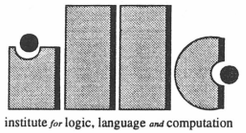
Proceedings of the Tenth Amsterdam Colloquium

December 18 — 21, 1995

Part II

Paul Dekker and Martin Stokhof
(eds.)

ILLC/Department of Philosophy
University of Amsterdam



For further information about ILLC-publications, please contact

Institute for Logic, Language and Computation
Universiteit van Amsterdam
Plantage Muidergracht 24
1018 TV Amsterdam
phone: +31-20-5256090
fax: +31-20-5255101
e-mail: illc@wins.uva.nl

Proceedings of the

Tenth Amsterdam Colloquium

Printing by CopyPrint 2000, Enschede

Cover design by Instand, Malden

ISBN: 90-74795-52-8

Contents

Preface v
Contents vi
The Papers 1

Quantification over Interdependent Variables
Natasha Alechina 1

Mathematical Treatments of Discourse Contexts
Nicholas Asher 21

Modal Correspondence for Models
Jon Barwise and Larry Moss 41

Local Satisfaction Preferred
David Beaver 57

Degree Questions, Maximal Informativeness, and Exhaustivity
Sigrid Beck and Hotze Rullmann 73

Discourse Grammar and Dynamic Logic
Martin H. van den Berg 93

Discourse Structure and Discourse Interpretation
Martin H. van den Berg and Livia Polanyi 113

Predicate Logic Unplugged
Johan Bos 133

The Role of Context in the Interpretation of Generic Sentences
Lawrence Cavedon and Sheila Glasbey 143

Generics and Frequency Adverbs as Probability Judgments
Ariel Cohen 163

Quantifiers, Contexts and Anaphora
Jaap van der Does 183

Aspect and Quantification: an Iterative Approach
Markus Egg 203

An Undecidability Result for Polymorphic Lambek Calculus
Martin Emms 223

Non-Monotonic Consequences of Preferential Contextual Dismbiguation
Tim Fernando 243

A Process Algebraic Approach to Situation Semantics
Tsutomu Fujinami 263

Three Theories of Anaphora and a Puzzle from C. S. Peirce
Brendan S. Gillon 283

Dynamic Epistemic Logic
Willem Groeneveld 299

Semantic Properties of Interrogative Generalized Quantifiers
Javier Gutierrez Rexach 319

Links without Locations. Information Packaging and Non-Monotone Anaphora
Herman Hendriks and Paul Dekker 339

Safety for Bisimulation in General Modal Logic
Marco Hollenberg 359

set aside the problem that $\{\psi : \top \mathrel{|\!\sim} \psi\}$ is infinite, there remains the question of establishing the equivalence

$$(\ddagger) \qquad \varphi \mathrel{|\!\sim} \varphi' \quad \text{iff} \quad \{\psi : \top \mathrel{|\!\sim} \psi\}, \varphi \vdash \varphi' \,.$$

(As usual, $\vdash$ can be extended to $\mathsf{Pow}(\Phi) \times \Phi$ by

$$\Gamma \vdash \varphi \quad \text{iff} \quad (\exists \text{ finite } \Gamma_0 \subseteq \Gamma) \bigwedge \Gamma_0 \vdash \varphi \,,$$

with $\Gamma, \varphi \vdash \varphi'$ written instead of $\Gamma \cup \{\varphi\} \vdash \varphi'$.) It suffices for the forward direction $\Rightarrow$ of $(\ddagger)$ that

$$\varphi \mathrel{|\!\sim} \varphi' \quad \text{implies} \quad \top \mathrel{|\!\sim} \varphi \supset \varphi' \,,$$

which is a special case of the "hard half of the deduction theorem" called rule (S) in Kraus, Lehmann and Magidor [9] (page 191). Avoiding this rule altogether, let us disambiguate c instead as $\{\psi \supset \psi' : \psi \mathrel{|\!\sim} \psi'\}$. But there is still the backward direction $\Leftarrow$ of $(\ddagger)$, for which the relativization c^{δ}_{φ} of the disambiguation of c to φ, given by (C2), is useful. In particular, let us trim down the disambiguation further by setting our sights on the theory

$$\Gamma_\varphi \;=\; \{\varphi \supset \varphi' \;:\; \varphi \mathrel{|\!\sim} \varphi'\} \,,$$

which enjoys the following property.

Lemma 3.[4] *Assuming $\mathrel{|\!\sim}$ verifies* $\mathbf{C}^-$,

$$\varphi \mathrel{|\!\sim} \varphi' \quad \textit{iff} \quad \Gamma_\varphi, \varphi \vdash \varphi'$$

for all $\varphi, \varphi' \in \Phi$.

Proof. Suppose $\varphi \mathrel{|\!\sim} \varphi'$. Then by the definition of Γ_φ, $\Gamma_\varphi \vdash \varphi \supset \varphi'$, and we are done by the deduction theorem for $\vdash$. The converse requires a bit more work. Assume $\Gamma_\varphi, \varphi \vdash \varphi'$. That is, there are $\varphi_1, \ldots, \varphi_n$ such that

$$(*) \qquad \varphi \;\mathrel{|\!\sim}\; \varphi_i \quad \text{for every } i \in \{1, \ldots, n\} \,,$$

4. This is the key lemma, corresponding to Lemma 3.18 in Kraus, Lehmann and Magidor [9]. It seems that (LLE) is unnecessary, unless, as suggested in §3.3, (Cut) is replaced by the preferential rule (Or) described there.

and

$$(**) \qquad \bigwedge_{i=1}^{n} (\varphi \supset \varphi_i) \wedge \varphi \;\vdash\; \varphi' \,.$$

Now, $(**)$ implies

$$(\bigwedge_{i=1}^{n} \varphi_i) \wedge \varphi \;\vdash\; \varphi' \,,$$

whence (since $\vdash$ is contained in $\mathrel{|\!\sim}$)

$$(\bigwedge_{i=1}^{n} \varphi_i) \wedge \varphi \;\mathrel{|\!\sim}\; \varphi' \,.$$

Also, by the rule (And), it follows from $(*)$ that

$$\varphi \;\mathrel{|\!\sim}\; \bigwedge_{i=1}^{n} \varphi_i \,.$$

Thus, (Cut) yields $\varphi \mathrel{|\!\sim} \varphi'$, as required. $\dashv$

Next, let us attend to the problem that the set Γ_φ is infinite. This is where Δ and $\prec$ come to play. Assuming Φ is countable, fix an enumeration $\{\psi_0, \psi_1, \ldots\}$ of Φ, and build a set $\Delta = \{\delta_0, \delta_1, \ldots\}$ of disambiguations, where (for every $\varphi \in \Phi$) δ_i approximates Γ_φ up to $\{\psi_0, \ldots, \psi_i\}$ in the sense that

$$c^{\delta_i}_{\varphi} \;=\; \bigwedge\{\varphi \supset \varphi' \;:\; \varphi \mathrel{|\!\sim} \varphi' \text{ and } \varphi' \in \{\psi_0, \psi_1, \ldots, \psi_i\}\} \,.$$

Arrange

$$\delta_i(\varphi, \varphi') \;=\; \varphi'$$

for every i and for all $\varphi, \varphi' \in \Phi$. Then define $\prec$ by

$$\delta_i \prec \delta_j \quad \text{iff} \quad i > j \,.$$

Let $\vdash_\prec$ be $\vdash^{\mathcal{D}_\prec}$, where $\mathcal{D}_\prec$ is the family of stably $\prec$-dense sets.

Theorem 4 (Representation). *Let Φ be countable, and $\mathrel{|\!\sim} \subseteq \Phi \times \Phi$ verify* $\mathbf{C}^-$. *Then for Δ and $\prec$ constructed as above,*

$$\varphi \mathrel{|\!\sim} \varphi' \quad \textit{iff} \quad c^{\wedge}\varphi \vdash_\prec \varphi'$$

for all $\varphi, \varphi' \in \Phi$. *Furthermore,* $(\Delta, \prec)$ *is* $\mathbf{C}^-$*-sufficient. Similarly, for* $\mathbf{C}$.

Proof. For the equivalence, it suffices by Lemma 3 to establish that

$$\Gamma_\varphi, \varphi \vdash \varphi' \quad \text{iff} \quad c^\wedge\varphi \vdash_\prec \varphi' ,$$

which is easy because $\prec$ is a linear order, and whenever $\delta \prec \delta'$, $c^\delta_\varphi \vdash c^{\delta'}_\varphi$.

Turning to $\mathbf{C}/\mathbf{C}^-$-sufficiency, use Proposition 1 and the definition of c^δ_φ for parts (i) and (ii)[5] respectively of $\mathbf{C}^-$-sufficiency. Part (iii) and the additional condition for $\mathbf{C}$-sufficiency are proved by reversing the argument for soundness. In particular, observe that

$$\frac{\{\delta \in \Delta \ : \ c^\delta_\varphi \wedge \varphi \vdash \varphi'\} \in \mathcal{D}_\prec}{\{\delta \in \Delta \ : \ c^\delta_\varphi \wedge \varphi \vdash c^\delta_{\varphi\wedge\varphi'}\} \in \mathcal{D}_\prec}$$

follows from (Ref), (Cut), (And) and the definition of c^δ_φ, while

$$\frac{\{\delta \in \Delta \ : \ c^\delta_\varphi \wedge \varphi \vdash \varphi'\} \in \mathcal{D}_\prec}{\{\delta \in \Delta \ : \ c^\delta_{\varphi\wedge\varphi'} \wedge (\varphi^\wedge\varphi') \vdash c^\delta_\varphi\} \in \mathcal{D}_\prec}$$

is a consequence of (Ref), (CM), (And) and the definition of c^δ_φ. ⊣

3.3 Extensions

Insofar as neither Δ nor $\prec$ need mention models, it is perhaps unsuitable to call Theorem 4 a "completeness theorem." Moreover, §3.2 is hardly "complete" in that there are at least two obvious directions in which to extend the analysis. Further rules might be added to $\mathbf{C}^-$, and the set Φ might be expanded to cover more of E. Let us take these up briefly, in turn.

The equivalence stated in Theorem 4 can be maintained for any extension of $\mathbf{C}^-$ by rules (over the same set Φ). If necessary, the rules can be translated mechanically into conditions on Δ and $\prec$. Take, for instance, what Kraus, Lehmann and Magidor [9] call *preferential reasoning*, which is characterized by adding to $\mathbf{C}$ the rule

$$\text{(Or)} \qquad \frac{e_1 \mathrel{|\!\sim} e \qquad e_2 \mathrel{|\!\sim} e}{e_1 \vee e_2 \mathrel{|\!\sim} e} .$$

5. Observe that the reversal of (LLE) would not have been possible had $c^{\delta_i}_\varphi$ been defined instead as $\bigwedge(\Gamma_\varphi \cap \{\psi_0, \ldots, \psi_i\})$.

Put in terms of $(\Delta, \prec)$, this rule becomes simply

$$\text{(OR)} \qquad \frac{\{\delta \in \Delta \ : \ c^\wedge e_1 \vdash^\delta e\} \in \mathcal{D}_\prec \qquad \{\delta \in \Delta \ : \ c^\wedge e_2 \vdash^\delta e\} \in \mathcal{D}_\prec}{\{\delta \in \Delta \ : \ c^\wedge(e_1 \vee e_2) \vdash^\delta e\} \in \mathcal{D}_\prec} ,$$

and if we are content with such a "syntactic" requirement on $(\Delta, \prec)$, we need work no further for a representation of $\mathbf{P}$. (A parallel here might be drawn with the manner in which the completeness theorem for first-order logic applies to different first-order theories.) Of course, a little labor is always good, and a simple restatement of (OR) might be illuminating. Or perhaps with a bit more effort, we might even strengthen (OR) by a sufficient condition met by the model constructed in the proof of Theorem 4, adapted to $\mathbf{P}$.[6] Following a suggestion of D. Lehmann's (made in a conversation with me) that (Or) be regarded as a basic rule, consider the system $\mathbf{P}^-$ obtained from $\mathbf{P}$ by replacing (CM) by (And). As proved in page 191 of Kraus, Lehmann and Magidor [9], $\mathbf{P}^-$ enjoys *cut-elimination* — which is to say that $\mathbf{P}^-$ can be redefined as $\mathbf{C}^-$ with (Cut) replaced by (Or)

$$\mathbf{P}^- \ = \ \{\text{(Ref), (LLE), (RW), (And), (Or)}\} .$$

Now, improving on condition (iii) of $\mathbf{C}^-$-sufficiency, define a pair $(\Delta, \prec)$ to be $\mathbf{P}^-$*-sufficient* if

(i) $\prec$ is transitive,

(ii) for all $\varphi, \varphi' \in \Phi$, $\delta \in \Delta$ and $e \in E$,

$$\frac{\varphi \equiv \varphi'}{\delta(\varphi, e) \equiv \delta(\varphi', e)}$$

and

(iii) for all $\varphi_1, \varphi_2, \in \Phi$, and $\delta \in \Delta$,

$$\delta(\top, c^\wedge(\varphi_1 \vee \varphi_2)) \ \vdash \ \delta(\top, c^\wedge\varphi_1) \vee \delta(\top, c^\wedge\varphi_2) .$$

The arguments in §§ 3.1 and 3.2 can be modified easily to prove

6. Notice, for instance, that condition (C3) (in the definition of $\mathbf{C}^-$-sufficiency) is stronger than the mechanical translation of (LLE), but that we have taken pains to construct, in the proof of Theorem 4, an interpretation $(\Delta, \prec)$ satisfying (C3).

Theorem 5. *For every* $\mathbf{P}^-$*-sufficient pair* $(\Delta, \prec)$*, the relation* $\{(\varphi, \varphi') \in \Phi \times \Phi : c^\frown\varphi \vdash_\prec \varphi'\}$ *verifies* $\mathbf{P}^-$*. Conversely, if* Φ *is countable, then for every* $\mathrel{|\!\sim} \subseteq \Phi \times \Phi$ *verifying* $\mathbf{P}^-$*, there is a* $\mathbf{P}^-$*-sufficient pair* $(\Delta, \prec)$ *such that*

$$\varphi \mathrel{|\!\sim} \varphi' \quad \textit{iff} \quad c^\frown\varphi \vdash_\prec \varphi'$$

for all $\varphi, \varphi' \in \Phi$*.*

A second direction in which to push the present analysis of $\mathrel{|\!\sim}$ is to enlarge the set of formulas to E. This is, in my mind, no mean enterprise, requiring as it does some fundamental reflection on the nature of ambiguity, and on the plausibility that $\vdash^{\mathcal{D}}$ should be "logical" (in any recognizable sense). Two assumptions made in the present paper, for instance, deserve more careful consideration: (i) the claim that the "partial" character of (the "actual" process of) disambiguation can be modeled through sets of total disambiguations, and (ii) the assumption that $\mathcal{D}$ is closed under intersections. In confining its attention to the disambiguation of a rogue context c, the present paper takes the tiniest bite at the monster that is ambiguity; the reader with an appetite for more is referred to Fernando [4].

4 Discussion

The basic hypothesis about non-monotonicity explored above is, very briefly, that non-monotonicity arises from the non-persistent interpretation (— identified here with disambiguation —) of implicit (background) conditions. To help us digest that point, let us try to put things in their proper context.

4.1 Contexts: between models and formulas

Few would dispute the claim that contexts are partial, but just what the consequences of this claim are varies widely from researcher to researcher. For the approach developed above, it is crucial that this partiality *not* be confused with uncertainty *about* contexts, as such a distinction is necessary in formulating the following principle of *persistence*

$$\frac{\varphi \vdash \varphi'}{\delta(\varphi, e) \vdash \delta(\varphi', e)} \,.$$

The failure (above) of persistence gives rise to the essential asymmetry between the rules (LLE) and (RW) of Kraus, Lehmann and Magidor [9] (— an asymmetry that all other rules can be viewed as being devised to diminish). In fact, whereas the rule (RW) is, in the present work, an immediate consequence of the innocuous assumption that only "large" sets can contain "large" sets, the rule (LLE) must, weak as it is, be introduced through condition (C3). (C3) can be restated (as in the definition of $\mathbf{P}^-$-sufficiency) as

$$\frac{\varphi \equiv \varphi'}{\delta(\varphi, e) \equiv \delta(\varphi', e)} \,,$$

asserting that, so far as disambiguation is concerned, an unambiguous formula matters only up to its set of models. That is, a perfectly finitary formula can be replaced by an infinite set (if not proper class) of its possibly infinite models. Of course, only the most rabid semanticist could (shamelessly) describe such a move as a "reduction." On the other hand, there is no denying the widespread bias for models over formulas, manifested, for instance, by the fact that preferences were initially defined on models (rather than on formulas). We may well expect preferences on infinite sets of formulas (corresponding to theories of models) to be reducible to preferences on (single) formulas (corresponding to basic open sets of models), if only because compactness has become so familiar.[7] But defining preferences between states "labeling" the same sets of formulas (as in the preferential models constructed in §5.3 of Kraus, Lehmann and Magidor [9]) is a different matter altogether, requiring a shift in perspective. Staying with our interpretation of $\mathrel{|\!\sim}$ in terms of Δ and $\prec$, it perhaps bears repeating that in view of the dispensability of models to that interpretation, it is more appropriate to describe Theorem 4 as a "representation theorem," rather than a "completeness theorem." Indeed, I conceived Theorem 4 originally to serve a certain *representationalist* end, establishing the eliminability of models (poorly suited, as they are, for mechanization).

7. Passing to effectively presented sets, there is also Kleene's theorem stating that a recursively enumerable first-order theory is equivalent to a first-order sentence, allowing an expansion in vocabulary.

4.2 Discourse representation

Although a model-theoretic interpretation of a notion $\vdash \subseteq \Phi \times \Phi$ of consequence allows formulas (in Φ) to be replaced by models, it need not imply that $\vdash$ is classical. In the case of *Discourse Representation Theory* (DRT, Kamp and Reyle [8]), for instance, an equivalence between model-theoretic and syntactic accounts of Φ was more or less obvious before a classical consequence relation $\vdash$ was isolated. The simple reason was that the model-theoretic interpretation commonly associated with the DRT merge $^\wedge$ is not commutative. More precisely, a certain first-order fragment Φ of DRT can be supplied a semantics $[\![\cdot]\!]$, under which

(i) every $\varphi \in \Phi$ is interpreted as a binary relation $[\![\varphi]\!] \subseteq S \times S$ on a certain fixed set S of states (constructed from models), with the intuition that

$$s[\![\varphi]\!]s' \quad \text{iff} \quad \text{given an initial context } s, \varphi \text{ can return the context } s'$$

for all $s, s' \in S$,

and

(ii) $^\wedge$ is relational composition

$$[\![\varphi^\wedge\varphi']\!] \;=\; \{(s,s') \in S \;:\; (\exists s'')\; s[\![\varphi]\!]s'' \text{ and } s''[\![\varphi']\!]s'\}\;,$$

for all $\varphi, \varphi' \in \Phi$.

Nevertheless, it turns out that the relational semantics $[\![\cdot]\!]$ is determined by $^\wedge$ and a set $\Phi_\perp \subseteq \Phi$ of "absurd" formulas (Fernando [2]). The situation is completely analogous to the way conjunction and negation determine Boolean connectives. Hence, it is perhaps not terribly surprising that, for this case, relational semantics can be reconciled with Boolean-valued semantics (Fernando [3]). That is to say, a classical notion $\vdash$ of consequence can be defined for an unambiguous fragment Φ of DRT.

Far more interesting, however, are parts of DRT that go beyond Φ. Two examples are *Segmented* DRT (SDRT, Asher [1]) and treatments of plurals. In SDRT, a non-monotonic entailment relation $\mid\!\sim_{KB}$ relative to some knowledge base KB lies at the heart of the discourse interpretation process (involving so-called discourse relations). The idea is to put in the lefthand side of $\mid\!\sim_{KB}$ not only the natural language utterance to be interpreted, but also the "common ground" — that is, the discourse context. Now, it is natural

to hypothesize that "conversational leaps" in the common ground give rise to the rogue context c (above). In particular, the presupposition associated with the utterance of a sentence φ at context K might be expressed as c so that

$$K, \varphi \mid\!\sim_{KB} K' \quad \text{iff} \quad K, c, \varphi \vdash_{KB} K'$$

for all (candidate consequences) K'. As for plural statements, ambiguity is a notorious problem, motivating, for example, the work of Reyle [13]. And whether or not the difficulty is labeled to be one of "ambiguity," interpretation does not become any easier where there is no explicit quantification, as in the case of

```
Birds fly.
```

In contrast to the case of $\mid\!\sim_{KB}$, it would seem useful to formulate such "bare plurals" as conditional expressions, rather than as judgments about $\mid\!\sim$. In any event, a closer investigation of $\mid\!\sim$, including its relation with conditionals, is clearly called for.

4.3 Smoothness, optimal disambiguations and generics

The assumption in abduction that there is a most preferred explanation, the *limit assumption* in Stalnaker [15], and the "smoothness condition" in Kraus, Lehmann and Magidor [9] correspond in the present approach to the following notion. A pair $(\Delta, \prec)$ is said to be *smooth at* $(\varphi, e) \in \Phi \times E$ if

$$(\forall \delta \in \Delta)(\exists \delta' \preceq \delta)(\forall \delta'' \preceq \delta') \qquad \delta'(\varphi, e) \equiv \delta''(\varphi, e)\;.$$

That is, there is a set $\Psi_{\varphi,e} \subseteq \Phi$ such that $\{\delta \in \Delta : (\exists \psi \in \Psi_{\varphi,e})(\forall \delta' \preceq \delta)\ \delta'(\varphi, e) \equiv \psi\}$ is stably $\prec$-dense. (Note that the particular pair $(\Delta, \prec)$ constructed in §3.2 is smooth at (φ, c) iff Γ_φ is equivalent to a formula.) Specializing to the case $e = c$ relevant to our analysis of $\mid\!\sim \subseteq \Phi \times \Phi$, call a pair $(\Delta, \prec)$ *c-smooth* if it is smooth at (φ, c), for every $\varphi \in \Phi$.

Proposition 6. *If $(\Delta, \prec)$ is c-smooth, then for every $\varphi \in \Phi$, there is a set $\Psi_\varphi \subseteq \Phi$ of unambiguous formulas such that*

$$c^\wedge\varphi \vdash_\prec \varphi' \quad \textit{iff} \quad (\forall \psi \in \Psi_\varphi)\ \psi \vdash \varphi \supset \varphi'\;,$$

for every $\varphi' \in \Phi$.

Proposition 6 not only reduces $\mid\!\sim$ to material implication, but also characterizes the background condition as a disjunction $\bigvee \Psi_\varphi$ (which can be formalized, assuming the disjunction can be formed; e.g., if Ψ_φ is finite). Observe that the background condition is defined relative to the antecedent φ, and that the more general case where antecedents and/or succedents might be ambiguous requires a notion of an "optimal disambiguation" that checks every ambiguous expression (not just c). Whether or not optimal choices at every pair (φ, e) (for all $e \in E$ as well as $\varphi \in \Phi$) can be assembled in a disambiguation presumably depends on the application at hand.

Returning to the simple case of c, the following example is instructive, particularly for a comparison with Kraus, Lehmann and Magidor [9].

Example. Let Φ be the set of propositions generated by a countable set $\{p_0, p_1, \ldots\}$ of propositional variables, and let $\mathcal{L}$ be the first-order language induced by a countable set $\{U_0, U_1, \ldots\}$ of unary relation symbols. Fix a constant symbol a, and define the translation $\cdot^a$ from Φ to $\mathcal{L}(a)$ as follows

$$\begin{aligned} {p_n}^a &= U_n(a) \\ (\varphi \wedge \psi)^a &= \varphi^a \wedge \psi^a \\ (\neg\varphi)^a &= \neg(\varphi^a) \, . \end{aligned}$$

Next, let M be the $\mathcal{L}$-model with universe ω $(= \{0, 1, \ldots\})$ that interprets U_n as $\{m \in \omega : m \geq n\}$, for every $n < \omega$. Form the $\mathcal{L}(a)$-theory T by adding to the $\mathcal{L}$-theory of M the $\mathcal{L}(a)$-sentences $U_n(a)$, for every $n < \omega$:

$$T \;=\; \{\chi \in \mathcal{L} : M \models \chi\} \;\cup\; \{U_0(a), U_1(a), \ldots\}$$

(which is consistent, by compactness). Define

$$\varphi \mid\!\sim \psi \quad \text{iff} \quad T \vdash_{\mathcal{L}(a)} \varphi^a \supset \psi^a \, ,$$

for all $\varphi, \psi \in \Phi$, and observe that $\mid\!\sim$ satisfies **C** (including (CM)), but cannot be captured by a c-smooth pair $(\Delta, \prec)$, since T is not satisfiable in M (but only finitely satisfiable there). [End of Example]

By contrast, note that in Kraus, Lehmann and Magidor [9], a similar "smoothness condition is necessary to ensure the validity of Cautious Monotonicity" (page 182). A notion central to the representation arguments there is that

of a *normal model of* φ, by which is meant a model of $\{\varphi' : \varphi \mid\!\sim \varphi'\}$. This leads to so-called *strongly cumulative models* (the essential property of which is that every antecedent φ has a minimum state), because normal models contain infinite information, whereas disambiguations only encode single formulas (in values assigned to pairs $(\varphi, e) \in \Phi \times E$). As it turns out, it is fruitful to introduce *types* (as they are known in model theory) to express more information than can be packaged in single first-order formulas. The question is, however, what are these types *of*? As has already been mentioned, preferences in Kraus, Lehmann and Magidor [9] can be defined between states labeling the same sets of formulas. (This, at least, is the case for the preferential models in §5.3, though not, as noted in Voorbraak [16], for the cumulative models in §3.5.) In conceptual terms, what do such states reflect? Toward an answer, Fernando [5] treats these states as objects in a first-order model, explores a notion of *generic* object (or better: *type*) that reduces defeasibility to material implication, and presents a proposal relating entailment $\mid\!\sim$ to a connective producing conditional expressions. (At the risk of sounding obscure, suffice it to say that what is gained by introducing a preferential relation is the possibility of *omitting the type*.)

Acknowledgments

My thanks to Nicholas Asher, Hans Kamp, Emil Weydert and Daniel Lehmann (who suggested $\mathbf{P}^-$ in §3.3, among other things) for helpful discussions.

References

[1] Nicholas Asher. *Reference to Abstract Objects in Discourse.* Kluwer Academic Publishers, Dordrecht, 1993.

[2] Tim Fernando. What is a DRS? In R. Cooper and J. Groenendijk, editors, *Integrating Semantic Theories II.* Dyana deliverable R2.1.B, 1994. Presented in a Computational Semantics workshop at Tilburg (December 1994).

[3] Tim Fernando. A persistent notion of truth in dynamic semantics. In D. Westerståhl and J. Seligman, editors, *Language, Logic and Computation: The 1994 Moraga Proceedings.* CSLI, Stanford, To appear.

[4] Tim Fernando. Ambiguity under changing contexts. Manuscript, 1995. Presented in *Mathematics of Language 4*, Philadelphia, October 1995.
[5] Tim Fernando. Defeasible generic types. Manuscript, 1995.
[6] Dov M. Gabbay. Theoretical foundations for non-monotonic reasoning in expert systems. In K.R. Apt, editor, *Proc. NATO Advanced Study Institute on Logics and Models of Concurrent Systems*. Springer Verlag, Berlin, 1985.
[7] J. Hobbs, M. Stickel, D. Appelt, and P. Martin. Interpretation as abduction. *Artificial Intelligence*, 63, 1993.
[8] H. Kamp and U. Reyle. *From Discourse to Logic*. Kluwer Academic Publishers, Dordrecht, 1993.
[9] S. Kraus, D. Lehmann, and M. Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence*, 44, 1990.
[10] David Makinson. General theory of cumulative inference. In M. Reinfrank, J. de Kleer, M.L. Ginsberg, and E. Sandewall, editors, *Proc. Second International Workshop on Non-monotonic Reasoning*, Lecture Notes in Computer Science. Springer Verlag, Berlin, 1989.
[11] Charles Sanders Peirce. Abduction and induction. In J. Buchler, editor, *Philosophical Writings of Peirce*. Dover Books, New York, 1955.
[12] Raymond Reiter. A logic for default reasoning. *Artificial Intelligence*, 13, 1980.
[13] Uwe Reyle. Dealing with ambiguities by underspecification: construction, representation and deduction. *Journal of Semantics*, 10(2), 1993.
[14] Yoav Shoham. *Reasoning about Change*. MIT Press, Cambridge, MA, 1988.
[15] Robert Stalnaker. A theory of conditionals. In *American Philosophical Quarterly Monograph Series 2*. Blackwell, Oxford, 1968.
[16] Frans Voorbraak. Preference-based semantics for non-monotonic logics. In *Proc. 13th IJCAI*. Morgan Kaufmann, San Mateo, CA, 1993.

A Process Algebraic Approach to Situation Semantics

Tsutomu Fujinami (IMS, Universität Stuttgart)

Abstract: We propose a way to base Situation Semantic to a computational ground of concurrency and to linear logic. One of the core ideas of Situation Semantics is ecological realism, the idea that meaning arises from the interaction between a cognitive agent and his/her environments. We model both the agent and environments as a process and study the interaction as a system of communicating processes. We turn to the π-calculus to construct semantic objects employed in Situation Theoretic Discourse Representation Theory (*ST-DRT*). The construction helps us relate *ST-DRT* with linear logic, through our translation of the calculus to a combinatory intuitionistic linear logic. The multiplicative conjunction then enables us to build up various semantic objects as a theory. By conceiving of linear logic as a theory of information flow, we can establish a connection between Channel Theory and *ST-DRT*.

1 Introduction

Process algebra has been being developed in Computer Science to study communication and concurrency.[1] The paper proposes one way to apply the technique to natural language semantics. Philosophically, the approach may be justified as a development from Situation Theory [Barwise and Perry 1983]. It is argued in the theory that the meaning of sentences is the relation between the situation where the sentence is uttered and the situation it describes. Channel Theory, a recent development from Situation Theory, enables us to refine the relation to a pair of notions, *connection* and *constraint* [Barwise 1993, Seligman and Barwise 1993]. That is, the relation between a particular utterance and the situation described can be regarded as a connection. As a constraint, of which the connection is an instance, we can take the relation between the sentence type of the utterance and the situation type of the described situation. Statically, classifying the connection as an instance of a constraint is enough to determine the meaning of a sentence, but here we are also interested in the operational aspect, that is, how the connection can be classified by an agent.

To investigate the aspect, there are several points to be considered. First of all, the model must allow to capture the meaning of multimodal expressions such as utterances accompanied by a picture or figure. Recall Situation Theory claims that natural language is *a* way to convey the meaning [Barwise and Perry 1983]. The theory of meaning must be general so that it can capture information flow by various means, not limited to natural language. The other point is to do with ecological realism, the idea that the meaning arises from the interaction between a

[1] See for example, [Milner 1989].

cognitive agent and his/her environments. Given that various information sources are available in dialogue, the agent can interact with them in parallel to extract information. Based on the observation, we argue that process algebra is useful to model information flow in which utterances are involved.

The algebra is expressive enough to model many aspects of linguistic actions such as parsing, interpretation, and evaluation,[2] but before exploiting the possibilities we are concerned about how such an enterprise can be related with other theories of natural language semantics. To answer the question, the modelling provides us with an operational model for semantic objects employed in various theories. The central issue for semantics has been to develop frameworks to describe the meaning of sentences, while less attentions have been paid to their implementations. By 'implementation', we do not mean to implement a program using a particular programming language such as *prolog* or *Lisp*. We are rather interested in identifying a certain class of computations needed to model a theory and investigating their properties, abstracted away from actual implementations. In the paper, we propose a way to construct semantic objects as a system of communicating processes. Semantic objects are then regarded as a (declarative) definition of such a system. Suppose we have a semantic object, α, in our theory and implement it as a system of processes, P. To express the relation, we write $P \models \alpha$ and read it as P *satisfies* α. The first half of the paper is devoted to working out the relation, where we turn to the π-calculus [Milner et al. 1992, Milner 1993] for model and Situation-Theoretic Discourse Representation Theory(*ST-DRT*) [Cooper 1993a, Cooper 1993b] for semantic theory.

The construction of semantic objects employed in *ST-DRT* can not be completed only with the π-calculus. Obviously, logical connectives and quantifiers are missing in the algebra, for which we have to base our construction to some logic. For the purpose, we turn to linear logic [Girard 1987], especially to its intuitionistic and combinatory version [Lafont 1988], and propose one way to define our model in the logic. The grounding in turn benefits us for its close connection with Channel Theory. It will be turned out that, as a theory of information flow, they share much in common. The grounding is also useful to investigate computational properties of semantic objects because the logic has been studied well in Computer Science.

The paper consists of four parts. The first part explains our view on semantic objects modelled as processes (§2), the second part presents the construction of semantic objects (§3), and the third part shows the grounding of the construction to linear logic (§4). We will discuss finally how the logic can be related with Channel Theory (§5). The paper however remains to be a sketch due to limited

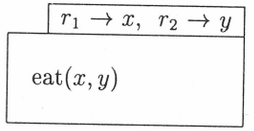
[2]Throughout the paper, *interpretation* means to construct semantic representations based on syntactic information, and *evaluation* to relate a particular semantic representation to (a part of) the world.

space. For more detail, the reader is invited to consult the author's dissertation [Fujinami 1995].

2 Semantic objects modelled as processes

2.1 Situation-Theoretic *DRT*

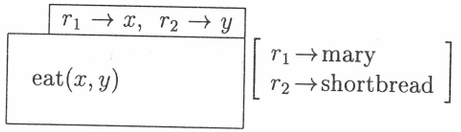
To describe the meaning, the approach taken in *ST-DRT* is to capture it as a sort of function. For instance, the meaning of a sentence, *"Mary eats a piece of shortbread."*, can be described in *ST-DRT* graphically as:[3]



(1)

or in linear notation as $\lambda[r_1 \to x, r_2 \to y]\langle\langle \text{eat}, x, y; 1\rangle\rangle$. The formula, $\langle\langle \text{eat}, x, y; 1\rangle\rangle$, expresses an item of information composed of three elements with a polarity, 1, indicating the item as positive. The first element denotes a relation of eating, the second eater, and the third foods. The second and third elements are however yet to be filled with some concrete objects, thus parameterised with x and y, respectively. We call such an object *parametric infon*. The object must be formalised mathematicaly and is regarded as an abstract object called *infon abstract*, abstracted over x and y by adopting the λ-abstraction. We can therefore conceive of it as function, but it is more flexible in that abstractions w.r.t. x and y can occur simultaneously. That is, it does not matter in which order the object is abstracted over, e.g., either x then y or y then x. To compensate it, another object called *role* is introduced to index parameters. In the infon abstract (1), roles are r_1 and r_2.

In accordance with the extension, the assignment to parameters must be defined with roles, too. Suppose we anchor x to a particular person, 'mary', and y to a particular piece of biscuit, 'shortbread'. The anchoring could be expressed as below, where 'mary' is indexed with r_1 and 'shortbread' with r_2:



(2)

[3]For simplicity, we disregard the information that x is named Mary and y is a piece of shortbread. As for the graphical representation, the reader is also referred to [Barwise and Cooper 1991,Barwise and Cooper 1993].

When applied to it, the infon abstract (1) turns into another object called *infon* such as:

$$\boxed{\text{eat(mary, shortbread)}} \tag{3}$$

which expresses an item of information such that a particular person called Mary eats a particular piece of shortbread.

2.2 Processes as generalised function

The extension of functional calculus by Situation Theory is linguistically motivated. Robin Cooper [Cooper 1993a], for example, proposes to use utterances as roles. Pushing the idea further, we can include other factors contributing towards indexing such as referring actions as a composite of roles, where roles become more complex objects. What is suggested through these extensions is that the way a semantic object is abstracted over and anchored to environments could be more lengthy and complex than we hope to capture as function. Thus, backing to our intuition, we recapture the way an agent extracts the meaning from an utterance as a *process* rather than function.

Let us reinterpret the infon abstract with the assignment (2) as a process. We regard the roles, r_1 and r_2, as a port through which items of information can be exported and imported (Figure 1). For the infon abstract, the roles are used to import the individuals, 'mary' and 'shortbread', while they are used to export them for the assignment.

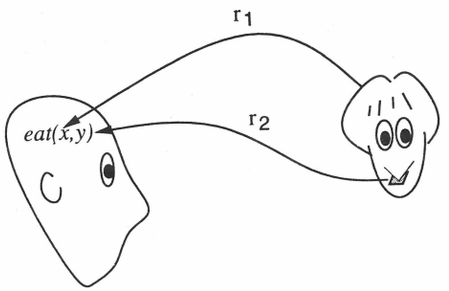


Figure 1: Roles as an information port

In the π-calculus, the action to import an item of information through the port, r_1, to replace a parameter, x, is expressed as $r_1(x)$. Let P be the state of the

process after importing an item. The change caused to the process by the action can then be expressed as $r_1(x).P$, where '.' indicates sequential order of the states. On the other hand, the action to export the item, 'mary', is expressed as $\overline{r_1}\langle\text{mary}\rangle.\mathbf{0}$, where the overline indicates it is the dual of the input action. The symbol, $\mathbf{0}$, means termination and can be often suppressed for readability. When these two processes are concurrently active, the individual can be imported to P upon the interaction through the port. As the result, the parameter, x, is substituted by 'mary', whose transition we express as:

$$\begin{array}{rrl} & r_1(x).P & | \quad \overline{r_1}\langle\text{mary}\rangle.\mathbf{0} \\ \longrightarrow & P\{^{\text{mary}}/_{x}\} & | \quad \mathbf{0} \end{array}$$

where $\{^{\text{mary}}/_{x}\}$ expresses the substitution environment such that 'mary' substitutes x. In this formulation, the process, P, can be said to be abstracted over x indexed with r_1. We may even express it as $\lambda[r_1 \rightarrow x].P$, adopting the syntax of *ST-DRT*. As hinted with the example, the process algebra can be regarded to be a generalisation of functional calculi[4] and can be a useful foundation to study *ST-DRT*.

2.3 Process algebraic *ST-DRT*

The flexibility provided by the algebra enables us to model the complex cases of abstractions and anchorings more naturally than employing composite roles. Suppose for example we want to express the meaning of the sentence, *"Mary eats a piece of shortbread."*, as an abstract object whose parameters x and y are substituted by individuals determined jointly by utterances and referring actions. In *ST-DRT*, such an object may be expressed as:

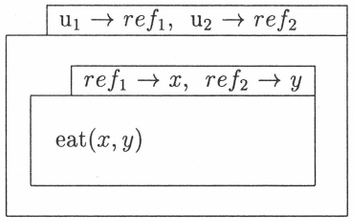
$$\boxed{\begin{array}{|l} \langle \text{ref}_1, \text{u}_1\rangle \rightarrow x,\ \langle \text{ref}_2, \text{u}_2\rangle \rightarrow y \\ \hline \text{eat}(x, y) \end{array}} \tag{4}$$

where $\langle \text{ref}_1, \text{u}_1\rangle$ is a composite role, composed of a referring action, ref_1, and an utterance, u_1. The parameter, x, may be substituted by an individual determined by an utterance of *"Mary"* and the reference to her at that time.

When we regard the object as a process, we can further analyse the relation between referring actions and utterances. Suppose we regard the referring action, ref_1, as a parameter to be substituted by a role imported upon the utterance of

[4] See, for example, [Milner 1992].

"Mary", u_1, and the parameter, x, will be substituted by an individual imported through the role. By extending the notation of *ST-DRT*, it can be expressed as:



(5)

This is a clarification of the assumption that an utterance can convey a reference and can be expressed in the calculus as a process such as $u_1(ref_1).ref_1(x).P$.[5] When the process is accompanied by other processes providing it with a role, r_1, through u_1, and the individual, 'mary', through r_1, it can get access to it as follows:

$$\begin{array}{rrrr} & u_1(ref_1).ref_1(x).P \mid & \overline{u_1}\langle r_1\rangle.\mathbf{0} \mid & \overline{r_1}\langle \text{mary}\rangle.\mathbf{0} \\ \longrightarrow & r_1(x).P\{^{r_1}/_{ref_1}\} \mid & \mathbf{0} \mid & \overline{r_1}\langle \text{mary}\rangle.\mathbf{0} \end{array}$$

In the above formulation, we can conceive of the $\overline{r_1}$ of $\overline{r_1}\langle \text{mary}\rangle$ as a *reference* to the person called Mary. The way the reference is conveyed to the process, P, was simplified by the assumption, but it could be conveyed to through other means as well. Also, if each utterance of noun phrases may import a reference to its corresponding part of semantic objects, it must be passed to another part subsuming it so as to construct the meaning of the sentence as a whole. The way references are passed around between processes may be determined by syntax. The ability of the calculus to model the exchange of references between processes is then useful to model such a dynamic aspect of syntax/semantics interface, too.[6]

3 The construction of semantic objects

3.1 Interaction graphs

The π-calculus provides for a useful foundation, but we need to extend it so as to model semantic objects employed in *ST-DRT* due to the strict sequential composition. Suppose we would like to model the infon abstract (1) as a process such as $(r_1(x) \mid r_2(y)).P$, the process that receives an item simultaneously through r_1 and r_2 to replace it for x and y, respectively, and to turn into another state,

[5]For simplicity, we disregard the information flow through u_2 and ref_2.

[6]Interested readers are referred to the author's thesis.

P, encoding the parametric infon. The construction is, however, not allowed for in the calculus because the sequential composition must proceed to the parallel one. We lift the restriction by proposing *interaction graphs* as our model.

Interaction graphs (*IG*) represent graphically the structure of interactions between actions, where the sequential order is not specified explicitly. In the representation, a system such as $(\overline{a}\langle b\rangle \mid a(x))$ is depicted as shown in Figure 2. Since the node, a, is used as *port*, the below half is shaded, to which two arcs are attached. The *exporting arc* links b to a, depicting the action, $\overline{a}\langle b\rangle$, while the *importing arc* links a to x to depict the action, $a(x)$. The different sorts are depicted with different heads of arcs.[7]

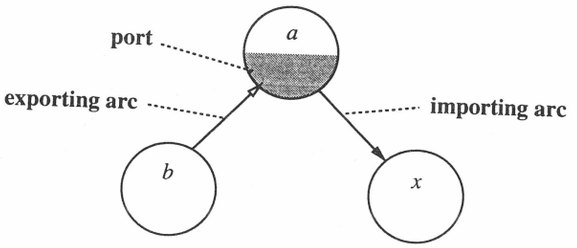


Figure 2: An interaction graph of $(\overline{a}\langle b\rangle \mid a(x))$

To illustrate how the computation is performed by *IG*, we consider a more complex case than the above, a system such as $(\overline{a}\langle b\rangle.\overline{b}\langle c\rangle \mid a(x).x(y))$, whose *IG* representation is shown in the leftmost of Figure 3. By a chemical metaphor,[8] we conceive of the arcs as molecules moving freely in the solution, whose surface is indicated by the shadowed horizontal line. Nodes shaded in the below half are then regarded as a catalyst to activate the interaction between molecules. The interaction always occurs in the surface. The two molecules, $\overline{a}\langle b\rangle$ and $a(x)$, are activated by a to interact with each other. Upon the interaction, they are evaporated, leaving x renamed to b (the second leftmost of Figure 3). Then, the other molecules come up to the surface and move to contact with each other through b (the second rightmost of the figure). The two parts are merged because the two nodes above the surface bear the same name. Finally, they interact with each other to leave y renamed to c (the rightmost of the figure).

With the representation, we do not have to specify the sequential order as it is naturally enforced by the constraint that the interaction must occur in the surface.

[7]Interaction Graphs are a variant of Pi-nets, a graphical form of the π-calculus proposed by Milner [Milner 1994], but they can depict possible interactions in clearer form. The idea to shade only nodes serving as port is due to Robin Cooper.

[8]Such a metaphor was first proposed by [Berry and Boudol 1992].

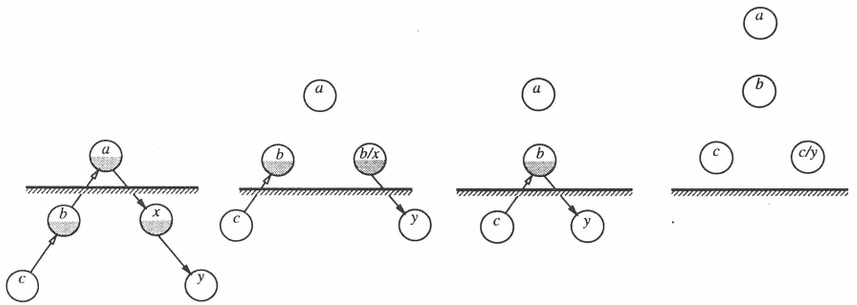


Figure 3: Modelling mobility in *IG*

Concurrency can be depicted in the representation since unlimited number of molecules can be in the surface to interact with each other at the same time. Apart from the advantage, the representation is easier to read than algebraic formulas. We will threfore employ *IG* in the next section to construct semantic objects of *ST-DRT*.

3.2 Encoding *ST-DRT* into *IG*

As an example, we construct the infon abstract (1), $\lambda[r_1 \rightarrow x, r_2 \rightarrow y]\langle\langle \text{eat}, x, y; 1\rangle\rangle$, using *IG*. The encoding of simultaneous abstractions is trivial; They are encoded as two molecules, $r_1(x)$ and $r_2(y)$. The encoding of the parametric infon, $\langle\langle \text{eat}, x, y; 1\rangle\rangle$, is however not straightforward. In Situation Theory, an infon such as $\langle\langle \text{eat, mary, shortbread}; 1\rangle\rangle$ is actually an abbreviation of $\langle\langle \text{eat}, r_1 \rightarrow \text{mary}, r_2 \rightarrow \text{shortbread}; 1\rangle\rangle$ because relations such as 'eat' are a particular kind of abstract, which means that they have indexed roles and restrictions on their assignments. The parametric infon is therefore to be regarded as $\langle\langle \text{eat}, r_1 \rightarrow x, r_2 \rightarrow y; 1\rangle\rangle$. Since the information that x and y are indexed with r_1 and r_2 is already in the encoding of abstractions, we only need to represent the information that the relation, 'eat', has two arguments indexed with r_1 and r_2. The easiest way is to encode it as an exporting action such as $\overline{eat}\langle r_1, r_2\rangle$ by regarding the relation as port and the roles as the items to be exported. Then, the infon abstract can be depicted in *IG* as is shown in Figure 4.[9]

The assignment, $[r_1 \rightarrow \text{mary}, r_2 \rightarrow \text{shortbread}]$, is encoded as output actions,

[9]We do not depict the order between r_1 and r_2 for simplicity. To be precise, we had to number the arcs pointing to *eat* to distinguish the graph for $\overline{eat}\langle r_1, r_2\rangle$ from another for $(\overline{eat}\langle r_1\rangle \mid \overline{eat}\langle r_2\rangle)$. We also disregard the polarity, 1. It can be encoded as an additional item such as $\overline{eat}\langle r_1, r_2, 1\rangle$.

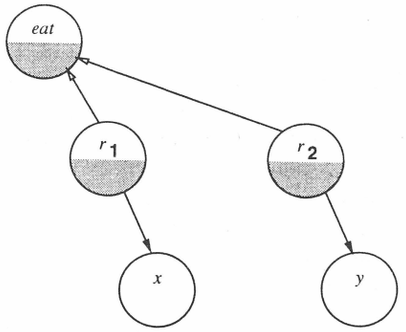


Figure 4: An *IG* representation of $\lambda[r_1 \rightarrow x, r_2 \rightarrow y]\langle\langle \text{eat}, x, y; 1\rangle\rangle$

$\overline{r_1}\langle\text{mary}\rangle$ and $\overline{r_2}\langle\text{shortbread}\rangle$. Combined with them, the infon abstract is depicted as is shown in Figure 5. When the structure is accessed from others, we can first observe at the location of 'eat' in the surface that r_1 and r_2 are extracted upon the interaction. Then we can see at the location of r_1 and r_2 that x is substituted by 'mary' and y by 'shortbread', respectively. The notion of observability determined by the structure of graphs leads to a strong equivalence relation between processes. A weak notion, on the other hand, is obtainable by introducing negligible interactions such as the ones under the surface if we allow for any. We do not investigate the equivalence relations here, but will show how the structure can be used to study the complexity of information states in the next section.

