

## 5. Die Theorie der Fragebedeutung von Groenendijk & Stokhof

Groenendijk & Stokhof (1984, 1988, #) haben eine Variante der Propositionsmengen-Theorie der Fragebedeutung entwickelt (vgl. auch Higginbotham & May 1981). Literatur: G&S §6, Bäuerle & Zimmermann §5, G&S 1990: Partitioning Logical Space.

### 5.1 Exhaustive und nicht-exhaustive Fragebedeutung

Die Propositionsmengen-Theorie der Frage nach Hamblin führt zu Fragebedeutungen folgender Art (angenommen, es gibt zwei Personen, Hans und Maria). Dies identifiziert zwei Propositionen direkt und zwei andere auf indirekte Weise.

$$(49) \quad [Wer\ kommt?]=\{\lambda i[x\ kommt\ in\ i] \mid x \in \{Hans, Maria\}\}$$



Bemerke: die Antwort *Maria kommt* schließt nicht aus, dass auch Hans kommt. Das heisst, Antworten in der Theorie von Hamblin sind **nicht exhaustiv**.

Oft werden Antworten jedoch exhaustiv verstanden. Die Antwort *Maria kommt* ist unvollständig, wenn Hans auch kommt, und der Sprecher das weiß. G&S schlagen deshalb vor, daß Fragen die Mengen aller möglichen Welten **partitioniert**, d.h. in nicht-überlappende und die Menge aller Welten ausschöpfende Zellen zerlegt:

$$(50) \quad [Wer\ kommt?]\sim$$

Hamblin-Theorie		G&S-Theorie	
'Hans und Maria kommen.'		'Hans und Maria kommen.'	
'Hans kommt.'	'Maria kommt.'	'Hans kommt.'	'Maria kommt.'
'Niemand kommt.'		'Niemand kommt.'	

Eine Antwort wie *Hans kommt*. ist dann im Zusammenhang der Frage *Wer kommt?* exhaustiv zu verstehen.

Es ist allerdings fragwürdig, ob Antworten immer exhaustiv verstanden werden können (selbst wenn sie nicht, etwa durch progrediente Prosodie oder durch Zusätze wie *zum Beispiel*, als unvollständig markiert werden).

- (51) A: *Wo kann ich die New York Review of Books in Berlin bekommen?*  
B: *Am Bahnhof Friedrichstrasse.*

Eine Alternative zu G&S, die wir im Auge behalten sollten: Fragebedeutungen sind nicht exhaustiv, aus Gründen der pragmatischen Informationsmaximierung werden sie aber oft als exhaustiv verstanden (skalare Implikatur beruhend auf der Konversationsmaxime der Quantität von Grice: Make your contribution as informative as is required.

### 5.2 Fragebedeutungen als Äquivalenzrelationen und Partitionen

Exhaustive Fragebedeutungen werden als Relationen konstruiert:

- (52) a. [*Wer kommt?*]  
b. G&S-Fragebedeutung:  $\lambda j\lambda i[\lambda x[x\ kommt\ in\ i] = \lambda x[x\ kommt\ in\ j]]$

Das ist eine Relation zwischen möglichen Welten  $j$  und  $i$  sodass gilt: Die Menge der Individuen  $x$ , die in  $i$  kommen, ist gleich der Menge der Individuen  $x$ , die in  $j$  kommen, d.h.  $i$  und  $j$  sind in Bezug auf die Menge der Kommenden nicht unterschieden.

Diese Relation ist ein **Äquivalenzrelation**: Sie ist reflexiv, symmetrisch und transitiv. Jede Äquivalenzrelation definiert eine Partition einer Menge. In unserem Beispiel sind zwei Welten  $i, j$  in derselben Zelle einer Partition gdw. wenn sie in der G&S-Fragerelation stehen.

Beispiel Entscheidungsfrage: zwei Welten  $i, j$  stehen in der Relation, wenn sie sich bezüglich des Wahrheitswertes von *Hans kommt* nicht unterscheiden. Dies führt zu derselben Bedeutung wie bei Hamblin.

- (53) a. [*Kommt Hans?*]  
b. G&S-Fragebedeutung:  $\lambda j\lambda i[[Hans\ kommt\ in\ i] = [Hans\ kommt\ in\ j]]$

'Hans kommt.'
'Hans kommt nicht.'

Beispiel mehrfache Ergänzungsfrage: zwei Welten  $i, j$  stehen in der Relation, wenn sie sich bezüglich der Kennen-Relation nicht unterscheiden:

- (54) a. [*Wer kennt wen?*]  
b. G&S-Fragebedeutung:  $\lambda j\lambda i[\lambda y\lambda x[x\ kennt\ y\ in\ i] = \lambda y\lambda x[x\ kennt\ y\ in\ j]]$

Bei 2 Personen erhalten wir bereits eine Partition von 16 Zellen.

