Baum abdecken, auf den sie appliziert.

Beispiel:

(i) Angenommen, es gäbe eine Transformation T mit der SB in (18):

(18) Transformation T:

| | | | |
|-----|-----|-----|---|
| | NP | V | |
| SB: | 1 | 2 | → |
| SW: | ... | ... | |

(ii) Ohne die Konvention oben würde T auf (19) applizieren können, denn in (19) gibt es eine NP, die einem V vorangeht (*Raccoons* vor *ate*).

(iii) Mit der Konvention kann T aber nicht auf (19) angewandt werden, da die Strukturbeschreibung von T nicht den gesamten Baum in (19) abdeckt: die NP *the garbage* fehlt.

(19) [S [NP Raccoons] [VP [V ate] [NP the garbage]]]

Eine konkrete Transformation:

Es ist oft angenommen worden, dass (20-b) aus (20-a) durch die Transformationsregel der Extraposition von S in (21) abgeleitet wird (siehe Ross 1967/86, 2).

(20) *Tiefen- und Oberflächenstruktur der Extraposition von S*

- a. [NP A gun [S which I had cleaned]] went off
- b. [NP A gun] went off [S which I had cleaned]

(21) *Extraposition von S (optional)*

| | | | | | | |
|-----|---|--------------------|---|---|---|---|
| | X | [_{NP} NP | S |] | Y | |
| SB: | 1 | | 2 | | 3 | → |
| SW: | 1 | | ∅ | | 3 | 2 |

Erläuterungen:

- (i) Extraposition von S verschiebt einen Relativsatz von der NP, die er modifiziert, weg und plaziert ihn am Ende des Satzes, in dem diese NP steht.
- (ii) Die Regel (21) indiziert die Teile des Inputs mit Ziffern: 1 für den linken Rand und die NP, 2 für den Relativsatz, 3 für den rechten Rand. Man nennt dies auch die strukturelle Beschreibung (SB).
- (iii) Im Output der Regel hat sich die Nummer 2, die für den Relativsatz steht, an den rechten Rand verschoben. Man nennt dies auch strukturellen Wandel (SW).
- (iv) Die Stelle, welche durch ∅ im Output der Regel markiert ist, zeigt an, wo der Relativsatz vor der Extraposition stand.

Seitenbemerkung 1:

Perlmutter & Soames definieren Transformationen oft nicht nach einem Schema wie in (21) sondern eher umgangssprachlich. Das erfüllt aber auch seinen Zweck.

Seitenbemerkung 2:

- (i) X und Y in (21) sind Variablen, sie können für alles mögliche an Struktur stehen, auch für nichts.
- (ii) Dies ist so gemacht, damit (21) auch z.B. Fälle wie in (22) ableiten kann.

- (22)
- a. I gave a gun which I had cleaned to my brother
 - b. I gave a gun to my brother which I had cleaned
 - c. He let the cats which were meowing out
 - d. He let the cats out which were meowing

Problem:

Die Regel mit den Variablen in (21) übergeneriert: es werden auch Sätze erzeugt, die nicht wohlgeformt sind, siehe (23-b).

- (23)
- a. that a gun which I had cleaned went off surprised no one
 - b. *that a gun went off surprised no one which I had cleaned

Lösung:

(i) Ross (1967/86) schlug daher vor, dass man in der Syntax Beschränkungen braucht, welche die Macht von Variablen in Transformationen eindämmen.

(ii) Die Beschränkung, die (23-b) blockiert ist der "Right Roof Constraint" von Ross, welcher besagt, dass Extraposition eine Konstituente nur bis zum Ende ihres Satzes bewegen darf, aber nicht ans Ende eines übergeordneten Satzes (siehe auch Perlmutter & Soames 1979, 275ff.).

1.2.3. Phrasenstrukturelle Relationen

Vorweg:

Im folgenden bezeichnen A, B, C, etc. Knoten in einem Baum.

(24) *Dominanz:*

A dominiert B genau dann, wenn es einen uniform abwärtsgerichteten Pfad von A nach B gibt.

Beachte:

Dominanz ist eine transitive Relation zwischen Knoten.

(25) *Transitivität:*

Eine Relation R ist transitiv, wenn aus (25-a) und (25-b) (25-c) folgt.

a. A steht in Relation R zu B.

b. B steht in Relation R zu C.

c. A steht in Relation R zu C.

(26) *Unmittelbare Dominanz:*

A dominiert B unmittelbar genau dann, wenn (a) und (b) gelten.

a. Es gibt einen uniform abwärtsgerichteten Pfad von A nach B.

b. Zwischen A und B liegt kein anderer Knoten C.

(27) *Schwesterschaft*

A ist Schwester von B genau dann, wenn A und B vom selben Knoten C unmittelbar dominiert werden.

Beachte:

Schwesterschaft ist eine symmetrische Relation zwischen Knoten.

(28) *Symmetrie*

Eine Relation R ist symmetrisch, wenn aus (28-a) (28-b) folgt.

a. A steht in Relation R zu B.

b. B steht in Relation R zu A.

(29) *Konstituenz*

A_1, A_2, \dots, A_n bilden eine Konstituente genau dann, wenn (a) und (b) gelten.

a. Es gibt ein C, welches A_1, A_2, \dots, A_n dominiert.