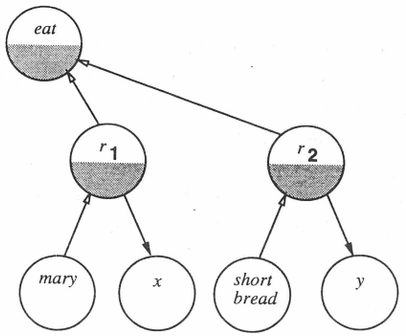


Figure 5: The infon abstract with the assingment $[r_1 \rightarrow \text{mary}, r_2 \rightarrow \text{shortbread}]$

3.3 Situations as references

We have so far not looked into the notion of *situation*, the most important one in the theory. To begin with, we investigate what roles the notion may play in Situation Semantics. Firstly, a situation, s, can support items of information called infons. Let σ be an infon supported by s, which we express as $s \models \sigma$. We may form then a proposition such that σ is factive at s, written as $(s \models \sigma)$, which may be true or false depending on s. The role of situations to determine the truth value of propositions cannot be captured by *IG* because algebra has nothing to do with such a logical notion.[10]

Another role that situations can play is to be a reference to an information source, which can be captured by our approach. Such a use of situations can be found, for example, in the study of common knowledge by shared-situation approach due to Barwise [Barwise 1989]. Suppose we would like to represent a case where an agent a knows σ, another agent b knows σ, a knows b knows σ, b knows a knows σ, a knows b knows a knows σ, and so on. The situation can be defined by the following three axioms:

- $s \models \sigma$
- $s \models a$ know s
- $s \models b$ know s

where s is a shared situation between a and b. The point of the definition is in the self-referential use of situations observed in the second and third formulas. We can conceive of the situation, s, in the arguments of 'know' as a reference to an information source.

The shared situation can be encoded in *IG* as is shown in Figure 6, where the part for the infon, σ, is indicated by the dotted box. The graph means that at s one can get access to the two relations, know$_a$ and know$_b$, the knowledge states of a and b, in addition to the access to the infon, σ. The items of information available at know$_a$ are r_{a1} and r_{a2}, the access to the agent and the shared situation, s.[11] Computationally speaking, the point of the construction is in treating the situation as an *address* and the content of the second argument of know$_a$ as a *pointer*. Note such an encoding is made possible by the ability to exchange references of the π-calculus.

[10] We will investigate the issue in the next part by grounding the graphical representation to linear logic.

[11] We are again sloppy here in not depicting the order between r_{a1} and r_{a2}. The lack of numbering between arcs pointing to s is however intentional; One can randomly extract these items of information. The non-deterministic choice will be defined in the next part as the additive conjunction, &, in linear logic.

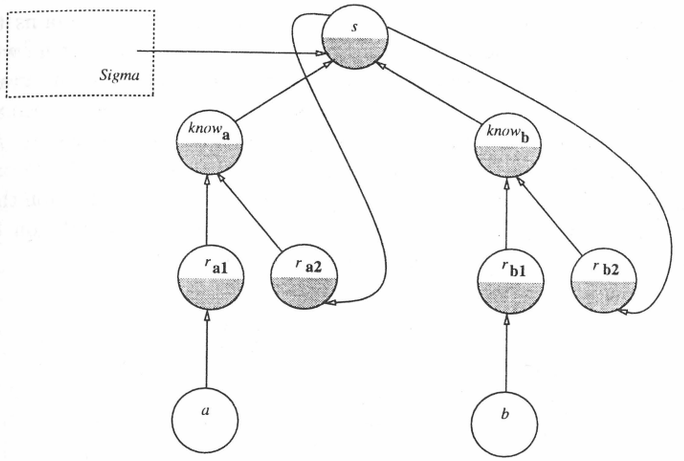


Figure 6: An *IG* of a shared situation

To see how *IG* helps us study varieties of shared situations, we consider a more complex shared situation, which is defined with the following axioms:

- $s_1 \models \sigma$
- $s_1 \models a$ know s_1
- $s_1 \models b$ know s_2
- $s_2 \models \sigma$
- $s_2 \models a$ know s_1
- $s_2 \models b$ know s_2

Here we have two shared situations, s_1 and s_2, both of which support the infon, σ. The agent, a, however gets access to them only through s_1, while b only through s_2. The questions are how the case is complex and how it can be related to the first case.

Backing to Figure 6, we can observe the information flow through s consists of two self-referential flows: $s \rightarrow r_{a2} \rightarrow know_a \rightarrow s$ and $s \rightarrow r_{b2} \rightarrow know_b \rightarrow s$. The flow becomes more complex for the second case, depicted in Figure 7. Both a and b retrains the same flows, say $s_1 \rightarrow r_{a2} \rightarrow know_a \rightarrow s_1$ and $s_2 \rightarrow r_{b2} \rightarrow know_b \rightarrow s_2$. In addition to these, one can observe a new flow linking s_1 and s_2:

$s_1 \rightarrow r_{a2} \rightarrow know_a \rightarrow s_2 \rightarrow r_{b2} \rightarrow know_b \rightarrow s_1$. The additional flow explains the difference between the first and second cases in complexity. To relate them with each other, we can think of an operation on graphs, which enables one graph to simulate another. Observe the graph for the second case (Figure 7) becomes identical with the one for the first (Figure 6) when both nodes, s_1 and s_2, are mapped to s. Such an operation does not exist for the first to simulate the second. We can therefore conclude that the second graph contains richer information than the first. The notion of *simulation* sketched here may shed a new light on the notion of *bisimulation*, but we leave the research for a future project.

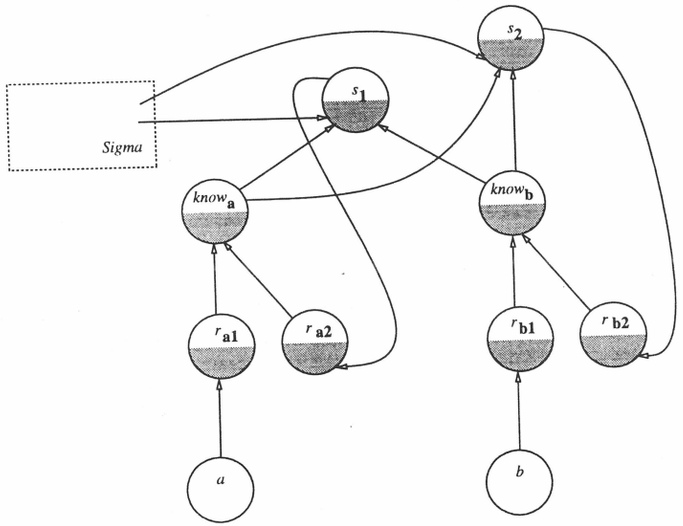


Figure 7: An *IG* of more complex shared situations

4 Grounding *IG* to linear logic

4.1 Combinatorial intuitionistic linear logic

The π-calculus and *IG* have been useful to study *ST-DRT*, but they fall short of capturing logical aspects of the theory. Firstly, it lacks with the notion of *truth*. Secondly, logical connectives such as conjunction and disjunction are missing. Quantifiers have not been considered, too. These notions can only be investigated by grounding *IG* to a logic. For the purpose, we turn to linear logic because it

provides us with a good starting point to capture some of the most important aspects of *IG*: *concurrency* and *state changes*. The linear logic we adopt here is a *combinatorial intuitionistic linear logic* (*C-ILL*) [Lafont 1988] extended with the equivalence relation, $=$, to express the substitution environments.

The table 1 shows the axioms and rules for C-ILL$_0$, the most basic parts of the logic without the exponential, the equivalence relation, and quantifiers. To interpret the axioms and rules, we conceive of the relation, $\Rightarrow$, as indicating state changes, where propositions define the resources available at the state. Terms are then regarded as the action causing the change. In the interpretation, the cut rule is regarded as the rule for sequential composition in the sense that when the two actions, ϕ and ψ, are combined sequentially, which change a state A into B and B into C, respectively, the composed action changes a state A into C. To clarify the meaning, we write $\phi \circ \psi$ rather than $\psi \circ \phi$. The identity axiom, id_A, is regarded to define *idling* or to express unchanged parts in transitions. The rule and axiom for the multiplicative conjunction are on the other hand regarded as defining parallel composition. That is, if ϕ changes A into B and ψ C into D, the state change caused by performing ϕ and ψ concurrently is defined as $A \otimes B \Rightarrow C \otimes D$. The symbol, $\mathbf{1}$, is the unit of $\otimes$.

The rule for the additive conjunction expresses an external non-determinism. That is, if you have two options, ϕ and ψ, at the state of A, then the product of them forms a basis to forecast that you will end up with either B or C, depending on your choice. The projection, π, expresses the choice you make between them. The rule for the additive disjunction on the other hand expresses an internal non-determinism. That is, you know the state of A can be achieved either by performing ϕ at B or ψ at C. When these arè composed as coproduct, it does not matter which is done actually. The coprojection, κ, expresses something unpredictable could always happen beyond your control.

Let C-ILL$_e$ be a system obtained by adding the rules and axioms for the exponential (Table 2) to C-ILL$_0$. The motivation is to incorporate *contraction* into the logic (Table 3). In defining *IG* using the logic, the operation is only applied to the formulas of the equivalence relation, which is defined as is shown in Table 4.[12] We will shortly show in the next section how these rules and axioms are used.

4.2 Defining *IG* in *C-ILL*

We show how we can define *IG* using *C-ILL*, C-ILL$_e$ plus $=$. We first show how actions can be defined when no interactions occur between them. Backing to the

[12]More succinctly, one can construct the axioms for equivalence relation from those without the exponential, given another axiom, $(t = t) \Rightarrow \mathbf{1}$.

Cut:

$$\frac{\phi : A \Rightarrow B \quad \psi : B \Rightarrow C}{\phi \circ \psi : A \Rightarrow C} \qquad id_A : A \Rightarrow A$$

Multiplicative conjunction:

$$\frac{\phi : A \Rightarrow B \quad \psi : C \Rightarrow D}{\phi \otimes \psi : A \otimes C \Rightarrow B \otimes D} \qquad 1 : \mathbf{1} \Rightarrow \mathbf{1}$$

Symmetry, associativity, and unit:

$$\gamma_{A,B} : A \otimes B \Rightarrow B \otimes A$$

$$\alpha_{A,B,C} : A \otimes (B \otimes C) \Rightarrow (A \otimes B) \otimes C$$

$$\alpha^{-1}_{A,B,C} : (A \otimes B) \otimes C \Rightarrow A \otimes (B \otimes C)$$

$$\lambda_A : \mathbf{1} \otimes A \Rightarrow A \qquad \lambda_A^{-1} : A \Rightarrow \mathbf{1} \otimes A$$

Additive conjunction:

$$\frac{\phi : A \Rightarrow B \quad \psi : A \Rightarrow C}{\langle \phi, \psi \rangle : A \Rightarrow B \& C} \qquad \pi_{A,B,i} : A_0 \& A_1 \Rightarrow A_i \ (i \in \{0,1\})$$

$$\top_A : A \Rightarrow \top$$

Additive disjunction:

$$\frac{\phi : B \Rightarrow A \quad \psi : C \Rightarrow A}{[\,\phi, \psi\,] : B \oplus C \Rightarrow A} \qquad \kappa_{A,B,i} : A_i \Rightarrow A_0 \oplus A_1 \ (i \in \{0,1\})$$

$$\mathbf{0}_A : \mathbf{0} \Rightarrow A$$

Table 1: Axioms and rules for Combinatorial Intuitionistic Linear Logic, C-ILL$_0$

Exponential:

$$\frac{\phi : A \Rightarrow B}{!\phi : !A \Rightarrow !B} \qquad \mathbf{s}_A : !A \Rightarrow !!A \qquad \mathbf{r}_A : !A \Rightarrow A \qquad \mathbf{t} : !\top \Rightarrow \mathbf{1}$$

$$\mathbf{p}_{A,B} : !(A \& B) \Rightarrow !A \otimes !B \qquad \mathbf{p}^{-1}_{A,B} : !A \otimes !B \Rightarrow !(A \& B)$$

Table 2: Axioms and rules for the exponential

Contraction:

$$\frac{A \otimes \ (!C \otimes !C) \Rightarrow D}{A \otimes !C \Rightarrow D}$$

Table 3: The rule for contraction

Equivalence relation:

$$\mathbf{eq}_1 : \mathbf{1} \Rightarrow !(t = t)$$

$$\mathbf{eq}_2 : !(s = t) \Rightarrow !(t = s)$$

$$\mathbf{eq}_3 : !(s = t) \otimes P[t/x] \Rightarrow P[s/x],$$

Table 4: The axioms for equivalence relation

previous example, each action is assigned a term, e.g., u_i and v_i where $i \in \{1, 2\}$ (Figure 8). To reflect the constraint that actions must occur from the surface, we decorate each arc with its precondition in the upper part. In the lower part, they are decorated with their postconditions, that is, resources made available as the result. For example, u_1 is defined as $a(x) \Rightarrow x(y)$.[13] When an action is followed by no actions underneath, its postcondition is defined as $\mathbf{1}$. The actions are thus defined as follows:

- $u_1 : a(x) \Rightarrow x(y)$
- $u_2 : x(y) \Rightarrow \mathbf{1}$
- $v_1 : \overline{a}\langle b \rangle \Rightarrow \overline{b}\langle c \rangle$
- $v_2 : \overline{b}\langle c \rangle \Rightarrow \mathbf{1}$

To define interactions, we have to predict which pairs may interact with each other. Our graphical representation is useful for the purpose since any pairs are in the same distance from the common parent. In the figure, for example, u_1 and v_1 are in the distance of zero from the node a, and u_2 and v_2 are in the distance of one from the same node. Let t_1 and t_2 be the interactions between these actions. Then, they are defined as follows:

[13] To define states, we borrow the π-calculus notation. To be precise, we should use special predicates such as $importing(a, x)$ or $exporting(a, b)$, but we avoid to complicate the notation.

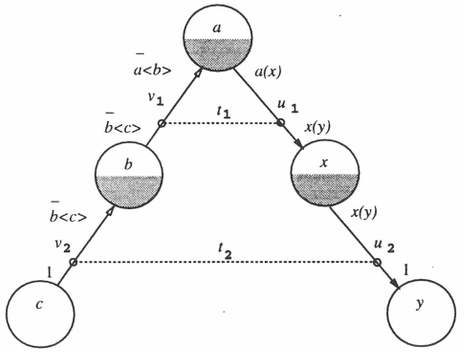


Figure 8: The decorated interaction graph with resources

- $t_1 : a(x) \otimes \overline{a}\langle b\rangle \Rightarrow x(y) \otimes \overline{b}\langle c\rangle \otimes\ !(b = x)$
- $t_2 : x(y) \otimes \overline{b}\langle c\rangle \otimes\ !(b = x) \Rightarrow !(c = y)$

Since they are concurrently active when they interact with, we use the multiplicative conjunction, $\otimes$. One point of the definition is in the postcondition of t_1; We add $!(b = x)$ to express the effect of the interaction. This enables us to express the correct precondition of t_2; The interaction becomes possible only if x is substituted by b.

With these special axioms, we can infer in *C-ILL*, for example, that the computation starts initially from the state where resources for $a(x)$ and $\overline{a}\langle b\rangle$ are available and ends up with the state where x and y are substituted by b and c, respectively, as below. One can see in the derivation that the contraction rule is used to distribute the information on the substitution, $b = x$. In our definition, the use of the exponential is restricted to expressing substitution environments.

$$\dfrac{t_1 : a(x) \otimes \overline{a}\langle b\rangle \Rightarrow x(y) \otimes \overline{b}\langle c\rangle \otimes\ !(b = x) \qquad \dfrac{t_2 : x(y) \otimes \overline{b}\langle c\rangle \otimes\ !(b = x) \Rightarrow !(c = y) \qquad \dfrac{id : (b = x) \Rightarrow (b = x)}{!id : !(b = x) \Rightarrow !(b = x)}\ !}{t_2 \otimes !id : x(y) \otimes \overline{b}\langle c\rangle \otimes\ !(b = x) \otimes\ !(b = x) \Rightarrow !(c = y) \otimes\ !(b = x)}\ \otimes\ con}{t_1 \circ (t_2 \otimes !id) : a(x) \otimes \overline{a}\langle b\rangle \Rightarrow !(c = y) \otimes\ !(b = x)}\ cut$$

4.3 Defining *ST-DRT* in *C-ILL*

In the previous section, we have used cut, $\circ$, and multiplicative conjunciton, $\otimes$, to define the computation specified by *IG*. Since *IG* enables us to construct infons,

propositions, and abstract objects employed in *ST-DRT*, as shown already (§3), we only need the two to define those objects in *C-ILL*. We are still left with two connectives unused, the additive conjunction, &, and disjunction, $\oplus$, which we will use to define conjunctive and disjunctive objects of infons or propositions. The translation rules are given as follows:

- $[\![A \wedge B]\!] := [\![A]\!]\ \&\ [\![B]\!]$
- $[\![A \vee B]\!] := [\![A]\!] \oplus [\![B]\!]$

where A and B are either infons, propositions, or composite objects with the connectives.

To capture the constraint relation in *ST-DRT*, $\rightarrow$, we have to extend the logic with linear implication, $\multimap$, by internalising $\Rightarrow$ to the system. We can then define quantifiers, $\forall$ and $\exists$, in the system. We do not go into the detail, however, due to the limited space. Interested readers are referred to the author's thesis.

Once we have based *ST-DRT* to *C-ILL* in this way, the notion of *truth* can be obtained through the logic. That is, an expression, $\phi : A \Rightarrow B$, is true if and only if it is constructed in *C-ILL* from the given axioms by applying the rules, where the formula expresses a proposition in Situation Theory. The logic can also be seen as a proof theory for *ST-DRT*. If you conceive of the logic as a type system, then it gives you a computational account of *ST-DRT* though it is yet to be investigated what the underlying calculus is.

5 Relating *C-ILL* to Channel Theory

The grounding to *C-ILL* also enables us to relate our study of *ST-DRT* with Channel Theory, a theory of information flow developed from Situation Theory. The table 5 shows the principle of information flow proposed by Barwise [Barwise 1993]. We can relate these postulates with the axioms and rules for *C-ILL* as follows:

- 'Xerox Principle' corresponds to the cut rule, $\circ$.
- 'Logic as Information Flow' is inherit in our system, too, though there is a slight difference in that we write $\phi : t_1 \Rightarrow t_2$ rather than $s : t_1 \longrightarrow s : t_2$, where ϕ is a connection.
- 'Addition of Information' corresponds to the rule for multiplicative conjunction, $\otimes$.

- 'Exhaustive Cases' can be derived using the rules for additive disjunciton and cut as below:

$$\frac{\phi : t_1 \Rightarrow t_2 \oplus t_2' \qquad \dfrac{\psi_1 : t_2 \Rightarrow t_3 \quad \psi_2 : t_2' \Rightarrow t_3}{[\psi_1, \psi_2] : t_2 \oplus t_2' \Rightarrow t_3}}{\phi \circ [\psi_1, \psi_2] : t_1 \Rightarrow t_3}$$

- 'Contraposition' does not hold in our system because the logic is intuitionistic. It will, however, hold if we make the logic classical by introducing $\parr$, the dual of $\otimes$ with its neutral element, $\bot$.

---

1. Xerox Principle:

$$\frac{s_1 : t_1 \longrightarrow s_2 : t_2 \quad s_2 : t_2 \longrightarrow s_3 : t_3}{s_1 : t_1 \longrightarrow s_3 : t_3}$$

2. Logic as Information Flow:

$$\frac{t_1 \vdash t_2}{s : t_1 \longrightarrow s : t_2}$$

3. Addition of Informaition:

$$\frac{s_1 : t_1 \longrightarrow s_2 : t_2 \quad s_1 : t_1' \longrightarrow s_2 : t_2'}{s_1 : t_1 \wedge t_1' \longrightarrow s_2 : t_2 \wedge t_2'}$$

4. Exhaustive Cases:

$$\frac{s_1 : t_1 \longrightarrow s_2 : t_2 \vee t_2' \quad s_2 : t_2 \longrightarrow s_3 : t_3 \quad s_2 : t_2' \rightarrow s_3 : t_3}{s_1 : t_1 \longrightarrow s_3 : t_3}$$

5. Contraposition:

$$\frac{s_1 : t_1 \longrightarrow s_2 : t_2}{s_2 : \neg t_2 \longrightarrow s_1 : \neg t_1}$$

Table 5: The principle of information flow: Barwise's postulates

---

Our system is actually different from Channel Theory in that it is intuitionistic and linear, but otherwise they share much in common. Barwise in fact proposes to view linear logic as a theory of information flow [Barwise 1992]. We think therefore our approach is in track of Channel Theory and we have shown a way to relate Channel Theory with *ST-DRT*, which have been hitherto developed rather independently.

6 Conclusion

In the paper, we have shown how to construct semantic objects employed in *ST-DRT* as systems of communicating processes. We have devised a graphical representation called *interaction graphs* to model processes and have shown how they can be grounded to linear logic. The grounding enables us to study *ST-DRT* on one hand and to relate our study with Channel Theory on the other hand. Computationally, the system is still problematic because it is known linear logic is undecidable [Lincoln et al. 1992]. But the author believes we have shown a starting point to investigate computational aspects of semantic theories.

It is also hoped that the work will contribute to establishing dialogue semantics. The framework presented here should allow to describe the information update by utterances in dialogue when we model it as a reactive system. In the long ran, we also hope to study linguistic phenomena where information update and interactions with environments are involved. Some languages seem to have a way to indicate whether or not the meaning of a sentence part is in the common ground. A Japanese phrase final particle, 'tte', is an example [Fujinami 1995]. There are also many languages that have 'evidential' particles or affixes indicating degree of direct evidence for assertions [Levinson 1988]. Levinson mentions as an example that Kwakiutl, a southern American language, requires all noun phrases to be affixed with indicators of visibility/invisibility to speaker. We believe our framework can be extended to study such phenomena.

Acknowledgement The work presented here was done while the author was at the Centre for Cognitive Science, University of Edinburgh. His sincere thanks to Robin Cooper and Jonathan Ginzburg for their supervision. He also thanks to Peter Ruhrberg for his comments. The author is working for and supported by *Verbmovil* project at Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart.

References

Aczel, P., Israel, D., Katagiri, Y., and Peters, S. (eds.): 1993, *Situation Theory and its Applications*, Vol. 3, Center for the Study of Language and Informaiton, Stanford, California

Barwise, J.: 1989, On the model theory of common knowledge, in *The Situation in Logic*, pp 201–220, Center for the Study of Language and Informaiton

Barwise, J.: 1992, Information links in domain theory, in S. Brookes, M. Main, A. Melton, M. Mislove, and D. Schmidt (eds.), *Proceedings of the Mathematical*

Foundations of Programming Semantics Conference, Vol. 598 of *LNCS*, pp 168–192, Springer Verlag

Barwise, J.: 1993, Constraints, channels, and the flow of information, in Aczel et al. 1993, pp 3–27

Barwise, J. and Cooper, R.: 1991, Simple situation theory and its graphical representation, in J. Seligman (ed.), *Partial and Dynamic Semantics III*, pp 38–74, Centre for Cognitive Science, University of Edinburgh, DYANA Report R2.1.C

Barwise, J. and Cooper, R.: 1993, Extended kamp notation: a graphical notation for situation theory, in Aczel et al. 1993, pp 29–53

Barwise, J. and Perry, J.: 1983, *Situations and Attitudes*, MIT Press, Cambridge, Mass.

Berry, G. and Boudol, G.: 1992, The chemical abstract machine, *Theoretical Computer Science* 96, 217–248

Cooper, R.: 1993a, Integrating different information sources in linguistic interpretation, in *First International Conference on Linguistics at Chosun University*, pp 79–109, Foreign Culture Research Institute, Chosun University, Kwangju, Korea

Cooper, R.: 1993b, Towards a general semantic framework, in R. Cooper (ed.), *Integrating Semantic Theories*, ILLC/Department of Philosophy, University of Amsterdam, Deliverable R2.1.A, Dyana-2

Fujinami, T.: 1995, *A process algebraic approach to computational linguistic*, Ph.D. thesis, Centre for Cognitive Science, University of Edinburgh, Edinburgh, avaibable from ftp.ims.uni-stuttgart.de

Girard, J.-Y.: 1987, Linear logic, *Theoretical Computer Science* 50(1), 1–102

Lafont, Y.: 1988, The linear abstract machine, *Theoretical Computer Science* 59, 157–180

Levinson, S.: 1988, *Informal notes on pragmatics and situation semantics*

Lincoln, P., Mitchell, J., Scedrov, A., and Shankar, N.: 1992, Decision problems for propositional linear logic, *Annuals of Pure and Applied Logic* 56, 239–311

Milner, R.: 1989, *Communication and Concurrency*, Prentice Hall, New York

Milner, R.: 1992, Functions as processes, *Journal of Mathematical Structures in Computer Science* 2(2), 119–141

Milner, R.: 1993, The polyadic π-calculus: a tutorial, in F. L. Bauer, W. Brauer, and H. Schwichtenberg (eds.), *Logic and Algebra of Specification*, pp 203–246, Springer Verlag

Milner, R.: 1994, Pi-nets: a graphical form of π-calculus, in *Proceedings of ESOP '94*, Vol. 788 of *LNCS*, pp 26–42, Springer Verlag

Milner, R., Parrow, J., and Walker, D.: 1992, A calculus of mobile processes, parts I and II, *Information and Computation* 100, 1–40 and 41–77

Seligman, J. and Barwise, J.: 1993, *Channel Theory: toward a mathematics of imperfect information flow*, Unpublished ms.

Brendan S. Gillon
Department of Linguistics
McGill University

1 Introduction

Thirty years ago, Peter Geach (1962) brought to our attention a kind of sentence, the statement of whose truth conditions puzzled Medieval thinkers. They have come to be known as donkey sentences, for reasons the two prototypical examples given below make clear:

(1.1) Every farmer who owns a donkey beats it.
(1.2) If a farmer owns a donkey, he beats it.

Sentences such as these continue to be a puzzle to philosophers and semanticists, for they seem to escape any straightforward syntactic and semantic treatment.

In particular, they show the inconsistency of three *prima facie* plausible assumptions: first, that the antecedent of the third person personal pronoun is the indefinite noun phrase; second, that indefinite noun phrases are restricted existential quantifiers of classical quantificational logic (CQL, hereafter); and third, that a third person personal pronoun whose antecedent is a quantified noun phrase functions in natural language in a way analogous to the way a bound variable functions in CQL.

Consider the first sentence. Either the indefinite noun phrase, *a donkey*, has scope wider than *every farmer*, or it has narrower scope. And the principal connective is either $\wedge$ or $\rightarrow$. The alternatives are given in notation below:

(2.1) [a donkey]$_y$ [every farmer]$_x$ [x owns y $\wedge$ x beats y]
(2.2) [every farmer]$_x$ [a donkey]$_y$ [x owns y $\wedge$ x beats y]
(2.3) [a donkey]$_y$ [every farmer]$_x$ [x owns y $\rightarrow$ x beats y]
(2.4) [every farmer]$_x$ [a donkey]$_y$ [x owns y $\rightarrow$ x beats y]

Suppose, on the one hand, that the principal connective is $\wedge$. Then there are two possibilities: either the indefinite noun phrase *a donkey* has scope wider than the noun phrase *every farmer* or it has narrower scope. In the first case (2.1), the sentence would mean that there is at least one donkey which every farmer owns and beats. This, in turn, implies that every farmer has a donkey, indeed, the very same donkey, which clearly the sentence in (1.1) does not imply. In the second case (2.2), the sentence would mean that, for every farmer, there is a donkey which he owns and beats. This implies that each farmer has a donkey, though not necessarily the same one. But clearly this too is not implied by the sentence in (1.1).

Suppose, on the other hand, that the principal connective is $\rightarrow$. Again, there are the same two possibilies pertaining to the scope of the quantified noun phrases. If *a donkey* has scope wider than *every farmer* (2.3), then the sentence can be true, regardless of what the farmers do to the donkeys they actually own, provided that there is at least one unowned donkey. If *a donkey* has scope narrower than *every farmer* (2.4), then the sentence can be true, regardless of what the farmers do to the donkeys they actually own, provided that each farmer fails to own at least one. Neither of these is a suitable construal of the first sentence.

Three responses to the enigma of donkey anaphora have enjoyed some currency. All three responses accept the assumption that the indefinite noun phrase is the antecedent of the pronoun. The most widely accepted response gives up the assumption that indefinite noun phrases are restricted existential quantifiers, and maintains, instead, that indefinite noun phrases introduce restricted free variables. This approach was advocated independently by Kamp (1981) and Heim (1982). A second response gives up the assumption that anaphoric third person personal pronouns are functioning on the analogy with bound variables of CQL. This second approach, due to Evans (1977)[1], has been recently elaborated in detail by Neale (1990). A third response, advocated by Groenendijk and Stokhof (1991), gives up on two assumptions: that indefinite noun phrases are restricted existential quantifiers of CQL; and that a third person personal pronoun whose antecedent is a quantified noun phrase functions in a way analogous to the way a bound variable functions in CQL.

What I want to do here is to examine the ramifications for these approaches of a puzzle, due to Charles Sanders Peirce (Hartshorne and Weiss (eds) 1933 v. 4, §546 and §580), which Stephen Read (1992) has recently brought to light.

2 Peirce's Puzzle

It is a routine exercise in CQL to show that the following formulae are logically equivalent (Appendix 1).

(3.1) $\forall\nu\,\phi(\nu) \rightarrow \exists\nu\,\psi(\nu)$
(3.2) $\exists\nu\,(\phi(\nu) \rightarrow \psi(\nu))$

Yet, the following sentences, which apparently instantiate these formulae,

(4.1) Someone will win $1,000, if everyone takes part.
(4.2) Someone will win $1,000, if he takes part.

are not, intuitively speaking, analytically equivalent. To see this, consider these circumstances of evaluation: There is a sweepstakes in which only one thousand people are eligible to participate. Tickets are sold for $1 each. No participant is permitted to buy more than one ticket. And the winner will take the total of the stakes. Clearly, under such circumstances, the first sentence is true and the second is false.

Read (1992: p. 10) diagnoses the problem to lie with material implication as a model of the English subordinating conjunction *if*. But this diagnosis is not borne out by the data.

To be sure, there are uses of the subordinating conjunction *if* which are not well-modelled by material implication, the most notorious cases being so-called Austinian Conditionals:

(5) If it snows, there is a shovel in the trunk.

The evidence for this is quite striking. Recall two of the standard logical equivalents for a proposition of the form $\alpha \rightarrow \beta$: $\neg\alpha \vee \beta$ and $\neg(\alpha \wedge \neg\beta)$. To the extent that the adverb *not* and the co-ordinating conjunctions *and* and *or* are well-modelled by classical propositional negation, conjunction, and disjunction respectively, one expects the following: any sentence of the form *If A, B*, which is well-modelled by material implication, should be judged intuitively equivalent to other, suitably constructed, English compound sentences of the form, *Either it is not the case that*

[1]The insight seems to have been anticipated by Quine (1960: §23).

A or B and *It is not the case that both A and not B*. The recasting of the sentence in (5) into such forms yields results which cannot be judged in any way as analytically equivalent to the sentences in (6).

(6.1) Either it will not snow or there is a shovel in the trunk.
(6.2) It is not the case that it will snow and there is no shovel in the trunk.

It is equally clear, however, that there are uses of the subordinating conjunction *if* which are well-modelled by the material conditional. The evidence is the fact that they observe precisely the logical equivalences which the Austinian conditionals do not. The following sentences, I submit, ordinary intuition judges to be analytical equivalents.

(7.0) If London is in China, I am a monkey's uncle.
(7.1) Either London is not in China or I am a monkey's uncle.
(7.2) It is not the case that London is in China and I am not a monkey's uncle.

Let us return to Peirce's puzzle. Consideration of a broader range of data shows that ascription of the failure of analytic equivalence to material implication is precipitate. To begin with, notice that one necessary ingredient to the problem is the presence of quantified noun phrases: the replacement by proper names of the quantified noun phrases in the sentences in (5) yields a sentence which one judges to be analytically equivalent to its disjunctive and conjunctive counterparts.

(8.0) John will win $1,000, if he takes part.
(8.1) Either John will win $1,000 or he will not take part.
(8.2) It is not the case that John will take part and he will not win $1,000.

Moreover, the mere presence of quantified noun phrases in such a sentence is not sufficient to give rise to the puzzle, since one judges the disjunctive and conjunctive counterparts of the sentence in (4.1) as analytically equivalent.

(9.0) Someone will win $1,000, if everyone takes part.
(9.1) Either someone will win $1,000 or it is not the case that everyone will take part.
(9.2) It is not the case that both someone will not win $1,000 and everyone will take part.

Indeed, Peirce's puzzle can be reproduced without any occurrence of the subordinating conjunction, or any occurrence of any of its lexical or syntactic counterparts. To see this, recall the following well-known logical equivalence of CQL[2]:

(10.1) $\exists\nu\,\phi(\nu) \vee \exists\mu\,\psi(\mu)$
(10.2) $\exists\nu\,(\phi(\nu) \vee \psi(\nu))$

The following sentences appear to be instances of these formulae:

(11.1) Either someone will win $1,000 or someone will not take part.
(11.2) Either someone will win $1,000 or he will not take part.

And Read's circumstances of evaluation yield the judgement that the sentence in (11.1) is true while the one in (11.2) is false.

It is clear, then, that the lack of entailment does not accrue to taking material implication as a model of *if*. Indeed, the problem seems to accrue to those sentences in which the indefinite noun phrase *someone* serves as an antecedent to the third person personal pronoun. Such is the configuration evinced by so-called donkey

[2]Indeed, this equivalence holds even in minimal quantificational logic.

sentences. Let us see, then, what challenge, if any, the anaphoric configurations evinced by the sentences of Peirce's puzzle pose for the three theories of anaphora designed to address precisely such anaphoric configurations.

3 Free Variables and Peirce's Puzzle

I begin with the theory of anaphora advocated by Kamp (1981) and Heim (1982: ch. 2). Recall that they retain the assumption that anaphoric pronouns are well-modelled as bound variables of CQL, but reject the assumption that indefinite noun phrases are well-modelled as restricted existential quantifiers of CQL. This rejection requires that certain compensating assumptions be adopted. Kamp and Heim have different compensating assumptions.

Consider, for example, the sentence in (13).

(12.0) Some man arrived.
(12.1) x is a man $\wedge$ x arrived
(12.2) $\exists x$ (x is a man $\wedge$ x arrived)

According to Heim, the sentence in (12.0) has a syntactic analysis which can be rendered by the formula in (12.1). This is an open formula, and hence cannot be assigned a truth-value by the semantics of CQL. To ensure that it has a truth-value, Heim posits that all free variables in a formulae are closed by existential closure over the formula. The result in this case is the formula in (12.2).

In Discourse Representation Theory, the truth conditions of a sentence are determined, not with respect to the sentence's syntactic analysis, but rather with respect to structures – so-called *discourse representation structures* – constructed from the sentence's syntactic analysis, where the construction proceeds top to bottom and left to right. These structures comprise a list of free variables and a set of conditions. A discourse representation structure determines not only the truth conditions of the sentence from which it is constructed but also which noun phrases might serve as antecedents for which pronouns.

Now, treatment of the antecedence relation must specifiy the morphological and syntactic constraints on the relation and the semantic mechanism whereby appropriate values are assigned to the relata of the relation. Discourse Representation Theory does both. The antecedence relation obtains, according to Discourse Representation Theory, when an identity condition is created for the variables associated with the relevant noun phrases. Whether or not the appropriate identity condition occurs depends on whether or not the variable corresponding to the antecedent is accessible to the variable corresponding to the pronoun. This, in turn, depends on two factors: the definition of the accessibility relation (essentially, an augmentation of the dominance relation defined over the discourse representation structure), and the points in the discourse structure where the variables are introduced.

The sentence in (12.0), once syntactically analyzed, has the discourse representation structure in (13).

(13) $\langle\, x : x$ is a man , x arrived $\rangle$

This structure comprises is a list of variables (in this case, a list of one variable), and a set of conditions (in this case, two conditions).

To obtain an equivalence between the structure in (13) and the quantificational formula in (12.2), Kamp avails himself of two facts of elementary model theory[3]:

[3]See Kamp and Reyle 1993 ch. 1.5 for details

First, each formula in a finite set of formulae is true in a model if, and only if, the conjunction formed from the formulae in the set is true in it. Second,

(14) $\models_M \exists\nu\, \phi(\nu)$, iff, for some variable assignment g, $\models_{M,g} \phi(\nu)$

What happens in the case of sentences such as those in (11)? On Heim's approach, the sentences have a syntactic analysis which can be rendered as follows:

(15.1) x will win \$1,000 $\vee$ y will not take part.
(15.2) x will win \$1,000 $\vee$ x will not take part.

Existential closure of these formulae and routine quantifier equivalences yield instances of the schemata in (10).

Discourse Representation Theory provides a different treatment of the sentences in (11). It assigns to the first one the discourse representation structure in (16.1), which is equivalent to the formula in (16.2) CQL.[4]

(16.1) $\langle\, \langle\, x : x$ will win \$1,000 $\rangle \vee \neg \langle\, y : y$ will take part $\rangle\, \rangle$
(16.2) $\exists x\ x$ will win \$1,000 $\vee\ \exists y \neg\ y$ will take part.

To obtain a discourse representation structure for the sentence in (11.2) requires that some modification be made either in how clauses co-ordinated by the connector *or* yield discourse representation structures or in how the relation of accessibility is defined. As pointed out by Evans (1977 p. 530), noun phrases without definite reference occurring in the first of two clauses connected by *or* do not usually serve as antecedents for pronouns occurring in the second.

(17) *Either John owns a donkey or he keeps it well hidden.

Aware of such facts, Kamp and Reyle (1993: ch. 2.3.1) configure the discourse representation structure created by clauses co-ordinated by the connector *or* and define the accessibility relation so that the variable introduced by an indefinite noun phrase is inaccessible to the variable introduced by a pronoun when the former occurs in the first clause and the latter in the second. But, as Kamp and Reyle (1993 ch. 2.3.1) recognize, cases do occur where an indefinite noun phrase in the first clause may serve as an antecedent to the pronoun in the second. To take such cases into account, Kamp and Reyle suggest that the variable for the indefinite noun phrase may be introduced in the most superordinate list of variables, thereby making it accessible to the variable introduced by the pronoun. The application of this alternative to the sentence in (11.2) gives rise to the discourse representation structure in (18.1), whose equivalent formulation in CQL is given in (18.2).

(18.1) $\langle\, x : \langle\, : x$ will win \$1,000 $\rangle \vee$
$\neg \langle\, : y$ will take part , $x = y \,\rangle\, \rangle$
(18.2) $\exists x$ (x will win \$1,000 $\vee \neg\ x$ will take part)

But the formulae in (16.2) and (18.2) are just instances of the schemata in (11). Thus, on either approach, that of Kamp and Reyle or that of Heim, the truth conditions for the sentences in (11) are the same as those given them by their *prima facie* treatment by CQL. In other words, the sentences remain logically equivalent, contrary to intuitions.

Let us now turn to the sentences in (5). The syntactic analysis of these sentences yields, according to Heim's view, the following formulae.

(19.1) $\forall x\ x$ take part $\rightarrow\ y$ will win \$1,000.
(19.2) x take part $\rightarrow\ x$ will win \$1,000.

[4]See Kamp and Reyle (1993) ch. 2.3.1 and ch. 3.7.3. It should be borne in mind that the operations denoted by the symbols '$\vee$' and '$\neg$' in discourse representation structures are distinct from, though similar to, those denoted by the same symbols in the formulae of CQL.

Now, she holds that subordinate conditional clauses serve as restrictions on adverbial unselective quantifiers.[5] When no overt quantificational adverb is present, she posits the presence of a phonetically null universal quantificational adverb. The subordinate clause in (19.1) contains no free variables, so the free variable in the main clause is bound by phonetically null existential quantifier. As a result, its truth conditions are those provided by CQL. The subordinate clause in (19.2), however, does contain a free variable. The tacit unselective universal adverbial quantifier then binds it and the second occurrence of the same variable in the main clause to yield truth conditions which the following rendition in CQL of the sentence in (4.2) have, namely,

(20) $\forall x$ (x take part $\rightarrow$ x will win \$1,000)

In other words, Heim's analysis implies that the sentence in (4.2) has the same truth conditions as the one in (21),

(21) Everyone who takes part will win \$1,000.

But this is clearly wrong. Again, Discourse Representation Theory provides a different treatment. It assigns to the first sentence in (4) the discourse structure in (22.1), which is equivalent to the formula in (22.2) of CQL.

(22.1) $\langle : \langle : \langle\, x : x$ is a person $\rangle \rightarrow \langle\, y : y$ takes part , $x = y \,\rangle$
$\rightarrow \langle\, z : z$ will win \$1,000 $\rangle\,\rangle$

(22.2) $\forall x\ x$ takes part $\rightarrow \exists z, z$ will win \$1,000.

But, to obtain a discourse representation structure for the sentence in (4.2) requires that some modification be made in the treatment of conditionals by Kamp and Reyle.

The treatment of conditionals by Kamp and Reyle begins with the observation, accepted by everyone, that indefinite noun phrases occurring in the protasis of a conditional sentence may serve as antecedents for pronouns in the apodosis, as exemplified by the sentence in (1.2). It is often the case indefinite noun phrases occurring in the apodosis of a conditional sentence may not serve as antecedents for pronouns in the protasis, as exemplified below:

(23.1) If Jones owns a book, he reads it.
(23.2) *If Jones owns it, he reads a book.

As a result, Kamp and Reyle have defined the accessibility relation so that the variables introduced by indefinite noun phrases in the protasis of a conditional are accessible to the variables introduced by pronouns in the apodosis, whereas those introduced by pronouns in the protasis are inaccessible to those introduced by indefinite noun phrases in the protasis. This implies, however, that the variable introduced by the indefinite noun phrase in (4.2) is inaccessible to the one introduced by the pronoun. But, clearly the indefinite noun phrase serves as the pronoun's antecedent.

Two ways to accommodate the facts suggest themselves: to introduce the variables in the usual way and to let the variable introduced by the indefinite noun phrase be accessible to the variable introduced by the pronoun; or, to have the variable introduced by the indefinite noun phrase be placed in the most superordinate list of variables, thereby making it accessible to the variable introduced by the pronoun – that is, to redeploy the modification adopted above for the treatment of the sentence in (11.2). Neither suggestion overcomes the difficulty posed by Peirce's puzzle.

[5]The notion of unselective adverbs of quantification is taken from Lewis (1975).

Using the first option, one obtains the discourse representation structure in (24.1)[6], whose equivalent in CQL is given by the formula in (24.2):

(24.1) $\langle\, x : x$ will take part $\rangle \rightarrow \langle\, y : y$ will win \$1,000 , $x = y \,\rangle$
(24.2) $\forall x$ (x takes part $\rightarrow$ x will win \$1,000).

But, these are the same truth conditions as those provided by Heim's analysis, which were seen earlier to be clearly incorrect.

Using the second option, one obtains the discourse representation structure in (25.1), whose equivalent in CQL is given by the formula in (25.2):

(25.1) $\langle\, x : \langle : x$ will take part $\rangle \rightarrow \langle\, y : y$ will win \$1,000 ,
$x = y \,\rangle\,\rangle$
(25.2) $\exists x$ (x takes part $\rightarrow$ x will win \$1,000).

But the formulae in (22.2) and (25.2) are just instances of the schemata in (3). In other words, the sentences in (4) are logically equivalent, contrary to intuitions. In short, Discourse Representation Theory does not provide a way to improve on the analysis for the sentences in (4) resulting from the naive application of CQL.

4 Dynamic Predicate Logic and Peirce's Puzzle

Dynamic Predicate Logic (hereafter, DPL) offers another way to address the problem of indefinite noun phrases serving as antecedents for pronouns which are not, at least *prima facie*, within their scope. To see how it differs from the previous treatment, let us begin by recalling how the relation of antecedence is modelled in CQL.

CQL is very limited in its capacity to mimic the antecedence relation of natural language. To mimic the relation, the variable which corresponds to the pronoun must be the same as the variable which corresponds to the pronoun's antecedent, and both instances of the variable must be bound by one and the same quantifier. It is this syntactic configuration of CQL which alone permits the values of the variable corresponding to the pronoun to depend on the values of the variable corresponding to the pronoun's antecedent. But it is common for sentences of natural language to fail to have their renditions in the language of CQL meet these conditions. Consider, for example, the following sentence:

(26) Some man arrived and he was hungry.

The formula of CQL which most closely corresponds to the pair of sentences above is the one below.

(27) $\exists x$ (x is a man $\wedge$ x arrived) $\wedge$ x is hungry.

However, the fourth occurrence of the variable x is not within the scope of the quantifier which binds the third occurrence. As a result, the choice of value for the variable x in the formula, *x is hungry*, is independent of the choice of value for the variable x in the formula, *x is a man $\wedge$ x arrived*.

The attraction of DPL is that it offers a way to permit values of one occurrence of one variable to depend on the choice of value of another occurrence of the same variable, even though these occurrences are not bound by the same quantifier. To appreciate better how DPL manages to do this, let us first go into some of the details of why CQL fails to do so.

[6]See Kamp and Reyle (1993) ch. 2.1, ch. 2.2 and ch. 3.7.3. It should be borne in mind that the operation denoted by the symbol '$\rightarrow$' in discourse representation structures is distinct from, though similar to, that denoted by the same symbol in the formulae of CQL.

Let us recall the rudiments of the semantics of CQL. In it, a model M comprises a pair $\langle U, i\rangle$, where U is a non-empty set, the universe, and i is a function, having as its domain the predicates[7], where, if Π is an n-place predicate, then $i(\Pi) \in Pow(U^n)$. Let V be the set of all variable assignments. Then one defines $[\ ]_M$ as a function from formulae into $Pow(V)$ as follows:

(28) Truth Clauses for a Model in CQL

1. $[\ \Pi\nu_1 \ldots \nu_n\ ]_M = \{g\colon \langle g(\nu_1), \ldots, g(\nu_n)\rangle \in i(\Pi)\}$
2. $[\ \neg\alpha\ ]_M = V - [\ \alpha\ ]_M$
3. $[\ \alpha \wedge \beta\ ]_M = [\ \alpha\ ]_M \cap [\ \beta\ ]_M$
4. $[\ \alpha \vee \beta\ ]_M = [\ \alpha\ ]_M \cup [\ \beta\ ]_M$
5. $[\ \alpha \rightarrow \beta\ ]_M = (V - [\ \alpha\ ]_M) \cup [\ \beta\ ]_M$
6. $[\ \exists\nu\,\alpha\ ]_M = [\ \alpha\ ]_M \cup \{g\colon g[\nu]h \text{ and } h \in [\ \alpha\ ]_M\}$
7. $[\ \forall\nu\,\alpha\ ]_M = [\ \alpha\ ]_M$, if, for each $h \in [\ \alpha\ ]_M$, $g \in [\ \alpha\ ]_M$ where $h[\nu]g$;
   $\emptyset$, otherwise.

(where '$g[\nu]h$' means that the variable assignments g and h differ at most in the value they assign to ν.) A formula α is true in a model M if and only if $[\ \alpha\ ]_M = V$; and a formula α is false in a model M if and only if $[\ \alpha\ ]_M = \emptyset$.

Let us return to the sentence in (26). There are circumstances of evaluation which, intuitively speaking, render the sentence in (26) true, but whose model, together with the truth clauses in (28), fails to render the formula in (27) true; and inversely, there are circumstances of evaluation which, intuitively speaking, render the sentence in (26) false, but whose model, together with the truth clauses in (28), fails to render the formula in (27) false.

Consider, on the one hand, the circumstances in which there are at least two people, one of whom is not hungry, the other of whom is a man, has arrived, and is hungry. Intuitively speaking, the sentence in (26) is true, but the formula in (27) fails to be true in the corresponding model, when evaluated in accordance to the clauses in (28). Of course, in the model for these circumstances, there is a variable assignment which renders *x is a man* $\wedge$ *x arrived* true; and so, by (28.6), $[\ \exists x\,(x \textit{ is a man } \wedge \textit{ x arrived})\ ]_M = V$; that is, in this model, the first conjunct in (27) is true. At the same time, however, there is a variable assignment which renders the second conjunct, *x is hungry*, false. So, $[\ x \textit{ is hungry}\ ]_M \neq V$. Thus, by (28.3), $[\ \exists x\,(x \textit{ is a man } \wedge \textit{ x arrived}) \wedge x \textit{ is hungry}\ ]_M \neq V$. In other words, the formula in (27) is not true in the model, contrary to one's intuitions about the sentence in (26).