### 5.3 Kompositionaler Aufbau von G&S-Fragebedeutungen:

- Aufbau einer strukturierten Fragebedeutung SFB von einer Proposition (bei Ergänzungsfragen, bei Entscheidungsfragen wird die Proposition selbst gewählt).
- Umwandlung in eine Relation zwischen möglichen Welten nach der folgenden Vorschrift:  $\lambda j\lambda i[SFB[i] = SFB[j]]$ , wobei  $i, j$  die Stelle der möglichen Welten in SFB füllt.

Beispiele; der Operator LE macht lediglich das letzte Argument einer Funktion zum ersten (bei G&S nicht nötig, da das Argument der möglichen Welt ohnehin das erste ist):

$$(55) \quad \text{Wenn } R = \lambda X_1 \dots \lambda X_n[\alpha], \text{ wobei } \alpha \text{ nicht funktional, dann } LE(R) = \lambda X_n \lambda X_1 \dots \lambda X_{n-1}[\alpha].$$

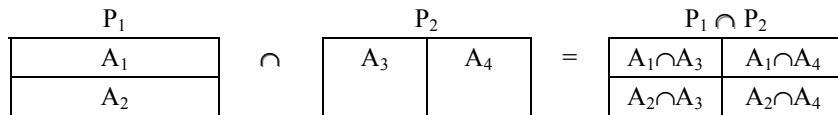
$$(56) \quad \text{Fragebedeutung: } \lambda j\lambda i[LE(SFB)(i) = LE(SFB)(j)]$$

- (57) a. [*Kommt Hans?*]  
 $= \lambda j\lambda i[LE([Hans\ kommt])(i) = LE([Hans\ kommt])(j)]$   
 $= \lambda j\lambda i[[Hans\ kommt](i) = [Hans\ kommt](j)]$
- b. [*Wer kommt?*]  
 $= \lambda j\lambda i[LE([Wer\ \lambda t_1 [t_1\ kommt]])](i) = LE([Wer\ \lambda t_1 [t_1\ kommt]])](j)]$   
 $= \lambda j\lambda i[\lambda i\lambda x \in PERSON[x\ kommt\ in\ i](i) = \lambda i\lambda x \in PERSON[x\ kommt\ in\ i](j)]$   
 $= \lambda j\lambda i[\lambda x \in PERSON[x\ kommt\ in\ i] = \lambda x \in PERSON[x\ kommt\ in\ j]]$

### 5.4 Operationen auf Partitionen und Operationen auf Fragen

Die Partitionen einer Menge genügen mengentheoretischen Beziehungen. Insbesondere gilt, dass der **Schnitt** der Mengen zweier Partitionen wieder eine Partition ergibt:

$$(58) \quad \text{Schnitt zweier Partitionen } P_1 \cap P_2 = \{X \cap Y \mid X \in P_1 \text{ und } Y \in P_2\}$$



Dem Schnitt von Partitionen entspricht die Konjunktion von Fragen:

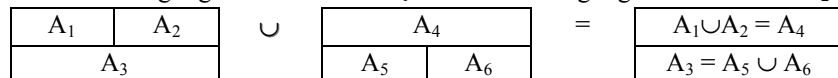
(59) *Wer kommt und wer geht?* = [*Wer kommt?*] ∩ [*Wer geht?*]

Man kann daher jede Partition als einen Schnitt von Zweier-Partitionen verstehen. D.h. man kann jede Ergänzungsfrage als Konjunktion von Entscheidungsfragen verstehen:

- (60) a. *Wer kommt?* ~ *Kommt Hans und kommt Maria?*  
 b. *Peter weiß, wer kommt.* ~ *Peter weiß, ob Hans kommt und ob Maria kommt.*

Man kann ebenfalls eine **Vereinigung** auf Partitionen definieren. Eine Definition im Stile von (58) garantiert allerdings nicht, dass das Resultat wiederum eine Partition ist.

- (61)  $P_1 \cup P_2 =$  die Menge aller Mengen Z, die kleinste nicht-leere Mengen sind, die sich sowohl Vereinigung von Zellen von P<sub>1</sub> wie als Vereinigung von Zellen von P<sub>2</sub> sind.

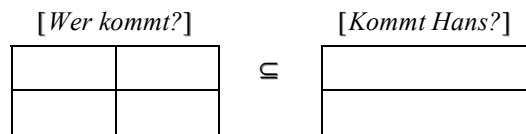


Dieser Vereinigung entspricht aber nicht unmittelbar einer linguistischen Operation. Beispielsweise ergibt die Vereinigung von [*Kommt Hans?*] und [*Kommt Maria?*] nur die tautologische Partition mit einem Element, {I} (I: die Menge aller möglicher Welten).

Man kann eine Relation der **Inklusion** für Partitionen definieren:

(62)  $P_1 \subseteq P_2$  gdw.  $P_1 \cup P_2 = P_2$

Damit kann man den Begriff modellieren, dass eine Frage spezifischer ist als eine andere:

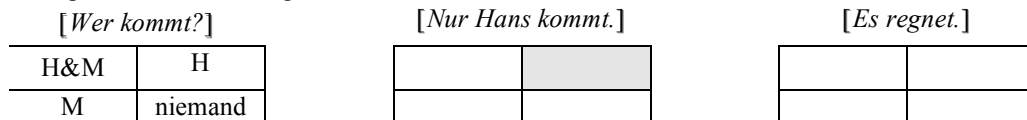


Extreme Partitionen: Neben der tautologischen Partition {I} die totale Partition  $\{\{i\} \mid i \in I\}$ .