- b. Es gibt kein terminales D, so dass (i) und (ii) gelten.
 - (i) D wird von C dominiert.
 - (ii) D wird nicht von einem der A_i , $1 \leq i \leq n$, dominiert.

1.3. Syntaktisches Argumentieren

Vorweg:

- (i) In empirischen Wissenschaften (wie der Linguistik) gibt es keine beweisbaren Erkenntnisse über den Gegenstand der Forschung (die Sprache).
- (ii) Man kann nur Hypothesen darüber aufstellen, wie die Sprache funktioniert.

Aber:

- (i) Man kann versuchen, Argumente zu geben, die dafür sprechen, dass eine Theorie T_1 besser ist als eine andere Theorie T_2 .
- (ii) Solche Argumente setzen meist gewisse Annahmen voraus und erreichen unter diesen Annahmen bestimmte Schlussfolgerungen.
- (iii) Wenn man aber diese Annahmen nicht akzeptiert, dann ist man auch nicht gezwungen, die Schlussfolgerung zu akzeptieren.
- (iv) Daher kann man linguistische Theorien nicht beweisen.

Beispiel:

Im folgenden wird ein Argument wiedergegeben, das innerhalb der sogenannten Rektions-Bindungstheorie (Chomsky 1981) dafür angeführt wurde, dass die Bindungstheorie (oder ein Teil davon) auf der Oberflächenstruktur appliziert.

- (30) a. You said he liked the pictures that John took Oberfläche
 b. [_{α} How many pictures that John took] did you say he liked Oberfläche

Ausgangsbeobachtungen:

- (i) In (30-a) c-kommandiert das Pronomen *he* die NP *John* auf der Oberflächenstruktur und kann daher nicht mit *John* koreferent sein.
- (ii) Auf der Oberflächenstruktur (30-b) liegt kein solches c-Kommando zwischen *he* und *John* vor; daher kann *he* mit *John* koreferent sein.

Annahmen:

- (i) Neben der Oberflächenstruktur gibt es noch eine weitere syntaktische Repräsentationsebene: die Ebene der logischen Form (LF).
- (ii) Elemente können auf der Ebene der LF an anderen Positionen stehen, als auf der Oberflächenstruktur, ohne, dass man das direkt sehen (bzw. hören) kann.
- (ii) Beispielsweise werden in Mehrfachfragen des Englischen auf LF alle kompletten Fragephrasen an den Satzanfang gestellt, obwohl auf der Oberflächenstruktur nur eine Fragephrase am Satzanfang steht.

- (31) a. Who said he liked [_{α} how many pictures that John took] Oberfläche

b. [_α how many pictures that John took] who said he liked Logische Form

Argument:

- (i) Auf der LF in (31-b) c-kommandiert *he* die NP *John* nicht.
- (ii) Würde die Bindungstheorie also auf der LF-Repräsentation in (31-b) applizieren, dann würde sie Koreferenz von *he* und *John* auf der entsprechenden Oberflächenstruktur (31-a) erlauben.
- (iii) Diese Koreferenz ist aber in (31-a) nicht möglich. Also darf die Bindungstheorie nicht auf LF applizieren, sondern muss auf der Oberfläche applizieren.

Chomsky (1993, 25f.):

“The argument is not conclusive, however. We might reject the last assumption [...]”.
(Chomsky meint hier die Annahme, dass auf LF alle Fragephrasen, also auch die α-Phrase in (31-a), an den Satzanfang gestellt werden.)

Konsequenz:

Wenn man diese Annahme nicht macht, dann hat man kein Argument dafür, dass die Bindungstheorie auf der Oberflächenstruktur angewandt werden muss.

Abstrakte Beispiele für Argumente:

- (i) Gegeben die Annahmen A, B und C, legen die Daten es nahe, dass man die Hypothese D annimmt.
- (ii) Gegeben die Annahmen A, B und C, suggerieren die Daten, dass man die Hypothese D durch die Hypothese E ersetzt.
- (iii) Unter der Annahme von A, B und C sind die Daten sowohl kompatibel mit D als auch mit E.
- (iv) Die Daten machen es unmöglich alle drei Annahmen A, B und C aufrechtzuerhalten. Eine davon muss aufgegeben werden.

Literatur

Chomsky, Noam (1965): *Aspects of the Theory of Syntax*. MIT Press, Cambridge, Massachusetts.

Chomsky, Noam (1981): *Lectures on Government and Binding*. Foris, Dordrecht.

Chomsky, Noam (1993): A Minimalist Program for Linguistic Theory. In: K. Hale & S. J. Keyser, eds, *The View from Building 20*. MIT Press, Cambridge, Massachusetts, pp. 1–52.

Partee, Barbara H., Alice ter Meulen & Robert E. Wall (1990): *Mathematical Methods in Linguistics*. Vol. 30 of *Studies in Linguistics and Philosophy*, Kluwer, Dordrecht.

Perlmutter, David & Scott Soames (1979): *Syntactic Argumentation and the Structure of English*. The University of California Press, Berkeley.

Ross, John Robert (1967/86): Constraints on Variables in Syntax. PhD thesis, MIT, Cambridge, Massachusetts. Appeared in 1986 as: *Infinite Syntax*. Ablex Publishing Corporation, Norwood, New Jersey.