Consider, on the other hand, the circumstances in which there are at least two people, one of whom is hungry but is not a man, the other of whom is a man and has arrived, but is not hungry. Intuitively speaking, the sentence in (26) is false, but the formula in (27) fails to be false in the corresponding model, when evaluated in accordance to the clauses in (28). As before, there is a variable assignment which renders *x is a man* $\wedge$ *x arrived* true, and so, by (28.6), $[\ \exists x\,(x \textit{ is a man } \wedge \textit{ x arrived})\ ]_M = V$; that is, in this model, the first conjunct in (27) is true. At the same time, however, there is a variable assignment which renders the second conjunct, *x is hungry*, true. So, $[\ x \textit{ is hungry}\ ]_M \neq \emptyset$. Thus, by (28.3), $[\ \exists x\,(x \textit{ is a man } \wedge \textit{ x arrived}) \wedge x \textit{ is hungry}\ ]_M \neq \emptyset$. In other words, the formula in (27) is not false in the model, contrary to one's intuitions about the sentence in (26).

The problem is not only that open formulae which are not substitution instances

[7] I omit individual constants from the language of CQL in order to simplify the presentation.

of the tautologies and contradictions of classical propositional logic do not have a truth value (relative to a model), but also that such formulae never have the values of their free variables determined by the values of variables in closed formulae. The reason for the latter fact is simple: when a closed formula is true, the variable assignments which rendered it so are, as it were, washed out with the addition of new, as it were, irrelevant variable assignments. Groenendijk and Stokhof (1991) have found a way to circumvent these problems. Their basic idea is to redo the clauses of (28) so that not only is the truth of a formula reckoned, but the variable assignments whereby an open formula is rendered true are kept track of.

This is how they do it. As before, a model is a pair $\langle U, i\rangle$, where U is a non-empty set, the universe, and i is a function whose domain comprises the predicates[8], where, if Π is an n-place predicate, then $i(\Pi) \in Pow(U^n)$. Let V be the set of all variable assignments. Then one defines $|\ |_M$ as a function from formulae into $Pow(V^2)$ as follows:

(29) Truth Clauses for a Model in DPL

1. $|\,\Pi\nu_1 \ldots \nu_n\,|_M = \{\langle g, g\rangle\colon \langle g(\nu_1), \ldots, g(\nu_n)\rangle \in i(\Pi)\}$
2. $|\,\neg\alpha\,|_M = \{\langle g, g\rangle\colon \langle g, h\rangle \in |\alpha|_M, \text{ for no } h\}$
3. $|\,\alpha \wedge \beta\,|_M = \{\langle g, h\rangle\colon \langle g, k\rangle \in [\alpha] \text{ and } \langle k, h\rangle \in |\beta|_M, \text{ for some } k\}$
4. $|\,\alpha \vee \beta\,|_M = \{\langle g, g\rangle\colon \text{either } \langle g, h\rangle \in |\alpha|_M \text{ or } \langle g, h\rangle \in |\beta|_M, \text{ for some } h\}$
5. $|\,\alpha \rightarrow \beta\,|_M = \{\langle g, g\rangle\colon \text{for each } h \text{ such that } \langle g, h\rangle \in |\alpha|_M, \text{ there is a } k \text{ such that } \langle h, k\rangle \in |\beta|_M\}$
6. $|\,\exists\nu\,\alpha\,|_M = \{\langle g, h\rangle\colon g[x]k \text{ and } \langle k, h\rangle \in |\alpha|_M\}$
7. $|\,\forall\nu\,\alpha\,|_M = \{\langle g, g\rangle\colon \text{for each } h \text{ such that } g[x]h, \text{ there is a } k \text{ such that } \langle h, k\rangle \in |\alpha|_M\}$

A formula α is true in a model M if and only if $\{g\colon \text{for some } h \in V, \langle g, h\rangle \in |\alpha|_M\} = V$; and a formula α is false in a model M if and only if $|\,\alpha\,|_M = \emptyset$. The first co-ordinates in the pairs of variable assignments which are assigned to formulae in DPL serve the same purpose as sets of simple variable assignments in CQL, namely, to keep track of truth and falsity; while the second co-ordinates serve to furnish values to free variables in certain open formulae.

Let us reconsider the sentence in (26) and see how this aspect of DPL serves to help the formula in (27) to mimic better the relation of antecedence in (26). Consider again the circumstances in which there are at least two people, one of whom is not hungry, the other of whom is a man, has arrived, and is hungry. Again, there is a variable assignment in the model of the circumstances which renders the formula, *x is a man* $\wedge$ *x arrived*, true. The clause in (29.6) assigns the formula, $\exists x\,(x$ *is a man* $\wedge$ *x arrived*), a set of pairs of variable in which every variable assignment in V is paired with every variable assignment which renders the formula, *x is a man* $\wedge$ *x arrived*, true. Thus, the first conjunct in (27) is true. Moreover, there is a variable assignment in the same model which renders the second conjunct, *x is hungry*, true. Thus, by (29.1), it is assigned the identity graph over these variable assignments. And finally, by hypothesis, the set of variable assignments which render the open formula *x is a man* $\wedge$ *x arrived* true is not disjoint from the set of variable assignments which render *x is hungry* true. Therefore, by (29.3), the conjunction of the existential closure of the first formula with the second formula is true, since the composition of the former set of pairs of variable assignments with the second set yields a set of pairs such that each variable assignment in V appears in some pair. In other words, the formula in (27) is true in the model, in harmony

[8] Again, individual constants are omitted in order to simplify the presentation.

with one's intuition about the sentence in (26).

Consider again the circumstances in which there are at least two people, one of whom is hungry but is not a man, the other of whom is a man and has arrived, but is not hungry. Again, as before, there is a variable assignment in the model for these circumstances which renders x *is a man* $\wedge$ x *arrived* true, and so, the clause (29.6) assigns the formula $\exists x$ (x *is a man* $\wedge$ x *arrived*), a set of pairs of variable assignments in which every variable assignment in V is paired with some variable assignment which renders the formula, x *is a man* $\wedge$ x *arrived*, true. Thus, the first conjunct in (27) is true. The second conjunct also has a variable assignment which renders it true. However, the variable assignments which render the formula, x *is a man* $\wedge$ x *arrived*, true and the ones which render the formula, x *is hungry*, true are disjoint. The conjunction of these two formulae is false according to (29.3), since the composition of the former set of pairs of variable assignments with the second set yields the empty set. In other words, the formula in (27) is false in the model, again in harmony with one's intuition about the sentence in (26).

Before returning to the question of how DPL treats the pairs of sentences in (4) and (11), one should bear in mind that DPL and CQL agree, in every model, on the truth and falsity of closed formulae and open formulae which are substitution instances of classical propositional tautologies and contradictions; they disagree, insofar as CQL fails to assign truth or falsity to the remaining formulae, whereas DPL does assign truth and falsity to some of them.

Let us now turn to the sentences in (11). Their renditions into the language of CQL are these:

(30.1) $\exists x$ x will win \$1,000 $\vee$ $\exists z$ $\neg$ z takes part
(30.2) $\exists x$ x will win \$1,000 $\vee$ $\neg$ x will take part

Since the first formula has no free variables, its truth-conditions are precisely those of CQL. The second formula, however, has a free variable. Clearly, the pronoun in the sentence in (11.2) has *someone* as its antecedent. By the semantics of DPL, the permissible values of the pronoun, or the third occurrence of the variable x, should be those values which render the open formula corresponding to the first clause true. However, clause (29.4) does not make the choice of variable assignment for the second disjunct dependent on the choice of variable assignment for the first. In this way, DPL fails to permit the formula in (30.2) to mimic the antecedence relation for the sentence in (4.2).

Like Kamp and Reyle, Groenendijk and Stokhof (1991: §5) are aware of the fact that noun phrases without definite reference occurring in the first of two clauses connected by *or* may, sometimes, serve as antecedents for pronouns occurring in the second. To handle such cases, the authors suggest an alternative for clause (29.4) which renders the choice of variable assignment for the second disjunct dependent on the choice of variable assignment for the first.[9] The essence of the solution is to define a dynamic negation and then to use the usual equivalences to define the dynamic version of other static connectives. Dynamic negation, which we shall denote by the symbol '$\sim$', is interpreted as complementation on $Pow(V^2)$. Internally dynamic disjunction can then be defined as $\alpha \sqcup \beta = \sim (\sim \alpha \wedge \sim \beta)$.

The sentence in (11.2) is then rendered, not as in (30.2), but as in (31) below:

(31) $\exists x$ x will win \$1,000 $\sqcup \sim x$ will take part

However, such a solution has the result that, when the semantics of DPL renders the sentence in (11.1) true, it renders that sentence in (31) true, contrary to intuitions.

[9] For a full treatment of these and other related problems using DPL, see Dekker 1993.

(See Appendix 2.)

Let us now turn to the sentences in (4). Their closest renditions into the language of CQL are these:

(32.1) $\forall x$ x takes part $\rightarrow$ $\exists z$ z will win \$1,000
(32.2) x takes part $\rightarrow$ $\exists x$ x will win \$1,000

Since the first formula has no free variables, its truth-conditions are precisely those of CQL. The second formula, however, has a free variable. Clearly, the pronoun in the sentence in (4.2) has *someone* as its antecedent. Again, however, the semantics of DPL does not permit the formula in (32.2) to mimic the antecedence relation found in (4.2); for, the clause (29.5) does not make the choice of variable assignment for the protasis dependent on the variable assignment for the apodosis.

The obvious solution is to define a new connective which permits the suitable dependence. Again, one can use dynamic negation to define such a new connective: $\alpha \leftarrow \beta = \sim (\sim \alpha \wedge \beta)$.

The sentence in (4.2) is then rendered, not as in (32.2), but as in (33) below:

(33) $\exists x$ x will win \$1,000 $\leftarrow$ x takes part

Such a solution, however, renders both sentences in (4) as semantically equivalent, contrary to intuitions. (See Appendix 3.)

In brief, then, DPL offers no improvement over the naive application of CQL to the sentences in (4) and (11).

5 Descriptive Pronouns and Peirce's Puzzle

What happens when the pronouns in the sentences in (4.2) and (11.2) are treated as descriptive pronouns? To answer this question, one must first know what descriptive pronouns are. As their name suggests, they are definite descriptions, albeit degenerate ones. What that amounts to here is this: a grammatically singular, descriptive pronoun denotes the unique individual satisfying the term of its antecedent and the open clause obtained by deleting its antecedent from the clause in which it occurs.[10]

To see how descriptive pronouns work, consider the following sentences.

(34.1) Harry bought *a carpet* and John cleaned *it*.
(34.2) Harry bought a carpet which John cleaned.

The first sentence implies that there is exactly one carpet relevant, while the second does not. The semantics for descriptive pronouns delivers this contrast. In (34.1), the antecedent of *it* is *a carpet*. The required open clause is *Harry bought* __. So, the denotation of *it* is the unique carpet such that Harry bought it. This rule of interpretation implies that the following is a good, though admittedly long-winded, paraphrase of the sentence in (34.1):

(35) Harry bought a carpet and John cleaned the carpet Harry bought.

How does this analysis apply to Peirce's puzzle? Consider again the sentence in (5.2). Here, the open clause is __ *will win \$1,000*. The denotation of the pronoun *he* is the unique person who will win \$1,000. Under the circumstances of evaluation

[10] Only as much of the theory of descriptive pronouns is presented here as is necessary for the treatment of sentences under examination. For a comprehensive presentation, see Neale 1990 ch. 5.

specified by Read, the sentence in (4.2) need not be true: after all, the participation in the sweepstakes of the person who will be the winner of a $1,000 should everyone take part, is not itself sufficient for there to be a winner of a $1,000, since someone else may decide not to take part, thereby reducing the winnings to less than $1,000.

The same considerations show that the sentence in (11.2) may be false, though under the circumstances of evaluation the one in (11.1) is true, for the person who will win the $1,000 should everyone take part, may participate and win, but not win $1,000, because someone else does not participate.

In summary, the analysis of the pronouns in (4.2) and (11.2) as descriptive pronouns shows neither the sentence in (4.2) is entailed by the one in (4.1) nor is the one in (11.2) entailed by the one in (11.1), and thus Peirce's puzzle is furnished with an intuitively satisfying solution.

But the solution is not without its disappointing consequences. Recent advocates of descriptive pronouns – for example, Neale (1990) – have maintained, with a great deal of plausibility, that descriptive pronouns appear in syntactic configurations disjoint from those in which bound pronouns appear.[11]

Now, for a wide range of data, this disjointness seems to be the case. However, if the correct analysis of the pronouns in the examples under investigation here is that of descriptive pronouns, then this additional claim must be given up. The following sentences are perfectly isomorphic syntactically,

(36.1) Each person will win if he buys a ticket.
(36.2) Some person will win if he buys a ticket.

yet the pronoun in the first is a bound pronoun, while the one in the second is a descriptive pronoun.

As appealing as the would-be empirical generalization that bound pronouns and descriptive pronouns occur in disjoint syntactic configurations might be, independent evidence, brought to light by McKay (1991), has already overturned such a generalization.[12]

(37.1) Each man in the department thinks that he should attend the meeting.
(37.2.1) Each man in the department thinks that they should meet.
(37.2.2) *Each man in the department* thinks that *the men in the department* should meet.

These sentences too are, in all relevent respects, isomorphic – in each case the pronoun is c-commanded by its antecedent; yet the pronoun in the sentence in (37.2.1) is a descriptive one, as borne out by its paraphrase in (37.2.2), whereas the one iin the sentence in (37.1) is a bound pronoun.

Another disappointing consequence is that the descriptive content of the descriptive pronouns found in the sentences in (4.2) and (11.2) is not exhausted by the content of their antecedent clauses: in each case, the descriptive content of the pronoun is subject to a subjunctive condition, namely, the condition that everyone take part. Now, Evans himself recognized that descriptive pronouns may be used without their content being exhausted by the content of their antecedent clauses. Nonetheless, invoking such additional material without grounding its invocation in empirically justified principles is certainly a weakness of the analysis. However, Emmon Bach has pointed out, independently of the facts under discussion here, uses

[11] The claims are formulated in terms of the syntactic relation of c-command and involve details of syntactic analysis which need not detain us here. See Neale 1990 Ch. 5.3 for details.
[12] See McKay (1991) for different examples.

of definite noun phrases whose meaning cannot be determined without invoking a subjunctive, and even a counterfactual, condition.

(38.1) Jill died before she finished her dissertation.
(38.2) Jill died before she finished the dissertation she would have written had she lived long enough.

(39.1) Fred prevented the fire.
(39.2) Fred prevented the fire which would have occurred had he not acted in some way (evident from the context of use).

6 Conclusion

Three competing treatments of the problem of donkey anaphora have been examined in light of a puzzle due to Charles Peirce. I showed that this puzzle is not one pertaining to the semantics of the English subordinating conjunction *if*, but one pertaining to the correct analysis of pronouns with indefinite noun phrases as antecedents, precisely the configuration of central interest to the study of donkey anaphora. Peirce's puzzle has brought to light inadequacies in the competing approaches to so-called donkey anaphora, though on the brief examination offered here, it seems that the analysis in terms of descriptive pronouns has the advantage over an analysis which employs the anaphoric devices of either DRT or DPL.

Appendix 1: The Equivalence of the Formulae in (3)

This proof involves three well-known equivalences:

1. $\neg(\alpha \wedge \neg\beta) \leftrightarrow \neg(\alpha \rightarrow \beta)$
2. $\forall\mu\,\neg\psi(\mu) \leftrightarrow \neg\,\exists\mu\,\psi(\mu)$
3. $\forall\nu\,(\,\phi(\nu) \wedge \neg\psi(\nu)\,) \leftrightarrow \forall\nu\,\phi(\nu) \wedge \forall\mu\,\neg\psi(\mu)$

The proof is as follows:

(3.1)	$\forall\nu\,\phi(\nu) \rightarrow \exists\mu\,\psi(\mu)$	
	$\neg\,(\,\forall\nu\,\phi(\nu) \wedge \neg\,\exists\mu\,\psi(\mu)\,)$	by (1)
	$\neg(\,\forall\nu\,\phi(\nu) \wedge \forall\mu\,\neg\psi(\mu))$	by (2)
	$\neg\forall\nu\,(\,\phi(\nu) \wedge \neg\psi(\nu)\,)$	by (3)
	$\exists\nu\neg\,(\,\phi(\nu) \wedge \neg\psi(\nu)\,)$	by (2)
(3.2)	$\exists\nu\,(\,\phi(\nu) \rightarrow \psi(\nu)\,)$	by (1)

Appendix 2: DPL and the formulae in (30.1) and (31)

Exactly the same models render the formula in (30.1) true, whether it is evaluated in terms of DPL or it is evaluated in terms of CQL, since the formula is closed. In CQL, exactly the same models render the formula in (30.1) and the formula $\exists x\,(\,x$ *will win $1,000* $\vee \neg\, x$ *will take part*$)$. Finally, the truth of this last formula in CQL

entails the truth of the formula in (31), as shown below. (In the proof below, 'W' stands for *will win $1,000* and 'T' stands for *will take part*).

Suppose $\exists x\,(Wx \vee \neg Tx)$ is true in CQL. Then, $[\exists x\,(Wx \vee \neg Tx)]_M = V$. By (28.6), $[Wx \vee \neg Tx]_M \neq \emptyset$, and hence contains a variable assignment w. According to (28.4), either $w \in [Wx]_M$ or $w \in [\neg Tx]_M$.

CASE 1: Suppose that $w \in [Wx]_M$. Suppose further that, for some v in V, $\langle v, w\rangle \notin |\exists x\, Wx \sqcup \sim Tx|_M$. By the definition of $\sqcup$, $\langle v, w\rangle \notin |\sim(\sim \exists x Wx \wedge \sim\sim Tx)|_M$. Since dynamic negation is interpreted as complementation on $Pow(V^2)$, it follows that $\langle v, w\rangle \in |\sim \exists x Wx \wedge \sim\sim Tx|_M$, and further that $\langle v, w\rangle \in |\sim \exists x Wx \wedge Tx|_M$. Since, by (29.1), $|Tx|_M \subseteq I_V$ (where I_V is the identity graph on V^2), it follows by (29.3) that $\langle v, w\rangle \in |\sim \exists x Wx|_M$, and thus that $\langle v, w\rangle \notin |\exists x Wx|_M$. By (29.6), one concludes that $\langle w, w\rangle \notin |Wx|_M$. However, by hypothesis, $w \in [Wx]_M$. And so, by (29.1), $\langle w, w\rangle \in |Wx|_M$. ¿From this contradiction, it follows that, for each v in V, $\langle v, w\rangle \in |\exists x Wx \sqcup \sim Tx|_M$. Therefore, $\exists x Wx \sqcup \sim Tx$ is true in DPL.

CASE 2: Suppose that $w \in [\neg Tx]_M$. Then, by (28.2), $w \notin [Tx]_M$. And, by (29.1), $\langle w, w\rangle \notin |Tx|_M$. Since dynamic negation is interpreted as complementation on $Pow(V^2)$, $\langle w, w\rangle \notin |\sim\sim Tx|_M$. Thus, for each $v \in V$, $\langle v, w\rangle \notin |\sim \exists x Wx \wedge \sim\sim Tx|_M$. And so, for each $v \in V$, $\langle v, w\rangle \in |\sim(\sim \exists x Wx \wedge \sim\sim Tx)|_M$. By the definition of $\sqcup$, for each $v \in V$, $\langle v, w\rangle \in |\exists x Wx \sqcup \sim Tx|_M$. Therefore, $\exists x Wx \sqcup \sim Tx$ is true in DPL.

Therefore, whether $w \in [Wx]_M$ or $w \in [\neg Tx]_M$, $\exists x Wx \sqcup \sim Tx$ is true in DPL.

Appendix 3: DPL and the formulae in (32.1) and (33)

The formulae in (32.1) and (33) are rendered true by exactly the same models. Since the formula in (32.1) is closed, precisely the same models render it true, whether it is evaluated in terms of DPL or it is evaluated in terms of CQL. Moreover, in CQL, exactly the same models render the formula in (32.1) and the formula $\exists x\,(x$ *will take part* $x \rightarrow x$ *will win $1,000*$). Finally, precisely the same models render true this formula in CQL and the formula in (33) in DPL, as shown below. (In the proof below, 'W' stands for *will win $1,000* and 'T' stands for *will take part*).

$\Rightarrow$ Suppose $\exists x\,(Tx \rightarrow Wx)$ is true in CQL. Then, $[\exists x\,(Tx \rightarrow Wx)]_M = V$. By (28.6), it follows that $[Tx \rightarrow Wx]_M \neq \emptyset$, and hence contains a variable assignment w. According to (28.4), either $w \in [Wx]_M$ or $w \in [\neg Tx]_M$.

CASE 1: Suppose that $w \in [Wx]_M$. Suppose further that, for some v in V, $\langle v, w\rangle \notin |\exists x Wx \leftarrow Tx|_M$. By the definition of $\leftarrow$, $\langle v, w\rangle \notin |\sim(\sim \exists x Wx \wedge Tx)|_M$. Since dynamic negation is interpreted as complementation on $Pow(V^2)$, it follows that $\langle v, w\rangle \in |\sim \exists x Wx \wedge Tx|_M$. Since, by (29.1), $|Tx|_M \subseteq I_V$ (where I_V is the identity graph on V^2), it follows by (29.3) that $\langle v, w\rangle \in |\sim \exists x Wx|_M$, and thus that $\langle v, w\rangle \notin |\exists x Wx|_M$. By (29.6), one concludes that $\langle w, w\rangle \notin |Wx|_M$. However, by hypothesis, $w \in [Wx]_M$. And so, by (29.1), $\langle w, w\rangle \in |Wx|_M$. ¿From this contradiction, it follows that, for each v in V, $\langle v, w\rangle \in |\exists x Wx \leftarrow Tx|_M$. Therefore, $\exists x Wx \leftarrow Tx$ is true in DPL.

CASE 2: Suppose that $w \in [\neg Tx]_M$. Then, by (28.2), $w \notin [Tx]_M$. And, by (29.1), $\langle w, w\rangle \notin |Tx|_M$. Thus, for each $v \in V$, $\langle v, w\rangle \notin |\sim \exists x Wx \wedge Tx|_M$. And so, for each $v \in V$, $\langle v, w\rangle \in |\sim(\sim \exists x Wx \wedge Tx)|_M$. By the definition of $\leftarrow$, for each $v \in V$, $\langle v, w\rangle \in |\exists x Wx \leftarrow Tx|_M$. Therefore, $\exists x Wx \leftarrow\sim Tx$ is true in DPL.

Therefore, whether $w \in [Wx]_M$ or $w \in [\neg Tx]_M$, $\exists x Wx \leftarrow Tx$ is true in DPL.

$\Leftarrow$ Suppose $\exists x\,(Tx \rightarrow Wx)$ is false in CQL. Then, $[\exists x\,(Tx \rightarrow Wx)]_M = \emptyset$. So, $[Tx]_M = V$ and $[Wx]_M = \emptyset$. On the one hand, since $[Tx]_M = V$, it follows by (29.1) that $|Tx|_M = I_V$ (where I_V is the graph of the identity relation on V). On the other hand, since $[Wx]_M = \emptyset$, it follows, by (28.6), that $|\exists x\, Wx|_M = \emptyset$, and it follows further, by the definition of dynamic negation, that $|\sim \exists x\, Wx|_M = V^2$. Now, since $|\sim \exists x\, Wx|_M = V^2$ and $|Tx|_M = I_V$, it follows by (29.3), $|\sim \exists x\, Wx \wedge Tx|_M = I_V \circ V^2 = V^2$. And so, according to the definition of dynamic negation, $|\sim(\sim \exists x Wx \wedge Tx)|_M = \emptyset$. But, by the definition of $\leftarrow$, it follows that $|\exists x Wx \leftarrow Tx)|_M = \emptyset$. Therefore, $\exists x Wx \leftarrow Tx$ is false in DPL.

References

Dekker, P.: 1993, *Transsentential Meditations: Ups and downs in dynamic semantics.* dissertation, University of Amsterdam, The Netherlands: (ILLC dissertation series: 1993, n. 1)

Evans, G.: 1977, Pronouns, Quantifiers, and Relative Clauses, *Canadian Journal of Philosophy* 7(3), 467-536

Geach, P.: 1962, *Reference and Generality*, Ithaca, New York: Cornell University Press (2nd ed. revised, 1968)

Groenendijk, J., Janssen, T. and Stokhof, M. (eds): 1981, *Formal Methods in the Study of Language*, Amsterdam, The Netherlands: Mathematical Center, reprinted in Groenendijk *et al* (eds) 1984, 1-41

Groenendijk, J., Janssen, T. and Stokhof, M. (eds): 1984, *Truth, Interpretation and Information*, Dordrecht, The Netherlands: Foris

Groenendijk, J. and Stokhof, M.: 1991, Dynamic Predicate Logic. *Linguistics and Philosophy*, 14(1), 39-100

Hartshorne, C. and Weiss, P. (eds): 1933, *Collected Papers of Charles Sanders Peirce*, 4 vols. Cambridge, Massacusetts: Harvard University Press

Heim, I.: 1982, *The Semantics of Definite and Indefinite Noun Phrases*, Ph.D. thesis, University of Massachusetts, Amherst, Published: Heim 1987.

Heim, I.: 1987, *The Semantics of Definite and Indefinite Noun Phrases*, New York, New York: Garland Press, reprint of Heim 1982

Kamp, H.: 1981, A Theory of Truth and Semantic Representation, In Groenendijk *et al.* (eds) 1981, 1-40.

Kamp, H. and Reyle, U.: 1993, *From Discourse to Logic. Introduction to Model Theoretic Semantics of Natural Language, Formal Logic, and Discourse Representation Theory*, Dordrecht, The Netherlands: Kluwer Publishing Company (*Studies in Linguistics and Philosophy* 42)

Keenan, E.L. (ed): 1975, *Formal Semantics of Natural Language*, Cambridge, England: Cambridge University Press.

Lewis, D.: 1975, Adverbs of Quantification, in Keenan (ed) 1975, 3-15.

McKay, T.: 1991, Unbound Pronouns, a paper presented at the December 1991 meeting of the Eastern Division of the American Philosophical Association.

Neale, S.: 1990, *Descriptions*, Cambridge, Massachusetts: The MIT Press.

Quine, W. van Ormen: 1960, *Word and Object.* Cambridge, Massachusetts: The MIT Press.

Read, S.: 1992, Conditionals Are Not Truth-Functional: An Argument from Peirce, *Analysis* 52(1), 5-12.

Dynamic Epistemic Logic

Willem Groeneveld

University of Amsterdam

1 Introduction

Epistemic Logic, broadly conceived, studies the logical properties of expressions like 'to know that', 'to believe that', 'to have the information that'.[1] The normal way of approaching this problem is to extend propositional logic with a unary sentential operator. The logical behavior of this operator can then be fixed by a set of axioms, or by a semantics for the operator, and preferably by both. The objective of this paper is to extend epistemic logic to a system that takes account of *changes* in knowledge, or belief, or information.

There are two main motivations why such an extension of epistemic logic is desirable. First, it is just a fact of life that knowledge, or belief, or information, is not constant, but changes. Without this dynamic aspect there would almost be no point in reading, or in doing an experiment, or in teaching (and so on). Thus, an epistemic logic that also takes account of the aspect of change can be expected to be applicable to a wider range of problems.

The second motive I have for a dynamic extension of epistemic logic is based on the expectation that such a logic will be a suitable basis for a system of formal pragmatics. By the latter I mean a logical framework for studying *information exchange*, as it occurs in human conversation, or in message transmission by electronic devices. The theory of information exchange of the kind I envisage will provide at least the following three features. It will be

- **Dynamic:** sentences are interpreted as functions on information states
- **Multi-Agent:** there are at least two agents
- **Higher-Order:** information states of actors not only specify information about the world but also information about the other actors information states

This full program, of developing an epistemic logic that is multi-agent, dynamic and higher-order, followed by a use of this logic in a system of information exchange, will not be realized in this paper. Here I will only deal with the first stage, and even there I will mainly discuss the problem of defining a reasonable notion of update over higher-order information states of one actor. It will turn out that this is already a non-trivial problem.

I propose the following syntax of the language of dynamic epistemic logic (DEL).

Definition 1.1 (Language of DEL) Let $\mathcal{A}, \mathcal{P}$ be non-empty sets, of actors and propositional atoms, respectively. Then the language of Dynamic Epistemic Logic, $DEL(\mathcal{A}, \mathcal{P})$, consists of the formulae ϕ defined by

$$\phi ::= p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \Box_a\phi \mid [\phi_1]_a\phi_2$$

where $p \in \mathcal{P}$, $a \in \mathcal{A}$. The classical fragment of this language, which consists precisely of the formulae built up from atoms, $\neg$, and $\wedge$ only, is called $\mathcal{L}_0$. The connectives $\bot, \top, \vee, \rightarrow, \leftrightarrow, \Diamond$ have their usual classical definitions. □

The intended interpretation of $\Box_a\phi$ is that agent a has the information that ϕ. The intended meaning of $[\phi_1]_a\phi_2$ is that an update of a's information with ϕ_1 results

[1] The classic of epistemic logic is [Hin62]; see [FHMV95] for a recent monograph.

in a situation where ϕ_2 is true. Note that the language of DEL is simple extension of the standard language of epistemic logic, and that it is also very similar to the language of Propositional Dynamic Logic.

The remainder of this paper is devoted to the problem of finding a suitable semantics for the language DEL.[2]

2 Constraints on Updates

Before we face the task of defining a semantics for DEL we take an abstract perspective and try to formulate desiderata for a general theory of Dynamic Epistemic Logic. I will do this by making some minimal assumptions about the form of the truth definition for formulae of DEL. I will then list some principles in the language of DEL that come out as valid under these minimal assumptions. Finally I discuss some extra postulates on updates that seem reasonable, and try to determine which extra assumptions on the semantics are needed to validate the extra principles.

The language of DEL of definition 1.1 contains only one extra construction compared to the standard vocabulary of modal logic: the formulae of the form $[\phi]\psi$. The intended interpretation of this formula is that after the update with ϕ, ψ is true. The following minimal assumptions about the semantics will make this work. Consider structures of the form

$$(\mathcal{S}, (\mathcal{R}_\phi)_{\phi \in DEL}, \mathcal{R}, \mathcal{V})$$

The set $\mathcal{S}$ consists of 'informational situations', that determine facts and information. Relative to these situations we assign truth conditions to formulae. $\mathcal{V}$ is a valuation function that (totally) interprets atomic formulae in states in $\mathcal{S}$. The Boolean connectives receive the classical truth conditions. $\mathcal{R}$ is a binary relation over $\mathcal{S}$, to which the meaning of the $\Box$ relates in the same way as in standard modal logic. And for each ϕ, $\mathcal{R}_\phi$ is also a binary relation over $\mathcal{S}$, which is called the update relation of ϕ. The truth conditions for update modalities are then given by

$$s \models [\phi]\psi \text{ iff } \forall t: \text{ if } s\mathcal{R}_\phi t \text{ then } t \models \psi$$

The idea is that $s\mathcal{R}_\phi t$ means that s is just like t except that the information in t is the information in s updated with ϕ. The real problem of this paper is of course to arrive at a concrete definition of $\mathcal{R}_\phi$. However, in the present section we will take an abstract approach, by reviewing ways in which axioms in the language of DEL can constrain these relations $\mathcal{R}_\phi$.

The first constraint we consider does not put a restriction on the class of structures: the $\Box$, as well as the expressions '$[\phi]$' behave as normal modal operators:

Normality

- if $\models \psi$ then $\models \Box\psi$
- $\models \Box(\psi \to \chi) \to (\Box\psi \to \Box\chi)$
- if $\models \psi$ then $\models [\phi]\psi$
- $\models [\phi](\psi \to \chi) \to ([\phi]\psi \to [\phi]\chi)$

(Here $\models$ means true in all situations in all structures.) For this reason we will from now on refer to the expressions '$[\phi]$' as 'update modalities'.

Next consider the constraint that updates are functions. This is expressed by

[2]The present paper is a shortened version of chapter 4 of [Gro95]. For proofs I refer the reader to that chapter.

Functionality $\models \neg[\phi]\psi \leftrightarrow [\phi]\neg\psi$

From now on we will assume this constraint, and use $[\![\phi]\!]$ to refer to the update function for ϕ. The semantic clause for update modalities can then be rewritten as

$$s \models [\phi]\psi \text{ iff } s[\![\phi]\!] \models \psi$$

where we have used the postfix notation $s[\![\phi]\!]$ for the result of applying the function $[\![\phi]\!]$ to s. It is then clear that $[\phi]\psi$ means that after *the* update with ϕ, ψ is true.

And if we want updates to change information only, and not facts, we will have

Information Change Only $\models \psi \leftrightarrow [\phi]\psi$, if ψ is classical

Here ψ is classical if it does not contain $\Box$ or $[\cdot]$. We will refer to the three principles of Normality, Functionality, and Information Change Only, as the Minimal Principles.

If we want more constraints, one line of thought is to look at Veltman's update semantics [Vel91], which defines a notion of updates over first-order information states (sets of possible worlds), and generalize some of the properties of these updates to a higher-order setting.

Eliminativity $\models [\phi]\Diamond\psi \to \Diamond\psi$

Success $\models [\phi]\Box\phi$

Consistent Elimination $\models [\phi]\Diamond\psi \to \Diamond(\phi \wedge \psi)$

Minimality $\models \Diamond(\phi \wedge \psi) \to [\phi]\Diamond\psi$

Ramsey Test $\models \Box(\phi \to \psi) \leftrightarrow [\phi]\Box\psi$

Information Increase $\models \Box\phi \to [\psi]\Box\phi$

Descriptive Information Increase $\models \Box\phi \to [\psi]\Box\phi$, if ϕ is classical

Informational Equivalence $\models \Box(\phi \leftrightarrow \psi) \to ([\phi]\chi \leftrightarrow [\psi]\chi)$

The Eliminativity constraint expresses that a possibility that is present after an update, must have been there before the update. Success expresses that after an update with ϕ you have the information that ϕ. The import of Consistent Elimination is that only possibilities remain that are consistent with the new information. And Minimality expresses that updates are minimal information changes: they preserve all possibilities that are consistent with the new information.

Some connections between these principles are:

Lemma 2.1 Given the Minimal Principles,

1. Eliminativity and Success imply Consistent Elimination, and vice versa.
2. Minimality and Consistent Elimination are equivalent to the Ramsey Test
3. The Ramsey Test implies Success
4. Consistent Elimination implies Success □

However, although this list of extra principles may be reasonable if we only consider a change of first-order information, in case of higher-order information the situation is different. Consider the principle of Information Increase, which expresses the most central idea of updates, that they increase information. But in the case that ϕ itself expresses higher-order information, especially some lack of

information as in the case that $\phi = \neg\Box p$, this odd. We certainly have a strange agent if he cannot learn that is raining, for the very reason that he initially is aware of the fact that his information about the weather is inconclusive! This shows that we may want to weaken the principle to Descriptive Information Increase, which is unproblematic.

It is possible to extend the minimal assumptions we have been making on the structures for DEL in such a way that the postulates all come out as valid. In fact we can see the full list of the above principles as giving a characterization of the notion of *eliminative update*. A concrete example of such a logic, called Eliminative K, will be developed in section 3. In this semantics the simple eliminative update strategy of Update Semantics is generalized to a setting with higher-order information. As we just observed, this strategy has some undesirable effects. In section 4 we will therefore look at an alternative semantics, which will rely on a notion of *conscious update*. Such an update can be informally described as a process of 'knowingly changing one's mind', in which the new information is also reflected at higher-order levels.

3 Eliminative Updates

For an eliminative semantics for DEL the most obvious choice of structures to consider are the standard Kripke models for epistemic logic. However, it turns out that Kripke models are problematic for a definition of update. The reason is that in a Kripke model, worlds or arrows (or whole constellations of these) may play more than one role. For example, a world (or an arrow) may be relevant for the determination of first order information and at the same time for the determination of higher order information. This means that an update with a piece of information of order k may have an undesirable side effect on the information of order n.[3] Thus a type of structure in which the different levels of higher-order information are clearly separated seems more suitable. The *modal structures* developed by Fagin and Vardi in [FV85] (also see [FHV91]) provide such a hierarchical structure.[4]

By way of informal introduction to modal structures, consider the notion of information in update semantics, where an information state is a set of possible worlds. This set of world is intended to model the information an agent has of the real world. A generalization of this schema could be: information about an object x of type α is a set of things of type α, namely those things you cannot distinguish from x on your current information. We can use this general scheme to define what higher-order information is. Given a set of possible worlds $\mathcal{W}$, define

$$\mathcal{I}_1 = Pow(\mathcal{W})$$

$$\mathcal{I}_{n+1} = Pow(\mathcal{I}_n)$$

So first-order information is information about the real world, and consists of a set of possible worlds. Second-order information is information about first-order information, and consists of a set of first-order information states, so it is a set of sets of possible worlds. And so on. We can then define an information state as an information function that determines for each natural number n an n-th order information set:

$$f(n) \in \mathcal{I}_n \text{ for all } n \in \omega$$

Unfortunately this proposal is not correct, since it does not deal properly with connections between higher-order and lower-order information, as was observed in

[3] See[Gro95], pp. 129–140 for an extensive discussion of the problems with Kripke models.

[4] See [HD89] for some useful observations on the connections between Kripke models and modal structures; also see [Gro95], pp. 141-149.

[FHV91].[5] For example, a 'mixed' sentence of the form $p \vee \Box q$ will be problematic. If f is an information function, then the first disjunct p will relate to $f(1)$, but the second disjunct $\Box q$ will relate to $f(2)$. But $f(1)$ and $f(2)$ are in principle unrelated. This is wrong because the most natural reading of $p \vee \Box q$ is as a connection between first and second order information. [6]

A possible repair is to add the missing links between higher and lower order information. As the example suggests, second order information should be seen as information about the individual worlds in the first order information set rather than as information about the first-order set as a whole. This can be achieved by defining second order information as

$$\mathcal{I}_2 = Pow(W \times Pow(\mathcal{W}))$$

So second order information is information not only about first order information, but about possible constellations of facts and first order information.[7]

This view on higher-order information can be generalized to higher levels: $n+1$-th order information is an $n+1$-ary relation between facts and 1st order information and ... and n-th order information. Formally:

$$\mathcal{I}_1 = Pow(\mathcal{W})$$

$$\mathcal{I}_{n+1} = Pow(\mathcal{W} \times \mathcal{I}_1 \times \cdots \times \mathcal{I}_n)$$

Now this recursion matches exactly the 1 agent case of the definition of modal structures. For the Multi-agent case the definition runs as follows.

Definition 3.1 Let P be a set of propositional variables, $\mathcal{A}$ be a set of actors. The set of possible worlds for P is $W = Pow(P)$. *Informational situations* of any finite order, and *Information assignments* of finite order ≥ 1, are inductively defined as follows:

- $S_0 = W$
- $I_{n+1} = Pow(S_n)^{\mathcal{A}}$
- $S_{n+1} = \{(s_0, i_1, \ldots, i_{n+1}) \in S_0 \times I_1 \times \ldots \times I_{n+1} | EXT_{n+1}(s_0, i_1, \ldots, i_{n+1})\}$

Here the clause $EXT_{n+1}(s_0, i_1, \ldots, i_{n+1})$ is:

Extension for all $a \in \mathcal{A}$, and for all k with $1 \leq k \leq n+1$:
$(j_0, \ldots, j_{k-2}) \in i_{k-1}(a)$ iff $\exists j_{k-1} : (j_0, \ldots, j_{k-2}, j_{k-1}) \in i_k(a)$

A *information structure* is an ω-sequence $(s_0, i_1, \ldots, i_k, \ldots)$ such that for each n, its initial $(s_0, i_1, \ldots, i_n)$ is an n-th order situation, that is, a member of S_n. S_ω is the set of all information structures. □

Notice that the objects we call information structures are formally identical to what are called modal structures in [FV85, HD89]. This is motivated by the informational perspective of our investigation. I also chose for the term 'situation'

[5] The authors mention this in reference to the paper [vEBGS81], where essentially this notion of higher-order information is employed.

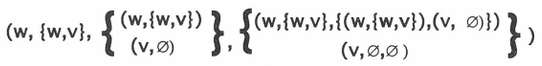
[6] This does not strictly follow, but is nevertheless almost unavoidable. Suppose we define a join on information functions by pointwise union: $f \sqcup g = \lambda i \cdot f(i) \cup g(i)$. Suppose moreover we use this join to interpret disjunction: $f[p \vee \Box q] = f[p] \sqcup f[\Box q]$. Finally suppose that the p-update only effects $f(1)$ and the $\Box q$-update only effects $f(2)$, and that both updates result in a subset of the level on which they act. Then it follows that for all f, $f[p \vee \Box q] = f$, so $p \vee \Box q$ is always accepted. This is clearly wrong.

[7] An update with a mixed formula of the form $p \vee \Box q$ will now be unproblematic. From a second order information state $\Sigma \subseteq \mathcal{W} \times Pow(\mathcal{W})$, the update will eliminate those pairs (w, σ) such that either $w \not\models p$ or $\exists v \in \sigma : v \not\models q$.

rather than 'world' since I think the phrase 'informational situation' sounds more natural than 'informational world'. However, we will not conceive of situations as being partial in same the sense as in Situation Semantics (see [BP83]). That is, our situations completely determine what the facts are.

Also note that the indexing regime in definition 3.1 works as follows: a situation of order k is a $k+1$-tuple, and specifies the facts and the information up to and including order k. So the order of a situation corresponds to the maximal order of the information that is specified in that situation.[8]

Here is a situation s of order 3:

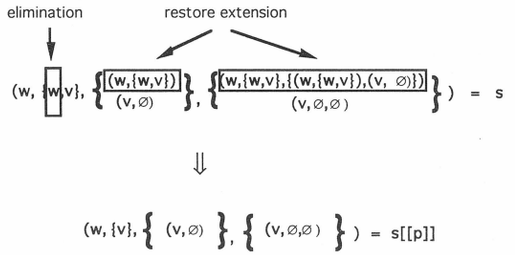


For convenience we have chosen a one agent case, so we in fact need not mention the agent. Suppose p is false in w but true in v. Since w is the first constituent of s and determines the facts of s, p is not true in s. The second constituent of s, the set $\{w, v\}$, determines the first-order information that our anonymous agent has in s. This first-order information consists of the ways in which she thinks the facts of the situation she is in may be. Because p is false in w but true in v, she does not have the information whether p. Her second-order information is determined by the second constituent of s, $\{(w, \{w, v\}), (v, \emptyset)\}$. This information concerns the ways in which she thinks the facts and her first-order information may be. She sees two possibilities: one in which p is false and she does not have the information whether p (the first-order situation $(w, \{w, v\})$), and one in which p is true but her first-order information is inconsistent (the option $(v, \emptyset)$.).

The idea of the Extension Constraint in definition 3.1 is that a higher-order informational option refines some lower-order option, and that any lower-order option is refined by some higher-order option. This is also be illustrated in the example. Consider the second-order option $(w, \{w, v\})$. This option extends the first-order option w, and is extended by the third order option $(w, \{w, v\}, \{(w, \{w, v\}), (v, \emptyset)\})$. Likewise, the second-order option $(v, \emptyset)$ extends v and is extended by $(v, \emptyset, \emptyset)$.

In the multi-agent case, the structures become a bit more complex. Roughly, whenever we have a k-th order information set in the one agnent case, we now have a function from actors to k-th order information sets.

Before we turn to a formal definition of eliminative updates over information structures, we briefly discuss an example of such an update (again for the one-agent case). Consider the following third order situation, with $w \not\models p$, $v \models p$. Then an update with p can be pictured as follows:



The update with p is first carried out eliminatively on level 1, and then the higher-order levels that have lost their first order basis are also removed. Likewise, an update with $\Box p$ will first act on level 2, but consequently the options at level 1 that have lost their higher-order extension, as well as the options at levels higher than 2 that have lost their basis, have to be removed by a extension restorement operation.

We now turn to a formal definition of eliminative updates.

Definition 3.2 (Notations)

- If s is an n-tuple and $i < n \leq \omega$ then s_i is the $i+1$-th constituent of s.
- If s is an n-tuple and $k < n \leq \omega$ then $(s)_k = (s_0, \ldots, s_k)$.
- If s is an ω-sequence then s will also be written as $(s)_\omega$. □

So for an $n+1$-tuple s we have $s = (s_0, \ldots, s_n) = (s)_n$.

Definition 3.3 (Eliminative Information ordering) Let $s, t \in S_k$, $a \in \mathcal{A}$. then $s[a]t$ iff $s_0 = t_0$ and for all $b \in \mathcal{A}$ with $b \neq a$, and all i with $1 \leq i \leq k$, $s_k(b) = t_k(b)$. Then define s is at least as strong as t for a, notation $s \sqsubseteq_a t$, if $s[a]t$ and for all i with $1 \leq i \leq k$, $s_k(a) \subseteq t_k(a)$. □

So $s[a]t$ means that s and t differ at most in the information assigned to a. $s \sqsubseteq_a t$ means that the information of actor a in s is at least as strong as the information of a in t, while in all other respects, s and t are the same. Again this information ordering corresponds to an eliminative view on information growth: to have more information is to have less options.

Definition 3.4 Let $s \in S_k$, $t \in S_n$, $k \leq n$. Then t extends s, notation $s \leq t$, if $(s_0, \ldots, s_k) = (t_0, \ldots, t_k)$. □

The operation $\bullet_n^a$ of the next definition does the following: given an $i \subseteq S_{n-1}$, i.e. i is a piece of n-th order information, we set the n-th order information of actor a to i, and then restore the extension constraint.

Definition 3.5 (Revision and Extension Restorement) let $s \in S_k$, $i \subseteq S_{n-1}$, $n \leq k$, $a \in \mathcal{A}$. Then s revised by i for a (at level n) is defined as the sequence $s \bullet_n^a i = (t_0, \ldots, t_k)$, where

- $s[a](t_0, \ldots, t_k)$
- if $1 \leq j < n$ then $t_j(a) = \{s' \in s_j(a) \mid \exists t' \in i : s' \leq t'\}$
- $t_n(a) = i$
- if $n < j \leq k$ then $t_j(a) = \{s' \in s_j(a) \mid \exists t' \in i : t' \leq s'\}$ □

Lemma 3.6 let $s \in S_k$, $t \in S_m$, $i \subseteq S_{n-1}$, $n \leq k \leq m$, $a \in \mathcal{A}$.

1. If $i \subseteq s_n(a)$ then $(s \bullet_n^a i) \in S_k$.
2. If $i \subseteq s_n(a)$ then $s \bullet_n^a i \sqsubseteq_a s$.
3. If $s \leq t$ then $s \bullet_n^a i \leq t \bullet_n^a i$. □

Definition 3.7 (Modal Degree) $\#(p) = 0$, $\#(\neg\phi) = \#(\phi)$, $\#(\Box_a\phi) = \#(\phi) + 1$, $\#(\phi \wedge \psi) = max\{\#(\phi), \#(\psi)\}$, $\#([\phi]_a\psi) = max\{\#(\phi) + 1, \#(\psi)\}$. □

We are now ready to define the semantics:

Definition 3.8 (Semantics) Let $s \in S_k$. Then truth in s for formulae of degree $\leq k$ is inductively defined by:

[8] This differs from the usual way of indexing, for example in [FHV91], in which a k-ary world is a k-tuple that specifies the facts and the knowledge up to and including order $k-1$.

1. $s \models p$ iff $p \in s_0$
2. $s \models \neg\phi$ iff $s \not\models \phi$
3. $s \models \phi \wedge \psi$ iff $s \models \phi$ and $s \models \psi$
4. $s \models \Box_a\phi$ iff for all $t \in s_k(a) : t \models \phi$
5. $s \models [\phi]_a\psi$ iff $s \bullet_n^a [\![\phi]\!]_s^a \models \psi$, where $n = \#\phi + 1$, and $[\![\phi]\!]_s^a = \{t \in s_{\#\phi+1}(a) \mid t \models \phi\}$ □

Cluases 1 to 4 are just the standard semantics for modal logic on modal structures (see [FV85] or [FHV91]). Clause 5 expresses, modulo some notation, that $[\phi]_a\psi$ is true if and only if ψ is true after the update of agent a's information with ϕ.

The following is a simple generalization to the language of DEL of the Extension lemma of [FHV91].

Lemma 3.9 (Dynamic Extension Lemma)
For all formulae ϕ, all k, n with $\#\phi \leq k \leq n$, and all $s \in S_n$: $(s)_k \models \phi$ iff $s \models \phi$. □

So the truth of a sentence of degree k is settled at level k, and the levels of order higher than k are irrelevant.