In G&S führt eine Frage zu einer Partition der möglichen Welten, d.h. zu einer Vergrößerung der Menge I: Es interessieren nur mehr diejenigen Unterschiede zwischen möglichen Welten, die für die Frage relevant sind. Der tautologischen Partition entspricht einer trivialen Frage, der totalen Partition eine totale Frage (*Wie ist die Welt beschaffen?*).

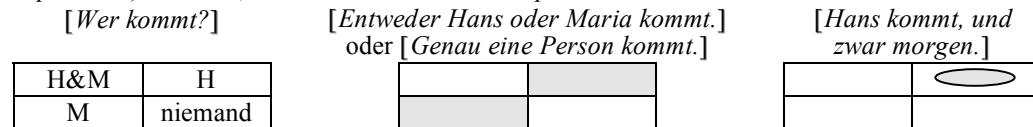
### 5.5 Antworten

Beispiel einer vollständigen Antwort und einer irrelevanten Antwort:



Zwischen diesen Extremen gibt es andere Antwort-Typen: **Partielle** Antworten, welche die Zahl der Zellen reduzieren, und **über-informative** Antworten (G&S: “a proposition that

gives an answer to a question”, im Gegensatz zu “a proposition that is an answer to a question”). Beachte, dass eine Antwort sowohl partiell als auch über-informativ sein kann.



Die Menge der möglichen Welten I ist typischerweise eingeschränkt auf eine Teilmenge I', die den gegenwärtigen Kenntnisstand von Sprecher und Hörer darstellt (**common ground**). Eine Frage ist relevant, wenn sie innerhalb von I' zu einer echten Partition (mit mehr als zwei Zellen) führt.

### 5.6 Exhaustivität

Wie oben bemerkt, nehmen G&S an, dass Antworten primär als exhaustiv interpretiert werden; Nicht-Exhaustivität ist für sie markiert. Sie nehmen daher an, dass Antworten einer Exhaustivierungsoperation unterliegen. Diese Operation wird über Strukturen definiert, in welchen die Kurzantwort identifizierbar ist:

(63) *Wer kommt?* — *Hans.* / *Hans kommt:*  $\forall x[x \text{ kommt} \rightarrow x=\text{Hans}]$

Exhaustivierungsoperation: Eine Operation, die aus einer Menge von Mengen (z.B. einem Quantor) die kleinsten Mengen herausfiltert. Beispiel, in einem nicht-intensionalen Rahmen:

(64)  $EXH = \lambda Q \lambda P(Q(P) \wedge \neg \exists P'[Q(P') \wedge P' \subset P])$

(65) a. *Hans* =  $\lambda P[P(\text{Hans})]$   
 b.  $EXH(\lambda P[P(\text{Hans})])(\text{kommt})$   
 =  $\lambda Q \lambda P[Q(P) \wedge \neg \exists P'[Q(P') \wedge P' \subset P]](\lambda P[P(\text{Hans})])(\text{kommt})$   
 =  $\text{kommt}(\text{Hans}) \wedge \neg \exists P'[P'(\text{Hans}) \wedge P' \subset \text{kommt}]$   
 (D.h. die Menge der Kommenden enthält nur Hans.)

(66) a. *Hans oder Maria* =  $\lambda P[P(\text{Hans}) \vee P(\text{Maria})]$   
 b.  $EXH(\lambda P[P(\text{Hans}) \vee P(\text{Maria})])(\text{kommt})$   
 =  $[\text{kommt}(\text{Hans}) \vee \text{kommt}(\text{Maria})] \wedge \neg \exists P'[[P'(\text{Hans}) \vee P'(\text{Maria})] \wedge P' \subset \text{kommt}]$   
 (D.h. die Menge der Kommenden enthält entweder nur Hans oder nur Maria.)

Diese Analyse der Exhaustivität funktioniert für viele Quantoren, für viele (sog. abwärts-implizierende wie *niemand*, *höchstens zwei Gäste*) jedoch nicht. Vgl. Stechow & Zimmermann 1984 für eine Verallgemeinerung mit Fallunterscheidung von abwärts-implizierenden und aufwärts-implizierenden Quantoren. Ein Problem bleiben gemischte Quantoren wie *zwei Jungen und höchstens drei Mädchen*.

### 5.7 Probleme

G&S stellen eine Mischtheorie vor, in der propositionale Bedeutungen von funktionalen abgeleitet werden. Die Entscheidung, die exhaustive Antwort als grundlegend zu betrachten, ist fragwürdig. Es bleibt unklar, wie exhaustive Bedeutungen kompositional gewonnen werden, und wie der Exhaustionsoperator für bestimmte Quantoren anzuwenden ist.