Definition 3.10 (Validity)

1. If $s \in S_\omega$, then $s \models \phi$ iff $(s)_{\#\phi} \models \phi$
2. Let Γ be a set of formulae. Then $\Gamma \models_{MEK} \phi$ if and only if for all $s \in S_\omega$, if $s \models \psi$ for all $\psi \in \Gamma$, then also $s \models \phi$. □

It turns out that $\models_{MEK}$ has a fairly simple axiomatization.

Definition 3.11 (MEK) The system MEK (Multi-Agent Eliminative K) is defined by the following axioms and rules.

Axioms

A1 $\vdash \phi$, if ϕ is valid in classical propositional logic
A2 $\vdash \Box_a(\phi \to \psi) \to (\Box_a\phi \to \Box_a\psi)$
A3 $\vdash [\chi]_a(\phi \to \psi) \to ([\chi]_a\phi \to [\chi]_a\psi)$
A4 $\vdash \neg[\phi]_a\psi \leftrightarrow [\phi]_a\neg\psi$
A5 $\vdash p \leftrightarrow [\phi]_a p$, if p is an atom
A6 $\vdash [\phi]_a\Diamond_a\psi \to \Diamond_a(\phi \wedge \psi)$
A7 $\vdash \Diamond_a(\phi \wedge \psi) \to [\phi]_a\Diamond_a\psi$
A8 $\vdash \Box_a(\phi \leftrightarrow \psi) \to ([\phi]_a\chi \leftrightarrow [\psi]_a\chi)$
A9 $\vdash [\phi]_a\Box_b\psi \leftrightarrow \Box_b\psi$, if $a \neq b$

Rules

MP $\phi, \phi \to \psi \vdash \phi$
Nec$\Box$ if $\vdash \phi$ then $\vdash \Box_a\phi$
Nec$[\cdot]$ if $\vdash \phi$ then $\vdash [\psi]_a\phi$ □

Note that axioms $A2$, $A3$, and the rules $Nec\Box$ and $Nec[\cdot]$ together are the Normality constraint from section 2; $A4$ is the Functionality principle, $A5$ is Information Change Only, $A6$ is Consistent Elimination, $A7$ is Minimality, and $A8$ is Informational Equivalnce. Axiom $A9$ is a Privacy Axiom; its content is that if a and b are different actors, then a change of a's information will not change the information of b. For the one agent case axiom $A9$ becomes void and drops out. In that case we can also omit all subscripts on both the static modalities and the dynamic modalities. We leave it to the reader to check that all principles of section 2 that are not listed as axioms here, are derivable.

Theorem 3.12 (Completeness) $\Gamma \vdash_{MEK} \phi$ if and only if $\Gamma \models \phi$. □

We sketch the proof of the completeness theorem, because surprisingly, MEK appears to be translatable to multi-agent K. We first define the embedding degree of update modalities as follows.

Definition 3.13

1. $e(p) = 0$
2. $e(\neg\phi) = e(\phi)$
3. $e(\phi \wedge \psi) = max(e(\phi), e(\psi))$
4. $e(\Box_a\phi) = e(\phi)$
5. $e([\phi]_a\psi) = max(e(\phi), e(\psi) + 1)$ □

Next we reduce the embedding degree e.

Definition 3.14

1. $p^* = p$
2. $(\neg\phi)^* = \neg(\phi)^*$
3. $(\phi \wedge \psi)^* = \phi^* \wedge \psi^*$
4. $(\Box_a\phi)^* = \Box_a\phi^*$
5. $([\phi]_a\psi)^*$ is defined via a subinduction on ψ:
   (a) $([\phi]_a p)^* = p$
   (b) $([\phi]_a\neg\psi)^* = \neg([\phi]_a\psi)^*$
   (c) $([\phi]_a(\psi_1 \wedge \psi_2))^* = ([\phi]_a\psi_1)^* \wedge ([\phi]_a\psi_2)^*$
   (d) $([\phi]_a\Box_b\psi)^* = \begin{cases} \Box_b(\psi)^* & \text{if } a \neq b \\ \Box_a(\phi^* \to \psi^*) & \text{if } a = b \end{cases}$
   (e) $([\phi]_a[\psi_1]_b\psi_2)^* = [\phi^*]_a([\psi_1]_b\psi_2)^*$ □

This operation preserves equivalence:

Lemma 3.15 For all ϕ, $\vdash_{MEK} \phi \leftrightarrow \phi^*$. □

The operation also reduces the embedding degree:

Lemma 3.16 For all ϕ, if $e(\phi) \geq 1$ then $e(\phi^*) < e(\phi)$ □

Repeated application of the previous lemma shows that we can get rid of update modalities altogether.

Lemma 3.17 For each ϕ there exists a ϕ' that does not contain update modalities, and $\vdash_{MEK} \phi \leftrightarrow \phi'$. □

From this lemma we can derive theorem 3.12 as a corollary, using the fact that multi-agent K is sound and complete for information structures (see [FV85]). For details of the proofs I refer again to [Gro95].

We conclude this section listing two Permutation Principle for MEK.

Proposition 3.18

1. $\vdash_{MEK} [\phi \wedge \psi]_a \chi \leftrightarrow [\psi \wedge \phi]_a \chi$
2. $\vdash_{MEK} [\phi]_a [\psi]_b \chi \leftrightarrow [\psi]_b [\phi]_a \chi$

□

These observations show that MEK is not dynamic in the sense of either Dynamic Predicate Logic ([GS91]) or Update Semantics ([Vel91]), since 1 shows that conjunction is not ordersensitive in MEK, and 2, which also holds if $a = b$, shows that the order of updates is not important in MEK.

4 Conscious Updates

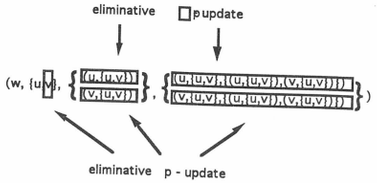
If an agent updates her information with p she will discard all 'ways the situation might be' in which p is false, according to the eliminative view on updates. In fact such an agent will do nothing more than just that. We have for example that

$$\vdash_{MEK} \Box(\neg\Box p \wedge \neg\Box\neg p) \rightarrow [p]\Box\bot$$

In short, this expresses that an agent who knows that his information whether p is inconclusive cannot consistently learn p. But for an agent who is aware of the fact that she is updating her information there will be no point in also preserving those options in which p is true but in which she does not have the information that p. Such a *conscious update* can be seen as an update that is also reflected at higher-order levels.

One way to explain this higher-order reflection, is by an 'iterated eliminative update', in which the higher-order adaptation is also purely eliminative. Such an update with p can modeled as an eliminative update with p followed by an eliminative update with $\Box p$. This could be called a conscious eliminative update of level 1. A conscious eliminative update of level 2 would also take account for the extra higher-order adaptation and could be modeled as the composition of eliminative updates with p, $\Box p$ and $\Box\Box p$. The limit of this sort of awareness is of course an adaptation through all finite levels.

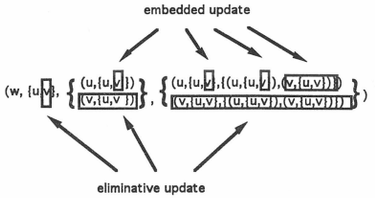
An example of a conscious eliminative update of level 2 is;



were we assume that $u \vDash p$, $v \nvDash p$. After restoring the Extension Constraint the resulting situation will be $(w, \emptyset, \emptyset, \emptyset)$. Of course these conscious updates of finite level can simply be modeled as a finite iteration of the eliminative updates we defined in the previous section.

However, this proposal cannot be a sensible explanation of conscious update. Consider an initial situation in which both $\neg\Box p$ and $\neg\Box\neg p$ are true. So our agent's information does not determine whether p. Moreover, assume she has the information that this is so: $\Box\neg\Box p$ and $\Box\neg\Box\neg p$ are also true. Then the agent has, in a sense, introspection about the proposition p, but we need not assume she has introspection in general. It just happens to be the case that for this particular proposition p, she is well informed about her own (lack of) information about it. This is a situation in which the agent is aware of the fact that she lacks the information whether p. It is also the typical situation in which we want to grant her the possibility to come to have the information that p and be aware of it. Now an 'eliminative conscious update' denies her the possibility of consciously coming to have the information that p: since initially all her informational options will satisfy $\neg\Box p$ (because $\Box\neg\Box p$ is true), the eliminative adaptation to $\Box p$ will remove all options, leaving her with nothing. This cannot be right.

So we have to look at a non-eliminative way of incorporating the higher-order reflection of the update. This can be achieved by explaining the higher-order adaptation not as a simple elimination but rather as an adaptation of all remaining options. In such an update with p, all $\neg p$ options will be discarded, but the remaining p-options will also be updated with p, as in:



were we again assume that $u \vDash p$, $v \nvDash p$. The resulting situation will then be $(w, \{u\}, \{(u, \{u\})\}, \{(u, \{u\}, \{(u, \{u\})\})\})$. This operation is not eliminative anymore since we internally modify the options. However, elimination still serves as a guiding idea, since the modification is in fact a form of 'embedded' elimination.

It is possible to define a notion of conscious update on arbitrary information structures. Nevertheless, we will not do this, but restrict ourselves to the case of an agent that has introspection. The reason for this is that is that it is a strange agent that has a 'dynamic awareness' in the sense that she is able to reflect new information correctly at higher-order levels, but has no 'static awareness' in the sense that she may be disinformed about her own information. For one thing, how is she able to *correctly* reflect the new information at higher-order levels if she can be totally mistaken about what here higher-order information actually is?

It is possible to use information structures to model introspective situations. However, at least for the one agent case, there is no real advantage in doing so. If s and t are information structures that satisfy Introspection, in which the facts and the first-order information are the same (so $s_0 = t_0$ and $s_1 = t_1$), then in fact $s = t$. This greatly simplifies the way in which we can model the higher-order adaptations for conscious updates: they are in fact completely determined by the first-order level. That is, if a notion of conscious update preserves introspection, only the first two constituents of the structure will be relevant. Then we do not have to bother about defining an infinitary adaptation mechanism through all levels, but can confine ourselves to the first-order level.

At a intuitive level, it is correct to assume that a conscious update will preserve introspection. An agent that is currently correctly informed about her information, and is also capable of a conscious strategy of dealing with new information, in the

sense that she reflects the new information correctly at higher levels, will remain introspective.

Paradoxically, then, an introspective agent can correctly reflect new information on higher levels by not representing these higher levels at all. The intuitive idea is that an introspective agent is not so much characterized by having some particular *state* of mind. Rather, we should see an introspective agent as having the *ability* of correctly reflecting on what her information is. She can therefore always calculate correctly what her higher-order information is from her first-order information. Since she knows that she has this ability, she has no need for explicitly keeping track of her higher-order information. When she gets new information, even if this new information is itself higher-order, she can just calculate what this means for her first-order information, and adapt that if needed. After that, she will still have her special ability, and introspection is 'automatically' preserved.

These considerations show that for conscious updates for one agent with introspection we can in fact make do with very simple structures.

Definition 4.1 (Semantics of CK45) As usual, let $\mathcal{P}$ be the set of atoms, $\mathcal{W} = Pow(\mathcal{P})$ be the set of possible worlds, $\Sigma = Pow(\mathcal{W})$ be the set of information states. A situation is a pair (w, σ) of a possible world and an information state. We define the truth relation $\models$ between situations and formulae, and functions $[\![\phi]\!]$ from information states to information states by a simultaneous induction.

1. (a) $(w,\sigma) \models p$ iff $p \in w$
   (b) $\sigma[\![p]\!] = \{w \in \sigma \mid p \in w\}$.
2. (a) $(w,\sigma) \models \neg\phi$ iff $(w,\sigma) \not\models \phi$
   (b) $\sigma[\![\neg\phi]\!] = \sigma \setminus \sigma[\![\phi]\!]$
3. (a) $(w,\sigma) \models \phi \wedge \psi$ iff $(w,\sigma) \models \phi$ and $(w,\sigma) \models \psi$
   (b) $\sigma[\![\phi \wedge \psi]\!] = \sigma[\![\phi]\!] \cap \sigma[\![\psi]\!]$
4. (a) $(w,\sigma) \models \Box\phi$ iff $\forall v \in \sigma : (v,\sigma) \models \phi$
   (b) $\sigma[\![\Box\phi]\!] = \begin{cases} \sigma & \text{if } \sigma[\![\phi]\!] = \sigma \\ \emptyset & \text{otherwise} \end{cases}$
5. (a) $(w,\sigma) \models [\phi]\psi$ iff $(w, \sigma[\![\phi]\!]) \models \psi$
   (b) $\sigma[\![[\phi]\psi]\!] = \begin{cases} \sigma & \text{if } \forall v \in \sigma : (v, \sigma[\![\phi]\!]) \models \psi \\ \emptyset & \text{otherwise} \end{cases}$

Validity is defined as preservation of truth: $\Gamma \models_{CK45} \phi$ if and only if for all situations (w,σ) such that $(w,\sigma) \models \psi$ for all $\psi \in \Gamma$, also $(w,\sigma) \models \phi$. □

Basically, $CK45$ is Veltman's definition of update semantics, embedded in a semantics for the richer language of DEL. We leave it to the reader to verify that defined connectives like $\Diamond$ and $\vee$ receive their natural meaning, both statically and dynamically. So the $\Diamond$ is statically the dual of the $\Box$, and also dynamically, since it is the consistency test dual to the test $\Box\phi$. And $\vee$ corresponds to 'or' statically, and to union dynamically.

An important distinction with the semantics for MEK is that we have defined the updates directly on the syntactical structure, instead of defining them indirectly via the truth conditions by the stipulation

$$\sigma[\![\phi]\!] = \{w \in \sigma \mid (w,\sigma) \models \phi\} \qquad (TC)$$

We made this choice, firstly, because it is possible, due to the nice Boolean structure over the information states. Secondly, it almost makes no difference. And thirdly, the case in which it does make a difference is intuitively justified.

Lemma 4.2 Define an operation ^{s} by: $p^s = p$; $(\neg\phi)^s = \neg(\phi^s)$; $(\phi \wedge \psi)^s = \phi^s \wedge \psi^s$; $(\Box\phi)^s = \Box\phi^s$; $([\phi]\psi)^s = \Box[\phi]\psi$. Then: $\sigma[\![\phi]\!] = \{w \in \sigma \mid (w,\sigma) \models \phi^s\}$. □

The only thing that the operation ^{s} does is to prefix update modalities with a $\Box$. Hence if ϕ does not contain update modalities, $\phi^s = \phi$. So for these formulae, the correspondence (TC) holds.

The update clause for formulae of the form $[\phi]\psi$ is the reason that (TC) does not hold in general. But this update clause fits the intuitive conception of the kind of agent we intend to describe. This is an agent with the ability to correctly reflect upon her information, and who also knows that she has this ability. Therefore, she will interpret a sentence of the form 'after an update with ϕ, ψ is true' as a test. She will simply check whether all situations she may be in, are such that after she learns ϕ in that situation, ψ will be true. Due to the nature of introspection, in an information state σ, the set of all pairs (v,σ) for $v \in \sigma$ are the situations she think possible.

A second issue about definition 4.1 that is worth some discussion is the extent to which the updates can really be called 'conscious'. After all, the updates look very eliminative, since we always have $\sigma[\![\phi]\!] \subseteq \sigma$. But note that the semantics realizes a higher-order reflection in the following sense. For formulae ϕ that do not contain $\Box$ or update modalities, updates are Hyper Successful:

$$\text{if } \phi \in \mathcal{L}_0, \text{ then for all } n \geq 1; \models [\phi]\Box^n\phi$$

where $\Box^n$ is $\Box$ iterated n times. On the other hand Success does not hold generally:

$$\text{not for all } \phi, \models [\phi]\Box\phi$$

We do have that if ϕ is Successful, then it is Hyper Successful:

$$\models [\phi]\Box\phi \rightarrow [\phi]\Box^n\phi$$

The latter is due to the fact that the semantics validates the Positive Introspection axiom, from which it follows that $\models \Box\phi \rightarrow \Box^n\phi$, and then by $[\phi]$-Necessitation and $[\phi]$-distribution we have $\models [\phi]\Box\phi \rightarrow [\phi]\Box^n\phi$. This contrasts with Eliminative K, where we do have Success for all formulae, but we do not have Hyper Success, not even for $\mathcal{L}_0$ formulae.

That Success fails, may be seen by considering the formula $\neg p \wedge \Diamond p$. This formula is statically consistent, since there are w, σ with $(w,\sigma) \models \neg p \wedge \Diamond p$. It is also dynamically consistent: there are σ with $\sigma[\![\neg p \wedge \Diamond p]\!] \neq \emptyset$. On the other hand, there are no w, $\sigma \neq \emptyset$ with $(w,\sigma) \models \Box(\neg p \wedge \Diamond p)$. So we have here a sentence that can be true, can be 'learned', but cannot be 'known' consistently. This may seem odd, but it is just right. First, it is clear that an introspective agent cannot have the information that both $\neg p$ and $\Diamond p$ (by positive introspection). Second, an update with $\neg p \wedge \Diamond p$ can have two effects: either $\sigma[\![\neg p \wedge \Diamond p]\!] = \emptyset$ or $\sigma[\![\neg p \wedge \Diamond p]\!] = \sigma[\![\neg p]\!]$. In the former case, the information has become inconsistent, hence we have a trivial higher-order reflection. In the latter case, the information that $\neg p$ will of course obstruct a consistent second update with $\Diamond p$.

One of the consequences of the failure of Success is that the Ramsey Test, $\Box(\phi \rightarrow \psi) \leftrightarrow [\phi]\Box\psi$, will not be valid either, since we will of course have $\models \Box(\phi \rightarrow \phi)$. Now the Ramsey Test was one of the driving forces in the reduction of MEK to K. This means that if there is a reduction of $CK45$ to $K45$, then this reduction will be of a different kind. It is at present unclear if such a reduction exists. However, a direct completeness proof for $CK45$ is possible.

Also observe that the formula $\phi = \neg p \wedge \Diamond p$ shows that we do not have Idempotency:

$$\text{not for all } \phi, \sigma[\![\phi]\!][\![\phi]\!] = \sigma[\![\phi]\!]$$

On the other hand for $\phi = \neg p \wedge \Diamond p$ we do have that either $\sigma[\![\phi]\!][\![\phi]\!] = \sigma[\![\phi]\!]$ or $\sigma[\![\phi]\!][\![\phi]\!] = \emptyset$. From this it follows that for this particular ϕ we do have that 'two times is enough', that is, $\sigma[\![\phi]\!][\![\phi]\!][\![\phi]\!] = \sigma[\![\phi]\!][\![\phi]\!]$. Can it get any worse?

Definition 4.3 Define $[\![\phi]\!]^0 = \lambda\sigma \cdot \sigma$, $[\![\phi]\!]^{n+1} = [\![\phi]\!]^n \circ [\![\phi]\!]$. Then a formula ϕ is k-stable if for all σ, $\sigma[\![\phi]\!]^{k+1} = \sigma[\![\phi]\!]^k$. □

$\top$ is 0-stable. All formulae that do not contain the modalities $\Box$ or $[\cdot]$ are 1-stable. All formulae that do not contain the conjunction symbol $\wedge$ are 1-stable. $\neg p \wedge \Diamond p$ is 2-stable, but not 1-stable. And here is a formula that is 3-stable but not 2-stable:[9]

$$(\Diamond p \vee \neg q) \wedge (\Diamond q \vee r) \wedge \neg p$$

The idea of this example is that a formula of the form $(\Diamond\phi \vee \neg\psi)$ 'postpones' an update with $\neg\psi$ until the test $\Diamond\phi$ fails:

$$\sigma[\![\Diamond\phi \vee \neg\psi]\!] = \begin{cases} \sigma & \text{if } \sigma[\![\Diamond\phi]\!] = \sigma \\ \sigma[\![\neg\psi]\!] & \text{otherwise} \end{cases}$$

Now suppose that in the first update of the example formula, both tests $\Diamond p$ and $\Diamond q$ succeed. The effect of the third conjunct $\neg p$ will be that in the second update, the test $\Diamond p$ will fail, and the update with $\neg q$ becomes activated. Thus in the third update, the test $\Diamond q$ fails, and the update with r is activated. Finally the fourth update will not change anything.

The following proposition establishes a finite upper bound on the stability of a formula.

Proposition 4.4 If $|\phi|$ is the number of propositional atoms that occur in ϕ, then ϕ is $2^{|\phi|}$-stable. □

The worst case actually arises.

Proposition 4.5 For each $n \geq 1$ there exists a formula ϕ with n propositional atoms, such that ϕ is not $(2^n - 1)$-stable. □

We will now turn our attention to a complete axiomatization of $CK45$.

Definition 4.6 (Axiomatization of CK45)

Axioms

1. ϕ, if ϕ is a substitution instance of a valid formula of classical propositional logic
2. $\Box(\phi \to \psi) \to (\Box\phi \to \Box\psi)$
3. $\Box\phi \to \Box\Box\phi$
4. $\neg\Box\phi \to \Box\neg\Box\phi$
5. $[\phi](\psi \to \chi) \to ([\phi]\psi \to [\phi]\chi)$
6. $\neg[\phi]\psi \leftrightarrow [\phi]\neg\psi$
7. $[\phi]p \leftrightarrow p$
8. $\Box\bot \to [\phi]\Box\bot$
9. $[\phi]\Diamond\psi \to \Diamond\psi$, if $\psi \in \mathcal{L}_0$
10. $[p]\Diamond\psi \leftrightarrow \Diamond(p \wedge \psi)$, for p atomic, $\psi \in \mathcal{L}_0$

[9] Thanks to David Beaver for finding this nice example.

11. $[\phi_1 \wedge \phi_2]\Diamond\psi \leftrightarrow [\phi_1]\Diamond\psi \wedge [\phi_2]\Diamond\psi$, for $\psi \in \mathcal{L}_0$
12. $\Diamond\psi \to ([\phi]\Diamond\psi \vee [\neg\phi]\Diamond\psi)$, $\psi \in \mathcal{L}_0$
13. $[\phi \wedge \neg\phi]\Box\bot$
14. $[\Box\phi]\Diamond\top \to ([\Box\phi]\psi \leftrightarrow \psi)$
15. $[\Box\phi]\Diamond\top \to ([\phi]\psi \leftrightarrow \psi)$
16. $\Box[\phi]\psi \to ([[\phi]\psi]\chi \leftrightarrow \chi)$
17. $\neg\Box[\phi]\psi \to [[\phi]\psi]\Box\bot$

Rules

MP From ϕ and $\phi \to \psi$ infer ψ

Nec$\Box$ From ϕ infer $\Box\phi$

Nec$[\cdot]$ From ϕ infer $[\psi]\phi$

Stability Let $\{\delta_1, \ldots, \delta_n\}$ be a complete set of diagrams for ϕ. Then from $\Diamond\top, (\Diamond\delta_1 \to [\phi]\Diamond\delta_1), \ldots, (\Diamond\delta_n \to [\phi]\Diamond\delta_n)$ infer $[\Box\phi]\Diamond\top$.

The relation $\Gamma \vdash_{CK45} \phi$ is defined as usual, with the normal weak interpretation of the Necessitation Rules. □

Axioms 1–4 are the axioms of $K45$, axioms 5–7 are the axioms of the minimal (functional) dynamic epistemic logic. Axiom 8 expresses that there is no escape from an inconsistent information state by an update. Axiom 9 is a weak form of Elimination axiom of EK. The remaining axioms reflect the definition of the updates, by explaining the properties of update modalities $[\phi]$ in terms of the syntactic structure of ϕ.

The role of the Stability Rule may not be completely obvious yet, but observe that it is a kind converse of the axiom $[\Box\phi]\Diamond\top \to ([\phi]\psi \leftrightarrow \psi)$. This axiom expresses that if the update with $\Box\phi$ is consistent, then the update with ϕ does not change anything. A real converse of this axiom would give the infinitary rule: if for all ψ, $([\phi]\psi \leftrightarrow \psi)$, then $[\Box\phi]\Diamond\top$. The condition expressed by the premises of this infinitary rule is that an update with ϕ does not change anything. The Stability Rule shows that we only need finitely many formulae to express this.

Theorem 4.7 (Completeness) $\Gamma \vdash_{CK45} \phi$ if and only if $\Gamma \models_{CK45} \phi$.
Proof: see [Gro95], pp. 169–174. □

We end this section with a short discussion on the Multi-Agent version of Conscious $K45$ we expect to develop. First observe that in the Multi-Agent case we will not be able to 'flatten' the information structures to its first two constituents (the facts and first-order information). An agents higher-order information about his own information will indeed be totally determined by his first-order information, just as in the one agent case. And even, agent a's k-th order information about agent b's n-th order information will be completely determined by a's second order information: from introspection for b, a-necessitation and introspection for a it follows that $\Box_a^k\Box_b^n\phi \leftrightarrow \Box_a\Box_b\phi$ will be valid. But a formula of the form $\Box_a\Box_b\Box_a\Box_b\phi$ will in general not be reducible to a formula of lower degree. This means that in the multi-agent case, we will have to return to the full information structures. One of the consequences will be a significant complication of the definition of the information ordering. The nice Boolean structure to which the information ordering gives rise in the one agent case will be lost. By consequence, it will not be clear that we can define the updates directly on the syntactic structure of the formulae, in the same way as we did in definition 4.1.[10] We leave it at these brief remarks, and hope the solve this problem in the near future.

[10] But see [Gro93] for an inductive definition of updates for the simpler case in which we only require information to be consistent.

5 Information and its structures

We compare the systems we developed in the previous sections to some other proposals from the literature. The discussion will focus on the notion of information and its structure. In the course of this chapter we have encountered Kripke models, and in the development of eliminative updates, we used information structures. For the one agent case of conscious updates we used a kind of 'flattened' information structure. But besides Kripke models and modal structures (or information structures, as we have called them), there are at least two different types of structures that have been proposed in the literature.

In [Bar89], situation theory and Aczel's theory of non-well-founded sets ([Acz88]) are used to obtain a compelling analysis of common knowledge via non-well-founded informational dependencies. The theory developed by Jaspars [Jas94], which is congenial to ours, uses Kripke models with partial valuations for defining constructive notions of update and downdate in a multi-agent setting. Both proposals differ from ours in being instances of partial semantics, whereas we have been working with total interpretations. Also, Jaspars uses a constructive notion of update, whereas we have been using a basically eliminative view on updates. Instead of a detailed discussion of the differences, I will point out the strong similarity between them. I think that all can be seen as based upon the following intuitive conception of informational situations:

- An informational situation specifies facts and information of a group of agents
- The information of a group of agents assigns to each actor an individual information state
- An individual information state is a set of situations

A straightforward translation into set theory, for a given set of truth value assignments W and a given set of actors A, gives the three equations

$$S = W \times G$$

$$G = I^A$$

$$I = Pow(S)$$

Observe that due to Cantor's theorem, S, G, I can only form a solution to the equations if all three of S, G, I are proper classes (we assume $\mathcal{W}$ and $\mathcal{A}$ non-empty). This is true in ZF as well as in Aczel's ZFC/AFA. Also, it appears that these three clauses by themselves cannot serve as a definition, since the definition would be circular. However, in Aczel's set theory ZFC/AFA, this is no problem, and we *can* define S, G, I coinductively as the largest classes that satisfy these three equations. If we apply this scheme to a set of actors A and a set of partial truth value assignments $\mathcal{W}$, we roughly get Barwises non-well-founded situations. Although this definitional scheme does not provide infons, and other formal details of Barwises approach are also missing, I think the scheme can at least be seen as the backbone of his approach. Also see [Ger95] for an elaboration of this definitional scheme.

But Kripke models (partial or total) and modal structures can be seen as as solutions to similar equations. For Kripke models, we have to weaken the identity between S and $W \times G$ by to the existence of a bijective correspondence between S and $W \times G$. Then take $\mathcal{S}$ to be the class of all rooted Kripke models. The natural bijective correspondence is then given by 'the root with its valuation, and the function that assigns to each actor a the set of generated models that a sees from the root'. So we can take $I = Pow(S)$ and $G = I^A$, and we have

$$S =_1 W \times G$$

$$G = I^A$$

$$I = Pow(S)$$

Whether valuations are total or partial is inessential here.

Finally, modal structures can be conceived as a solution to the equations

$$S =_1 W \times G$$

$$G = I^A$$

$$I \subseteq Pow(S)$$

in which S, G, I actually are (well-founded) sets. For sake of simplicity, consider the one agent case. Take $S = \mathcal{S}_\omega$, which is a set. I consists of all subsets of $\mathcal{S}$ that are of the form $\{t \in \mathcal{S}_\omega \mid s\mathcal{R}t\}$ for some $s \in \mathcal{S}_\omega$. It is then clear what the bijective correspondence is.[11]

In conclusion, I think that the differences between the four types of structures are best seen as different but similar elaborations of the same intuitive conception of informational situations.

6 Conclusions and Further Research

In the introduction we sketched a research program that consists of two parts: (a) the development of a dynamic epistemic logic; (b) a utilization of this logic as a logic for information exchange. The second part of this program has remained beyond the scope of this paper, but we hope to make a start with its realization in the near future. We have made considerable progress in the realization of the first part, the development of dynamic epistemic logic. But also there, several issues remain to be resolved by further research. We mention three of these issues.

First, a further development of conscious updates by giving a semantics for a Multi-Agent version of $CK45$, preferably accompanied by a complete axiomatization.

Then an investigation is due of dynamic (eliminative or conscious) variants of other modal logics that have been used in epistemic logic, such as $S4$, $KD45$, and $S5$.[12] This question involves another important issue, which can be called the Coherency Issue, and can be explained as follows. The dynamic logics we have been considering are Eliminative K and Conscious $K45$. Now the two combinations of eliminative updates with K, and conscious updates with $K45$, are natural combinations. In eliminative K, a minimal kind of static awareness is coupled with a minimal strategy of incorporating new information. And in Conscious $K45$, introspection, the actors 'static' ability to correctly represent her information, is coupled to a conscious way of incorporating new information, in the sense that the new information is also reflected at higher-order levels. Thus both systems are 'coherent' combinations of representational and dynamic abilities. This means that it is not immediate clear that either eliminative or conscious updates as we have developed them are the right notions of update in the context of logics such as $S4$, $KD45$, and $S5$. Although I do not intend to solve this problem here, I do expect that the notion of conscious update, with only minor modifications, will be the most suitable one for the three logics just mentioned.

[11] That I will not contain all subsets of $\mathcal{S}_\omega$ can be seen as follows. Let $X \subseteq \mathcal{S}_\omega$. Then it may well be that there is some $t \in \mathcal{S}_\omega$ such that for each n, there is some $t' \in X$ such that t and t' are the same up to level n, though $t \notin X$. However, any s that is $\mathcal{R}$-related to all structures in X will also be $\mathcal{R}$-related to t.

[12] An interesting question is whether the dynamic $S5$ system of [vEdV95] is equivalent to Conscious $S5$.

Finally, a third issue for further research is the extension of the multi-agent systems with the concept of shared information. This can be seen as a further preparation for the intended application of the logic for a system of information exchange. In many places in the literature it has been argued that shared information, or the related concepts of mutual belief and common knowledge, plays a prominent part in communication.[13]

The results of this paper can be summarized as follows. We have developed and completely axiomatized three actual systems of dynamic epistemic logic (EK, MEK, and $CK45$). We have achieved this by using structures and techniques that are familiar in modal and epistemic logic, in a formal language that extends the standard vocabulary of epistemic logic. Thus dynamic epistemic logic as we have developed is an extension of epistemic logic, rather than a revision.

Furthermore, the investigation led to the consideration of two fundamentally different types of update operations over higher-order information models. This warrants the conclusion that there are at least two different families of dynamic epistemic logics. We also observed in section 4.3.3 that within one such family, the axiomatic organization will not be as smooth as in modal logic, since restrictions on the model class need not induce an extension of the axiomatics.

Finally, our discussion on introspection lead to a special perspective on this concept. We conceived of introspection not so much as a particular *state* of mind in which higher-order information is correctly represented, but rather as the *ability* to correctly reflect upon what the information is. And for an agent that has this ability, there is actually no need at all to represent the higher-order information.

But maybe the most striking, and slightly disturbing, conclusion of the research in this paper is that the number of possible definitions of update, as a function of time, is neither increasing nor decreasing.

References

[Acz88] Peter Aczel. *Non-well-founded sets.* CSLI Lecture Notes Number 14. CSLI Publications, Stanford, 1988.

[Bar89] Jon Barwise. On the model theory of common knowledge. In *The Situation in Logic*, chapter 9. CSLI Lecture Notes No.17, Stanford, 1989.

[BP83] Jon Barwise and John Perry. *Situation Semantics.* MIT Press, Cambridge, USA, 1983.

[FHMV95] Ronald Fagin, Joseph Y. Halpern, Yoram Moses and Moshe Y. Vardi. *Reasoning About Knowledge.* MIT Press, Cambridge, USA, 1995.

[FHV91] Ronald Fagin, Joseph Y. Halpern, and Moshe Y. Vardi. A model-theoretic analysis of knowledge. *Journal of the ACM*, 38(2):382–428, 1991.

[13] The literature on common knowledge and related concepts is extensive, and I can only mention a few references here. The concept plays a prominent role in the coordination problems of [Lew69]. And in Grice's theory of implicature, common knowledge plays an important role (see [Gri89] and [Lev83]). A collection of essays on the linguistic significance of common knowledge is [Smi82]. Further, the notion of context in the context-change theories of [Sta74, Gaz79] is clearly related. Common knowledge is crucial for a proper understanding of the Conway Paradox ([Bar89]). Finally, there is a vast literature on common knowledge in computer science, pertaining to the communication in distributed computer systems; see [HM90], and the series of Proceedings of the TARK conferences.

[FV85] Ronald Fagin and Moshe Y. Vardi. An internal semantics for modal logic. In *Proceedings of the 17th Annual Symposium on Theory of Computing (Providence, R.I., May 6-8)*, pages 305–315, New York, 1985. ACM.

[Gär88] P. Gärdenfors. *Knowledge in Flux. Modelling the Dynamics of Epistemic States.* Bradford Books / MIT Press, Cambridge (Mass.), 1988.

[Gaz79] G. Gazdar. *Pragmatics. Implicature, Presuppostion, and Logical Form.* Academic Press, 1979.

[Ger95] Jelle Gerbrandy. On coming to know that you are dirty when you are dirty. Manuscript. Institute for Language, Logic and Computation, Department of Philosophy, University of Amsterdam, 1995.

[Gri89] P. Grice. Meaning. In *Studies in the Way of Words.* Harvard University Press, Cambridge, MA and London, England, 1989. First published 1957.

[GS91] J. Groenendijk and M. Stokhof. Dynamic predicate logic. *Linguistics and Philosophy*, 14:39–100, 1991.

[Gro93] Willem Groeneveld. Shifting attitudes. In Katalin Bimbó and András Máté, editors, *Proceedings of the 4th Symposium on Logic and Langauge*, pages 105–122, Budapest, 1993. Eötvös University Budapest, Aron Publishers.

[Gro95] Willem Groeneveld. *Logical Investigations into Dynamic Semantics.* PhD thesis, Institute for Logic, Language and Computation, Department of Philosophy, University of Amsterdam, 1995. iLLC Dissertation Series 1995-18.

[HM90] J. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, 37(3):549–587, 1990. First published in J. Halpern and Y. Moses (eds.), *Proceedings of the Third ACM Conference on Principles of Distributed Computing*, ACM, New York, 1984.

[HD89] S.J. Hamilton and J.P. Delgrande. An investigation of modal structures as an alternative semantic basis for epistemic logics. *Computational Intelligence*, 5:82–96, 1989.

[Hin62] J. Hintikka, editor. *Knowledge and Belief.* Cornell University Press, Ithaca, N.Y., 1962.

[Jas94] Jan Jaspars. *Calculi for Constructive Communication. A Study of the Dynamics of Partial States.* PhD thesis, ITK, Katholieke Universiteit Brabant, 1994.

[Lev83] S.C. Levinson. *Pragmatics.* Cambridge University Press, Cambridge, 1983.

[Lew69] David Lewis. *Convention.* Harvard University Press, Cambridge, Mass., 1969.

[Smi82] N.V. Smith, editor. *Mutual Knowledge.* Academic Press, London / New York, 1982.

[Sta74] R. Stalnaker. Pragmatic presuppositions. In M. Munitz and P. Unger, editors, *Semantics and Philosophy*. New York UP, New York, 1974.

[vEBGS81] P. van Emde Boas, J. Groenendijk, and M. Stokhof. The Conway paradox: its solution in an epistemic framework. In J. Groenendijk and M. Stokhof, editors, *Formal Methods in the Study of Language*. Mathematical Centre Tracts, Amsterdam, 1981. Also in: Groenendijk and Stokhof (eds.), Truth, Interpretation, Information. Grass 2, Foris, Dordrecht, 1983.

[vEdV95] Jan van Eijck and Fer-Jan de Vries. Reasoning about update logic. *Journal of Philosophical Logic*, 24:19–45, 1995.

[Vel91] Frank Veltman. Defaults in update semantics. ITLI Prepublication Series LP-91-02, Department of Philosophy, University of Amsterdam, 1991. To appear in the *Journal of Philosophical Logic*, 1996.

Authors Address:
Institute for Logic, Language and Computation
Department of Philosophy
University of Amsterdam
Nieuwe Doelenstraat 15
1012 CP Amsterdam
The Netherlands
E-mail: groenev@illc.uva.nl

Semantic Properties of Interrogative Generalized Quantifiers

Javier Gutiérrez Rexach [1]
Department of Linguistics, UCLA

1 Questions as Functions

In this paper an extensional theory of questions which attempts to characterize the contribution of interrogative quantifiers to the meaning of matrix interrogative sentences is presented. Questions are defined as functions from sets of objects to truth values. Matrix interrogative sentences denote such functions. Specifically, a question of type $<< \alpha, t >, t >$ is a function (of a certain sort) from sets of objects in type α to truth values. This corresponds to the intuition that a speaker, in asking a question of type $<< \alpha, t >, t >$, is asking a question of the corresponding type, namely, he is asking the hearer to identify a unique object of type $< \alpha, t >$, i.e. a unique set of objects of type α.

Definition 1 *A* question *of type* $<< \alpha, t >, t >$ *is a function* $f \in [\mathcal{P}(\alpha) \rightarrow 2]$ *such that* $\exists! x \in \mathcal{P}(\alpha)$ *such that* $f(x) = 1$. *We will call such an* x *the* answer A_f *of* f. *We write* $[_q \mathcal{P}(\alpha) \rightarrow 2]$ *for the set of questions of type* $<< \alpha, t >, t >$.

From the above definition, it follows that for an arbitrary question f its answer exists and is unique (no question has more than one answer). The intuition here is that the unique x a question f is true of is the complete true answer to f. So we are taking the fact that questions are "strongly exhaustive" (Groenendijk and Stokhof (henceforth G&S), 1984) as the essential ingredient of the definition of a question. Another important point is that answers, as we treat them here, are in a higher type than expected. Consider the question in (1a):

(1) a. Who came to the party?
b. John, Mary and Bill.
c. John and Mary.

At this point and for the sake of simplicity, we think of (1b) as denoting a three element set, and (1c) a two element set. In a state of affairs in which John, Mary and Bill are exactly the individuals who came to the party, the constituent response in (1b) denotes the answer set of the question that (1a) denotes. So the question denoted by the interrogative sentence (1a) maps {JOHN,MARY,BILL} to True, and the rest of the objects in $\mathcal{P}(E)$ to False. A *partial answer*, as the one denoted by the expression in (1c), is a subset of the answer set of the question. There are also other responses to (1a) which provide some pragmatic or semantic information about the answer set but do not constitute proper or even partial answers. Although partial, non-canonical, and uninformative answers are possible answers to a question, they should not be considered as equal in status to complete true answers. The intuition that my approach builds on is that only complete true answers are the objects that fulfill the information gap represented by the question. On many concrete occasions of everyday life, we are forced to give partial or uninformative answers to questions, either because we do not have enough information or we want to hide something, etc., but in doing so we actually are not logically answering the question. An additional motivation for treating questions as functions from sets of objects (for instance, individuals) to truth values rather than as merely sets of individuals is that otherwise we are conflating the denotations of questions and relative clauses or free relatives (Cooper, 1983; Jacobson, 1995). In the NP *the*

[1] I wish to thank Ed Keenan for his very helpful comments and suggestions on earlier versions of this paper. I would also like to thank Anna Szabolcsi and the audiences at the Fourth CSLI Workshop on Logic, Language and Computation and the Amsterdam Colloquium for their comments.

students who came to the party we may treat the relative clause *who came to the party* as denoting a function mapping the set of students to the set of students who came to the party. But the unit constituted by a question and its answer is of a propositional nature, it should extensionally denote a truth value (or proposition) rather than a set. Interestingly, the fact that we are one type higher does not mean that we have an increase in expressive power, as the following observation shows:

Fact 2 $|[_q\mathcal{P}(\alpha) \rightarrow 2]| = |\mathcal{P}(\alpha)|$

An interrogative sentence may denote one question in a possible world (situation), and a different question in other possible world. For example, the denotation of the expression in (1c) is one element of the answer space of the question which can constitute its answer set in a different world. Let I be an index set. An *indexed question* f is a function from indices to questions: $f \in [I \rightarrow [_q\mathcal{P}(\alpha) \rightarrow 2]]$. There are other ways to let possible worlds enter the picture, such as those proposed by Karttunen (1977) and G&S (1984). In this paper we will restrict ourselves to non-indexed questions. Taking English as our object language we are going to consider two basic types of questions: argument questions and modifier questions.

2 Argument Questions

Definition 3 *An element of* $[_q\mathcal{P}(E) \rightarrow 2]$ *is a* (unary) argument question.

In general, argument interrogative quantifiers are functions from n-ary relations to questions. Unary argument interrogative quantifiers are functions from sets to unary argument questions. Unary argument interrogative determiners are functions from sets to unary argument interrogative quantifiers.

Definition 4 *(***Argument interrogative GQs***)*
$[\mathcal{P}(E) \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]$ *is the set of* (unary) argument interrogative quantifiers.
$[\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]]$ *is the set of* (unary) interrogative determiners.

For example, consider the following interrogative sentences:

(2) a. Who is smoking?
b. Which student is smoking?

The *wh*-word *who* denotes an argument interrogative quantifier, as illustrated in (3a,b). *Which* denotes an argument interrogative determiner (3c,d).

(3) a. $[\![Who]\!] \in [\mathcal{P}(E) \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]$
b. $[\![Who]\!]([\![\lambda x.Smoke(x)]\!]) \in [_q\mathcal{P}(E) \rightarrow 2]$
c. $[\![Which]\!] \in [\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]]$
d. $[\![Which]\!]([\![\lambda x.Student(x)]\!])([\![\lambda x.Smoke(x)]\!]) \in [_q\mathcal{P}(E) \rightarrow 2]$

Using uppercase letters to represent denotations in a fixed universe E we can define the following English argument interrogative quantifiers:

Definition 5 *(***English Interrogative Quantifiers***) For all* $Z, Y, X \subseteq E$*:*
$WHO(Y)(X) = 1$ *iff* $PERSON \cap Y = X$
$WHAT(Y)(X) = 1$ *iff* $E \cap Y = X$
$WHICH_n^Z(Y)(X) = 1$ *iff* $Z \cap Y = X \,\&\, |X| = n$
$WHICH_ONES^Z(Y)(X) = 1$ *iff* $Z \cap Y = X \,\&\, |X| \geq 2$

Applying the above definition, we see that sentence (4a) denotes a function that maps the set $PERSON \cap IN_THE_CORRIDOR$ to 1 and any other set of individuals to 0. The calculation of the truth conditions of the interrogative sentence/response pair in (4) is as in (5):

(4) a. Who is in the corridor?
b. Fred and Bill

(5) $WHO(\{x|x \in IN_THE_CORRIDOR\})(\{[\![Fred]\!], [\![Bill]\!]\}) = 1$ iff
$PERSON \cap (\{x|x \in IN_THE_CORRIDOR\}) = \{[\![Fred]\!], [\![Bill]\!]\}$

The functions $WHICH_n$ and $WHICH_ONES$ are inherently restricted to context sets. Therefore, they cannot be uttered in "out of the blue situations". Consider the interrogative sentence (6a) and the answer (6b) in a context where we are talking about female students in our deparment. Therefore, the relevant context set is $Z = \{x|x \in FEMALE_STUDENT\}$ and the interpretation as in (7).

(6) a. Which three like tacos?
b. Jill, Jodie and Jennie

(7) $WHICH_THREE^Z(\{x|x \in LIKE_TACOS\})(\{[\![Jill]\!], [\![Jodie]\!], [\![Jennie]\!]\}) = 1$ iff
$\{x|x \in FEMALE_STUDENT\} \cap \{x|x \in LIKE_TACOS\} = \{[\![Jill]\!], [\![Jodie]\!], [\![Jennie]\!]\}$

Definition 6 *(***English Interrogative Determiners***)*
For all $Z, Y, X, W \subseteq E, x, y \in E, m \in \mathcal{N}$*:*
$WHAT_{sg}(Z)(Y)(X) = 1$ *iff* $Z \cap Y = X \,\&\, |X| = 1$
$WHAT_{pl}(Z)(Y)(X) = 1$ *iff* $Z \cap Y = X \,\&\, |X| \geq 2$
$WHICH_{sg}{}^W(Z)(Y)(X) = 1$ *iff* $(W \cap Z) \cap Y = X \,\&\, |X| = 1$
$WHICH_{pl}{}^W(Z)(Y)(X) = 1$ *iff* $(W \cap Z) \cap Y = X \,\&\, |X| \geq 2$
$WHICH_n{}^W(Z)(Y)(X) = 1$ *iff* $(W \cap Z) \cap Y = X \,\&\, |X| = n$
$HOW_MANY(Z)(Y)(\{m\}) = 1$ *iff* $|Z \cap Y| = m$
$WHOSE(Z)(Y)(X) = 1$ *iff* $Z \cap Y = X \,\&\, \exists x \forall y \in Z[Poss(x, y)]$ [2]

Which determiner expressions also denote context dependent functions. The difference between the argument interrogative determiners $WHICH_{sg}$ and $WHICH_{pl}$ lies in the additional condition imposed on the cardinality of their answer sets. Here we treat the difference as the semantic correlate of grammatical number in parallel to the contrast between singular and plural declarative determiners (THE_{sg} vs. THE_{pl}). Consider sentence (8). Informally, the question denoted by (8) either poses a query about the set of students in the model or about a subset of those students that the speaker has in mind.

(8) Which students came to the party?

The latter reading is sometimes called a "partitive" reading. The descriptive intuition behind the term relies on the equivalence between the interpretation of *which students* and the interpretation of *which of the students*, as noted by Heim (1987). Let us consider first the interpretation of (8):

(9) Let $W = \{x|x \in LINGUIST\}$
$WHICH_{pl}{}^W(STUDENT)(COME)(X) = 1$ iff
$\{x|x \in LINGUIST\} \cap \{x|x \in STUDENT\} \cap \{x|x \in COME\} = X \,\&\, |X| \geq 2$

As observed in Barwise & Cooper (1981) and Keenan & Stavi (1986), declarative partitive determiners obey the restriction that only definite plural determiners can follow the preposition *of*. The same constraint surfaces in the interrogative domain (10).

(10) a. *Which of some/most/many/every/three students came to the party?
b. Which of the (ten)/John's (ten)/these ten students came to the party?

This fact suggest an analysis of *which of the* as a complex determiner, along the lines proposed by Keenan and Stavi for the declarative counterpart. One can check immediately that for $\alpha \in \{sg, pl\}, WHICH_OF_THE_\alpha{}^W(Z)(Y)$, as defined below, and $WHICH_\alpha{}^{W'}(Z)(Y)$ are the same question function when the contextual restrictions of the determiners are equal (W = W'),

[2] Here we understand *Poss* as a possession relation between individuals, the possessor and the possessee.

(11) a. $WHICH_OF_THE_{sg}{}^{W}(Z)(Y)(X) = 1$ iff $(W \cap Z) \cap Y = X \,\&\, |X| = 1$

b. $WHICH_OF_THE_{pl}{}^{W}(Z)(Y)(X) = 1$ iff $(W \cap Z) \cap Y = X \,\&\, |X| \geq 2$

3 Plural Questions

We say that an interrogative sentence denotes a plural question iff it maps a collection of sets of individuals to 1 and any other collection to 0.

Definition 7 *$[_q\mathcal{P}(\mathcal{P}(E)) \rightarrow 2]$ is the set of* (unary) plural argument questions

Consider the following sentences:

(12) Which students gathered in the plaza?

(13) Who carried the piano upstairs?

As a plural question, sentence (12) is asking for the groups of students that gathered in the plaza. Similarly, (13) can be interpreted as a question about the group(s) of persons that collectively carried the piano upstairs. If we analyze plural determiners as functions from sets of individuals (properties) to collections of sets of individuals (plural properties) to truth values, we can extend the same treatment to the analysis of the plural readings associated with interrogative quantifiers. This line of analysis has been proposed for declarative determiners, among others, by van Benthem (1991) and van der Does (1992). Its extension to the interrogative domain seems to be straightforward. In other words, plural interrogative quantifiers do not seem to exclude any of the readings associated with declarative ones (distributive, collective or neutral (van der Does, 1992)). Let us consider first the collective lifts of interrogative quantifiers and determiners. For any quantifier Q or determiner D we write $\mathbf{C}(Q)$,$\mathbf{C}(D)$ for the collective lift of the interrogative quantifier and determiner respectively. Here are two examples:

Definition 8 **(Collective lifts)** *Let $Z, W \subseteq E, Y, X \subseteq \mathcal{P}(E)$. Then,*
(i) $\mathbf{C}(WHO) \in [\mathcal{P}(\mathcal{P}(E)) \rightarrow [_q\mathcal{P}(\mathcal{P}(E)) \rightarrow 2]]$
$\mathbf{C}(WHO)(Y)(X) = 1$ *iff* $X = \{W | W \subseteq PERSON \,\&\, W \in Y\}$
(ii) $\mathbf{C}(WHICH_{pl}) \in [\mathcal{P}(E) \rightarrow [\mathcal{P}(\mathcal{P}(E)) \rightarrow [_q\mathcal{P}(\mathcal{P}(E)) \rightarrow 2]]]$
$\mathbf{C}(WHICH_{pl})(Z)(Y)(X) = 1$ *iff* $X = \{W | W \subseteq Z \,\&\, W \in Y\}$

The collective lift of WHO, $\mathbf{C}(WHO)$, is a function from collections of sets of individuals to plural questions. The collective lift of $WHICH_{pl}$, $\mathbf{C}(WHICH_{pl})$, is a function that maps a property Z to a collective interrogative quantifier $\mathbf{C}(WHICH_{pl})(Z)$. The truth conditions of (12) and (13) are as follows:

(14) a. $\mathbf{C}(WHICH_{pl})(STUDENT)(GATHER)(X) = 1$ iff
$X = \{W | W \subseteq STUDENT \,\&\, W \in GATHER\}$

b. $\mathbf{C}(WHO)(CARRY_THE_PIANO)(X) = 1$ iff
$X = \{W | W \subseteq PERSON \,\&\, W \in CARRY_THE_PIANO\}$

The answer set of the plural question denoted by (12) is the collection of subsets of students in the extension of the plural property GATHER. Therefore, in a situation where John gathered with Bill and Sam, and Susan gathered with Pam and Joe, the collection {{JOHN, BILL, SAM},{SUSAN, PAM, JOE}} would be the answer set of (12). Consider now (15):

(15) Which students ate pizza?

The distributive interpretation of sentence (15) is a plural question true of the collection of singletons of students who ate pizza. Again, for Q an interrogative quantifier and D an interrogative determiner function, we write $\mathbf{D}$(Q) and $\mathbf{D}$(D) for the distributive lifts of Q and D respectively.

Definition 9 **(Distributive lifts)** *Let $Z, W \subseteq E, Y, X \subseteq \mathcal{P}(E)$, and $AT(Z) = \{W | W \subseteq Z \,\&\, |W| = 1\}$. Then,*
(i) $\mathbf{D}(WHO) \in [\mathcal{P}(\mathcal{P}(E)) \rightarrow [_q\mathcal{P}(\mathcal{P}(E)) \rightarrow 2]]$
$\mathbf{D}(WHO)(Y)(X) = 1$ *iff* $X = AT(PERSON) \cap Y$
(ii) $\mathbf{D}(WHICH_{pl}) \in [\mathcal{P}(E) \rightarrow [\mathcal{P}(\mathcal{P}(E)) \rightarrow [_q\mathcal{P}(\mathcal{P}(E)) \rightarrow 2]]]$
$\mathbf{D}(WHICH_{pl})(W)(Y)(X) = 1$ *iff* $X = AT(Z) \cap Y$

The intended interpretation of (15) is:

(16) $\mathbf{D}(WHICH_{pl})(STUDENTS)(EAT_PIZZA)(X) = 1$ iff
$X = AT(STUDENTS) \cap EAT_PIZZA$

4 Answers and Linguistic Responses

4.1 Determiner Questions

In principle, there are no logical restrictions as to what types can be suitable answer spaces of a question. One can construct languages in which there are questions over all the denotable types of the language. This is clearly not the case in natural languages, where only a subset of the denotable types are suitable answer sets. Let $[_q[\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow 2]] \rightarrow 2]$ be the set of determiner questions, and $[_q[\mathcal{P}(E) \rightarrow 2] \rightarrow 2]$ be the set of generalized quantifier questions. In English *how many* and *whose*-questions are the only candidates to be defined as determiner questions. Consider the following question/answer pairs:

(17) a. How many apples are in the bag?

b. Six /??At least three /*Most.

The example in (17) illustrates the fact that *how many*-questions can only be answered with cardinal determiners. Moreover, only cardinal determiners of the form $EXACTLY_n$ constitute genuine complete true answers. To see this point, consider *at least six* as the answer of (17). As discussed at the beginning of the paper answers of this sort do not resolve the question properly since they are compatible with there being exactly six apples in the bag or two thousand. In this respect, they are partial answers and do not resolve the question completely. Now we define $HOW_MANY(Z)(Y)$ as a determiner question:

Definition 10 *For all determiners $D \in \{EXACTLY_n : n \in \mathcal{N}\}$, all $Z, Y \subseteq E$:*
$HOW_MANY(Z)(Y) \in [_q\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow 2]]$ *and*
$HOW_MANY(Z)(Y)(D) = 1$ *iff* $D(Z)(Y) = 1$

In general, questions of these sort are not limited to answers in the form of determiner expressions. They can also be answered with noun phrases, like *six apples*. The function HOW_MANY as defined above does not have generalized quantifers in its domain. Theferefore, we have to extend it to a function HOW_MANY^* whose domain includes also generalized quantifiers of the form $EXACTLY_n(Z)$, as follows:

Definition 11 *Let $GQ^{EX} = \{EXACTLY_n(Z) | Z \subseteq E\}$ Then,*
$Dom(HOW_MANY^*(Z)(Y)) = Dom(HOW_MANY(Z)(Y)) \cup GQ^{EX}$,
$HOW_MANY^*(Z)(Y)(D) = HOW_MANY(Z)(Y)(D), \forall$D $\in$ Dom(HOW_MANY(Z)(Y))
$HOW_MANY^*(Z)(Y)(Q) = 1$ *iff*
$\exists n[Q = EXACTLYn(Z) \,\&\, HOWMANY(Z)(Y)(D) = 1]$

Consider now the case of *whose*-questions, where also determiner and quantifier expressions are good constituent responses (18). The determiner question function WHOSE and its extension $WHOSE^*$ would be defined in a similar fashion:

(18) a. Whose cats are on the mat?

b. John's / His / John's cats / *Every cat.

Definition 12 *Let* $POSS = \{x's | x \in E\}$. *For all* $D \in POSS$, *all* $Z, Y \subseteq E$,
$WHOSE(Z)(Y) \in [_q\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow 2]]$ *and*
$WHOSE(Z)(Y)(D) = 1$ *iff* $D(Z)(Y) = 1$

Definition 13 *Let* $GQ^{POSS} = \{D(Z) | Z \subseteq E \,\&\, D \in POSS\}$. *Then,*
$Dom(WHOSE^*(Z)(Y)) = Dom(WHOSE(Z)(Y)) \cup GQ^{POSS}$
$WHOSE^*(Z)(Y)(D) = WHOSE(Z)(Y)(D), \forall D \in Dom(WHOSE(Z)(Y))$
$WHOSE^*(Z)(Y)(Q) = 1$ *iff* $\exists D[Q = D(Z) \,\&\, WHOSE(Z)(Y)(D) = 1]$

4.2 Question resolution

We can also consider the relation between a question and its linguistic answer as indirect. It is mediated by a resolution relation. Question resolution is a natural mechanism since from the denotation of noun phrases we can recover sets (a noun phrase denotes a set of sets). Therefore, one can recover answer sets in $\mathcal{P}(E)$ from answers in $[\mathcal{P}(E) \rightarrow 2]$, and also from answers in $[\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow 2]]$. The three function types correspond to a single class of expressions: argument interrogatives, i.e., those that question one argument of the relation. Given a question f, we need to recover the answer set A_f from an expression ϕ, the linguistic answer, whose type does not match the type of the domain of f. All that answers do is to resolve the question by providing its answer set. A generalized quantifier $D(Z)$ -the denotation of an NP constituent response- resolves a question f iff one its elements which is a subset of the restrictor set is the answer set of the question.

Definition 14 *Let* $D(Z) \in [\mathcal{P}(E) \rightarrow 2]$ *be a generalized quantifier. Then, for all* $f \in [_q\mathcal{P}(E) \rightarrow 2]$, $Resolve(D(Z), f)$ *iff* $A_f \subseteq Z \,\&\, D(Z)(A_f) = 1$

As an illustration of the process involved, consider the question-answer pair in (19):

(19) a. What did you put on the table?
b. Three forks

Applying the above definition, the generalized quantifier denoted by *three forks* resolves question (20) if and only if one of its elements is the answer set of the question:

(20) $WHAT([\![\lambda x.You\ put\ x\ on\ the\ table]\!])(X) = 1$ iff $E \cap [\![\lambda x.You\ put\ x\ on\ the\ table]\!] = X$
$Resolve([\![three\ forks]\!], WHAT([\![\lambda x.You\ put\ x\ on\ the\ table]\!]))$ iff
$A_{WHAT([\![\lambda x.You\ put\ x\ on\ the\ table]\!])} \subseteq$ FORK &
THREE (FORK)($A_{WHAT([\![\lambda x.You\ put\ x\ on\ the\ table]\!])}$) $= 1$ iff
$[\![\lambda x.You\ put\ x\ on\ the\ table]\!] \subseteq FORK$ &
$THREE(FORK)([\![\lambda x.You\ put\ x\ on\ the\ table]\!]) = 1$ iff
$[\![\lambda x.You\ put\ x\ on\ the\ table]\!] \subseteq FORK$ &
$|FORK \cap ([\![\lambda x.You\ put\ x\ on\ the\ table]\!])| = 3$

Notice that more than one generalized quantifier can resolve the same question, as long as the answer set is an element of the resolving quantifiers and a subset of their respective restrictors. The definition of question resolution is also related to exhaustivity.

Fact 15 *(***Exhaustivity and resolution***)*
If $D(Z)$ *resolves* f *with* X, *then* $\neg\exists Y$ *such that* $X \subset Y$ *and* $D(Z)$ *resolves* f *with* Y

Question resolution by determiners seems to pose a problem, since they are not sets of sets. Therefore, one cannot recover a set from their denotation and check whether this set is the answer set of the question. We claim that this is precisely the reason why determiner responses are so scarce. In English, only *how many-* and *whose*-interrogatives admit them clearly. In Spanish and other Romance languages, there is a wider variety of determiners that can occur as constituent responses:

(21) a. ¿Quiénes vinieron a la fiesta?
'Who came to the party?'
b. Algunos/ Muchos/ todos...
some-pl. many-pl. all
'Some people/ many people/ Everybody/...'

Only context-dependent determiners occur as constituent responses to argument interrogatives. These determiners are relativized to context sets and behave like generalized quantifiers in disguise. A type lowering operation of pronominalization(Pron) provides the restrictor of the generalized quantifier: for D a determiner, A a context set, $Pron(D) = D^A(A) = D(A)$. A question f is resolved by a pronominalized determiner *Pron(D)* iff the answer set of f, A_f, is a subset of the context set A and an element of $Pron(D)$. As in the case of standard GQs, we will say then that the pronominalized determiner resolves the question f.

Definition 16 *Let* $Pron(D) = D(A)$, *for A a context set, and* $f \in [_q\mathcal{P}(E) \rightarrow 2]$. *Then,* $Resolve(Pron(D), f)$ *iff* $A_f \subseteq A \,\&\, Pron(D)(A_f) = 1$

5 Modifier Questions

The standard analysis of modifiers, for instance in Keenan & Faltz (1985), is to treat them as denoting functions in $[\mathcal{P}(E^n) \rightarrow \mathcal{P}(E^n)]$, for $n \geq 0$. Nevertheless, modifer interrogative quantifier expressions like *where, when,* etc. behave more like true quantifiers. They quantify over different domains (times, places, manners) and are treated as variable binding operators in grammatical theories that posit logical form representations. Therefore, it seems that what is needed is to conceive modifiers not as maps from n-ary relations to n-ary relations but as arguments of the relation that we can question or quantify over (McConnell Ginet, 1982). Our representation language needs to be extended to a many sorted language with models $\mathcal{M} = << E, < \mathcal{D}_j >_{j \in S} >, \mathcal{I} >$, where S is an index set of sorts and for each $j \in S$,$\mathcal{D}_j$ is a non-empty set. For instance, $\mathcal{D}_l = PLACE$ is the set of locations in the model) , $\mathcal{D}_t = TIME$ is the set of times in the model, etc. It seems reasonable to assume that these domains have a rich underlying structure (see Szabolcsi & Zwarts (1993) for manners and times, Nam (1995) for locatives). Therefore, we have to shift the type of i-ary relations to $E^i \times \prod_j \mathcal{D}_j$. Writing s for $|S|$ here and later, relation-denoting expressions now denote sets of i-tuples of individuals and s-tuples of modifiers. By adopting this view we make the so-called adjuncts or modifiers into arguments. Therefore, adding modifier variables or constants to a relation increases its arity as follows:

Definition 17 *(***Argument extension of a relation by modifiers***) For all* $R \subseteq E^i$, $M \subseteq \prod_j \mathcal{D}_j$, *M is the* argument extension *of a relation R to a i+s-ary relation* $R' \subseteq E^i \times \prod_j \mathcal{D}_j$ *iff*
$R' = \{< \alpha_1, ..., \alpha_i, \mu_1, ..., \mu_s > | < \alpha_1, ..., \alpha_i > \in R \,\&\, < \mu_1, ..., \mu_s > \in M\}$

We are now in a position to introduce the notion of an argument extended question, and the definitions of argument extended interrogative quantifiers and determiners in the new type. For brevity we will also call this type of quantifiers and determiners modifier interrogative quantifiers and modifier interrogative determiners respectively.

Definition 18 *(***Unary modifier questions***)*
$\bigcup_{j \in S}[_q\mathcal{P}(\mathcal{D}_j) \rightarrow 2]$ *is the set of* (unary) modifier questions.

Definition 19 *(***Modifier interrogative quantifiers***) Let* $\mu_l \in PLACE$, $\mu_t \in TIME$, $\mu_m \in MANNER$, $\mu_c \in CAUSE$, $\mu_r \in REASON$. *Then, for all* $n \geq 1$, *all* $R' \subseteq E^i \times \prod_j \mathcal{D}_j$, *and all* $X \subseteq \bigcup_{j \in S} \mathcal{D}_j$:

$WHERE(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff* $\{\mu_l| < \alpha_1,...,\alpha_i,...,\mu_l,...,\mu_s >\in R'\} = X$
$WHEN(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff* $\{\mu_t| < \alpha_1,...,\alpha_i,...,\mu_t,...,\mu_s >\in R'\} = X$
$HOW(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff* $\{\mu_m| < \alpha_1,...,\alpha_i,...,\mu_m,...,\mu_s >\in R'\} = X$
$WHY_c(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff* $\{\mu_c| < \alpha_1,...,\alpha_i,...,\mu_c,...,\mu_s >\in R'\} = X$
$WHY_r(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff* $\{\mu_r| < \alpha_1,...,\alpha_i,...,\mu_r,...,\mu_s >\in R'\} = X$

For $j \in S$, and $X, Y \subseteq \mathcal{D}_j$, let $X \cap_j Y$ be the meet (glb) of X and Y in the lattice with domain $\mathcal{P}(\mathcal{D}_j)$.

Definition 20 *(***Modifier interrogative determiners***) For all* $Z \subseteq E$, *all* $R' \subseteq E^i \times \prod_j \mathcal{D}_j$, *and all* $X \subseteq \bigcup_{j\in S}\mathcal{D}_j$:
$IN_WHICH(Z)(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff*
$\rho(Z) \cap_l \{\mu_l| < \alpha_1,...,\alpha_i,...,\mu_l,...,\mu_s >\in R'\} = X$ [3]
$FOR_WHAT(REASON)(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff*
$REASON \cap_r \{\mu_r| < \alpha_1,...,\alpha_i,...,\mu_r,...,\mu_s >\in R'\} = X$
$AT_WHAT(TIME)(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff*
$TIME \cap_t \{\mu_t| < \alpha_1,...,\alpha_i,...,\mu_t,...,\mu_s >\in R'\} = X$
$IN_WHICH(MANNER)(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff*
$MANNER \cap_m \{\mu_m| < \alpha_1,...,\alpha_i,...,\mu_m,...,\mu_s >\in R'\} = X$

The truth conditions of the interrogative sentence (22a), in which *when* denotes in the type of modifier (argument extended) interrogative quantifiers, are in (22b):

(22) a. When did John arrive?
b. WHEN (ARRIVE')(JOHN)(X) = 1 iff
$\{\mu_l| < JOHN, \mu_1,...,\mu_l,...,\mu_s >\in ARRIVE'\} = X$

A qualification on exhaustivity is needed at this point. The fact that not only *at 10 am* but also *on Monday* or *last week* are also proper responses if John arrived at 10am on Monday last week does not count as evidence against exhaustivity, since all these responses are more or less specific descriptions of the moment of time in which John's arrival took place. Otherwise, they do not resolve the question. Not considered in the previous definition are degree questions, a special class of modifier questions. Assuming that degree predicates like *tall* denote functions from individuals to degrees, we say that a property G is *gradable* iff $G \subseteq E \times \mathcal{D}_d$, where $\mathcal{D}_d$ is a domain of degrees δ.

Definition 21 *(***Degree questions***) For all* $G \subseteq E \times \mathcal{D}_d$, *all* $R' \subseteq E^i \times \prod_j \mathcal{D}_j$, *all* $\delta \in \mathcal{D}_d$ *and all* $X \subseteq \bigcup_{j\in S}\mathcal{D}_j$:
$HOW(G)(R')(\alpha_1,...,\alpha_i)(X) = 1$ *iff*
$\{\delta|\exists x. < x, \delta >\in G\} \cap_d \{\delta| < \alpha_1,...,\alpha_i,...,\delta,...,\mu_s >\in R'\} = X$

Sentence (23a) is a degree question. Its answer set is the set of degrees in the meet(glb) of $\{\delta|\exists x. < x, \delta >\in FAST\}$ and $\{\delta| < JOHN,...,\delta,...,\mu_s >\in RUN'\}$.

(23) a. How fast did John run?
b. $HOW(FAST)(RUN')(JOHN)(X) = 1$ iff
$\{\delta|\exists x. < x, \delta >\in FAST\} \cap_d \{\delta| < JOHN,...,\delta,...,\mu_s >\in RUN'\} = X$

The approach to questions that I am developing gives a semantic solution to the problem of preposition stranding in interrogatives. The descriptive generalization for the contrast in (24) is that in sentence (24a) the preposition stays in its place and the *wh*-word "moves" to sentence initial position. In sentence (24b) the whole PP has moved to the initial position.

(24) a. Which box did you put my shoes in?

[3] I take ρ to be an operator mapping sets of individuals to the region of space (location) occupied by those individuals. For example, $\rho(BOX)$ is the region of space occupied by the denotation of *box* in the model. Note that, for all Z, $\rho(Z) \subseteq PLACE$. See Nam (1995) for further details.

b. In which box did you put my shoes?

Question (24a) is an argument question, whereas question (24b) is a modifier question. As a matter of fact, even their linguistic answers are different. (25a) is an adequate answer to (24a) and (25b) is an adequate answer to (24b), but they cannot be interchanged.

(25) a. This box.
b. In this box.

The truth conditions of the interrogative sentences in (24) are as follows:

(26) a. $WHICH([\![\lambda x.Box(x)]\!])([\![\lambda x.You\ put\ my\ shoes\ in(x)]\!])(X) = 1$ iff
$BOX \cap \{x|You\ put\ my\ shoes\ in\ (x)\} = X$
b. $IN_WHICH([\![\lambda x.Box(x)]\!])([\![\lambda \mu_l.You\ put\ my\ shoes\ (\mu_l)]\!])(X) = 1$ iff
$\rho([\![\lambda x.Box(x)]\!]) \cap_l \{\mu_l| < [\![you]\!], [\![myshoes]\!], ..., \mu_l, ..., \mu_s >\in PUT'\} = X$

The equivalence of the stranded preposition and non-stranded preposition interpretations is not immediate. It comes from the general equivalence between *which*-questions and *in which*-questions when the ρ operator is applied to the answer set of the first one:$\rho(A_{WHICH(Z)(Y)}) = A_{INWHICH(Z)(Y)}$. In other words, answers (25a) and (25b) are spatially equivalent.

6 Properties of Interrogative Determiners and Quantifiers

6.1 Conservativity and Intersectivity

Some characterizing properties of declarative quantifiers seem to hold of interrogative quantifiers. Here we will restrict ourselves to argument interrogative determiners and quantifiers, but most of the claims hold also for modifier interrogative determiners. Keenan & Westerståhl (1994) observe that interrogative quantifiers satisfy conservativity and extension, and give the following examples to illustrate this fact:

(27) a. Which roses are red? = Which roses are roses and are red?
b. Whose cat can swim? = Whose cat is a cat that can swim?

The claim holds not just for WHICH and WHOSE but for any argument interrogative determiner.

Definition 22 *(***Generalized Conservativity***) Let E be a universe and X any set. Then,* $D \in [\mathcal{P}(E) \to [\mathcal{P}(E) \to X]$ *is* conservative *iff for all* $A, B, B' \subseteq E$, *if* $A \cap B = A \cap B'$ *then* $D(A)(B) = D(A)(B')$. *Equivalently:* $D(A)(B) = D(A)(A \cap B)$

Conservativity of declarative and interrogative determiners follows from the above definition, since in the case of declarative determiners X is the set of truth values and in the case of argument interrogative determiners X is the set of argument questions. It also follows that if an interrogative determiner D satisfies conservativity, then $D(A)(B)$ and $D(A)(A \cap B)$ are the same question function. Applying the definition to (27a,b) we see that WHICH(ROSE)(RED) = WHICH(ROSE)(ROSE ∩ RED).

Fact 23 *All argument interrogative determiners are conservative*

Argument interrogative determiners all satisfy the property of extension.

Definition 24 *(***Generalized Extension***) For all* $D \in [\mathcal{P}(E) \to [\mathcal{P}(E) \to 2]]$, D *satisfies* extension *iff for all* $A, B \subseteq E \subseteq E', D_E(A)(B) = D_{E'}(A)(B)$

With respect to the property of permutation invariance (PI), it is interesting to note that WHO respects a local notion of it (Westerståhl, 1985), namely when we fix the set PERSON in all permutations. Context restricted determiners are also invariant under permutations that satisfy Locality, defined as follows:

Definition 25 *(**Locality**) Let $X \subseteq E$, PERM(E) be the set of permutations of E and D a determiner function. We say that D is* PI at X *iff* $\forall \pi \in PERM(E)$, *if* $\pi(X) = X$ *then* $D^X(A)(B) = D^X(\pi(A))(\pi(B))$.

The determiner WHOSE satisfies a more specific condition that van Benthem (1986) calls "quality". It requires that all permutations π preserve the possession relation induced by *Poss* (a possession relation).

Argument interrogative determiners satisfy a stronger invariance condition than conservativity. They are all intersective or generalized existential (Keenan 1987, 1993).

Definition 26 *(**Generalized Intersectivity**) Let X be any set. Then, $D \in [\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow X]$ is* intersective *iff for all $A, B, A', B' \subseteq E$, if $A \cap B = A' \cap B'$ then $D(A)(B) = D(A')(B')$. Equivalently: $D(A)(B) = D(A \cap B)(A \cap B)$*

6.2 Generalized Existential Interrogative Functions and Context Neutrality

If a determiner is intersective, then the denotation of D(A)(B) depends only on the intersection of the arguments. In the interrogative domain, we have seen that to determine the answer set of an argument question we only have to know the intersection of A and B. This set is precisely the answer of the question. Keenan (1987) and Lappin (1988) show that a conservative binary determiner is intersective iff it is existential iff it is symmetric. The generalized definition of the two latter notions for $< 1, 1 >$ determiners is as follows (their application to interrogative determiners follows again as a special case):

Definition 27 *(i) For all $D \in [\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow X]]$, D is* generalized existential *iff for all $A, B \subseteq E, D(A)(B) = D(A \cap B)(E)$.*
(ii) For all $D \in [\mathcal{P}(E) \rightarrow [\mathcal{P}(E) \rightarrow X]]$, D is symmetric *iff for all $A, B \subseteq E, D(A)(B) = D(B)(A)$.*

The fact that argument interrogative determiners satisfy intersectivity is equivalent to being existential. The question (28b) denoted by the interrogative sentence (28a) is equivalent to the one denoted by (28c). [4]

(28) a. How many students are vegetarians?
b. HOW MANY (STUDENT)(VEGETARIAN)
c. How many students who are vegetarians are there/exist?
d. HOW MANY(STUDENT ∩ VEGETARIAN)(E)

Since argument interrogative determiners and declarative determiners like SOME are generalized existential functions in their respective domain, it is not surprising that in many languages, like Chinese, Greek, Latin, Romance, etc., the declarative quantifier is

[4] There are some exceptions, though. Heim (1987) notices the marginal status of the following existential constructions with *which* and *which of the* determiners.

(i) a. ??Which one of the three men was there in the room?
b. ??Which actors were there in the room?

Notice that the functions $WHICH_{pl}$ and $WHICH_ONE_OF_THE_TWO$ are inherently context restricted. Therefore, they are related to a set of entities already present in the discourse and are not compatible with presentational/existential predicates that require "discourse novel" answer sets. Furthermore, the determiner function $WHICH_n_OF_THE_m$ defined as $WHICH_n_OF_THE_m(A)(B)(X) = 1$ iff $|A| = m \,\&\, X = A \cap B \,\&\, |X| = m$ is not intersective.

derived from the interrogative one by attaching a morpheme to it. In other languages the same word is used for some declarative and argument interrogative determiners

The equivalence between intersectivity and symmetry apparently poses some problems. Consider the interrogative sentence in (29a):

(29) a. How many vegetarians are students?
b. HOW MANY (VEGETARIAN)(STUDENT)

Since intersective determiners are symmetric, the questions in (28b) and (29b) should be equivalent. In fact, they are. The answer set of (29b) is the intersection set of the denotations of student and vegetarian which is also the answer set of (28b). However, the intuition remains that the two questions are "about" different things. Imagine a situation in which a school's cook wants to know the number of students who are vegetarians. Sentence (28a) would be felicitous in that situation whereas (29a) would not be so, despite the fact that their respective answer sets are the same. Higginbotham (1993) relates the contrast to the property of domain restriction. Since domain restriction is formally defined as conservativity + extension (Keenan & Westerståhl, 1994) and symmetry is a property satisfied only by a subset of the determiners that satisfy domain restriction, namely those which are intersective, it seems more reasonable to relate the "aboutness" problem with this latter property of interrogative determiners. From an information-based perspective one can easily conclude that the difference between the two sentences is that their respective topics or themes are different. In other words, they are not context neutral. We can generalize this new property as follows:

Definition 28 *(**Context Neutrality**) For all context sets $C \subseteq E$, all determiners D, D is* C-neutral *iff for all $A, B, X \subseteq E$, $D^C(A)(B) = D(A)(B)$.*

When a determiner is context neutral in a given context C its arguments can be inverted preserving truth values. This property only makes a difference in the case of intersective determiners. Co-intersective determiners (Keenan, 1993) are not symmetric nor are non-intersective determiners in general. Therefore, the sentences in (30) are only equivalent if STUDENT = VEGETARIAN.

(30) a. Every student is a vegetarian.
b. Every vegetarian is a student.

Since all argument interrogative determiners are intersective, context neutrality becomes a relevant issue. The property captures the idea that when a symmetric determiner is relativized to a non-empty context set then its arguments cannot be flipped in general, i.e., the determiner is not context neutral.

6.3 Monotonicity, Entailment and Negative Polarity Items (NPIs) Licensing

Argument interrogative determiners can be characterized as continuous in their monotonicity behaviour. Continuous functions are meets of increasing and decreasing functions.

Definition 29 *An argument interrogative quantifier $Q \in [\mathcal{P}(E) \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]$ is* continuous *iff for all $A, B, C \subseteq E$, if $A \subseteq B \subseteq C$ and $Q(A) = Q(C) = f$ then $Q(B) = f$.*

Fact 30 *Argument interrogative quantifiers are continuous*
Proof: Let D(A) be an argument interrogative quantifier. Assume for arbitrary sets W,Y,Z that $W \subseteq Y \subseteq Z$ and $D(A)(W) = D(A)(Z) = f$. Therefore, there is an X such that $D(A)(W)(X) = D(A)(Z)(X) = 1$. Then, $A \cap W = A \cap Z = X$ and, since $W \subseteq Y \subseteq Z, A \cap Y = X$ so $D(A)(Y)(X) = 1$ □

There is a grammatical fact associated with monotonicity which seems not to follow from this characterization. It is a fairly common observation that negative polarity items are licensed in interrogative sentences.

(31) a. Which student has *ever* been to Moscow?
b. Do you have *any* money?
c. Who has ever *lifted a finger* to help us?

The piece of data above does not follow from the properties of questions studied so far, since as observed for the first time by Fauconnier (1975) and Ladusaw (1979) negative polarity items are licensed in decreasing environments. Nevertheless, although interrogative determiners are continuous, questions can be considered as downward entailing with respect to their answer sets. Consider the following questions:

(32) a. Which guests smoked?
b. Which guests smoked cigars?
c. In which state do you have relatives?
d. In which state of the West Coast do you have relatives?
e. How many cars are parked in the garage?
f. How many red cars are parked in the garage?

There is a natural information-based relation between (32a) and (32b) above. Namely, a true complete answer to (32b) is a partial (and possibly complete) answer to (32a). Informally, (32b) asks for more specific information than (32a). In other words, if A_f is the answer set of (32a), then a subset of A_f is the answer set of (32b). The same applies to (32c) with respect to (32d) and to (32e) with respect to (32f). Let us call this relation between questions *subsumption*:

Definition 31 *Question* f subsumes *question* g ($f \leq g$) *iff* $A_g \subseteq A_f$.

Clearly, the subsumption relation is a partial order (reflexive, antisymmetric and transitive). Then, if we allow not only the monotonicity behaviour of the quantifier but also the subsumption relations between questions to enter the picture, interrogative determiners will exhibit the entailment pattern of declarative NO. As noted above, if question f subsumes question g, then a complete true answer to g is a partial or complete true answer to f but not necessarily viceversa. [5]

Definition 32 *(i) An interrogative quantifier* Q *is* decreasing *iff* $\forall A, B \subseteq E$ *if* $A \subseteq B$ *then* $Q(B) \leq Q(A)$
(ii) An interrogative determiner D *is* decreasing *iff* $\forall A, B, C \subseteq E$ *if* $A \subseteq B$ *then* $D(B)(C) \leq D(A)(C)$

Fact 33 *Argument interrogative quantifiers* Q *are decreasing*
Proof: Let $A, B, C \subseteq E$, $A \subseteq B, Q = D(C)$ *and* D = *WHICH, WHAT, etc . We have to show that for arbitrary* X, Y, *if* $Q(B)(X) = Q(A)(Y) = 1$, *then* $Y \subseteq X$. *Assume* $Q(B)(X) = Q(A)(Y) = 1$. *Since* $A \subseteq B, Y = C \cap A \subseteq C \cap B = X$. □

Fact 34 *Argument interrogative determiners* D *are decreasing*
Proof: Let $A, B, C \subseteq E$ *and* $A \subseteq B$. *We have to show that* $D(B)(C) \leq D(A)(C)$. *Let* X, Y *be such that* $D(B)(C)(X) = 1$ *and* $D(A)(C)(Y) = 1$. *Then,* $Y = A \cap C \subseteq B \cap C = X$. □

[5] The subsumption relation presented here is apparently different from the relation of entailment between questions in G&S(1989). For them the entailment relation holds between propositions and here subsumption holds between questions (it is the subset relation between answer sets). Notice, however, that if question f subsumes question g, then question f entails question g in G&S' (1989) sense, so the notion of subsumption could also be captured in their terms. Notice also that the notion of subsumption is identical to Higginbotham's (1993) notion of downward entailment for interrogatives.

The notion of subsumption given above predicts entailments between questions arising from their monotonicity pattern as the ones illustrated in (32a) to (32f) above. A complete (partial) answer to question (32b) will be a partial (complete) answer to (32a) since the answer set of (32b) is a subset of the answer set of (32a). Fact 34 also predicts that negative polarity items can occur in the first argument of interrogative determiners.

(33) Which students that have *ever* been to Moscow want to go back there?

The presence of NPIs in interrogative environments triggers a peculiar phenomenon observed, among others, by Linebarger (1991). In all the examples above involving NPIs the interpretetation of the questions as rhetorical is either available or strongly preferred. A rhetorical question is not a "well-behaved" question. The speaker knows already the answer and he asks it for rhetorical purposes (irony). For instance, in question (31c) the speaker knows already that the answer set of the question is empty but he asks it to highlight precisely this fact: that the set of persons who have done something to save us is empty. A sentence like (34) uttered as a rhetorical question has an empty answer set. Assume that the speaker knows that nobody came ($PERSON \cap COME = \emptyset$). Then, he would ask this question for rhetorical reasons.

(34) Who came?

Sentence (35) presents the opposite case. Assume that the speaker knows that everybody went to the party: $PERSON \subseteq COME$,i.e. $PERSON \cap COME = PERSON$. Therefore, for rhetorical reasons, he would ask:

(35) Who didn't come?

The answer set of (35) is $PERSON \cap *COME = \emptyset$, since everybody went to the party. In sum, for a speaker to be able to ask a rhetorical question he has to calculate the complement of an answer set and ask a question about it. He has to be able to go over the whole entailment set of a question and pick its smallest element. The presence of the NPI signals precisely this calculation (Fauconnier, 1975). Nevertheless, we are not claiming that rhetorical interpretations arise only when there are NPIs in the sentence. As observed in the literature, practically any question can be interpreted as rhetorical, depending on the circumstances and the speaker's intentions. What needs to be stressed is the close relationship between answer set entailment and the calculation of rhetorical questions. If question f is rhetorical ($Rhet(f)$), then$A_f = \emptyset$.

Definition 35 *(***Subsumption set of a question***)* $SU_f = \{g | f \leq g\}$

Fact 36 *If Rhet(f) then* $SU_f = \{f\}$.

Further evidence for the semantic treatment of rhetorical questions comes from the behaviour of *why* and *how*-questions. The occurrence of NPIs in these sentences does not trigger rhetorical readings (Lawler, 1971).

(36) a. Why did you tell *anybody* about us?
b. How did *anybody* buy that house?

Sentence (36a) presupposes that the addressee told somebody about them and (36b) presupposes that somebody bought the house. Neither of the questions has empty answer sets. Szabolcsi & Zwarts (1993) claim that manners and reasons are structured as join semilattices with no least element. They are closed under joins but not under complements. Being lattices without a bottom element, they cannot constitute proper denotations of rhetorical questions (there is no empty manner or reason). Therefore, the explanation of why there are no proper rhetorical why and how questions is mainly semantic. Since *why* and *how*-questions cannot have empty answer sets, they do not meet the essential denotational requirement to be a rhetorical question.

7 Multiple Questions

In previous sections we have only analyzed sentences with one interrogative quantifier. In order to give a proper semantics of multiple questions -sentences where more than one interrogative quantifier interact- we have to define the nominative, accusative and dative extensions of an interrogative quantifier. This will allow us to give a surface compositional semantics of English interrogative VPs like *bought what* or *bought for whom*.

Definition 37 *(***Extensions of Interrogative Quantifiers***)*
Let $R \subseteq E^n, , \alpha_1, ..., \alpha_n \in E, X \subseteq E, Q \in [\mathcal{P}(E) \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]$. *Then,*
(i) A nominative interrogative quantifier *(or the nominative extension of Q)is a function* $Q_1 \in [\mathcal{P}(E^n) \rightarrow [E^{n-1} \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]]$ *defined as follows:*
$Q_1(R)(\alpha_2, ..., \alpha_n)(X) = 1$ *iff* $Q(\{\alpha | < \alpha_1, ..., \alpha_n > \in R\})(X) = 1$
(ii) An accusative interrogative quantifier *(the accusative extension of Q) is a function* $Q_2 \in [\mathcal{P}(E^n) \rightarrow [E^{n-1} \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]]$ *defined as follows:*
$Q_2(R)(\alpha_1, \alpha_3, ..., \alpha_n)(X) = 1$ *iff* $Q(\{\alpha_2 | < \alpha_1, ..., \alpha_n > \in R\})(X) = 1$
(iii) A dative interrogative quantifier *(the dative extension of Q)is a function* $Q_3 \in [\mathcal{P}(E^n) \rightarrow [E^{n-1} \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]]$ *defined as follows:*
$Q_3(R)(\alpha_1, \alpha_2, \alpha_4, ..., \alpha_n)(X) = 1$ *iff* $Q(\{\alpha_3 | < \alpha_1, ..., \alpha_n > \in R\})(X) = 1$

The interpretation of the VP *buy what* is the following:

(37) $WHAT_2(BUY)(\alpha)(X) = 1$ iff $WHAT(\{\beta | < \alpha, \beta > \in BUY\})(X) = 1$ iff $X = E \cap \{\beta | < \alpha, \beta > \in BUY\}$.

The Chinese version of the VP is interpreted as above. In English, it would receive an "echo" interpretation, arising from the fact that *what* denotes $WHAT^X$ when it does not occur in its canonical fronted position. There is a significant difference between declarative and interrogative argument quantifiers. Declarative Qs behave as "arity reducers" (van Benthem, 1986; Keenan & Westerståhl, 1994). They take an n-ary relation as input and return an n-1-ary relation. Interrogative quantifiers are not arity reducers. They take an n-ary relation and return another n-ary relation. The output n-ary relation is not the same as the input one. The argument that has been "queried" is turned into an answer set. Consider the following sentence:

(38) Which men love which women?

In its most natural reading, question (38) asks for the sets of pairs in the love relation whose first coordinate is a member of the set of men and its second coordinate is a member of the set of women.

(39) $(WHICH_{pl}MAN, WHICH_{pl}WOMAN)(LOVE)(S) = 1$ iff $S = R \cap MAN \times WOMAN$

Sentence (38) denotes a binary question, i.e., a function mapping a binary relation S to true iff $S = R \cap MAN \times WOMAN$. Generalizing to the n-ary case, we define first the notion of a n-ary argument question and afterwards the polyadic $WHICH_{pl}$ interrogative quantifier induced by a sequence of n $WHICH_{pl}$ quantifiers:

Definition 38 *For* $n \geq 1, [_q\mathcal{P}(E^n) \rightarrow 2]$ *is the set of* n-ary argument questions.

Definition 39 *(***Polyadic resumption of** $WHICH_{pl}$ **quantifiers***)*
Let $R, S \subseteq E^n, Z_1, ..., Z_n \subseteq E$. *Then,* $\text{Res}(WHICH_{pl(1)}Z_1, ..., WHICH_{pl(n)}Z_n) \in [\mathcal{P}(E^n) \rightarrow [_q\mathcal{P}(E^n) \rightarrow 2]]$ *is defined as follows:*
$\text{Res}(WHICH_{pl(1)}Z_1, ..., WHICH_{pl(n)}Z_n)(R)(S) = 1$ *iff* $S = R \cap Z_1 \times ... \times Z_n$

All interrogative quantifiers can participate in multiple questions. We treat first the resumptions of arbitrary argument interrogative quantifiers except $WHICH_{sg}$.

Definition 40 *(***Resumption of argument interrogative quantifiers***)* *Let* $R, S \subseteq E^n, Z_1, ..., Z_n \subseteq E$, *for all* Q_i,

$$\text{let } Z_i = \begin{cases} Z & \text{if } Q_i = WHAT\ (Z_i) \text{ or } Q_1 = WHOSE(Z_i) \\ PERSON & \text{if } Q_i = WHO \\ E & \text{if } Q_i = WHAT \end{cases}$$

$\text{Res}(Q_1, ..., Q_n)(R)(S) = 1$ *iff*
$S = R \cap Z_1 \times ... \times Z_n$

The reducibility result that we prove states that the answer set that we get by an application of the prefix in definitions 39, 40 and the one that we get by successively applying the n interrogative quantifiers in the prefix are equivalent in a sense we make precise below. Therefore, what we really prove is equivalence of answer sets or, in less formal terms, the questions that we get with an absorbed interrogative quantifier and the iterated application of the members of the quantificational prefix are querying "about" the same object.

Fact 41 *The polyadic* $\text{Res}(Q_1, ..., Q_n)$ *is reducible to iterations*

Instead of a full proof we give here a worked out example. Consider the following sentence:

(40) Who gave what to whom?

Applying definition 37(iii), we see that $WHO_3(GIVE)$ is a function in $[E \rightarrow [E \rightarrow [_q\mathcal{P}(E) \rightarrow 2]]]$. Then,$WHO_3(GIVE)(\alpha, \beta)(X) = 1$ iff $X = PERSON \cap \{\gamma | < \alpha, \beta, \gamma > \in GIVE\}$. In the second step of the calculation, the accusative extension of WHAT, $WHAT_2$ applies to $WHO_3(GIVE)$, and we get the function $WHAT_2(WHO_3(GIVE)) \in [E \rightarrow [_q\mathcal{P}(E^2) \rightarrow 2]]$ defined as $WHAT_2(WHO_3(GIVE))(\alpha)(S) = 1$ iff $S = \{< \beta, \gamma > | \beta \in E \,\&\, \gamma \in WHO_3(GIVE)(\alpha, \beta)\}$ (writing $\gamma \in WHO_3(GIVE)(\alpha, \beta)$ for $\gamma \in A_{WHO_3(GIVE)(\alpha,\beta)}$). Finally, the nominative extension of WHO, WHO_1, applies to $WHAT_2(WHO_3(GIVE))$ yielding the function $WHO_1(WHAT_2(WHO_3(GIVE))) \in [_q\mathcal{P}(E^3) \rightarrow 2]$ such that $WHO_1(WHAT_2(WHO_3(GIVE)))(S) = 1$ iff $S = \{< \alpha, \beta, \gamma > | \alpha \in PERSON \,\&\, < \beta, \gamma > \in WHAT_2(WHO_3(GIVE))(\alpha)\}$. In general, $Q_1(...(Q_n(R))) \in [_q\mathcal{P}(E^n) \rightarrow 2]$. Let $Q_i = D(Z_i)$, for some $Z_i \subseteq E$. Then, $Q_1(...(Q_n(R)))(S) = 1$ iff $S = \{< \alpha_1, ..., \alpha_n > | \alpha_1 \in Z_1 \,\&\, < \alpha_2, ..., \alpha_n > \in Q_2(...(Q_n(R))(\alpha_1)\}$.

Resumptions of $WHICH_{sg}$ determiners deserve a more detailed analysis. Higginbotham & May (1981) claim that multiple $WHICH_{sg}$ questions have two interpretations: a singular interpretation and a bijective interpretation. Consider the following sentence:

(41) In *Gone with the wind*, which character admires which character?

The singular interpretation of the sentence is the one rendered by the answer in (42a). The answer in (42b) corresponds to the bijective reading.

(42) a. Ashley Wilkes admired Rhett Butler
b. Ashley Wilkes admired Rhett Butler and Melanie Wilkes admired Scarlett O'Hara.

Under the singular interpretation of the question, the answer set is a singleton, i.e., it consists of a unique pair. The bijective interpretation asks for a bijection between the restriction sets of the interrogative quantifiers. [6] The following definition characterizes H&M's intuition:

[6] According to H&M the bijective reading is only available when "the domain of quantification in the subject NP is disjoint from that of the object"(p.46). Nevertheless, the claim does not seem to be completely correct for all English dialects (Ed Keenan, p.c.). We assume, then, that the two readings are generally available, though accomodating H&M's disjointness condition is straightforward.

Definition 42 *(Singular reading of* $(WHICH_{sg}, ..., WHICH_{sg})$ **quantifiers)**
$Sg(WHICH_{sg(1)}(Z_1), ..., WHICH_{sg(n)}(Z_n))(R)(S) = 1$ *iff*
$S = R \cap Z_1 \times, ..., \times Z_n \,\&\, |S| = 1$
(The disjointness condition (H&M, 1981): $\bigcap_i Z_i = \emptyset$ *)*

The singular reading is the one that we get by iterated application of $WHICH_{sg}(Z)$ quantifiers. Therefore, it is reducible and does not lie "beyond the Frege boundary". We are able to get the interpretation above by applying a $WHICH_{sg(i)}(Z_i)$ quantifier to the function $WHICH_{sg(i+1)}(Z_{i+1})(...(WHICH_{sg(n)}(Z_n)(R))$. It can be conceived of as the polyadic resumption of $WHICH_{sg}(Z)$ quantifiers, so definition 42 corresponds to $Res(WHICH_{sg(1)}(Z_1), ..., WHICH_{sg(n)}(Z_n))$.

Fact 43 *The singular reading is derived from iterations of* $WHICH_{sg}(Z)$ *quantifiers.*
Proof: Let $R \subseteq E^n, Z_1, ..., Z_n \subseteq E$. *Then,*$WHICH_{sg(n)}(Z_n)(R)(\alpha_1, ..., \alpha_{n-1})(X) =$ *1 iff* $X = Z_n \cap \{\alpha_n | < \alpha_1, ..., \alpha_n >\in R\} \,\&\, |X| = 1$. *For all* i $(1 \leq i \leq n-1)$, $WHICH_{sg(i)}(Z_i)(...(WHICH_{sg(n)}(Z_n)(R)))(\alpha_1, ..., \alpha_{i-1})(S) = 1$ *iff* $S = \{< \alpha_i, ..., \alpha_n > | \alpha_i \in Z_i \,\&\, < \alpha_{i+1}, ..., \alpha_n >\in WHICH_{sg(i+1)}(Z_{i+1})(...(WHICH_{sg(n)}(Z_n)(R)))(\alpha_1, ..., \alpha_i)\}$ $\&\, |S| = 1$.□

In the bijective reading there is an apparent loss of the uniqueness condition imposed by $WHICH_{sg}$, due to the fact that the polyadic is not reducible to iterations of $WHICH_{sg}(Z)$ quantifiers.

Definition 44 *Let* $R, S \subseteq E^n, Z_1, ..., Z_n \subseteq E$. *Then,*
$Bij(WHICH_1(Z_1), ..., WHICH_n(Z_n)) \in [\mathcal{P}(E^n) \to [_Q \mathcal{P}(E^n) \to 2]]$ *and*
$Bij(WHICH_1(Z_1), ..., WHICH_n(Z_n))(R)(S) = 1$ *iff* $S = R \cap Z_1 \times ... \times Z_n \,\&$
$\forall \alpha_i \exists! < \alpha_1, ..., \alpha_{i-1}, \alpha_{i+1}, ..., \alpha_n >\in Z^1 \times ... \times Z^{i-1} \times Z^{i+1} \times ... \times Z^n$
such that $< \alpha_1, ..., \alpha_n >\in R$

8 Interactions of declarative and interrogative quantifiers

In the previous section we analyzed how interrogative quantifiers are combined (multiple questions). In this section we treat the combinations of interrogative and declarative quantifiers. Different interactions give rise to different readings of the interrogative sentence. The first reading to be considered is the *individual* (G&S, 1984) or "single constituent" reading (Chierchia, 1993). For example, the individual reading of sentence (43a) is reflected in the response (43b). In terms of the behaviour of the quantifiers, this reading corresponds to the iteration of the declarative and the interrogative quantifier, as (44) shows.

(43) a. What did every boy read?
b. *Tom Sawyer* and *The Jungle Book*

(44) $WHAT_2(EVERY BOY_1(READ))(X) = 1$ iff
$WHAT(\{\beta | BOY \subseteq \{\alpha | < \alpha, \beta >\in READ\}\})(X) = 1$ iff
$X = \{\beta | BOY \subseteq \{\alpha | < \alpha, \beta >\in READ\}\}$

Sentence (43a) denotes a unary question in its individual reading, as does (45a). In (43a) the accusative extension of the interrogative quantifier combines with the nominative extension of the declarative quantifier. In (45a) the nominative extension of the interrogative quantifier combines with the accusative extension of the declarative quantifier.

(45) a. Which students read more than three books?
b. John and Sam

(46) $WHICH STUDENTS_1(M_3 BOOKS_2(READ))(X) = 1$ iff
$WHICH STUDENTS_1(\{\alpha | |BOOK \cap \{\beta | < \alpha, \beta >\in READ\}| > 3\})(X) = 1$ iff
$X = STUDENT \cap \{\alpha | |BOOK \cap \{\beta | < \alpha, \beta >\in READ\}| > 3\} \,\&\, |X| \geq 2$

A second type of reading is called by G&S *pair-list* reading, since the answer specifies a set of pairs. The response in (47c) would be a pair-list answer for sentences (47a) and (47b).

(47) a. Which book did each boy read?
b. Which book did these boys (each) read?
c. Bill read *Tom Sawyer* and Joe read *The Jungle Book*

The problem with pair-list readings is that they are not available with any declarative quantifier. G& S(1984), Chierchia (1993) and Szabolcsi (1994) have observed that only quantifiers that denote principal filters give rise to pair-list readings, as the examples in (48), taken from Szabolcsi (1994), illustrate:

(48) a. Which boys did every dog bite? (pair-list o.k.)
b. Which boys did the dogs bite? (pair-list o.k.)
c. Which boys did two dogs bite? (pair-list o.k. if TWO DOGS is a principal filter)

A generalized quantifier Q is a principal filter iff it has a generator set GSET(Q) defined as follows:

Definition 45 *X is a* generator set *for Q (GSET(Q) = X) iff*
$Q(X) = 1 \,\&\, \forall Y[Q(Y) = 1$ *iff* $X \subseteq Y]$

Declarative quantifiers such as $MORE_THAN_THREE BOYS$ or $FEW BOYS$ are not principal filters and, as expected, sentences (49a) and (49b) lack pair-list readings.

(49) a. Which book did more than three boys read?
b. Which book did few boys read?
c. *Bill read *Tom Sawyer* and Joe read *The Jungle Book*

An interrogative sentence with one interrogative and one declarative quantifier denotes a binary question in its pair-list reading. It maps a unique set of pairs to true, the set specified by responses such as (47c). When the interrogative sentence has n declarative quantifiers and m interrogative quantifiers, the "$n + m - tuple$"-list reading is a $n + m$-ary question. For example, the triple-list readings of sentences (50a) and (50b) are ternary questions.

(50) a. Which book did each boy put on each desk?
b. Which book did each boy put on which desk?

Combinations of a declarative quantifier which is a principal filter and a modifier interrogative quantifier can be also conceived as binary questions. Therefore these combinations have pair-list readings (51b).

(51) a. When did each of your relatives arrive?
b. Uncle John arrived on Monday, Grandma arrived on Tuesday and my parents on Christmas eve.

Here we will treat only pair-list readings (binary questions) arising from the combination of an argument interrogative quantifier, a declarative quantifier and the denotation of a transitive verb. There are two cases to consider: when the nominative extension of the interrogative quantifier combines with the accusative extension of the declarative quantifier and the opposite case.

Definition 46 *(Pair-list lift)*
(i) Let $Q_1 \in [\mathcal{P}(E) \to [_q\mathcal{P}(E) \to 2]], Q_2 \in [\mathcal{P}(E) \to 2]$, *and* $R, S \subseteq E^2$. *Then,*
Pair-list$(Q_1, Q_2) \in [\mathcal{P}(E^2) \to [_q\mathcal{P}(E^2) \to 2]]$ *and* Pair-list$(Q_1, Q_2)(R)(S) = 1$ *iff*
$S = \{< \alpha, \beta > | \beta \in GSET(Q_2) \& \alpha \in Q_1(R)(\beta)\}$
(where $\alpha \in Q_1(R)(\beta)$ *abreviates* $\alpha \in X$ *such that* $Q_1(R)(\beta)(X) = 1\}$*)*
(ii) Let $Q_1 \in [\mathcal{P}(E) \to 2], Q_2 \in [\mathcal{P}(E) \to [_q\mathcal{P}(E) \to 2]]$, *and* $R, S \subseteq E^2$. *Then,*
Pair-list$(Q_2, Q_1) \in [\mathcal{P}(E^2) \to [_q\mathcal{P}(E^2) \to 2]]$ *and* Pair-list$(Q_2, Q_1)(R)(S) = 1$
iff $S = \{< \alpha, \beta > | \alpha \in GSET(Q_1) \,\&\, \beta \in Q_2(R)(\alpha)\}$

Not all polyadic pair-list lifts with English universal quantifiers seem to be denotable: ALL does not participate in pair-list polyadics at all, only the nominative extension of EVERY seems to participate in them, and EACH shows a much wider flexibility. In the end it is a question of economy: why having polyadic pair-list lifts denotable with EACH, EVERY and ALL when the function denoted would be the same? There is a clear specialization of morphological resources taking place: ALL is used for individual readings and EACH is used for pair-list constructions. Therefore, only distributive-key universal determiners (Gil, 1995) participate in this construction. [7] In some languages like Hungarian or Turkish, the pair-list reading is not expressible with a combination of a declarative and an interrogative quantifier. Only a combination of interrogative quantifiers can express it. The existence of the two alternatives is due to the fact that, for $i, j \in \{1, 2\}, Q_j$ a declarative quantifier and R a binary relation, when the answer set of the polyadic *Pair-list*$(WHICH_{sg(i)}(Z_i), Q_j)(R)$ is a bijective function (a set of pairs $\{< \alpha, \beta > | \forall \alpha \in GSET(Q_j) \exists! \beta$ such that $\beta \in WHICH_{sg(i)}(Z_i)(R)(\alpha)\}$), then *Pair-list*$(WHICH_{sg(i)}(Z_i), Q_j)(R) = Bij(WHICH_{sg(i)}(Z_i), WHICH_j(Z)_j)(R)$. In this case, (52a) and (52b) denote the same question: they have the same answer set (52c).

(52) a. Which boy likes which girl?
b. Which girl does each boy like?
c. John likes Mary, Bill likes Pam, Joe likes Sue.

Functional readings (G&S, 1984; Engdahl, 1986) of questions are specific to combinations of interrogative quantifiers and decreasing declarative quantifiers. In these cases the pair-list response is not possible.

(53) a. Which book did no student like to read?
b. Their last week assignment.

(54) a. Who do few Italian married men like? (Chierchia, 1993)
b. Their mother in law.

Functional readings are intensional renderings of pair-list answers (G&S, 1984; Chierchia, 1993). In clarification contexts, though, the pair-list answer can be explicitly obtained. Consider (55) as an answer to (53a):

(55) Their last week assignment. More explicitly, Bill did not like to read *El Quijote* and Joe did not like to read *Magic Mountain.*

How is this "extensionalization" possible?. Since NO BOY (B) = EVERY $BOY(\neg B)$ [8] and EVERY $BOY(\neg B)$ has a generator, namely BOY, then we may define the corresponding pair-list lift as follows:

(56) *Pair-list*$(NOBOY_1, WHICHBOOK_2)(R)(S) = 1$ iff
$S = \{< \alpha, \beta > | \alpha \in GSET(EVERYBOY_{\neg 1}) \,\&$
$\beta \in WHICHBOOK_2(\neg LIKETOREAD)(\alpha)\}$

[7] The position of EVERY is intermediate, at least for some dialects. Crosslinguistically, the most common scenario is that one universal determiner participates in individual readings (iterations) and a different one, the distributive-key universal determiner, in pair-list readings (polyadic).
[8] NO BOY(B) = EVERY $BOY_{\neg}$ (B) = EVERY BOY $(\neg B) = 1$ iff $BOY \not\subseteq B$ iff $BOY \cap B = \emptyset$.

Interrogatives like (57a) have an additional reading which has been called "cumulative" (Srivastav, 1992). A standard cumulative answer for (57a) would be one like (57b), whereas the pair-list answer would be (57c).

(57) a. Which books did the boys read?
b. They read *Tom Sawyer* and *The Jungle Book.*
c. Bill read *Tom Sawyer* and Joe read *The Jungle Book.*

Srivastav claim that the pair-list reading can be considered as the cooperative spell-out of the cumulative reading when the latter is the preferred interpretation. I perceive more of a semantic difference than of a gricean phenomenon. Cumulative questions are plural n-ary questions, functions that in a situation map a relation between sets to true. Notice also that there does not seem to be a branching lift in the interrogative domain.

Definition 47 *(Plural n-ary questions)*
$[_q\mathcal{P}(\mathcal{P}(E)^n) \to 2]$ *is the set of* plural n-ary questions.

Definition 48 *(Cumulative lift) Let* $i \neq j \in \{1, 2\}, Q_i \in [\mathcal{P}(E) \to [_q\mathcal{P}(E) \to 2]], Q_j \in GQ^{DEF}$ *(the set of definite generalized quantifiers), and* $R \subseteq E^2, S \subseteq \mathcal{P}(E)^2$ *(i.e.* $S \subseteq \mathcal{P}(E) \times \mathcal{P}(E)$*). Then,* Cum$(Q_i, Q_j) \in [\mathcal{P}(E^2) \to [_q\mathcal{P}(\mathcal{P}(E)^2) \to 2]]$ *and*
(i) if $i = 2$, $j{=}1$ *then* Cum$(Q_i, Q_j)(R)(S) = 1$ *iff* $S = A \times B$ *such that*
$A = GSET(Q_j) \,\&\, B = \{\beta | \exists \alpha \in A \,\&\, \beta \in Q_i(R)(\alpha)\}$
(ii) if $i = 1$, $j = 2$ *then* Cum$(Q_i, Q_j)(R)(S) = 1$ *iff* $S = A \times B$ *such that*
$B = GSET(Q_j) \,\&\, A = \{\alpha | \exists \beta \in B \,\&\, \alpha \in Q_i(R)(\beta)\}$

Another type of questions considered by G&S(1984) are choice questions. The name comes from the fact that sentences like (58a) can be paraphrased as "for two boys of your choice, which book did each read".

(58) a. Which books did two boys read?
b. Steve read *A Tale of Two Cities* and Mark read *The Never Ending Story*

There is common agreement that this reading is quite marginal in normal discourse. They are only natural in contests or quizzes and are also called "quiz" questions. The reason for its marginality might be that in the definition of the corresponding polyadic, the relevant domain set cannot be the generator of the declarative quantifier (it does not denote a principal filter) but one of its elements.

Definition 49 *("Choice" lift)*
Let $i \neq j \in \{1, 2\}, Z_j, W \subseteq E, Q_i \in [\mathcal{P}(E) \to [_q\mathcal{P}(E) \to 2]], D_j \in [\mathcal{P}(E) \to [\mathcal{P}(E) \to 2]], Q_j = D_j(Z_j)$ *and* $R, S \subseteq E^2$. *Then,* Choice$(Q_i, Q_j) \in [\mathcal{P}(E^2) \to [_q\mathcal{P}(E^2) \to 2]]$ *and*
(i) if $i = 2$, $j = 1$ *then,* Choice$(Q_i, Q_j)(R)(S) = 1$ *iff* $\exists W$ *such that* $Q_j(W) = 1$ *&*
$W \subseteq Z_j \,\&\, S = \{< \alpha, \beta > | \alpha \in W \,\&\, \beta \in Q_i(R)(\alpha)\}$
(ii) if $i = 1$, $j = 2$ *then,* Choice$(Q_i, Q_j)(R)(S) = 1$ *iff* $\exists W$ *such that* $Q_j(W) = 1$ *&*
$W \subseteq Z_j \,\&\, S = \{< \alpha, \beta > | \beta \in W \,\&\, \alpha \in Q_i(R)(\beta)\}$

The explanation of why choice questions are only possible with non-filter denoting declarative quantifiers is the result of the fact that if $|\{W | W \in Q_j \,\&\, W \subseteq Z_j\}| = 1$ then *Choice*(Q_i, Q_j) = *Pair-list*(Q_i, Q_j).

References

[1] Barwise, J. and Cooper, R.: 1981, Generalized quantifiers and natural language, *Linguistics and Philosophy* 4(1), 159-220.
[2] Benthem, J. v.: 1986, *Essays in Logical Semantics*, Reidel, Dordrecht.
[3] Benthem, J. v.: 1991, *Language in Action*, North-Holland, Amsterdam.
[4] Chierchia, G.: 1993, Questions with quantifiers, *Natural Language Semantics* 1, 181-234.

[5] Cooper, R.: 1983, *Quantification and Syntactic Theory*, Reidel, Dordrecht.

[6] Does, J. v. der: 1992, *Applied Quantifier Logics*, Ph.D. thesis, University of Amsterdam.

[7] Engdahl, E.: 1986, *Constituent Questions*, Reidel, Dordrecht.

[8] Fauconnier, G.: 1975, Polarity and the scale principle, in *Papers form the 11th Regional Meeting of the Chicago Linguistics Society*, 188-199.

[9] Gil, D.: 1995, Universal quantifiers and distributivity, in E. Bach et al.(eds.), *Quantification in Natural Languages*, Kluwer, Dordrecht.

[10] Groenendijk, J. & Stokhof, M.: 1984, *The Semantics of Questions and the Pragmatics of Answers*, Ph.D. thesis, University of Amsterdam.

[11] Groenendijk, J. & Stokhof, M.: 1989, Type-shifting rules and the semantics of interrogatives, in G. Chierchia et al. (eds.), *Properties, Types and Meaning, Vol 2: Semantic Issues*, Kluwer, Dordrecht.

[12] Heim, I.: 1987, Where does the definiteness restriction applies: evidence from the definiteness of variables, in E. Reuland et al. (eds.), *The Representation of (In)definiteness*,MIT Press, Cambridge (Mass.)

[13] Higginbotham, J.: 1993, Interrogatives, in K. Hale et al. (eds.), *The View from Building 20*, MIT Press. Cambridge (Mass.).

[14] Higginbotham, J. and May, R.: 1981, Questions, quantifiers and crossing, *The Linguistic Review* 1, 41-80.

[15] Jacobson, P.: 1995, On the quantificational force of English free relatives, in Bach, E. et al.(eds.) *Quantification in Natural Languages*, Kluwer, Dordrecht.

[16] Karttunen, L.: 1977, The syntax and semantics of questions, *Linguistics and Philosophy* 1, 3-44.

[17] Keenan, E.: 1987, A semantic definition of indefinite NP, in E. Reuland et al.(eds.), *The Representation of (In)definiteness*, MIT Press: Cambridge (Mass.)

[18] Keenan, E.: 1993, Natural language, sortal reducibility and generalized quantifiers, *Journal of Symbolic Logic* 58, 314-325.

[19] Keenan, E. and Faltz, L.: 1985, *Boolean Semantics for Natural Language*, Reidel, Dordrecht.

[20] Keenan, E. and Stavi, J.: 1986, A semantic characterization of natural language determiners, *Linguistics and Philosophy* 9, 253- 326.

[21] Keenan, E. and Westerståhl, D.: 1994, Generalized quantifiers in linguistics and logic, in Benthem, J. v. et al. (eds.), *Handbook of Logic and Linguistics*, Elsevier, Amsterdam.

[22] Ladusaw, W.: 1979, *Polarity Sensitivity as Inherent Scope Relations*, Ph.D. thesis, University of Texas at Austin.

[23] Lawler, J.: 1971, *Any* questions, *Papers from the 7th Regional Meeting of the Chicago Linguistic Society*, 163-173.

[24] Linebarger, M.: 1991, Negative polarity as linguistic evidence, *Papers from the 27th Regional Meeting of the Chicago Linguistic Society. Part Two: The Parasession on Negation*, 165-188.

[25] McConnell-Ginet, S.: 1982, Adverbs and logical form: a linguistically realistic theory, *Language* 58, 144-184.

[26] Nam, S.: 1995, *The Semantics of Locative Phrases*, Ph.D. thesis, UCLA.

[27] Srivastav, V.: 1992, Two types of universal terms in questions, *Nels* 22, 443-458.

[28] Szabolcsi, A.: 1994, Quantifiers in pair-list readings, and the non-uniformity of quantification, in Dekker, P. et al.(eds.), *Proceedings of the 9th Amsterdam Colloquium*. 645-664.

[29] Szabolcsi, A. and Zwarts, F.: 1993, Weak islands and an algebraic semantics for scope taking, *Natural Language Semantics* 1, 235- 284.

[30] Westerståhl, D.: 1985, Determiners and context sets, in Benthem, J. v. et al. (eds.),*Generalized Quantifiers in Natural Language*, Foris, Dordrecht.

Links without Locations

Information Packaging and Non-Monotone Anaphora

Herman Hendriks and Paul Dekker[1]

Utrecht University and University of Amsterdam

Abstract
In his work on information packaging—i.e., the structuring of propositional content in function of the speaker's assumptions about the hearer's information state—Vallduví (1992, 1993, 1994) identifies the informational primitives *focus*, *link* and *tail*, which are adapted from the traditional focus/ground and topic/comment approaches, and argues that the exploitation of information states of hearers by the information-packaging strategies of speakers reveals that these states have at least the internal structure of a system of Heimian file cards: links, which correspond to what are traditionally known as topics, say *where*–on what file card—the focal information goes, and tails indicate *how* it fits there. The present paper gives various reasons for not believing this and proposes to model information states as Kampian discourse representation structures, without locations. This requires and leads to a different perspective on the function of links. They signal non-monotone anaphora: their discourse referent Y is anaphoric to an antecedent discourse marker X such that $X \not\subseteq Y$. This idea will be shown to subsume 'non-identity' anaphora, contrastive stress, pronoun referent resolution, and restrictiveness of relatives and adjectives.

1 Information Packaging

The notion of information packaging is introduced in Chafe (1976):

> [The phenomena at issue] have to do primarily with how the message is sent and only secondarily with the message itself, just as the packaging of toothpaste can affect sales in partial independence of the quality of the tooth paste inside. (Chafe 1976: 28)

The basic idea is that speakers do not present information in an unstructured way, but that they provide a hearer with detailed instructions on how to manipulate and integrate this information according to their beliefs about the hearer's knowledge and attentional state:

> To ensure reasonably efficient communication, [... t]he speaker tries, to the best of his ability, to make the structure of his utterances congruent with his knowledge of the listener's mental world. (Clark and Haviland 1977: 5)

> On all levels the crucial factor appears to be the tailoring of an utterance by a sender to meet the particular assumed needs of the intended receiver. That is, information packaging in natural languge reflects the sender's hypotheses about the receiver's assumptions and beliefs and strategies. (Prince 1981: 224)

1. Sections 1 and 3 of the present paper have been written by the first author. Section 2 reflects joint work of the two authors.

For instance, sentences such as (1) and (2) are truth-conditionally equivalent in that they express the same proposition, but each of them 'packages' this proposition in a prosodically different way:[2]

(1) **The boss** *hates* BROCCOLI

(2) **The boss** HATES *broccoli*

Typically, speakers will use (1) if the hearer at the time of utterance knows nothing about or is not attending to the boss' relation to broccoli, while they will use (2) if the hearer at the time of utterance knows that there exists a relation between the boss and broccoli, is attending to this relation, but does not know what it is. Apparently, speakers are sensitive to such differences in the hearer's knowledge and attentional state, and hearers rely on this:

> speakers not using this device systematically give their listeners a harder time. (Nooteboom and Terken 1982: 317)

Truth-conditionally equivalent sentences that encode different information packaging instructions are not mutually interchangeable *salva felicitate* in a given context of utterance: e.g., of the above sentences, only the first one is a felicitous answer to the question *What does the boss hate?* It is this context-sensitivity that has traditionally placed information packaging within the realm of pragmatics.

Vallduví's account of information packaging (1992, 1993, 1994) is a combination of two influential earlier pragmatic approaches, the 'topic/comment' approach and the 'focus/ground' approach.

According to the focus/ground approach, sentences consist of a 'focus' and a 'ground'.[3] The focus is the informative part of the sentence, the part that (the speaker believes) makes some contribution to the hearer's mental state. The ground is the non-informative part of the sentence, the part that anchors the sentence to what is already established or under discussion in (the speaker's picture of) the hearer's mental state. Although sentences may lack a ground altogether, sentences without focus do not exist.

The topic/comment approach splits the set of subexpressions of a sentence into a 'topic', the—typically sentence-initial—part that expresses what the sentence is about, and a 'comment', the part that expresses what is said about the topic. Topics are points of departure for what the sentence conveys, they link it to previous discourse. Sentences may be topicless: so-called 'presentational' or 'news' sentences consist entirely of a comment.

In Reinhart (1982), it is argued that the dimension of 'old'/'new' information is irrelevant for the analysis of sentence topics. Instead, the notion of 'pragmatic aboutness' is is defined in terms of the organization of information. The set of Possible Pragmatic Assertions that can be made with a sentence S expressing proposition φ is defined as $\mathrm{PPA}_{(\mathrm{S})} = \{\varphi\} \cup \{\langle a, \varphi\rangle \mid a$ is the interpretation of an NP[4] in S$\}$ A pragmatic assertion $\langle a, \varphi\rangle$ is assumed to be *about* a.

Notice, by way of example (adopted from Dahl 1974), that the sentence **The boss** *hates* BROCCOLI gives rise to the parallel topic/comment and ground/focus

2. *Italics* are used for unaccented expressions; SMALL CAPS for expressions that bear a (focal) H* pitch accent; and **boldface** for expressions that bear a L+H* pitch accent. This is the terminology of Pierrehumbert (1980). H* accent and L+H* accent are called A accent and B accent, respectively, in Jackendoff (1972). We will assume that the relevant intonational unit for links is not the accent but rather the whole phrase, so that there is no such thing as a link-associated accent, but rather a link-associated tune.).
3. The ground is also known as 'presupposition' and as 'open proposition'.
4. Subject to further syntactic and semantic restrictions, cf. footnote 9 below.

partitions indicated in (3) if it answers the questions *What about the boss? What does he feel?*, whereas it induces the partitions specified by (4) in the interrogative context *What about the boss? What does he hate?*

(3)

topic	comment
The boss	*hates* BROCCOLI
ground	focus

(4)

topic	comment	
The boss	*hates*	BROCCOLI
ground		focus

The fact that the two informational articulations correspond to different partitions in (4) shows that neither of them is by itself capable of capturing all the informational distinctions present in the sentence. Therefore, Vallduví proposes to conflate the two traditional binomial articulations of focus/ground and topic/comment into a single trinomial and hierarchical one. The core distinction is the one between new information and anchoring, between focus and ground. In addition, the ground is further divided into the 'link', which corresponds approximately to the topic in the traditional topic/comment approach,[5] and the 'tail'.[6] In a picture:

(5) $\gg$

topic	comment		'aboutness'
link	tail	focus	
ground		focus	'old'/'new'

Given this articulation, the answer **The boss** *hates* BROCCOLI to the questions *What about the boss? What does he hate?* will receive the following analysis:

(6)

The boss	*hates*	BROCCOLI
link	tail	focus
ground		focus

Roughly speaking, the different parts—focus and ground, link and tail—of a sentence S have the following informational functions.

The focus encodes I_{S}, the *information* of S, which can be metaphorically described as ϕ_{S}, the proposition expressed by S, minus K_{h}, the information (the speaker presumes) already present in the hearer's information state.

The ground performs an *ushering* role—it specifies the way in which I_{S} fits in the hearer's information state: links indicate *where* I_{S} should go by denoting a location in the hearer's information state, and tails indicate *how* I_{S} fits there by signaling a certain mode of information update. Of course, talking about ushering information to some location in the hearer's information state presupposes that this information state has some sort of internal structure. In this respect, Vallduví purports to

5. To the extent that links correspond to the *topic* in the traditional topic/comment distinction, Vallduví's theory is quite similar to the analysis of sentence topics presented in Reinhart (1982), where a pragmatic assertion of φ about a is formalized as $\langle a, \varphi\rangle$, in that a functions as a kind of 'locus of update' for φ (cf. below). A difference is that Reinhart allows assertions without a topic (since also $\varphi \in \mathrm{PPA}_{(\mathrm{S})}$) and topics that express new information.
6. The hierarchy does not imply constituency or (even) continuity. In particular, the two parts (link and tail) of the ground may not constitute a linear unit at the surface. Moreover, sentences may have more than one link, and more than one element may constitute the tail.

> agree with Heim that there has to be some additional internal structure in the hearer's model of the common ground that plays an important role in natural language interpretation, even if this internal structure is of tangential relevance in truth value computation. It is this internal structure of information states which is, in fact, crucially exploited by the different information-packaging strategies used by speakers in pursuing communicative efficiency. (Vallduví 1994: 7)

In fact, Vallduví takes the metaphor of Heim's file change semantics (1982, 1983) literally, in that he assumes that the information in the hearer's model is organized in files, i.e., collections of file cards. Each file card represents a discourse entity: its attributes and its links with other discourse entities are recorded on the card in the form of conditions. Such a discourse entity may be known to the hearer but not salient at the time of utterance, it may be salient at the time of utterance, it may be completely new to the hearer, it may be inferable from what the hearer knows, etc. Discourse entities mediate between referring expressions (noun phrases) and entities in the real world: indefinite noun phrases prompt hearers to create a new file card, and definite noun phrases incite them to retrieve an already existing file card. Both definites and pronouns denote already existing file cards, but pronouns denote salient file cards, whereas (other) definites refer to non-salient ones.

File change comprises the above aspects of file card management, but it also involves content update, i.e., the incorporation of information conveyed by a given sentence into records on novel and familiar file cards, and this is where Vallduví lets information packaging come in.

Links are associated with so-called GOTO instructions. In file change semantics, the target location of such a declaration is a file card *fc*. A tail points at an information record—normally a (possibly underspecified) condition—on such a file card, RECORD(*fc*), and indicates that it has to be *modified* (or further specified) by the focus information I_S of the sentence. The associated instruction type is called UPDATE-REPLACE. In the absence of a tail, the focus information I_S of a sentence is simply *added* at the current location. The associated instruction type is called UPDATE-ADD.

Sentences may lack links and tails (recall that the focus is the only non-optional part of a sentence), so the following four sentence types can be distinguished:

(7) *a.* link-focus
b. focus
c. focus-tail
d. link-focus-tail

The above sentence types are associated with the below (compound) instruction types, respectively:

(8) *a.* GOTO(*fc*)(UPDATE-ADD(I_S))
b. UPDATE-ADD(I_S)
c. UPDATE-REPLACE(I_S,RECORD(*fc*))
d. GOTO(*fc*)(UPDATE-REPLACE(I_S,RECORD(*fc*)))

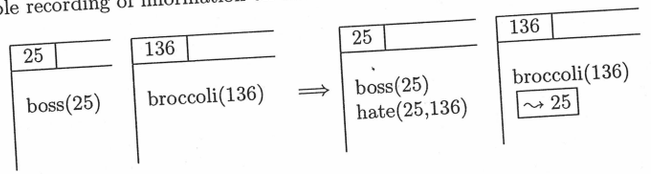
The sentence and instruction types in (7) and (8) can be illustrated with the following examples, where links, tails and foci are specified by means of [L...], [T...] and [F...] brackets, respectively, and accented expressions in foci and links are—as above—written in small caps (representing H* pitch accent) and boldface (for L+H* pitch accent), respectively:

(9) *a.* link-focus: [L**The boss**][F*hates* BROCCOLI]
GOTO(*fc*)(UPDATE-ADD(I_S))
b. focus: [F*He always eats* BEANS]
UPDATE-ADD(I_S)
c focus-tail: [F*He is* NOT][T*dead*]
UPDATE-REPLACE(I_S,RECORD(*fc*))
d. link-focus-tail: [L**The boss**][FHATES][T*broccoli*]
GOTO(*fc*)(UPDATE-REPLACE(I_S,RECORD(*fc*)))

As regards the first example, suppose that a newly appointed temp is ordering dinner for the boss and asks the executive secretary whether there is anything that should known about the boss' taste. The executive secretary gives the following answer:

(10) [L**The boss**][F*hates* BROCCOLI]

Example (10) is a link-focus construction, and as such it is associated with a GOTO(*fc*)(UPDATE-ADD(I_S)) instruction. The link subject *the boss* specifies a locus of update *fc*, viz., the card representing the boss—card #25, say. The focus verb phrase *hates broccoli* specifies the information I_S that has to be added to this card. Suppose that broccoli is represented by card #136. Then, passng over some formal details, the UPDATE-ADD(I_S) instruction associated with the focus *hates broccoli* amounts to adding the condition 'hates(25,136)' to the locus of update, i.e., the boss' card #25. Moreover, the record '$\boxed{\rightsquigarrow 25}$', a pointer to the locus of update, is added to card #136, rendering the condition 'hates(25,136)' on card #25 'accessible' from card #136: Vallduví says that this linking mechanism, which designates a unique location for content update, is 'much more efficient' than straightforward multiple recording of information on cards.



(11) [F*He always eats* BEANS]

Example (11), an all-focus construction, is simply associated with an UPDATE-ADD(I_S) instruction. Here, this instruction involves the addition of the focus information I_S that the value of the current card always eats beans. That is: if it is interpreted immediately after example (10) and if we leave its adverbially modified transitive verb phrase unanalyzed for simplicity, it amounts to adding the condition 'always eats beans(25)' to card #25.

The presence of a tail in a sentence signals a mode of update different from the straightforward UPDATE-ADD(I_S) instruction. A tail indicates that a (possibly underspecified) record on a file card has to be replaced (or specified further). The material in the tail serves the purpose of determining *which* record. Suppose, for example, that (12) is a reaction to the statement *Since John is dead, we can now split his inheritance*:

(12) *I hate to spoil the fun, but* [F*he is* NOT][T*dead*]

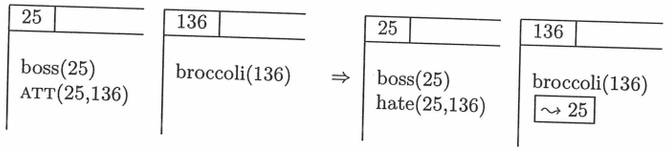
With this focus-tail example, the speaker instructs the hearer to replace the record on the current locus of update—card #17, say, for John—expressing that the value

of card #17 is dead by one saying that he is not dead. In short, the tail serves to highlight a condition on file card #17, the one saying its value is dead. This condition is then modified in the way specified by the material in the focus.

In addition to the option of replacing a record on a file card, there is the possibility of further specifying an underspecified record, something which is assumed to be going on in the link-focus-tail example (13) given below. Suppose now that the newly appointed temp asks the executive secretary whether it was a good idea to order broccoli for the boss, and that the executive secretary gives the following answer:

(13) [L **The boss**][FHATES][T*broccoli*]

The idea is that the temp has an underspecified record on his card for the boss, which says that the boss has some attitude towards broccoli. The lack of information about the nature of this attitude is reflected by the record 'ATT', and it is this record which is replaced by 'hate' after hearing the executive secretary's answer (13):



Different languages choose different structural means to spell out the same informational interpretations. Vallduví studies the manifestation of information packaging in several languages, with an emphasis on Catalan and English. Cross-language comparison shows that in expressing information packaging, languages exploit word order and prosody in various ways. Roughly speaking, English structurally realizes information packaging by means of alternative intonational contours of identical strings, whereas Catalan has a constant prosodic structure and effectuates information packaging by means of string order permutations. In fact, Vallduví argues that languages such as Catalan supply empirical support for the representation of information packaging sketched above, since these languages package their information in a much more salient way than, for example, English. Thus, while informational interpretations may be expressed exclusively by prosodic means in English, information packaging instructions in Catalan are straightforwardly reflected in syntax.

In English, the focus is associated with a H* pitch accent (written in small caps), links are marked by a L+H* pitch accent (written in boldface), and tails are structurally characterized by being deaccented. One and the same string may be assigned different intonational phrasings in order to realize different informational interpretations. In particular, the focal pitch accent may be realized on different positions in the sentence. This is illustrated by the sentences (15), (17) and (19), construed as answers to the questions (14), (16) and (18), respectively:

(14) *What did you find out about the company?*

(15) [F*The boss hates* BROCCOLI]

(16) *What did you find out about the boss?*

(17) [L**The boss**][F*hates* BROCCOLI]

(18) *What does the boss feel about broccoli?*

(19) [L**The boss**][FHATES][T*broccoli*]

In Catalan, the situation is as follows. Metaphorically speaking, one can say that Catalan focal elements remain within a so-called 'core clause', but that ground elements are 'detached' to a clause-peripheral position. In particular, links are detached to the left, and non-link ground elements undergo right-detachment. As a result of detaching both links and tails, the core clause (CC) is left containing only the focus of the sentence:

(20) LINKS [CC FOCUS] TAILS

Consider the Catalan counterparts (21), (22) and (23) of (15), (17) and (19), respectively. The all-focus sentence (21) displays the basic verb-object-subject word order. In (22) and (23), the link subject *l'amo* has been detached to the left. In (23), moreover, the tail direct object *el bròquil* has been detached to the right, leaving a clitic (*l'*) in the focal core clause. Note that intonational structure plays a part in Catalan too, albeit 'a rather lame one' (Vallduví 1993: 33): a focal H* pitch acent is invariably associated with the last item of the core clause.

(21) [F*Odia el bròquil* L'AMO]

(22) [L*L'amo*][F*odia el* BRÒQUIL]

(23) [L*L'amo*] [FL'ODIA][T*el bróquil*]

The above observations provide confirmation that information packaging involves syntax as well as prosody; hence any attempt to reduce information packaging to either syntax (for Turkish, cf. Hoffman 1995) or prosody (for English, cf. Steedman 1991, 1992, 1993) is inadequate from a cross-linguistic point of view.[7] Accordingly, Hendriks (*draft*) treats the range of variation in the structural realization of information packaging as displayed by Catalan and English by means of the sign-based categorial grammar formalism of Hendriks (1994). Basically, this formalism is a both intonationally/syntactically and semantically/informationally interpreted version of a double 'dependency' variant (see Moortgat and Morrill 1991) of the non-associative Lambek (1961) calculus, enriched with the unary operators of Moortgat (1994). The treatment of information packaging it accommodates differs from many of its predecessors (including other extensions of standard Lambek calculus such as Oehrle 1991, Van der Linden 1991, and Moortgat 1993), in that it does not employ focusing operators, but, instead, makes use of 'defocusing' operators that license the presence of links and tails.

Acccording to most approaches, focused constituents are semantic functors which take the non-focused part of the sentence as their argument. This analysis is based on such assumptions as made in Szabolcsi (1981, 1983) and Svoboda and Materna (1987), where focus is not only considered an information-packaging primitive but also an implicit truth-conditional exhaustiveness operator, and on semantic studies of the phenomenon of 'association with focus' as provided by Jacobs (1983), Rooth (1985), Krifka (1991), and others who have argued that the quantificational structure of so-called focus-sensitive operators is crucially determined by the traditional pragmatic focus-ground partition. However, Vallduví argues convincingly that 'the claim that focused constituents truth-conditionally entail exhaustiveness leads to extreme positions' (1992: 170), and Vallduví and Zacharski (1993) show that 'association with pragmatic focus' is not an inherent semantic property of

7. Note, moreover, that the structural realization of information packaging in Catalan involves *both* syntax *and* prosody.

'focus-sensitive' operators, which may express their semantics on partitions other than the focus-ground one—witness obvious cases of association with subsegments of the informational focus, with links, and with other parts of the ground.

This dissociation of the pragmatic focus-background distinction from issues of exhaustiveness and focus-sensitivity dispels the need of analyzing focused constituents as operators which semantically take scope over the non-focused parts of the sentence, which can be considered an advantage. As sentences may lack links and tails, such analyses do not immediately reflect the core status of the focus, which is the only non-optional part of a sentence. In some sense, then, all-focus sentences constitute the basic case, and the cases where there is a ground are derived from such basic all-focus structures.

2 Files in Focus

Vallduví has it that

> [...] a proper understanding of information packaging, i.e., of the actual strategies used by human agents in effecting information update by linguistic means, will help us gain further insight into the structural properties of the cognitive states these dynamic strategies manipulate. (Vallduví 1994: 24)

As we have seen, the basic idea of information packaging is that in discourse, speakers not only present information to their interlocutors, but also provide them with detailed 'instructions' on how to manipulate and integrate this information. With respect to the role of these instructions in the determination of those aspects of the structure of information states which are relevent to natural language interpretation, Vallduví claims the following:

> The use of these instructions reveals that speakers treat information states as highly structured objects and exploit their structure to make information update more efficient for their hearers. (Vallduví 1994: 3)

More specifically, concerning 'the internal structure of information states which is, in fact, crucially exploited by the different information-packaging strategies used by speakers in pursuing communicative efficiency' (1994: 7), it is argued that information packaging instructions contribute in two ways to the optimization of information update, since they provide means to

- designate a file card as the locus of information update and hence circumvent the redundancy of multiple update; and
- identify the information of the sentence and its relation to information already present in the hearer's model.

(Recall that the information of the sentence, I_S, is expressed by the focus, and that the ground has an ushering role with respect to I_S: links indicate where I_S goes, and tails indicate how it fits there.) So, summing up, Vallduví concludes that information states constitute systems that have at least the internal structure of a collection of file cards connected by pointers.

Though the presented arguments may appear to be intuitively quite appealing, it can be argued that, strictly speaking, they are not as compelling as they seem. Somehow, Vallduví is begging the question: 'talking about ushering I_S to a location in the hearer's model K_h [...] does not make much sense unless one assumes some

sort of rich internal structure for K_h' (Vallduví 1994: 7). The relevant question, however, is whether this assumption of 'some sort of rich internal structure' itself makes sense of anything besides the ushering function of links.

If file card systems are assumed, *then* the information-packaging instruction types apparently do contribute to efficient information exchange. And if this assumption is warranted, it may even serve as an explanation of the fact that we do appear to find these ways of packaging information in a variety of languages. Nevertheless, the more theoretical question is whether this assumption itself is warranted, and whether the organization of linguistic information exchange really presupposes such information states. After all, ushers can be very useful, but there are also halls that have unnumbered seats!

Maybe links really make no sense without files, but, for that matter, maybe we simply fail to understand what links do. The notion of 'ushering I_S to a location' may be just as metaphorical as the notion of 'file card collection'. For instance, files are, as Vallduví puts it, 'dimensionally richer' than the card-less discourse representation structures of Discourse Representation Theory (see Kamp 1981, Kamp and Reyle 1993), since each file card introduces its own 'representational space' where all its records are to be found while there is no sensible notion of location in discourse representation structures. Still, a hearer who employs discourse representation structures has an easier job from a bookkeeping perspective than a hearer whose information states are collections of file cards connected by pointers.

This can be illustrated as follows. Imagine an utterance made by Irene, a speaker who organizes her utterances on the basis of the assumption that her audience stores information using collections of file cards connected by pointers, to Hans, a hearer who employs discourse representation structures. Clearly, it would be inappropiate to say that Irene uses links to usher I_S to a location in the hearer's model K_S, since there is no sensible notion of location in Hans's discourse representation structures. Still, this does not at all preclude Hans from updating his discourse representation with the proposition that Irene attempts to get through. And worse, he has even got considerably less to do than a hearer who uses collections of file cards connected by pointers. Compare the following link-focus example:

(24) $[_L\textbf{Frank}_5][_F\text{flew from Amsterdam}_9\text{ to Oslo}_8\text{ via STUTTGART}_2]$

Neglecting various details, if a file clerk is to update her file in order to represent the information expressed by example (24) in the way sketched above, she has to carry out the following sequence of instructions:[8]

(25) GOTO(5)(UPDATE-ADD(flew(5,9,8,2)))
GOTO(9)(UPDATE-ADD(⌈⇝ 5⌉))
GOTO(8)(UPDATE-ADD(⌈⇝ 5⌉))
GOTO(2)(UPDATE-ADD(⌈⇝ 5⌉))
GOTO(5)

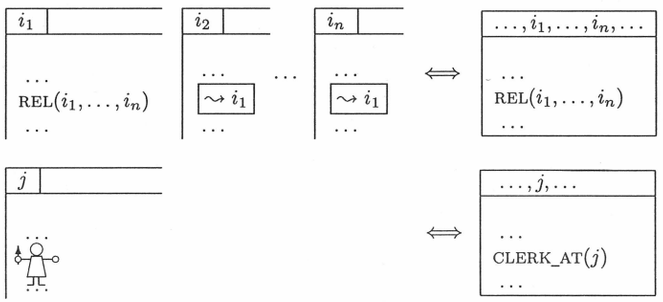
8. Assuming that establishing links to the locus of update is done via packaging instructions—of course, these links have to be established *some*how. Note, by the way, that the file clerk's task would not be made easier by structure sharing (something suggested by Enric Vallduví (personal communication)), because also the structure sharing will itself have to be established somehow—in the following way, for example:
GOTO(5)(UPDATE-ADD(⌈1⌉flew(5,9,8,2)))
GOTO(9)(UPDATE-ADD(⌈1⌉))
GOTO(8)(UPDATE-ADD(⌈1⌉))
GOTO(2)(UPDATE-ADD(⌈1⌉))
GOTO(5)

Hans, on the other hand, only has to carry out the following instruction:

(26) UPDATE-ADD(flew(5,9,8,2))

This example may serve as an indication that none of the data discussed above precludes the use of, say, Kampian discourse representation structures instead of Heimian files. Clearly, there may be evidence for assuming there to be files at work, and one of the last things we would like to claim is that people organize their information in simpler systems than collections of file cards (or discourse representation structures, for that matter). On the contrary. The only point is that the use of files does not appear to be imperative so far.

Notice that Vallduví's conclusion is, in some sense, unfalsifiable. Discourse representation structures can model precisely the same information as file card systems, except for one small difference. The only thing that discourse representation structures lack is a marked discourse referent corresponding to the file notion of 'current locus of update', i.e., the location where file clerk happens to find herself. If we assume that discourse representation structures have a way of marking such a discourse referent j—by a condition 'CLERK_AT(j)', say—, then the two systems differ only in the way in which they display their information: in one big box, or on several cards connected by pointers. But, moreover, one can show that given Vallduví's specific use of pointers to file cards, there is actually a bijective correspondence between his files and the class of discourse representation structures with atomic conditions and one marked discourse referent for the current locus of update. For note that conditions 'REL($i_1, \ldots, i_n$)' are invariably added on card i_1, inducing pointers '$\boxed{\rightsquigarrow i_1}$' on the cards $i_2, \ldots, i_n$. Hence the following correspondence can be established:



The idea that links specify a locus of update in information states that are collections of file cards connected by pointers is problematic for various reasons. First, it is unclear what locus of update must be associated with quantified, negative and disjunctive links, or—more in general—where and how quantified, negative and disjunctive information has to be put. Second, the existence of sentences with more than one link is enigmatic. Third, the replacement operation triggered by the presence of tails is complicated by the use of file cards. And fourth, the approach leads to the counterintuitive conclusion that pronouns form part of the focus. These issues will be addressed in the remainder of this section.

(*a*) Vallduví observes that files are 'dimensionally richer' than the discourse representation structures (DRSs) of Discourse Representation Theory. Now, this is true to the extent that each file card introduces its own 'representational space' where all records concerning that file card are to be found. In order to be actually richer, nonetheless, files must be adapted to model more than merely atomic

conditions—i.e., individuals having properties and standing in relations at various spatio-temporal locations. Among other things, they should be able to model quantified, negative and disjunctive information. Discourse Representation Theory allows the construction of complex conditions from sub-DRSs, and these conditions—by an appropriate semantic interpretation procedure—model precisely such information. Heim, who explicitly speaks of files and file cards as metaphors (1982: 276 and 302ff.), spells out quantified, negative and disjunctive information in purely semantic terms, i.e., in terms of the domains and satisfaction sets of files. However, it is not clear how such information must be expressed in the non-metaphorical file card set-up of Vallduví (1994).

For one thing, what loci of update are specified by the links of sentences such as (27), (28) and (29)?[9] On what file card(s)—if any—should the information expressed by these sentences be put?

(27) [$_{\text{L}}$**Every man**][$_{\text{F}}$WALKS]

(28) [$_{\text{L}}$**No man**][$_{\text{F}}$WALKS]

(29) [$_{\text{L}}$**John or Mary**][$_{\text{F}}$WALKS]

For another, how should this information be put? One might think of using sub-files, but then, where must these be put? Are they attached to a main file, or must they be attached to a main file's file card? Which one? Interestingly, Heim raises similar questions in her 1983 paper:

> Take a simple sentence [...]: *It is raining*. In the context of the file metaphor, one doesn't quite know how to deal with this sentence. As an informative sentence, it ought to call for an updating of the file somehow: but what exactly is the file clerk supposed to do? The information that it is raining does not belong on any particular file card, it seems, since each file card is a description of an individual, but *It is raining* is not about any individual. Should the file clerk perhaps write on some arbitrary card: 'is such that it is raining'? Or should he write that on all cards? And what if the file so far doesn't contain any cards yet? [...] Quantified and negated propositions are similarly puzzling if we are so ambitious as to want to say what exactly the file clerk does in response to them. Under the modest aspect of domain and satisfaction set change, however, they pose no problem. (Heim 1983: 183–184)

It should be noted here that such a 'modest' position cannot be retained in the set-up of Vallduví (1994), because there the entities to be updated must be *files*, *not* their domains and satisfaction sets.

(*b*) Vallduví (1992: 104) notes that there is no structural restriction on the number of links in Catalan. 'Sentences may have more than one link, as in the Catalan example (30).

9. Though 'links tend to be definite NPs' (1992: 77), Vallduví notes the 'restricted existence of indefinite links' (1992: 46). 'Sentences with *quantifier links* are' claimed to be 'less natural than others, causing raised eyebrows among some Catalan speakers. Sentences like *A tots els estudiants*$_i$ *els*$_i$ *donen un* CARNET t_i "To all students they give an ID" or *A tothom*$_i$ *no el*$_i$ *tracten* t_i IGUAL "Everybody they don't treat the same" are extremely natural, some other sentences sound odder. Most sentences, however, are felicitous once the right context is construed, although in some cases it may require some sophistication' (Vallduví 1992: 153). Analogously, Reinhart notes that if they 'can be interpreted (pragmatically) as denoting sets, universally quantified NPs, as well as specific and generic indefinite NPs, can serve as topics' (1982: 65–66).

(30) [L*El bròquil*] [L*a l'amo*] [F*l'hi van* REGALAR]
the broccoli to the boss *obj-iobj 3p-past* give
Approx.: 'The broccoli the boss (they) gave it to him (for free)'.

In these cases the speaker directs the hearer to go to two addresses and enter the information under both.' (Vallduví 1992: 60). So, assuming that 'they' have card #3 and that the boss and broccoli still possess their respective cards #25 and #136, this means that the sentence is *not* associated with the instruction (31),[10] but with an instruction along the lines of (32).

(31) *GOTO(136)(GOTO(25)(UPDATE-ADD(give(3,136,25)))

(32) GOTO(136)(UPDATE-ADD(give(3,136,25))
GOTO(25)(UPDATE-ADD(give(3,136,25))

But this raises questions. What is the current locus of update after (32) has been carried out? Is the file clerk suddenly simultaneously present on two file cards? If she isn't (suppose she lands on card #25), does this then mean that (32) is equivalant to (33), the instruction associated with the one-link sentence (34)?

(33) GOTO(25)(UPDATE-ADD(give(3,136,25))
GOTO(136)(UPDATE-ADD(⊡))
GOTO(25)

(34) [L*A l'amo*] [F*hi van regalar el* BRÒQUIL]]

But if (32) and (33) are equivalent, then why does Catalan allow multiple links at all? And how could (32) and (33) be non-equivalent—what sense could multiple loci of update make that pointers cannot?

(*c*) Above we gave an informal sketch of Vallduví's analysis of tail-containing sentences in terms of UPDATE-REPLACE instructions. It can be expected that various complications will arise when it comes to giving an explicit formalization of the replacement instructions associated with tails. Any attempt at giving an appropriate and fully general definition of these instructions will have to confront a number of questions. Thus, how exactly do you know which record has to be replaced or specified further? Is there guaranteed to be such a record? Is there a unique one, and what happens if there are more? Is it always one record that has to be replaced, or do we sometimes need to replace a group of records? What kind of match must there be between the material in a tail, and the material in the target record? Of course, these are tough nuts that have to be cracked when it comes to coming to theories of belief revision.

Here we will just present an example which illustrates that the replacement operation triggered by the presence of tails is specifically complicated by the idea that information is organized in file card systems. Suppose that Louis van Gaal utters (35), whereupon Johan Cruijff reacts with saying (36):

(35) [L**Ajax**][FWON]

(36) [F*No,* BARCELONA][T*won*]

Assume the file cards #1 and #2 for Ajax and Barcelona, respectively. Now, clearly, Johan Cruiff here instructs Louis van Gaal to replace his record according to which

10. Note, by the way, that the 'GOTO(3)' constitutes a superfluous detour in instruction (31).

Ajax won by one according to which Barcelona did. Presumably, this should not (only) be done on the card for Ajax. Instead of the straightforwardly simple (37), we seem to need the complex instruction given in (38).

(37) *UPDATE-REPLACE(won(2),won(1))

(38) UPDATE-REPLACE(,won(1))
GOTO(2)(UPDATE-ADD(won(2)))

(*d*) A typical example of the way in which Vallduví analyzes pronouns can be obtained by combining the above example sentences (10) and (11) into one text:

(39) [L**The boss**][F*hates* BROCCOLI]
[F*He always eats* BEANS]

The first sentence is a link-focus construction, and therefore associated with an instruction to go to the file card of the boss, thereby turning it into the current locus of update, and to enrich that file card with the information that the boss hates broccoli (and the broccoli file card with a pointer to the file card of the boss). The second sentence is an all-focus construction, associated with the simple instruction to add the focus information that the value of the current locus of update always eats beans to the current locus of update. Hence if it is interpreted immediately after the first sentence, it amounts to adding the information that the boss always eats beans to the card of the boss.

Note that the pronoun *he* obviously does not induce replacement or shift the locus of update. Hence it cannot be a link or a tail, and this inevitably leads to the conclusion that it forms part of the focus. This is a counterintuitive result, however, since it is also clear that the interpretation of the pronoun is provided by the value of the current locus of update—which does not constitute new information, but can be assumed to be already present in the hearer's information state.

3 Non-Monotone Anaphora

Let us wind up the discussion so far. We have argued that the data discussed above do not enforce the conclusion that information states have at least the structure of a collection of file cards connected by pointers. For that matter, the phenomena can also be accounted for in terms of discourse representation structures, and it is very well possible that circumventing file cards might lead to the avoidance of the complications that were outlined in the previous section.

In view of these considerations, a card-less alternative will be defended in the present section, according to which information states are modelled by means of discourse representation structures, which are ontologically less committed than the 'dimensionally richer' file card system, in that discourse representation structures do not come with locations.

But if, as we have argued, the use of files does not appear to be imperative, then we face a question: what purpose *do* links serve if they do not serve to specify a locus of update by ushering to locations? What does 'ushering to a location' mean if representations do not come with locations? Thus a different perspective on the function of links is required. We would like to suggest a tentative answer which we take to carry less presuppositions than the file metaphor.

The perspective we would like to offer has its heuristic starting point in Kamp and Reyle (1993), who note that processing a plural pronoun does not always involve equating the discourse referent it introduces with one introduced earlier through the processing of some other plural NP. They consider the following example:

(40) *John took Mary to Acapulco. They had a lousy time.*

Here, the plural pronoun *they* does not have a single NP for its antecedent. Rather, the 'antecedent' has to be 'constructed' out of various parts of the preceding text. Such examples, which are very common, seem to suggest that plural pronouns can pick up any antecedent that can be obtained from antecedent information by logical deduction. However, the deductive principles that are permitted in this context turn out to be subject to restrictions.

(41) *Eight of the ten balls are in the bag. They are under the sofa.*

The pronoun *they* in (41) cannot be understood as referring to the two balls that are missing from the bag. Apparently, subtracting one set from another is not a permissible operation for the formation of pronominal antecedents.

The permissible process of antecedent formation displayed by (40) is called Summation: a new discourse referent is introduced which represents the 'union' of individuals (John and Mary) and/or sets represented by discourse referents that are already part of the discourse representation structure. Other permissible processes are Abstraction, exemplified by (42), which allows the introduction of discourse referents for quantified NPs (compare also footnote 9 above), and Kind Introduction, which introduces discourse referents for a certain 'genus' explicitly mentioned in the text by a (simple or complex) noun.If *they* in (43) refers to the (few) men who joined the (conservative) party, we are dealing with Abstraction. The more natural reading, where *they* refers to men in general (and the party is presumably non-conservative), is a case of Kind Introduction.

(42) *I found every book Bill needs. They are on his desk.*

(43) *Few men joined the party. They are very conservative.*

In their discussion of the inferential processes available for the construction of antecedents for (plural) pronouns, Kamp and Reyle suggest the following generalization:

> What sets the admissible inference processes of Summation, Abstraction and Kind Introduction apart from an inadmissible inference pattern such as set subtraction is that the former are [...] strictly *positive* (Kamp and Reyle 1993: 344),

or

> 'cumulative' in the following sense: the newly created discourse referent represents an entity of which the discourse referents used in the application of the rule represent (atomic or non-atomic) parts. (Kamp and Reyle 1993: 394)

Notice that, when this generalization is taken in conjunction with a principle that anaphora invariably involves the addition of an equational condition 'X = Y' for an anaphoric expression with discourse referent Y and a—possibly inferentially created—antecedent discourse referent X (and such an equational approach is standard practice in Dicourse Representation Theory), the necessary result will be that anaphora is always (upward) monotone: if an expression with discourse referent Y is anaphorically dependent on an expression with discourse referent X, then $X \subseteq Y$.

The latter result, however, does not seem to be borne out by the facts. For example, Van Deemter (1992, 1994a) presents cases of 'non-identity anaphora' along the lines of (44), as well as minimal pairs such as (45) and (46):

(44) *Our neigbours are extremely nice* PEOPLE.
He *is a* TEACHER, **she** *is a* HOUSEWIFE.

(45) *John fed the* ANIMALS. *The cats were* HUNGRY.

(46) *John fed the* ANIMALS. *The* **cats** *were* HUNGRY.

It can be observed that the pronouns **he** an **she** are anaphorically dependent on *our neighbours* in (44), but that the discourse referents of the pronouns represent entities which are proper subsets of the entity represented by the discourse referent of the antecedent: obvious cases of non-monotone anaphora.

Moreover, whereas the reading of (45) where *the cats* is anaphoric to *the* ANIMALS strongly and monotonously suggests that all animals fed by John were cats, the reading of (46) where *the* **cats** is anaphoric to *the* ANIMALS does not. It even seems to imply that John fed at least one non-cat.[11] Again, we are dealing with non-monotone anaphora.

Note that the texts (45) and (46) differ only in the assignment of L+H* accent to the noun phrase *the cats*, which is the distinguishing mark of links in English. Hence our alternative hypothesis concerning links:

(47) *Non-Monotone Anaphora Hypothesis* (*NAH*):
Linkhood (marked by L+H* accent in English) serves to signal non-monotone anaphora. If an expression is a link, then its discourse referent Y is anaphoric to an antecedent discourse referent X such that $X \not\subseteq Y$.

As we will show, this hypothesis affects a range of phenomena. It subsumes not only the so-called 'non-identity' anaphora just exemplified and analyzed in Van Deemter (1992, 1994a), but also the cases of contrastive stress discussed in Rooth (1992) and Vallduví (1992, 1994). It contributes to an explanation of the effect of pitch accenting on pronoun referent resolution noted in Cahn (1995), Kameyama (1994), Vallduví (1994), among many others, and it sheds light on the distinction between restrictive and non-restrictive relative clauses and adjectives (see Kamp and Reyle 1993).

(*a*) The relationship between non-identity anaphora and linkhood can be demonstrated even more saliently with relational nouns:

(48) *Ten guys were playing basketball in the* RAIN.
The fathers were having FUN.

(49) *Ten guys were playing basketball in the* RAIN.
The fathers *were having* FUN.

Thus, while (48) has an 'identity' reading where *the fathers* is anaphoric to *ten guys* which suggests that all ten guys playing basketball in the rain were fathers who were having fun, and (49) has a 'subsectional' reading where **the fathers** is anaphoric to *ten guys* which suggests that the fathers who were having fun constitute a proper subset of the ten basketball-playing guys, the latter text also has a—non-monotone—'relational' reading where the fathers *of* the ten guys playing basketball in the rain were having fun.

11. 'Strongly suggests' and 'seems to imply' instead of 'entails', since though the effects are quite strong, they are of a pragmatic, rather than a logico-semantic, nature. See also (*c*), on pronoun referent resolution, below.

Observe, by the way, that Kamp and Reyle's example (40) of Summation, a case of monotone non-identity anaphora in which the pronoun *they* typically apears unaccented, shows that is not so much the 'non-identity' as the 'non-monotonicity' of the anaphora which is responsible for the L+H* accent (that is: the linkhood) of the anaphor.

(*b*) According to Rooth, contrast is the cornerstone of the interpretation of focus phenomena: 'Intonational focus has a semantic import related to the intuitive notion of contrast within a set of alternative elements' (1992: 113), and Vallduví gives the following example of 'contrastive' links (1993:14):

(50) *Where can I find the cutlery?*
The forks *are in the* CUPBOARD, *but*
the knives *I left in the* DRAWER.

However, note that contrast is not really necessary.[12] Mere non-monotonicity is sufficient for L+H* accent:

(51) *Where can I find the cutlery?*
The forks *are in the* CUPBOARD.

(*c*) Many authors have paid attention to the effect of pitch accenting on pronoun referent resolution. The examples below stem from Lakoff (1971).

(52) *Paul called Jim a Republican. Then he insulted him.*

(53) *Paul called Jim a Republican. Then* **he** *insulted* **him**.

For grammatical reasons (parallellism), the preferred antecedents for the unstressed pronouns *he* and *him* in (52) are *Paul* and *Jim*, respectively. The preferences are reverse for the stressed pronouns **he** and **him** in (53).[13]

In the theory of Kameyama (1994), this phenomenon is accounted for in the following way:

- A grammar subsystem represents the space of possibilities and a pragmatics subsystem represents the space of preferences;
- Stressed and unstressed pronouns have the same denotational range—the same range of *possible* values;
- *Complementary Preference Hypothesis* (*CPH*): A stressed pronoun takes the complementary preference of the unstressed counterpart.

However, the *NAH* formulated in (47) actually *predicts* the *CPH* effects: adding L+H* accent to pronouns means the addition of a pragmatic signal that the anaphora involved is non-monotone. In the case of singular antecedents with entity-representing discourse referents, this means that the anaphor does not *not* corefer with its antecedent. As a consequence, pronominal stress turns the grammatically determined preference for a certain antecedent into a pragmatic preference for non-reference with that antecedent.

12. Nor is contrariety (as proposed in Van Deemter 1994b), witness:

Where can I find the cutlery?
The forks *are in the* CUPBOARD, *and* **the knives** TOO.

13. The fact that (53) insinuates that calling someone a Republican is an insult is essentially due to the de-accenting of *insulted* in the second sentence of (53).

(*d*) The sentences (54) and (55) (taken from Kamp and Reyle 1993: 255) illustrate the familiar rule of English orthography that non-restrictive clauses are set apart from the surrounding text by commas, but that restrictive clauses are not.

(54) *The son who attended a boarding school was insufferable.*

(55) *The son, who attended a boarding school, was insufferable.*

Note that (54), in which the relative clause is used restrictively, suggests that there is more than one son, but only one who is boarding. In (55), where the relative clause is used non-restrictively, the suggestion is rather that there is only one son, of whom it is said not only that he was insufferable but also, parentheticlly as it were, that he attended a boarding school. If the prosody of these sentences is taken into account, it will be clear that this pragmatic difference is in keeping with the *NAH* as formulated in (47). Similar observations can be made with respect to the (non-)restrictiveness of the adjectives and nouns in (58) (Kamp and Reyle 1993: 372).

(56) **The son who attended a boarding school** *was* INSUFFERABLE.

(57) **The son**, *who attended a* BOARDING SCHOOL, *was* INSUFFERABLE.

(58) *John fed the* ANIMALS.
The young cats were HUNGRY.
The **young cats** *were* HUNGRY.
The **young** *cats were* HUNGRY.
The young **cats** *were* HUNGRY.

References

[1] Bosch, P., and R. van der Sandt (eds.) (1994). *Focus and Natural Language Processing. Proceedings of a Conference in Celebration of the 10th Anniversary of the Journal of Semantics*. Working Papers 6 (Vol. 1: Intonation and Syntax), 7 (Vol. 2: Semantics), and 8 (Vol. 3: Discourse) of the IBM Institute for Logic and Linguistics, Heidelberg.

[2] Cahn, J. (1995). 'The Effect of Pitch Accenting on Pronoun Referent Resolution'. Manuscript. MIT, Cambridge (Mass.).

[3] Chafe, W.L., (1976). 'Givenness, Contrastiveness, Definitness, Subjects, Topics and Point of View'. In C.N. Li (ed.) (1976), *Subject and Topic*, 25–55. Associated Press, New York.

[4] Clark, H.H., and S.E. Haviland (1977). 'Comprehension and the Given-New Contract'. In R.O. Freedle (ed.) (1977), *Discourse Production and Comprehension*, 1–40. Lawrence Erlbaum Asscoicates, Hillsdale (New Jersey).

[5] Dahl, Ö. (1974). (1974). 'Topic-Comment Structure Revisited'. In Ö. Dahl (ed.) (1974), *Topic and Comment, Contextual Boundedness and Focus. Papers in Text Linguistics 6*. Helmut Buske, Hamburg.

[6] Deemter, K. van (1992). 'Towards a Generalization of Anaphora'. *Journal of Semantics* **9**, 27–51.

[7] Deemter, K. van (1994a). 'What's New? A Semantic Perspective on Sentence Accent'. *Journal of Semantics* **11**, 1–31.

[8] Deemter, K. van (1994b). 'Contrastive Stress, Contrariety and Focus'. In P. Bosch and R. van der Sandt (eds.), 39–49.

[9] Dekker, P., and H. Hendriks (1994). 'Files in Focus'. In Engdahl (ed.), 27–38.

[10] Engdahl, E. (ed.) (1994). *Integrating Information Structure into Constraint-based and Categorial Approaches*. ESPRIT Basic Research Project 6852, Dynamic Interpretation of Natural Language. DYANA-2 Deliverable R1.3.B. ILLC, University of Amsterdam.

[11] Heim, I. (1982). *The Semantics of Definite and Indefinite Noun Phrases*. Ph.D. Dissertation University of Massachusetts, Amherst. Published in 1989 by Garland. New York.

[12] Heim, I. (1983). 'File Change Semantics and the Familiarity Theory of Definiteness'. In R. Bäuerle, C. Schwarze and A. von Stechow (eds.) (1983), *Meaning, Use and Interpretation of Language*. De Gruyter, Berlin, 164–189.

[13] Hendriks, H. (1994). 'Information Packaging in a Categorial Perspective'. In Engdahl (ed.), 89–116.

[14] Hendriks, H. (*draft*). 'Intonation, Derivation, Information'. Utrecht University.

[15] Hoffman, B. (1995). 'Integrating "Free" Word Order Syntax and Information Structure'. Manuscript, University of Pennsylvania.

[16] Jackendoff, R. (1972). *Semantic Interpretation in Generative Grammar* MIT Press, Cambridge (Mass.).

[17] Jacobs, J. (1983). *Fokus und Skalen: Zur Syntax und Semantik von Gradpartikeln im Deutschen*. Niemeyer, Tübingen.

[18] Kameyama, M. (1994). 'Stressed and Unstressed Pronouns: Complementary Preferences'. In P. Bosch and R. van der Sandt (eds.), 475–484.

[19] Kamp, H. (1981). 'A Theory of Truth and Semantic Representation'. In J. Groenendijk, T. Janssen and M. Stokhof (eds.) (1981), *Formal Methods in the Study of Language*. Mathematical Centre, Amsterdam. Reprinted in J. Groenendijk, T. Janssen and M. Stokhof (eds.) (1984), *Truth, Interpretation and Information. Selected Papers from the Third Amsterdam Colloquium*. Foris, Dordrecht.

[20] Kamp, H., and U. Reyle (1993). *From Discourse to Logic. Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer, Dordrecht.

[21] Krifka, M. (1991). 'A Compositional Semantics for Multiple Focus Constructions'. *Linguistische Berichte, Suppl.* **4**, 17–53.

[22] Lakoff, G., (1971). 'On Generative Semantics'. In D. Steinberg and L. Jacobovitz (eds.) (1971), *Semantics*. Cakbridge University Press, Cambridge, 232–296.

[23] Lambek, J. (1961). 'On the Calculus of Syntactic Types'. In R. Jakobson (ed.) (1961), *Structure of Language and its Mathematical Aspects*. Providence.

[24] Linden, E.-J. van der (1991). 'Accent Placement and Focus in Categorial Logic'. In S. Bird (ed.) (1991) *Declarative Perspectives on Phonology*. Edinburgh Working Papers in Cognitive Science. ECCS, Edinburgh.

[25] Oehrle, R. (1991). 'Prosodic Constraints on Dynamic Grammatical Analysis'. In S. Bird (ed.) *Declarative Perspectives on Phonology*. Edinburgh Working Papers in Cognitive Science. ECCS, Edinburgh.

[26] Moortgat, M. (1993). 'Generalized Quantifiation and Discontinuous Type Constructors'. In W. Sijtsma and A. van Horck (eds.) (1993), *Proceedings of the Tilburg Symposium on Discontinuous Dependencies*. De Gruyter, Berlin.

[27] Moortgat, M. (1994). 'Residuation in Mixed Lambek Systems'. In M. Moortgat (ed.) (1994), ESPRIT Basic Research Project 6852, Dynamic Interpretation of Natural Language, DYANA-2 Deliverable R1.1.B. ILLC, University of Amsterdam, and to appear in IGPL Bulletin.

[28] Moortgat, M., and G. Morrill (1991). 'Heads and Phrases. Type Calculus for Dependency and Constituent Structure'. OTS Research Paper, University of Utrecht.

[29] Nooteboom, S.G., and J.M.B. Terken (1982). 'What Makes Speakers Omit Pitch Accents?'. *Phonetica*. **39**: 317–336.

[30] Pierrehumbert, J. (1980). *The Phonology and Phonetics of English Intonation*. Ph.D. Disertation. MIT, Cambridge (Mass.). Distributed by the IULC.

[31] Prince, E. (1981). 'Toward a Taxonomy of Given-New Information'. In P. Cole, *Radical Pragmatics*. Academic Press, New York, 233–255.

[32] Reinhart, T. (1982). 'Pragmatics and Linguistics: An Analysis of Sentence Topics'. *Philosophica* **27**, 53–94.

[33] Rooth, M. (1985). *Association with Focus*. Ph.D. Dissertation University of Massachusetts, Amherst.

[34] Rooth, M. (1992). 'A Theory of Focus Interpretation'. *Natural Language Semantics* **1**, 75–116.

[35] Steedman, M. (1991). 'Structure and Intonation'. *Language* **67**, 260–296.

[36] Steedman, M. (1992). 'Surface Structure, Intonation and "Focus" '. In E. Klein and F. Veltman (eds.) *Natural Language and Speech. Symposium Proceedings, Brussels, November 1991*. Springer, Berlin.

[37] Steedman, M. (1993). 'The Grammar of Intonation and Focus'. In P. Dekker and M. Stokhof (eds.) (1993), *Proceedings of the Ninth Amsterdam Colloquium, December 14–17, 1993*, Part III. ILLC, University of Amsterdam.

[38] Svoboda, A., and P. Materna (1987). 'Functional Sentence Perspective and Intensional Logic'. In R. Dirven and V. Fried (eds.) *Functionalism in Linguistics*. John Benjamins, Amsterdam.

[39] Szabolcsi, A. (1981). 'The Semantics of Topic-Focus Articulation'. In J. Groenendijk, T. Janssen and M. Stokhof (eds.) (1981), *Formal Methods in the Study of Language*. Mathematical Centre, Amsterdam.

[40] Szabolcsi, A. (1983). 'Focussing Properties, or the Trap of First Order'. In *Theoretical Linguistics* **10**, 125–145.

[41] Vallduví, E. (1992). *The Informational Component*. Garland, New York.

[42] Vallduví, E. (1993). 'Information Packaging: A Survey'. Report prepared for *Word Order, Prosody, and Information Structure*. Centre for Cognitive Science and Human Communication Research Centre, University of Edinburgh.

[43] Vallduví, E. (1994). 'The Dynamics of Information Packaging'. In Engdahl (ed.), 1–27.

[44] Vallduví, E., and R. Zacharski (1993). 'Accenting Phenomena, Association with Focus, and the Recursiveness of Focus-Ground'. In P. Dekker and M. Stokhof (eds.) (1993) *Proceedings of the Ninth Amsterdam Colloquium, December 14–17, 1993*, Part III. ILLC, University of Amsterdam.

Safety for Bisimulation in General Modal Logic

Marco Hollenberg, Department of Philosophy, Utrecht University

Abstract

We define a natural generalisation of the notions of invariance and safety for bisimulation, and characterise the resulting notion of (n, m)-safety for bisimulation.

1 Introduction

Modal logic, when interpreted on models as opposed to frames, can be seen as a fragment of first order logic. This fragment is motivated mainly historically, but also by the well-known *invariance theorem* (Benthem 1976, Rijke 1993) that states that first order formulas $\phi(x_1)$ in a single free variable x_1 that are invariant for bisimulation are precisely the modal ones, modulo logical equivalence.

Invariance is a very natural notion from the viewpoint of modal logic. Propositional Dynamic Logic (PDL, (Harel 1984)) gave rise to another natural notion: *safety for bisimulation*. A formula $\phi(x_1, y_1)$ is safe for bisimulation iff whenever Z is a bisimulation between two models $\mathcal{M}$ and $\mathcal{N}$ then this same Z is also a bisimulation with respect to the relations defined by ϕ in $\mathcal{M}$ and $\mathcal{N}$ respectively. Van Benthem (Benthem 1993) proved that the safe formulas consist of the *programs* of PDL that do not contain the iteration operator $*$, which is, of course, not first order definable.

In the past, bisimulation was a notion defined just on structures with binary relations and unary predicates, so-called transition systems. The two theorems mentioned above would have sufficed in that era: the notion of invariance is just a generalisation of the bisimulation-clause for unary predicates, while that of safety is an obvious generalisation of the zig- and zag-clauses for binary relations. The study of *polyadic* modal operators, such as the binary composition operator in Arrow Logic (Venema 1992), whose corresponding first order translations use relation symbols of arity greater than two, created the need for a more general notion of bisimulation. So nowadays bisimulation is a relation that can hold between models (of the same type) with relations of any arity greater than zero. Thus, new notions of safety are called for.

The structure of the paper is as follows. The next section introduces the necessary concepts and states the relevant theorems that have occurred in the literature, namely the invariance theorems of (Benthem 1976) and (Rijke 1993) and the safety theorem of (Benthem 1993). An obvious generalisation of the notions that these theorems deal with is defined: (n, m)-*safety*, for $n, m \in \mathbb{N}$. The section after this introduces the languages $\text{PROG}_{(n,m)}$, all of whose formulas are (n, m)-safe. The section thereafter forms the technical heart of this paper: in it we prove that $\text{PROG}_{(n,m)}$ consists precisely of the (n, m)-safe formulas. We finish the paper with some conclusions and suggestions for future research.

The notion of invariance and safety also plays a role in defining process-algebraic operations on process graphs (or: rooted Kripke models) that *respect* bisimulation. This is discussed in detail in (Hollenberg 1995a). The proofs in that paper are very similar to those in the present one, but since the concerns of both papers are quite distinct, we have chosen to write two separate papers.

Notational Conventions

Whenever in the text we introduce a first order formula as in 'Consider $\phi(x_1, \ldots, x_n)$ ...' we mean: 'Consider ϕ, whose free variables are among $x_1, \ldots, x_n$ ...'. After having introduced a formula in this fashion, we may simply use ϕ to refer to it. Moreover $\phi(y_1, \ldots, y_n)$ will then refer to the formula $\phi[x_1 := y_1, \ldots, x_n := y_n]$, the result of simultaneously substituting in ϕ every occurence of x_i by y_i, for each

$1 \leq i \leq n$. Here we may need to apply *α-conversion* on ϕ first, making sure that none of the bound variables in ϕ occur among $x_1, \ldots, x_n, y_1, \ldots, y_n$. Finally, when interpreting ϕ in a model $\mathcal{M}$, we may write $\mathcal{M} \models \phi[s_1, \ldots, s_n]$, where $[s_1, \ldots, s_n]$ indicates an assignment sending $x_1, \ldots, x_n$ to $s_1, \ldots, s_n$ respectively.

2 Safety for bisimulation

Traditionally, the only structures of importance to modal logic were Kripke models. These are simply relational structures, for a signature (that is, a language) containing a single binary relation symbol to interpret the modal diamond $\Diamond$ and infinitely many unary predicate symbols to interpret the proposition letters of the modal language. In later years, this perspective was broadened somewhat by allowing more binary relation symbols in our signature. Structures for this type of signature (i.e. transition systems) can be used as models for a polymodal language. In recent years, however, even this has not sufficed. Arrow Logic (Venema 1992), for example, is a modal language that has a binary modality, which in the intended structures is interpreted by means of a ternary accessibility relation. Following this trend to the limit we are forced to view structures for any relational signature $\mathcal{L}$ as models for some modal language (where we assume that $\mathcal{L}$ does not contain relation symbols of arity zero: we will see why shortly).

Just as the signature of ordinary Kripke models easily gives rise to the standard modal language, an arbitrary $\mathcal{L}$ also determines a modal language. Its formulas are defined as follows:[1]

$$\phi := \top \mid \neg\phi \mid \phi \vee \phi \mid \langle R \rangle(\phi_1, \ldots, \phi_n)$$

where R is an $n+1$-ary relation symbol in $\mathcal{L}$. Such formulas are interpreted, as usual, at points in $\mathcal{L}$-structures $\mathcal{M}$. The notion of **truth of a modal formula** ϕ at a point s in some $\mathcal{L}$-model $\mathcal{M}$ (notation $\mathcal{M}, s \Vdash \phi$, or just $s \Vdash \phi$ if $\mathcal{M}$ is clear from the context) is given by the following truth-definition:

$s \Vdash \top$		always
$s \Vdash \neg\phi$	iff	$s \nVdash \phi$
$s \Vdash \phi \vee \psi$	iff	$s \Vdash \phi$ or $s \Vdash \psi$
$s \Vdash \langle R \rangle(\phi_1, \ldots, \phi_n)$	iff	there are $s_1, \ldots, s_n$ with $(s, s_1, \ldots, s_n) \in R^{\mathcal{M}}$ and $s_i \Vdash \phi_i$ for each $1 \leq i \leq n$.

Here $R^{\mathcal{M}}$ denotes of course the interpretation of the symbol R in $\mathcal{M}$.

Readers familiar with standard modal logic may not recognise their own systems in this setting. But note that when R is unary, the nullary diamond $\langle R \rangle$ behaves as a proposition letter: it is true at a point s if s is in the interpretation of R. As to why we restricted our relational signatures not to contain nullary symbols, note that the uniform schema above giving us a modal operator and its truth definition does not work for such nullary relation symbols.

Modal formulas can be seen as abbreviations for certain first order formulas. What they abbreviate is given by the so-called **standard translation** ST. This maps any modal formula ϕ to a first order formula $ST(\phi)$ that contains at most x free, where x_1 is supposed to indicate the point of evaluation of the modal formula.

1. This is also the way that the *basic modal language* is defined in (Rijke 1993). (Venema 1991) does things differently and first defines the modal language, and then assigns to each n-ary modality an $n+1$-ary relation symbol. Thus, different modalities may have the same relation as its accessibility relation, which cannot happen in our setup.

$$\begin{array}{lcl}
ST(\top) & := & \top \\
ST(\neg\phi) & := & \neg ST(\phi) \\
ST(\phi \vee \psi) & := & ST(\phi) \vee ST(\psi) \\
ST(\langle R \rangle(\phi_1, \ldots, \phi_n)) & := & \exists y_1 \ldots y_n . \left(\begin{array}{l} R(x_1, y_1, \ldots, y_k) \wedge \\ ST(\phi_1)(y_1) \wedge \ldots \wedge ST(\phi_n)(y_n) \end{array} \right)
\end{array}$$

Note that the standard translation simply copies the truth definition above. So it is not surprising that $\mathcal{M}, s \Vdash \phi$ iff $\mathcal{M} \models ST(\phi)[s]$.

Bisimulation is an important relation for comparing points in relational structures. Originally it was designed to fit well with modal logic: it was designed in such a way that bisimilar points satisfy the same modal formulas (Benthem 1976). But it has some independent motivation as well, from the area of process algebra (cf. (Baeten and Weijland 1994)), where it was developed independently from Van Benthem's work as a natural notion of process equivalence (by (Park 1981)). Applications of bisimulation are not restricted to computer science and modal logic however: it has also found its way into the fundamental area of non-wellfounded set theory (Barwise and Moss 1996).

The standard definition of bisimulation deals only with unary and binary predicates. This standard definition is a special case of the following general definition:

Definition 2.1 (Bisimulation) *Let $\mathcal{M}$ and $\mathcal{N}$ be two $\mathcal{L}$-models. A relation Z between $\mathcal{M}$ and $\mathcal{N}$ is an $\mathcal{L}$-bisimulation (or just 'bisimulation') if the following two clauses are satisfied:*

Zig *If sZt and $(s, s_1, \ldots, s_n) \in R^{\mathcal{M}}$ for some $R \in \mathcal{L}$ then there are $t_1, \ldots, t_n$ such that $(t, t_1, \ldots, t_n) \in R^{\mathcal{N}}$ and $s_1 Z t_1, \ldots, s_n Z t_n$.*

Zag *Vice versa: if sZt and $(t, t_1, \ldots, t_n) \in R^{\mathcal{N}}$ then there are $s_1, \ldots, s_n$ with $(s, s_1, \ldots, s_n) \in R^{\mathcal{M}}$ and $s_1 Z t_1, \ldots, s_n Z t_n$.*

We write $Z : \mathcal{M} \underline{\leftrightarrow} \mathcal{N}$ if Z is a bisimulation between these two models. When s is a point in $\mathcal{M}$ and t is in $\mathcal{N}$, we write $s \underline{\leftrightarrow} t$ if there is a bisimulation $Z : \mathcal{M} \underline{\leftrightarrow} \mathcal{N}$ connecting s and t. □

Bisimulations are sometimes also called **zigzagrelations**, which explains the names of the two conditions in the above definition. When the domain of the relation $Z : \mathcal{M} \underline{\leftrightarrow} \mathcal{N}$ is the whole domain of $\mathcal{M}$, we speak of a **total** bisimulation. Cases in point are functional bisimulations, known as **p-morphisms**, or **zigzagmorphisms**.

As bisimulations are closed under taking arbitrary unions, for any two models $\mathcal{M}$ and $\mathcal{N}$ there exists a **maximal bisimulation** $Z : \mathcal{M} \underline{\leftrightarrow} \mathcal{N}$. For such a maximal bisimulation Z, any other bisimulation $Z' : \mathcal{M} \underline{\leftrightarrow} \mathcal{N}$ is a subset of Z. The relation $\underline{\leftrightarrow}$ defined above is precisely this maximal bisimulation.

A natural question is: for which relations ρ do the zig- and zag-conditions carry over from the atomic relations of $\mathcal{L}$ to the relation ρ? In this paper we are only concerned with first order definable relations. So we are interested in the following notion: we call a first order $\mathcal{L}$-formula $\phi(x_1, y_1, \ldots, y_n)$ **n-safe for bisimulation** if whenever $Z : \mathcal{M} \underline{\leftrightarrow} \mathcal{N}$ then the zig- and zag-conditions are satisfied for the $n+1$-ary relations in $\mathcal{M}$ and $\mathcal{N}$ given by ϕ. That is: if sZt and $\mathcal{M} \models \phi[s, s_1, \ldots, s_n]$ then there are $t_1, \ldots, t_n$ such that $\mathcal{N} \models \phi[t, t_1, \ldots, t_n]$ and $s_i Z t_i$ for each $1 \leq i \leq n$, and similarly for the zag-condition.

This notion has been studied for various $\mathcal{L}$ and $n \in \mathbb{N}$ already. We list the two main results.

First, there's the notion of **invariance for bisimulation**. This corresponds to our notion of 0-safety. It is a generalisation of the bisimulation clauses for unary predicates, which tell us that bisimilar points should *agree* on all unary predicates. Let us spell it out: a formula $\phi(x_1)$ is invariant for bisimulation if whenever Z :

$\mathcal{M} \leftrightarrows \mathcal{N}$ and sZt then $\mathcal{M} \models \phi[s]$ iff $\mathcal{N} \models \phi[t]$.

(Benthem 1976) contains a characterisation of the first order formulas that are invariant for bisimulation, for the case where $\mathcal{L}$ has a single unary relation symbol and infinitely many unary predicate symbols. There, it is shown that invariance for bisimulation corresponds precisely with equivalence to a (translation of a) modal formula. (Rijke 1993) contains the same result for arbitrary relational signatures. (Rosen 1995) has a finite model theory version of Van Benthem's invariance theorem, that can easily be extended to the general case: if we restrict ourselves to finite models, still the only invariant formulas are the modal ones (modulo equivalence in the class of finite models).

These invariance theorems give us a rational reconstruction of the modal fragment of first order logic. For although historically bisimulations were designed for the specific needs of modal logic, we may also take bisimulation as the fundamental notion. If we then also take first order logic as fundamental, the study of modal logic is motivated by the invariance theorem: given first order logic and bisimulation, you get the modal fragment for free.

Second, (Benthem 1993) studies a notion of safety for bisimulation in a setting of transition systems. In this paper, a formula $\phi(x_1, y_1)$ is called safe for bisimulation if whenever $Z : \mathcal{M} \leftrightarrows \mathcal{N}$, sZt and $\mathcal{M} \models \phi[s, s']$ then we can find a t' with $s'Zt'$ and $\mathcal{N} \models \phi[t, t']$. The zag-clause for binary relations is automatically also satisfied by such ϕ, as bisimulations are closed under converse. This notion of safety clearly corresponds to our notion of 1-safety. It is demonstrated in (Benthem 1976) that the 1-safe first order formulas are precisely those equivalent to a formula in the fragment defined as follows.

Let PROG (PROG for 'programs') be the language containing as formulas:

$$\theta ::= R \mid \phi? \mid \theta; \theta \mid \theta \cup \theta$$

where R is a binary relation symbol in $\mathcal{L}$ and ϕ is a modal formula of $\mathcal{L}$. So PROG consists of the programs of PDL (Propositional Dynamic Logic, see (Harel 1984)) that do not use iteration $*$.

PROG-formulas are to be seen as abbreviations for first order formulas in two free variables, x_1 and y_2. This is again made clear by means of a standard translation ST (which we have already defined for the modal formulas):

$$\begin{array}{lcl} ST(R) & := & R(x_1, y_1) \\ ST(\phi?) & := & x_1 = y_1 \wedge ST(\phi)(x_1) \\ ST(\theta_1; \theta_2) & := & \exists z.(ST(\theta_1)(x_1, z) \wedge ST(\theta_2)(z, y_1)) \\ ST(\theta_1 \cup \theta_2) & := & ST(\theta_1) \vee ST(\theta_2) \end{array}$$

In (Hollenberg 1995b) a safety theorem is proved for the finite model case, thereby expanding on the work in (Rosen 1995): no new formulas become 1-safe, modulo equivalence in the class of finite models of course.

An obvious generalisation of the characterisations mentioned above would involve a characterisation of the n-safe formulas, for any signature $\mathcal{L}$. In fact, we will study an even more general notion, as a pre-emptive strike against possible future theorems on the subject of safety, namely (n, m)-safety, for $n, m \in \mathbb{N}$.

Definition 2.2 (**(n,m)-safety**) *A first order formula $\phi(x_1, \ldots, x_n, y_1, \ldots, y_m)$ of $\mathcal{L}$ (with $x_1, \ldots, x_n, y_1, \ldots, y_m$ all distinct) is called* **(n,m)-safe** *if whenever Z is a bisimulation between two $\mathcal{L}$-models $\mathcal{M}$ and $\mathcal{N}$ and $s_1, \ldots, s_n \in \mathcal{M}$, $t_1, \ldots, t_n \in \mathcal{N}$ such that $s_i Z t_i$ (for each $1 \leq i \leq n$) then:*

Zig *$\mathcal{M} \models \phi[s_1, \ldots, s_n, u_1, \ldots, u_m]$ implies the existence of $v_1, \ldots, v_m \in \mathcal{N}$ such that $\mathcal{N} \models \phi[t_1, \ldots, t_n, v_1, \ldots, v_m]$ and $u_j Z v_j$ (for each $1 \leq j \leq m$).*

Zag *$\mathcal{N} \models \phi[t_1, \ldots, t_n, v_1, \ldots, v_m]$ implies that there are $u_1, \ldots, u_m \in \mathcal{M}$ with $\mathcal{M} \models \phi[s_1, \ldots, s_n, u_1, \ldots, u_m]$ and $u_j Z v_j$ (for each $1 \leq j \leq m$).* □

To determine whether a formula ϕ is (n, m)-safe, we have to pick a sequence $x_1 \ldots x_n$ of distinct *input* variables and a disjoint sequence $y_1 \ldots y_n$ of distinct *output* variables, in order to have this question make sense. We solve this problem as follows. Divide the (infinite) set of variables into three disjoint sets, $X = \{x_1, x_2, \ldots\}$, $Y = \{y_1, y_2, \ldots\}$ and AUX of input, output and auxiliary variables, respectively. Now the question of (n, m)-safety can only be asked of formulas whose free variables occur among $x_1, \ldots, x_n, y_1, \ldots, y_m \subset X \cup Y$. For such formulas, the question is clear. There is no more need to first *pick* $x_1 \ldots x_n$ and $y_1 \ldots y_n$: these are fixed names for variables.

Let us consider a few examples.

- $R_a(x_1, y_1) \wedge R_b(x_1, y_2)$ is $(1, 2)$-safe (where R_a and R_b are two distinct binary predicates in $\mathcal{L}$).
- $R_a(x_1, y_1) \wedge R_b(x_1, y_1)$ (i.e. $R_a \cap R_b$) is not $(1, 1)$-safe.
- $x_1 = y_1$ is $(1, 1)$-safe.
- $x_1 = x_2$ is not $(2, 0)$-safe.
- $\top$ (the tautology) is $(n, 0)$-safe, for any n. It is *not* (n, m)-safe, for any $m > 0$. Note that $\top$ is thus an example where the question of (n, m)-safety is meaningful for different pairs (n, m). In general, if the question of (n, m)-safety is meaningful for some formula ϕ, it is also meaningful for (n', m'), with $n' \geq n$ and $m' \geq m$.
- $\bot$ is vacuously (n, m)-safe, for any n, m.
- Our earlier notion of n-safety corresponds to $(1, n)$-safety. Thus invariance for bisimulation corresponds to $(1, 0)$-safety and Van Benthem's notion of safety in the language of transition systems corresponds to $(1, 1)$-safety in our terminology.
- Studying (n, m)-safety instead of just n-safety has some support from the area of process algebra (Baeten and Weijland 1994). For example, the common process algebraic operator of *free merge* uses as part of its definition the formula:

$$(R_a(x_1, y_1) \wedge x_2 = y_2) \vee (x_1 = y_1 \wedge R_a(x_2, y_2))$$

  This formula is $(2, 2)$-safe, which explains why the free merge operator *respects* bisimulation. For a detailed investigation of this perspective on process algebra one may consult (Hollenberg 1995a) or (Benthem 1994), on which the former paper is based.

In the next section we will present fragments $\text{PROG}_{(n,m)}$ of the first order language all of whose formulas are (n, m)-safe. The section thereafter contains the proof of the converse: if a formula is (n, m)-safe then it must be equivalent to a $\text{PROG}_{(n,m)}$-formula.

3 Safe fragments

The languages $\text{PROG}_{(n,m)}$ are simultaneously defined as follows:

$\bot \in \text{PROG}_{(0,1)}$
$\text{ID} \in \text{PROG}_{(1,1)}$
$\Delta \in \text{PROG}_{(1,2)}$
$\pi \in \text{PROG}_{(2,2)}$
$R \in \text{PROG}_{(1,n)}$ if R is an $n+1$-ary relation symbol of $\mathcal{L}$.
$\phi;\psi \in \text{PROG}_{(n,m)}$ if $\phi \in \text{PROG}_{(n,k)}$ and $\psi \in \text{PROG}_{(k,m)}$.
$\phi \cup \psi \in \text{PROG}_{(n,m)}$ if $\phi, \psi \in \text{PROG}_{(n,m)}$.
$\sim \phi \in \text{PROG}_{(n,0)}$ if $\phi \in \text{PROG}_{(n,m)}$.
$\phi \bullet \psi \in \text{PROG}_{(n+k,m+l)}$ if $\phi \in \text{PROG}_{(n,m)}$ and $\psi \in \text{PROG}_{(k,l)}$.

The function ST sends a $\text{PROG}_{(n,m)}$-formula to an $\mathcal{L}$-formula of the form $\phi(x_1, \ldots, x_n, y_1, \ldots, y_m)$, which gives us its intended meaning:

$$\begin{array}{lcl}
ST(\bot) & := & \bot \\
ST(\text{ID}) & := & x_1 = y_1 \\
ST(\Delta) & := & x_1 = y_1 \wedge x_1 = y_2 \\
ST(\pi) & := & x_1 = y_2 \wedge x_2 = y_1 \\
ST(R) & := & R(x_1, y_1, \ldots, y_n) \\
ST(\phi;\psi) & := & \exists z_1 \ldots z_k. \left(\begin{array}{l} ST(\phi)(x_1, \ldots, x_n, z_1, \ldots, z_k) \wedge \\ ST(\psi)(z_1, \ldots, z_k, y_1, \ldots, y_m) \end{array} \right) \\
ST(\phi \cup \psi) & := & ST(\phi) \vee ST(\psi) \\
ST(\sim \phi) & := & \neg \exists y_1 \ldots y_m. ST(\phi) \\
ST(\phi \bullet \psi) & := & \left(\begin{array}{l} ST(\phi)(x_1, \ldots, x_n, y_1, \ldots, y_m) \wedge \\ ST(\psi)(x_{n+1}, \ldots, x_{n+k}, y_{m+1}, \ldots, y_{m+l}) \end{array} \right)
\end{array}$$

A more succinct way of stating the meaning of $\text{PROG}_{(n,m)}$-formulas is perhaps by means of relations on sequences of points. Given a model $\mathcal{M}$, we associate with each $\text{PROG}_{(n,m)}$-formula θ a relation $[\![\theta]\!]$ between $\mathcal{M}^n$ and $\mathcal{M}^m$. Instead of $(s_1 \ldots s_n, t_1 \ldots t_m) \in [\![\theta]\!]$ we write $s_1 \ldots s_n \xrightarrow{\theta} t_1 \ldots t_m$.

1. $\lambda \xrightarrow{\bot} s$ is *never* the case (where λ is the empty sequence).
2. $s \xrightarrow{\text{ID}} s$ for every $s \in \mathcal{M}$.
3. $s \xrightarrow{\Delta} ss$ for every $s \in \mathcal{M}$.
4. $st \xrightarrow{\pi} ts$ for every $s, t \in \mathcal{M}$.
5. $s \xrightarrow{R} s_1 \ldots s_n$ iff $(s, s_1, \ldots, s_n) \in R^{\mathcal{M}}$.
6. $\sigma \xrightarrow{\phi;\psi} \tau$ iff there is a ρ with $\sigma \xrightarrow{\phi} \rho$ and $\rho \xrightarrow{\psi} \tau$, where σ, τ and ρ are sequences of the appropriate length.
7. $\sigma \xrightarrow{\phi \cup \psi} \tau$ iff $\sigma \xrightarrow{\phi} \tau$ or $\sigma \xrightarrow{\psi} \tau$.
8. $\sigma \xrightarrow{\sim\phi} \lambda$ iff there is *no* τ with $\sigma \xrightarrow{\phi} \tau$.
9. $\sigma \xrightarrow{\phi \bullet \psi} \tau$ iff there are σ_i, τ_i $(i = 1, 2)$ such that $\sigma = \sigma_1\sigma_2$, $\tau = \tau_1\tau_2$, $\sigma_1 \xrightarrow{\phi} \tau_1$ and $\sigma_2 \xrightarrow{\psi} \tau_2$.

So in this setting ; denotes relational composition and $\cup$ denotes union, which explains the notation. Likewise, Δ is for 'diagonal' and π for 'permutation'.

To increase readability we introduce some notational conventions. First, ;, $\cup$ and $\bullet$ are treated as associative operations. This is not harmful, as they *are* associative on the semantic level. Second, the notation $\phi_1 \bullet \ldots \bullet \phi_n$ will sometimes be used. If $n \geq 2$, which formula is denoted by this should be clear (as $\bullet$ is associative). When $n = 0$, we let it denote the formula $\sim \bot$ and when $n = 1$ it stands for just ϕ_1.

The languages $\text{PROG}_{(n,m)}$ could be viewed as a single language, but with *sorts*, or *types*, (n, m). Such languages are not uncommon in the study of variable-free languages. An important example is Quine's variable-free predicate logic (Quine

1966). There, we see sorts $n \in \mathbb{N}$, where a formula of sort n is a formula having its free variables among the first n variables and which is *viewed* as an n-ary relation. In the present situation something similar occurs: a $\text{PROG}_{(n,m)}$-formula is seen as a relation from n-tuples to m-tuples.

Every formula in our setup has a unique *type* (n, m): there are no formulas that are both in $\text{PROG}_{(n,m)}$ and in $\text{PROG}_{(k,l)}$, with $(n, m) \neq (k, l)$. Yet there may be formulas of different types whose standard translations are equivalent semantically. An example is given by $\sim\sim \bot$ and $\bot$. The first of these has type $(0, 0)$, the second $(0, 1)$, yet they are equivalent as both are unsatisfiable.

Example: deterministic programming in $\text{PROG}_{(n,m)}$.
The name '$\text{PROG}_{(n,m)}$' points to the intuition that these formulas can be seen as non-deterministic programs, that on input of an n-sequence may output an m-sequence. To gain some experience with the new languages, let us write some *deterministic* programs in $\text{PROG}_{(n,m)}$. These programs will be useful in proofs later on as well.

- ID_n is the vacuous program: on input $s_1 \ldots s_n$ it outputs the same sequence. It is defined as $\text{ID} \bullet \ldots \bullet \text{ID}$ (n times).
- DEL_n is a more destructive program. It is defined thus: $\sim\sim \text{ID}_n$. On any sequence $s_1 \ldots s_n$, DEL_n gives us λ, the empty sequence.
- Projection is defined using the deletion program. If $1 \leq i \leq n$ we define π_i^n as $\text{DEL}_{i-1} \bullet \text{ID} \bullet \text{DEL}_{n-i}$. Thus π_i^n sends $s_1 \ldots s_n$ to s_i.
- The program RIGHT_n sends a sequence $st_1 \ldots t_n$ to $t_1 \ldots t_n s$:

$$\begin{array}{lcl} \text{RIGHT}_0 & := & \text{ID} \\ \text{RIGHT}_{n+1} & := & (\text{RIGHT}_n \bullet \text{ID}); (\text{ID}_n \bullet \pi). \end{array}$$

- Next we define a program $\text{COPY}_n \in \text{PROG}_{n,2n}$ that copies the input and places the result next to it.

$$\begin{array}{lcl} \text{COPY}_0 & := & \sim \bot \\ \text{COPY}_{n+1} & := & (\text{ID} \bullet \text{COPY}_n); (\Delta \bullet \text{ID}_{2n}); (\text{ID} \bullet \text{RIGHT}_n \bullet \text{ID}_n). \end{array}$$

  So COPY_n sends $s_1 \ldots s_n$ to $s_1 \ldots s_n s_1 \ldots s_n$.
- COPY_n^m does the previous procedure a fixed number of times: it produces m copies of the input sequence next to each other. Definition:

$$\begin{array}{lcl} \text{COPY}_n^0 & := & \text{DEL}_n \\ \text{COPY}_n^{m+1} & := & \text{COPY}_n; (\text{ID}_n \bullet \text{COPY}_n^m). \end{array}$$

- $\text{PERMUTE}(n, m)$ takes as input $s_1 \ldots s_n t_1 \ldots t_m$ and outputs $t_1 \ldots t_m s_1 \ldots s_n$:

$$\text{PERMUTE}(n, m) := \text{COPY}_{n+m}; (\text{DEL}_n \bullet \text{ID}_{n+m} \bullet \text{DEL}_m).$$

- Finally, we give the mother of all deterministic programs. Let f be a function from $\{1, \ldots, m\}$ to $\{1, \ldots, n\}$. Then $[f]$ will denote the program in $\text{PROG}_{(n,m)}$ that sends $s_1 \ldots s_n$ to $s_{f(1)} \ldots s_{f(m)}$. It is defined as follows:

$$[f] := \text{COPY}_n^m; (\pi_{f(1)}^n \bullet \ldots \bullet \pi_{f(m)}^n)$$

Here endeth the programming lesson.

A trivial theorem is the following:

Theorem 3.1 *$\text{PROG}_{(n,m)}$-formula are (n, m)-safe.* □

In the next section we will prove the converse to this: any (n,m)-safe formula ϕ is equivalent to a $\text{PROG}_{(n,m)}$-formula. The invariance theorem of (Benthem 1976, Rijke 1993) and the safety theorem in (Benthem 1993) already give us this converse for certain cases.

Lemma 3.2 $\text{PROG}_{(1,0)}$ *is (modulo equivalence) precisely the set of modal formulas.*

Proof.
Recall that $(1,0)$-safety is the same as 0-safety and thus the same as invariance. So $\text{PROG}_{(1,0)}$-formulas are invariant. By the invariance theorem of (Rijke 1993), they must then be equivalent to modal formulas. We could also give a more constructive proof, by means of a translation function that recursively translates $\text{PROG}_{(1,0)}$-formulas into equivalent modal ones.

For the other direction, we define the translation $(_)^0$ from modal formulas to $\text{PROG}_{(1,0)}$-formulas such that ϕ^0 is equivalent to ϕ:

$$\begin{array}{lcl} \top^0 & := & \sim\sim \text{ID} \\ (\phi \vee \psi)^0 & := & \phi^0 \cup \psi^0 \\ (\neg\phi)^0 & := & \sim \phi^0 \\ (\langle R\rangle(\phi_1,\ldots,\phi_n))^0 & := & R;((\phi_1)^0 \bullet \ldots \bullet (\phi_n)^0) \end{array}$$

□

Something similar can be done for PROG-formulas. As Van Benthem's safety theorem is limited to languages with just binary and unary relations, the following lemma is restricted as well.

Lemma 3.3 *Let $\mathcal{L}$ be a language with just binary and unary relations. Then* $\text{PROG}_{(1,1)}$ *is the same as* PROG*, modulo logical equivalence.*

Proof.
By the safety theorem and the fact that $(1,1)$-safety corresponds to Van Benthem's notion of safety, we conclude that $\text{PROG}_{(1,1)}$-formulas are equivalent to PROG-formulas.

The other direction is again proved by means of a translation, $(_)^1$, from PROG-formulas to $\text{PROG}_{(1,1)}$-formulas. Because the definition of PROG includes reference to the modal language, we also need the translation $(_)^0$.

$$\begin{array}{lcl} R^1 & := & R \\ (\phi?)^1 & := & \Delta;(\phi^0 \bullet \text{ID}) \\ (\phi;\psi)^1 & := & \phi^1;\psi^1 \\ (\phi \cup \psi)^1 & := & \phi^1 \cup \psi^1. \end{array}$$

□

4 Characterising (n,m)-safety

In this section we will prove our main theorem:

Theorem 4.1 (Main Theorem) *A first order formula $\phi(x_1,\ldots,x_n,y_1,\ldots,y_m)$ is (n,m)-safe iff it is equivalent to a* $\text{PROG}_{(n,m)}$*-formula.*

The 'if'-part of this theorem has already been stated in theorem 3.1. This section is therefore devoted solely to the 'only if'-part. The first few subsections deal with some modal techniques that we will need along the way: modal saturation and unravelling. After this, we define special sets of first order formulas, so-called

descriptions, which will be of use in the proof of the main theorem. Then the main theorem for the case that $n = 0$ is proved. For $n > 0$ the proof is modelled along the lines of the proof Van Benthem's safety theorem (Benthem 1993), but without his notion of *continuous* modal formulas.

4.1 Modal saturation

Definition 4.2 *Let $\mathcal{M}$ be some $\mathcal{L}$-model. We call $\mathcal{M}$* **modally saturated** *(or* m-saturated*, for short) if, for every $s \in \mathcal{M}$, every $n+1$-ary relation R in $\mathcal{L}$ and every sequence $\Phi_1,\ldots,\Phi_n$ of sets of modal formulas:*

For every sequence of finite subsets $\Delta_1 \subseteq \Phi_1,\ldots,\Delta_n \subseteq \Phi_n$ there are $s_1,\ldots,s_n$ such that $(s,s_1,\ldots,s_n) \in R^{\mathcal{M}}$ and $s_1 \Vdash \Delta_1,\ldots,s_n \Vdash \Delta_n$

implies that

there are $s_1,\ldots,s_n$ such that $(s,s_1,\ldots,s_n) \in R^{\mathcal{M}}$ and $s_1 \Vdash \Phi_1,\ldots,s_n \Vdash \Phi_n$. □

Modal saturation was first defined (under a different name) in (Fine 1975). It is studied in detail in (Hollenberg 1995c). We list a few basic facts, proofs of which may be found in the latter paper.

Fact 4.3 *If $\mathcal{M}$ and $\mathcal{N}$ are both m-saturated then the relation of modal equivalence*

$$Z := \{(s,t) \in \mathcal{M} \times \mathcal{N} \mid s \text{ and } t \text{ satisfy the same modal formulas}\}$$

is a bisimulation between them. In fact, Z is then the maximal bisimulation. □

Fact 4.4 *If $\mathcal{M}$ is ω-saturated then it is m-saturated.* □

Fact 4.5 *Let $Z : \mathcal{M} \leftrightarrows \mathcal{N}$ be a total bisimulation (i.e. $dom(Z) = \mathcal{M}$) and let $\mathcal{N}$ be m-saturated. Then $\mathcal{M}$ is also m-saturated.* □

4.2 Unravelling and multiplying

This section contains some useful operations on models, which will play key roles in the proof of our main theorem.

First, we define an *unravelling* construction, one that takes a model and turns it into a tree-like structure. It is a slight generalisation of the standard unravelling construction. This same generalisation can be found in (Rijke 1993).

Definition 4.6 *Let $\mathcal{M}$ be a model and suppose s is an element of M, the domain of $\mathcal{M}$. We define the model $\mathcal{M}_s^\nabla$ as follows:*

- *M_s^∇, the domain of $\mathcal{M}_s^\nabla$, contains pairs, the first coordinate an element of M, the second containing strings from the alphabet $M \cup \mathcal{L} \cup \{[]\}$. We always assume that the three sets making up this alphabet are disjoint. If σ and τ are two strings from this alphabet then $\sigma[\tau]$ is the result of replacing every occurrence of the 'gap' $[]$ in σ by the string τ. Now, M_s^∇ is construed thus:*
  - *$(s,[]) \in M_s^\nabla$;*
  - *If $(s_0,\sigma) \in M_s^\nabla$ and $(s_0,s_1,\ldots,s_n) \in R^{\mathcal{M}}$ for some $R \in \mathcal{L}$ of arity greater than 1, then*

$$(s_i, \sigma[Rs_0s_1\ldots s_{i-1}[]s_{i+1}\ldots s_n])$$

is also in M_s^∇, for every $1 \leq i \leq n$.

- *$n+1$-ary relation symbols $R \in \mathcal{L}$ are interpreted as follows:*

$$((s_0, \sigma), (s_1, \sigma_1) \ldots, (s_n, \sigma_n)) \in R^{\nabla}$$

iff $(s_0, s_1, \ldots, s_n) \in R^{\mathcal{M}}$ *and for every* $i \in \{1, \ldots, n\}$:

$$\sigma_i = \sigma[Rs_0 s_1 \ldots s_{i-1} [] s_{i+1} \ldots s_n].$$

□

There exists a zigzagmorphism (a totally functional bisimulation) from $\mathcal{M}_s^{\nabla}$ to $\mathcal{M}$, sending any $(s_0, \sigma) \in M_s^{\nabla}$ to s_0. Thus, if $\mathcal{M}$ is m-saturated then so is $\mathcal{M}_s^{\nabla}$ (use fact 4.5).

$\mathcal{M}_s^{\nabla}$ has many useful properties. To express them, we need to define, for any model $\mathcal{M}$ the binary relation $\rightarrow_{\mathcal{M}}$:

$$\rightarrow_{\mathcal{M}} := \{(s,t) \in \mathcal{M}^2 \mid \exists R \in \mathcal{L}. \exists s_1 \ldots s_k. (t \in \{s_1, \ldots, s_k\} \wedge (s, s_1, \ldots, s_k) \in R^{\mathcal{M}})\}.$$

We call a model $\mathcal{M}$ **rooted in** s (where $s \in \mathcal{M}$) if every $t \in \mathcal{M}$ is reachable from s via $\rightarrow_{\mathcal{M}}$: $s \rightarrow_{\mathcal{M}}^* t$. A model $\mathcal{M}$ is called **disjointly rooted in** $s_1 \ldots s_n$ (where $s_1 \ldots s_n$ is a nonempty sequence of distinct elements from $\mathcal{M}$) if for every $t \in \mathcal{M}$ there is a *unique* s_i in the sequence such that $s_i \rightarrow_{\mathcal{M}}^* t$.

We call a model $\mathcal{M}$ **unravelled** if it has the following properties:

1. $\rightarrow_{\mathcal{M}}$ is well-founded: there is no infinite sequence $s_0, s_1, s_2, \ldots$ in $\mathcal{M}$ with $s_{n+1} \rightarrow_{\mathcal{M}} s_n$ for all $n \in \mathbb{N}$.
2. If $(s_0, s_1, \ldots, s_k) \in R^{\mathcal{M}}$ then all the s_i are distinct.
3. If $t \in \mathcal{M}$ then there is at most one sequence $R, s_0, s_1, \ldots, s_k$ such that $(s_0, s_1, \ldots, s_k) \in R^{\mathcal{M}}$ and $t \in \{s_1, \ldots, s_k\}$.

Our model $\mathcal{M}_s^{\nabla}$ has all of these properties: it is rooted in $(s, [])$ and it is unravelled.

For present purposes, we need more than unravelling. So we present another model construction: *multiplying*, due to (Andréka et al. 1994).

Definition 4.7 *Let $\mathcal{M}$ again be a model and let $s \in M$, M being the domain of $\mathcal{M}$. Define the model $\mathcal{M}_s^{\triangle}$ as follows.*

- *$M_s^{\triangle}$, the domain of $\mathcal{M}_s^{\triangle}$ is a subset of $M \times \mathbb{N}^*$ defined inductively:*
  - *$(s, \lambda) \in M_s^{\triangle}$, where λ is again the empty sequence.*
  - *If $(s_0, \sigma) \in M_s^{\triangle}$ and $(s_0, s_1, \ldots, s_k) \in R^{\mathcal{M}}$ then $(s_i, \sigma n) \in M_s^{\triangle}$ for each $i \in \{1, \ldots, k\}$ and every $n \in \mathbb{N}$.*
- *We define the interpretation of R in $\mathcal{M}_s^{\triangle}$, $R^{\triangle}$:*

$$((s_0, \sigma_0), (s_1, \sigma_1), \ldots, (s_k, \sigma_k)) \in R^{\triangle}$$

iff $(s_0, s_1, \ldots, s_k) \in R^{\mathcal{M}}$ *and there is an* $n \in \mathbb{N}$ *with* $\sigma_i = \sigma_0 n$ *for each* $i \in \{1, \ldots, k\}$.

□

The map $(t, \sigma) \mapsto t$ defines a zigzagmorphism from $\mathcal{M}_s^{\triangle}$ to $\mathcal{M}$. This implies that m-saturation transfers from $\mathcal{M}$ to $\mathcal{M}_s^{\triangle}$. Furthermore, if $\mathcal{M}$ is unravelled, then $\mathcal{M}_s^{\triangle}$ will also be unravelled. Finally, $\mathcal{M}_s^{\triangle}$ is rooted in (s, λ).

We did of course not define this latest operation just for things to stay the same: the operation makes $\mathcal{M}$ **multiplied**, which means that whenever $(s_0, s_1, \ldots, s_k) \in R^{\mathcal{M}}$ (for $k \geq 1$) then there are infinitely many disjoint sequences $t_1 \ldots t_k$ with $s_1 \leftrightarrow t_1$, ..., $s_k \leftrightarrow t_k$ and $(s_0, t_1, \ldots, t_k) \in R^{\mathcal{M}}$.

We may thus define an operation $(_)_s^{\circ}$ as follows: $\mathcal{M}_s^{\circ} := (\mathcal{M}_s^{\nabla})_{(s,[])}^{\triangle}$. $\mathcal{M}_s^{\circ}$ is always unravelled, multiplied and rooted in $((s, []), \lambda)$. Furthermore, there is a

zigzagmorphism Z from $\mathcal{M}_s^{\circ}$ to $\mathcal{M}$ connecting $((s, []), \lambda)$ to s. So if $\mathcal{M}$ is m-saturated, then $\mathcal{M}_s^{\circ}$ will also be m-saturated.

One further operation will be of use in our proof: the familiar construction of *disjoint union*. This operation does nothing but put a number of models alongside each other in a single model.

Definition 4.8 *(Disjoint Union) Let $\mathcal{M}_1, \ldots, \mathcal{M}_n$ be models of the same relational signature $\mathcal{L}$, with $n \geq 1$. Then $\mathcal{M}_1 \oplus \ldots \oplus \mathcal{M}_n$ is their disjoint union, defined as follows:*

- *Domain: $\bigcup_{1 \leq i \leq n} (\{i\} \times \mathcal{M}_i)$.*
- *A $k+1$-ary R is interpreted as:*

$$\{((i, s_0), \ldots, (i, s_k)) \mid 1 \leq i \leq n \text{ and } (s_0, \ldots, s_k) \in R^{\mathcal{M}_i}\}$$

□

4.3 Descriptions

Certain sets of first order formulas will play a major role in the main theorem: we call these (I,O)-*descriptions*, where I and O are finite, disjoint sets of variables. (I,O)-descriptions are (infinitarily defined) relations, where I signifies the input and O the output. What an (I,O)-description describes is a *path* from I-points to O-points: it tells us how I and O are connected. We give a definition in this section and prove that certain finite descriptions are equivalent to $\text{PROG}_{(n,m)}$-formulas.

Definition 4.9 *A set of $\mathcal{L}$-formulas is an (I,O)-description if it is produced by the following rules:*

- *$\emptyset$ is an $(I,\emptyset)$-description, for any finite set of variables I.*
- *If Φ is an (I,O)-description, z is a variable in $I \cup O$ and Ψ is a set of (translations of) modal formulas then $\Phi \cup \Psi(z)$ is an (I,O)-description.*
- *If Φ is an (I,O)-description, $z \in I \cup O$, R is an $k+1$-ary relation symbol of our language $\mathcal{L}$ and $z_1, \ldots, z_k$ are fresh, distinct variables (i.e. not among $I \cup O$) then $\Phi \cup \{R(z, z_1, \ldots, z_k)\}$ is an $(I, O \cup \{z_1, \ldots, z_k\})$-description.* □

Lemma 4.10 *If Φ is a finite $(\{x_1, \ldots, x_n\}, \{y_1, \ldots, y_m\})$-description then $\bigwedge \Phi$ is equivalent to a $\text{PROG}_{(n,m)}$-formula.*

Proof.
We prove the lemma by induction on the construction of descriptions:

- $\bigwedge \emptyset \equiv \top$ is equivalent to DEL_n.
- Suppose Φ is produced by the second rule from Γ (also a finite $(\{x_1, \ldots, x_n\}, \{y_1, \ldots, y_m\})$-description) and $\Psi(z)$ (Ψ some finite set of modal formulas and z occurring among the x_i or y_j). By induction, $\bigwedge \Gamma$ is equivalent to some $\theta \in \text{PROG}_{(n,m)}$. Also $\bigwedge \Psi(z)$ is equivalent to $\psi(z)$, where ψ is some modal formula, as the set of modal formulas is closed under conjunction. So $\bigwedge \Phi$ is equivalent to $\theta \wedge \psi(z)$ There are two cases to distinguish:
  1. $z = x_i$ for some $i \in \{1, \ldots, n\}$. Then $\theta \wedge \psi(z)$ is equivalent to

$$(\text{ID}_{i-1} \bullet \psi? \bullet \text{ID}_{n-i}); \theta.$$

  where by $\psi?$ we really mean $(\psi?)^1$ (see page 8).
  2. $z = y_j$ for some $j \in \{1, \ldots, m\}$. Then $\theta \wedge \psi(z)$ is equivalent to

$$\theta; (\text{ID}_{j-1} \bullet \psi? \bullet \text{ID}_{m-j}).$$

- Suppose Φ is produced by the third rule from some finite (I, O')-description Γ and an atomic formula $R(z, z_1, \ldots, z_k)$. Clearly $O' = \{y_1, \ldots, y_m\} - \{z_1, \ldots, z_k\}$. Order O' according to indices: for $y_i, y_j \in O'$, $y_i < y_j$ if $i < j$. This gives us that $O' = \{u_1, \ldots, u_{m-k}\}$ with $u_1 < \ldots < u_{m-k}$. Define

$$\Gamma' := \Gamma[u_1 := y_1, \ldots, u_{m-k} := y_{m-k}].$$

Γ' is an $(\{x_1, \ldots, x_n\}, \{y_1, \ldots, y_{m-k}\})$-description constructed in the same number of steps as Γ. By the induction hypothesis we know that $\bigwedge \Gamma'$ is equivalent to a $\text{PROG}_{(n,m-k)}$-formula θ. So we have to find a $\text{PROG}_{(n,m)}$-formula equivalent to $\theta[y_1 := u_1, \ldots, y_{m-k} := u_{m-k}] \wedge R(z, z_1, \ldots, z_k)$. There are again two cases to consider:

1. $z = x_i$ for some i among $1, \ldots, n$. First construct the function $f : \{1, \ldots, m\} \to \{1, \ldots, m\}$:

$$f(j) := \begin{cases} l & \text{if } y_j = u_l \text{ for some } 1 \leq l \leq m-k; \\ l+m-k & \text{if } y_j = z_l \text{ for some } 1 \leq l \leq k. \end{cases}$$

The desired formula is then

$$(\text{ID}_{i-1} \bullet (\Delta; (\text{ID} \bullet R)) \bullet \text{ID}_{n-i}); (\text{ID}_i \bullet \text{PERMUTE}(k, n-i)); (\theta \bullet \text{ID}_k); [f].$$

2. $z = u_i$ for some $i \in \{1, \ldots, m-k\}$. Now the desired formula is:

$$\theta; (\text{ID}_{i-1} \bullet (\Delta; (\text{ID} \bullet R)) \bullet \text{ID}_{m-(k+i)}); (\text{ID}_i \bullet \text{PERMUTE}(k, m-(k+i))); [f]$$

where f is the same as in the first case. □

4.4 Proof of the main theorem: the zero case

If we examine the definition of (n, m)-safety again, we notice that $n = 0$ is a special case: if $n = 0$ the condition on a bisimulation Z that there are $s_1, \ldots, s_n \in \mathcal{M}$ and $t_1, \ldots t_n \in \mathcal{N}$ such that Z connects these points respectively is vacuous. Let us try to prove the main theorem for this simple case.

We will prove that any $(0, m)$-safe formula $\phi(y_1, \ldots, y_m)$ is either unsatisfiable or $m = 0$ and it is always satisfied.

For suppose $\phi(y_1, \ldots, y_m)$ is satisfiable, say $\mathcal{M} \models \phi[s_1, \ldots, s_m]$. Take any model $\mathcal{N}$ and let Z be the empty bisimulation between $\mathcal{M}$ and $\mathcal{N}$. As ϕ is $(0, m)$-safe we must conclude that there are $t_1, \ldots, t_m \in \mathcal{N}$ with $\mathcal{N} \models \phi[t_1, \ldots, t_m]$ and $s_i Z t_i$ for $i = 1, \ldots, m$. Since Z is empty this can only be the case if $m = 0$ and ϕ is thus a *closed* formula, a sentence. The argument above shows that ϕ is satisfied in *any* model $\mathcal{N}$. Thus, if ϕ is satisfied *some*where then it is satisfied *every*where.

This in fact shows that $(0, m)$-safe formulas are equivalent to $\text{PROG}_{(0,m)}$-formulas. For an unsatisfiable $(0, m)$-safe formula $\phi(y_1, \ldots, y_m)$ is equivalent to $\bot; \text{COPY}_1^m$ (see the definition on page 7) and a tautological $(0, 0)$-safe formula is equivalent to $\sim \bot$.

We seem to have demonstrated a case where the empty bisimulation is actually of importance. This is not really so: we can easily rearrange the argument such that only nonempty bisimulations are considered.

4.5 Proof of the main theorem: the nonzero case

Suppose $\phi(x_1, \ldots, x_n, y_1, \ldots, y_m)$ is (n, m)-safe. Consider the subset of $\text{PROG}_{(n,m)}$ containing only those formulas that by themselves imply ϕ:

$$\Psi := \{\psi \in \text{PROG}_{(n,m)} \mid \psi \models \phi\}.$$

We will prove that $\phi \models \bigvee \Psi$. That is, we prove that ϕ implies the infinite disjunction of Ψ, that $\mathcal{M} \models \phi[\vec{s}, \vec{t}]$ implies that for *some* $\psi \in \Psi$ (not necessarily the same ψ each time), $\mathcal{M} \models \psi[\vec{s}, \vec{t}]$. So suppose that $\mathcal{M} \models \phi[\vec{s}, \vec{t}]$.

We may assume that $\mathcal{M}$ is m-saturated. For every model $\mathcal{M}$ has an ω-saturated (hence m-saturated) elementary extension $\mathcal{M}^*$ (Keisler 1961). By elementary extension, $\mathcal{M}^* \models \phi[\vec{s}, \vec{t}]$, as ϕ is a first order formula. If we could now prove that $\mathcal{M}^* \models \psi[\vec{s}, \vec{t}]$ for some $\psi \in \Psi$ then, again by elementary extension, $\mathcal{M} \models \psi[\vec{s}, \vec{t}]$ would also hold, so we would be done.

Furthermore, we may assume the following (see subsection 4.2 for the relevant definitions):

- $\mathcal{M}$ is unravelled;
- $\mathcal{M}$ is multiplied and
- $\mathcal{M}$ is disjointly rooted in $s_1 \ldots s_n$.

For consider the model $\mathcal{M}^+ := \mathcal{M}^\circ_{s_1} \oplus \ldots \oplus \mathcal{M}^\circ_{s_n}$ (see again subsection 4.2). This model clearly has the desired properties, with respect to the points $u_i := (i, ((s_i, []), \lambda))$. We know that we have a zigzagmorphism Z_i from $\mathcal{M}^\circ_{s_i}$ to $\mathcal{M}$ for each i. Thus $Z : (i, a) \mapsto Z_i(a)$ is a zigzagmorphism from $\mathcal{M}^+$ to $\mathcal{M}$. This Z connects u_i to s_i. As ϕ is assumed (n, m)-safe, it must be the case that there are $v_1, \ldots, v_m$ in $\mathcal{M}^+$ such that $Z(v_j) = t_j$ for each $1 \leq j \leq m$ and $\mathcal{M}^+ \models \phi[\vec{u}, \vec{v}]$. If we could now prove that $\mathcal{M}^+ \models \psi[\vec{u}, \vec{v}]$, for some $\psi \in \Psi$, then by the (n, m)-safety of ψ and the functionality of Z, $\mathcal{M} \models \psi[\vec{s}, \vec{t}]$ should also hold, so we would be done. Note that as Z is a zigzagmorphism from $\mathcal{M}^+$ to $\mathcal{M}$, $\mathcal{M}^+$ must be m-saturated ($\mathcal{M}$ is m-saturated, so just use fact 4.5): we can thus combine the assumptions from this and the previous paragraph without restraint. Note that the argument used here would not work for the case where $n = 0$, as then $\mathcal{M}^+$ would have an empty universe. This is why we needed to deal with the zero case separately.

Define PATH to be the set of all points in $\mathcal{M}$ (now assumed to satisfy each of the properties described in the previous two paragraphs) of relevance to the path from each s_i to $t_1, \ldots, t_m$. Formally, define PATH to be the smallest subset of the domain of $\mathcal{M}$ such that:

- $\{s_1, \ldots, s_n, t_1, \ldots, t_m\} \subseteq \text{PATH}$.
- If $v \in \text{PATH}$, $R \in \mathcal{L}$, $(w_0, w_1, \ldots, w_k) \in R^{\mathcal{M}}$ and $v \in \{w_1, \ldots, w_k\}$ then $\{w_0, w_1, \ldots, w_k\} \subseteq \text{PATH}$.

For each $w \in \mathcal{M} - \{s_1, \ldots, s_n\}$ there is a unique choice of the sequence $R, w_0, w_1, \ldots, w_k$ such that $(w_0, w_1, \ldots, w_k) \in R^{\mathcal{M}}$ and $w \in \{w_1, \ldots, w_k\}$. This same choice is the unique one for each of the w_i (with $1 \leq i \leq k$). This is because $\mathcal{M}$ is assumed to be unravelled. PATH is arrived at by starting at $t_1, \ldots, t_m$ and going downwards as described and adding all the points encountered. As $\mathcal{M}$ is disjointly rooted in $s_1 \ldots s_n$, this will end at one of the s_i. PATH must therefore be a *finite* set.

We assign variables to points in PATH. The roots $s_1, \ldots, s_n$ are assigned the variables $x_1, \ldots, x_n$ respectively. To each point in $\text{PATH} - \{s_1, \ldots, s_n\}$ we assign a unique variable from $\{y_1, \ldots, y_k\}$, where $k = |\text{PATH} - \{s_1, \ldots, s_n\}| = |\text{PATH}| - n$.

We build a description for PATH in steps:

Step 1: Begin with $\Phi_{s_1}(x_1) \cup \ldots \cup \Phi_{s_n}(x_n)$ where Φ_{s_i} is the set of all modal formulas true at s_i. We consider the points s_i 'treated'.

Step 2: Suppose that w_0 has been treated and that Φ is the description built so far. If there are untreated $w_1, \ldots, w_k \in \text{PATH}$ ($k \leq 1$) such that $(w_0, w_1, \ldots, w_k) \in R^{\mathcal{M}}$, let the new description be:

$$\Phi \cup \{R(z_0, z_1, \ldots, z_k)\} \cup \Phi_{w_1}(z_1) \cup \ldots \cup \Phi_{w_k}(z_k)$$

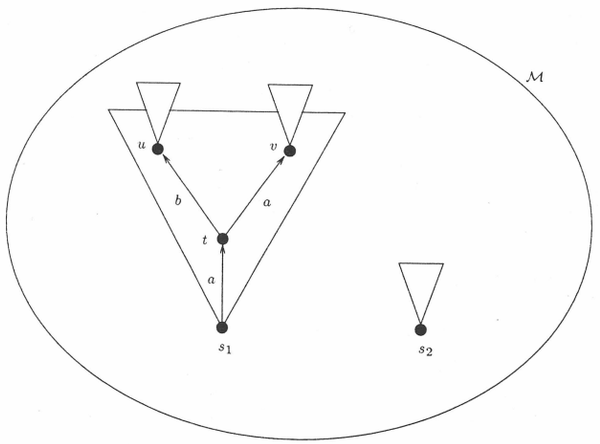


Figure 1: The PATH in $\mathcal{M}$ from s_1, s_2 to u, v.

where z_i is the variable assigned to w_i ($i \leq k$) and Φ_{w_i} is again the set of all modal formulas true at w_i. At this point, consider $w_1, \ldots, w_k$ treated. Repeat this step until there are no more untreated points in PATH.
Note that because that we are in an unravelled model, all the z_i are distinct. Furthermore, they do not occur freely in Φ, giving us a real description, as defined in subsection 4.3.

This process gives us an $(\{x_1, \ldots x_n\}, \{y_1, \ldots, y_k\})$-description Φ. Let $z_1, \ldots, z_m$ be the variables assigned to $t_1, \ldots, t_m$ respectively. We will prove that: $\Phi \models \phi(x_1, \ldots, x_n, z_1, \ldots, z_m)$. But first, let us consider an

Example:
Assume $n = 2$, $m = 4$ and consider a $(2,4)$-safe formula $\phi(x_1, x_2, y_1, y_2, y_3, y_4)$. Suppose $\mathcal{M} \models \phi[s_1, s_2, s_2, u, v, u]$, where $\mathcal{M}$ is as in figure 4.5. An arrow $\xrightarrow{a}$ in this figure corresponds to an R_a-step in $\mathcal{M}$, where R_a is some binary relation symbol in $\mathcal{L}$. Likewise for the $\xrightarrow{b}$-arrow in the diagram.

We want to describe the path from s_1, s_2 to u, v. First of all, PATH $= \{s_1, s_2, t, u, v\}$. Note that we have added the intermediate point t. Let us assign the variables x_1, x_2, y_1, y_2, y_3 to s_1, s_2, t, u, v respectively. The corresponding description Φ is:

$$\Phi_{s_1}(x_1) \cup \Phi_{s_2}(x_2) \cup \Phi_t(y_1) \cup \Phi_u(y_2) \cup \Phi_v(y_2)$$
$$\{R_a(x_1, y_1), R_b(y_1, y_2), R_a(y_1, y_3)\}$$

At this point in the proof, for this concrete example, we are trying to show that $\Phi \models \phi(x_1, x_2, x_2, y_2, y_3, y_2)$.
End of Example.

Back to the general case. Suppose $\mathcal{N} \models \Phi[\sigma]$ for some assignment to variables σ. Again we may assume that $\mathcal{N}$ is m-saturated. What's more, we may also assume that

- $\mathcal{N}$ is unravelled and multiplied.
- $\mathcal{N}$ is disjointly rooted in $\sigma(x_1) \ldots \sigma(x_n)$.
- σ is injective on the *relevant* variables (i.e. $x_1, \ldots, x_n, y_1, \ldots, y_k$).

To see this, consider $\mathcal{N}^+ := \mathcal{N}^\diamond_{\sigma(x_1)} \oplus \ldots \oplus \mathcal{N}^\diamond_{\sigma(x_n)}$. Clearly, this model is unravelled and multiplied. Furthermore, it may still be assumed m-saturated, as we have a zigzagmorphism Z from $\mathcal{N}^+$ to $\mathcal{N}$. We define an assignment τ such that the second and third requirements above are satisfied for τ and $\mathcal{N}^+$ and furthermore $\mathcal{N}^+ \models \Phi[\tau]$. We choose τ in such a way that $\sigma(z)$ and $\tau(z)$ are connected by Z, hence bisimilar, for any relevant z: this will ensure that they satisfy the same modal formulas. We build τ in steps:

Step 1: $\tau(x_i) := (i, ((\sigma(x_i), []), \lambda))$. Recall that $((\sigma(x_i), []), \lambda)$ is the root of $\mathcal{N}^\diamond_{\sigma(x_i)}$. The zigzagmorphism Z then connects $\tau(x_i)$ to $\sigma(x_i)$, so these are bisimilar.

Step 2: Suppose we have already assigned the value b_0 to v_0, where v_0 is the variable associated with $a_0 \in$ PATH and suppose there are $a_1, \ldots, a_l \in$ PATH with $(a_0, a_1, \ldots, a_l) \in R^{\mathcal{M}}$ whose associated variables $v_1, \ldots, v_l$ have not been assigned values by τ yet.
By definition, $R(v_0, v_1, \ldots, v_l) \in \Phi$, so $(\sigma(v_0), \sigma(v_1), \ldots, \sigma(v_l)) \in R^{\mathcal{N}}$. As $\sigma(v_0)$ and $\tau(v_0) = b_0$ are bisimilar there are $b_1, \ldots, b_l$ such that$(b_0, b_1, \ldots, b_l) \in R^{\mathcal{N}^+}$ with each b_i bisimilar to $\sigma(v_i)$. As $\mathcal{N}^+$ is multiplied, we must be able to choose $b_1, \ldots, b_l$ such that they do not not occur as values of τ yet. Define $\tau(v_i) := b_i$.
Repeat this step until τ assigns a value to every relevant variable.

If we could now prove $\mathcal{N}^+ \models \phi(\vec{x}, \vec{z})[\tau]$, i.e.

$$\mathcal{N}^+ \models \phi[\tau(x_1), \ldots, \tau(x_n), \tau(z_1), \ldots, \tau(z_m)]$$

then, because ϕ is (n,m)-safe, Z is functional and sends $\tau(z)$ to $\sigma(z)$, for all relevant variables z, we may conclude that $\mathcal{N} \models \phi[\sigma(x_1), \ldots, \sigma(x_n), \sigma(z_1), \ldots, \sigma(z_m)]$, hence $\mathcal{N} \models \phi(\vec{x}, \vec{z})[\sigma]$ as desired.

So we have an m-saturated, unravelled and multiplied model $\mathcal{M}$, disjointly rooted in $s_1, \ldots, s_n$. We have another m-saturated, unravelled and multiplied model $\mathcal{N}$, disjointly rooted in $\sigma(x_1), \ldots, \sigma(x_n)$. By fact 4.3 the relation Z of modal equivalence is a bisimulation between $\mathcal{M}$ and $\mathcal{N}$.

We want to prove that $\mathcal{N} \models \phi[\sigma(x_1), \ldots, \sigma(x_n), \sigma(z_1), \ldots, \sigma(z_m)]$. As $\mathcal{N} \models \Phi[\sigma]$, Z connects $s_1, \ldots, s_n$ to $\sigma(x_1), \ldots, \sigma(x_n)$ respectively. As ϕ is (n,m)-safe and $\mathcal{M} \models \phi[\vec{s}, \vec{t}]$ there must be $b_1, \ldots, b_m$ in $\mathcal{N}$ such that $t_j Z b_j$ for each $1 \leq j \leq m$ and $\mathcal{N} \models \phi[\sigma(x_1), \ldots, \sigma(x_n), \vec{b}]$. There is however no guarantee that $b_j = \sigma(z_j)$ for each $1 \leq j \leq m$. To achieve this, we must first restrict Z somewhat.
Consider the finite set:

$$L := \{(a, \sigma(z)) \mid a \in \text{PATH and } z \text{ is the variable associated with } a\}$$

By definition of Φ, $L \subseteq Z$, that is: L-connected points are modally equivalent. We will prove that

$$Z' := \{(a,b) \in Z \mid a \notin dom(L) \text{ and } b \notin rng(L)\} \cup L$$

is still a bisimulation between $\mathcal{M}$ and $\mathcal{N}$. $dom(L)$ is here intended to be the *domain* of the relation L (i.e. PATH), $rng(L)$ its range.

Zig: Suppose $aZ'b$ and $(a, a_1, \ldots, a_l) \in R^{\mathcal{M}}$, for some $R \in \mathcal{L}$. There are two cases:

- One of the a_i is in PATH. Then all of $a_1, \ldots a_l$ must be, as well as a. So $aLb = \sigma(v)$ (where v is the variable associated with a). If $v_1, \ldots, v_l$ are the variables assigned to $a_1, \ldots, a_l$ respectively then $R(v, v_1, \ldots, v_k) \in \Phi$ and thus $(\sigma(v), \sigma(v_1), \ldots, \sigma(v_l)) \in R^{\mathcal{N}}$. As $a_i L \sigma(v_i)$ we are done.
- Suppose none of the a_i occur in PATH. As $Z' \subseteq Z$, there must be $b_1, \ldots, b_l \in \mathcal{N}$ such that $a_i Z b_i$ (for i among $1, \ldots, l$) and $(b, b_1, \ldots, b_l) \in R^{\mathcal{N}}$. As $\mathcal{N}$ is

multiplied and $rng(L)$ is finite, we must be able to find $c_1, \ldots, c_l$, none of which occur in $rng(L)$, with $(b, c_1, \ldots, c_l) \in R^{\mathcal{N}}$ such that each b_i is bisimilar to c_i. As Z is the maximal bisimulation, $a_i Z c_i$ is still the case: these points will be modally equivalent. Furthermore, we have ensured that $a_i Z' c_i$.

Zag: For the other direction, assume $aZ'b$ and $(b, b_1, \ldots, b_l) \in R^{\mathcal{N}}$ for some $R \in \mathcal{L}$. Again, we have two cases:

- One of the b_i is in $rng(L)$. Then $b_i = \sigma(v)$ for v associated with some $c \in$ PATH. c cannot be one of the roots s_j, as then $v = x_j$ and so $\sigma(v) = b_i$ would also be one of the roots, which gives us a contradiction with the fact that b_i has a predecessor b and that $\mathcal{N}$ is unravelled. Thus, by the fact that $\mathcal{M}$ is unravelled, there must be a sequence $S, c_0, c_1, \ldots, c_p$ with $(c_0, c_1, \ldots, c_p) \in S^{\mathcal{M}}$ and $c \in \{c_1, \ldots, c_p\}$. As $c \in$ PATH, $c_0, c_1, \ldots, c_p$ must also be in PATH, say the variable v_j is assigned to c_j (for $j \leq p$). Then $S(v_0, v_1, \ldots, v_p) \in \Phi$, hence $(\sigma(v_0), \sigma(v_1), \ldots, \sigma(v_p)) \in S^{\mathcal{N}}$. As $c \in \{c_1, \ldots, c_p\}$, and v is assigned to c, v must occur among $v_1, \ldots, v_p$. Thus $\sigma(v) = b_i \in \{\sigma(v_1), \ldots, \sigma(v_p)\}$. As $\mathcal{N}$ is unravelled: $S = R$, $l = p$, $\sigma(v_0) = b$ and $\sigma(v_1) = b_1, \ldots, \sigma(v_l) = b_l$. b is therefore in the range of L. We assumed $aZ'b$, so aLb must be the case. This implies that $b = \sigma(z)$, where z is the variable assigned to a. But $b = \sigma(v_0)$ was stated earlier, so by the injectivity of σ: $z = v_0$, hence $a = c_0$. So we have proved that there are $c_1, \ldots, c_l \in$ PATH with $(a, c_1, \ldots, c_l) \in R^{\mathcal{M}}$ and $c_1 L b_1$, $\ldots$, $c_l L b_l$.
- The other case is entirely similar to the second case of the zig-part of this proof.

We can now prove $\mathcal{N} \models \phi[\sigma(x_1), \ldots, \sigma(x_n), \sigma(z_1), \ldots, \sigma(z_m)]$ as follows. $\mathcal{M} \models \phi[\vec{s}, \vec{t}]$ is given. For each $1 \leq i \leq n$: $s_i L \sigma(x_i)$ as x_i is the variable assigned to s_i. As $L \subseteq Z'$ and the latter is a bisimulation, there must be $b_1, \ldots, b_m$ in $\mathcal{N}$ such that $t_j Z' b_j$ for each $1 \leq j \leq m$ and $\mathcal{N} \models \phi[\sigma(x_1), \ldots, \sigma(x_n), \vec{b}]$. Since $t_1, \ldots, t_m \in$ PATH, t_j is actually connected to b_j via L. Hence $b_j = \sigma(z_j)$, as z_j is the variable associated with t_j.

So we have shown: $\Phi \models \phi(\vec{x}, \vec{z})$. By compactness, there is a finite $(\{x_1, \ldots, x_n\}, \{y_1, \ldots, y_k\})$-description $\Phi_0 \subseteq \Phi$ such that $\bigwedge \Phi_0 \models \phi(\vec{x}, \vec{z})$. By lemma 4.10, $\bigwedge \Phi_0$ is equivalent to a $\text{PROG}_{(n,k)}$-formula θ. Define a function $f : \{1, \ldots, m\} \to \{1, \ldots, n+k\}$ as

$$f(i) := \begin{cases} j & \text{if } z_i = x_j; \\ j+n & \text{if } z_i = y_j. \end{cases}$$

Then

$$\exists x_{n+1} \ldots x_{n+m}.(\theta(x_1, \ldots, x_{n+m}) \wedge y_1 = x_{f(1)} \wedge \ldots \wedge y_m = x_{f(m)}) \models \phi$$

But the formula before the $\models$ is equivalent to the $\text{PROG}_{(n,m)}$-formula $\psi := \text{COPY}_n; (\text{ID}_n \bullet \theta); [f]$. So we have found a $\text{PROG}_{(n,m)}$-formula ψ with $\psi \models \phi$ (i.e. $\psi \in \Psi$). By construction of Φ, $\mathcal{M} \models \Phi[s_1, \ldots, s_n, u_1, \ldots, u_k]$ for some $u_1, \ldots, u_k \in \mathcal{M}$ (simply take as u_i the element of PATH that y_i is assigned to) and hence $\mathcal{M} \models \psi[\vec{s}, \vec{t}]$. We have therefore achieved our aim: we have proved that $\phi \models \bigvee \Psi$.

Example:
Let us consider our example of page 14 again. For that specific example we have now proved that $\Phi \models \phi(x_1, x_2, x_2, y_2, y_3, y_2)$. By compactness, the infinite sets of modal formulas that occur in this Φ can be replaced by single modal formulas. Thus there are modal formulas $\phi_{s_1}, \phi_{s_2}, \psi_t, \psi_u, \psi_v$ such that:

$$\left(\begin{array}{c} \phi_{s_1}(x_1) \wedge \phi_{s_2}(x_2) \wedge \psi_t(y_1) \wedge \psi_u(y_2) \wedge \psi_v(y_3) \wedge \\ R_a(x_1, y_1) \wedge R_b(y_1, y_2) \wedge R_a(y_1, y_3) \end{array} \right) \models \phi(x_1, x_2, x_2, y_2, y_3, y_2)$$

But then:

$$\exists z. \left(\begin{array}{c} \phi_{s_1}(x_1) \wedge \phi_{s_2}(x_2) \wedge \psi_t(z) \wedge \psi_u(y_2) \wedge \psi_v(y_3) \wedge \\ R_a(x_1, y_1) \wedge R_b(y_1, y_2) \wedge R_a(y_1, y_3) \wedge y_1 = x_2 \wedge y_4 = y_2 \end{array} \right) \models \phi$$

and the formula on the left of the latter turnstyle is equivalent to the $\text{PROG}_{(n,m)}$-formula:

$$(\phi_{s_1}? \bullet \phi_{s_2}?); \pi; (\text{ID} \bullet [R_a; \phi_t?; \Delta; (R_b \bullet R_a); (\phi_u? \bullet \phi_v?); (\Delta \bullet \text{ID}); (\text{ID} \bullet \pi)])$$

End of Example.

By compactness again, we deduce that $\phi \models \psi_1 \vee \ldots \vee \psi_l$, where each $\psi_i \in \Psi$. This means that each ψ_i is a $\text{PROG}_{(n,m)}$-formula that implies ϕ. Thus ϕ is really equivalent to $\psi_1 \cup \ldots \cup \psi_l$, where we define the empty union of $\text{PROG}_{(n,m)}$-formulas as the inconsistent $\text{DEL}_n; \bot; \text{COPY}_1^m$. □

5 Conclusions and Further Research

We have proved that being (n, m)-safe for bisimulation is the same as being equivalent to a $\text{PROG}_{(n,m)}$-formula.

There are a few ways of looking at this result. It could be seen as just a generalisation of the well-known invariance theorem, placing this theorem within a certain landscape of similar theorems. But we could also value the languages $\text{PROG}_{(n,m)}$ themselves, as revealing the *fine-structure* of our modal languages. $\text{PROG}_{(n,m)}$-formulas could then be viewed as living *inside* modal formulas. They have been hidden until now, but theorem 4.1 reveals them.

Future research could consist of generalising the result along several lines. First, we could extend the theorem by considering other languages beside the first order one, such as higher order languages or infinitary ones. The case of invariance in the infinitary case is dealt with in (Benthem 1993): invariance coincides precisely with the infinitary modal formulas, modal formulas where infinitary disjunctions and conjunctions are permitted. Maybe in a fragment of infinitary logic where only certain effective disjunctions are allowed, invariance under bisimulation corresponds exactly to PDL. Transitive closure logic (see (Ebbinghaus and Flum 1995)) comes to mind in this connection.

Also, a finite model version could be investigated, thereby extending the mentioned result for invariance in (Rosen 1995) and the one for safety in (Hollenberg 1995b). These theorems strongly suggest that in the finite case nothing changes: a first order formula is (n, m)-safe in all finite models iff it is equivalent (in the class of finite models) to a $\text{PROG}_{(n,m)}$-formula.

References

Andréka, H., Benthem, J.F.A.K. van , and Németi, I.: 1994, *Back and Forth between Modal Logic and Classical Logic*, To appear in *Bulletin of the Interest Group in Pure and Applied Logics*

Baeten, J. and Weijland, W.: 1994, *Process Algebra*, Vol. 18 of *Cambridge Tracts in Theoretical Computer Science*, Cambridge University Press, 1990

Barwise, J. and Moss, L.: 1996, *Vicious Circles: On the Mathematics of Non-Wellfounded Phenomena*, CSLI Publications, Stanford University, to appear

Benthem, J. v.: 1976, *Modal Correspondence Theory*, Ph.D. thesis, Mathematisch Instituut & Instituut voor Grondslagenonderzoek, University of Amsterdam

Benthem, J. v.: 1993, *Programming Operations that are Safe for Bisimulation*, CSLI Report 93-179, Center for the Study of Language and Information, Stanford University, to appear in *Studia Logica*

Benthem, J. v.: 1994, *A Modal Perspective on Process Operations*, draft in progress

Ebbinghaus, H. and Flum, J.: 1995, *Finite Model Theory*, Springer, Berlin

Fine, K.: 1975, Some connections between elementary and modal logic, in *Proceedings of the Third Scandinavian Logic Symposium, Uppsala 1973*, pp 15–31, Nort-Holland, Amsterdam

Harel, D.: 1984, Dynamic Logic, in D. Gabbay and F. Guenther (eds.), *Handbook of Philosophical Logic, Vol. II*, pp 497–604, Reidel, Dordrecht

Hollenberg, M.: 1995a, *Bisimulation Respecting First Order Operations*, unpublished manuscript, to appear in the Logic Group Preprint Series, Department of Philosophy, Utrecht University

Hollenberg, M.: 1995b, *Bisimulation Safety over Finite Models*, unpublished manuscript

Hollenberg, M.: 1995c, Hennessey-Milner Classes and Process Algebra, in M. d. R. A. Ponse and Y. Venema (eds.), *Modal Logic and Process Algebra: a Bisimulation Perspective*, Vol. 53 of *CSLI Lecture Notes*, pp 187–216, CSLI Publications

Keisler, H.: 1961, Ultraproducts and elementary classes, *Indagationes Mathematicae* 23, 477–495

Park, D.: 1981, Concurrency and Automata on Infinite Sequences, in *Proceedings 5th GI Conference*, pp 167–183, Springer

Quine, W.: 1966, Variables Explained Away, in *Selected Logical Papers*, Random House, New York

Rijke, M. d.: 1993, *Extending Modal Logic*, Ph.D. thesis, ILLC-dissertation series 1993-4

Rosen, E.: 1995, *Modal Logic over Finite Structures*, Report ML-95-08, Institute for Logic, Language and Computation, University of Amsterdam

Venema, Y.: 1991, *Many-Dimensional Modal Logic*, Ph.D. thesis, University of Amsterdam

Venema, Y.: 1992, *A Crash Course in Arrow Logic*, to appear in *Arrow Logic and Multimodal Logics*, M. Marx, Sz. Mikulás, I. Németi (eds.)

HOW FREGE FAILED TO PROVE THE EQUIVALENCE OF 'NOTHING IS *F*' AND 'THE NUMBER OF *F*'s = 0' IN *GRUNDLAGEN*

Antoon Hurkmans, Vrije Universiteit Amsterdam

When Michael Dummett worked on his first, groundbreaking book on Frege [1], he believed, so he told us later [2], that his interpretation of Frege's work would not be a matter for serious controversy; it never occurred to him that the mere exegesis of that work (as opposed to its evaluation) could in any way be problematic. It is not difficult to understand how Dummett came to work under that assumption: Frege states the objective of his project plainly, its execution is largely formal, and the informal bits are written in an admirably clear prose style. But, as we know, Dummett's optimistic assumption proved hopelessly wrong; his own reading of Frege was fundamentally challenged by various people offering what were often again mutually incompatible alternatives; and now, 20 odd years after he set the ball rolling, the prospect of a more or less agreed understanding of even the most basic tenets of Frege's work would seem to be further away than ever: Frege's legacy has, indeed, been turned into a "battleground for exegetes" [3]. This state of affairs is not just baffling, given the *apparent* clarity of Frege's purpose and writing; it is also very frustrating for those of us who labour in the philosophical fields that Frege helped to mark out. Few people would doubt Frege's seminal role in the development of various new disciplines - philosophical or otherwise - that have helped shape the intellectual world we now inhabit; and in philosophy, at least, it pays to know your ancestors well; so the present state of confusion in Frege exegesis means that, in an important sense, we do not really know where we are: we lack a clear perspective on how we got to our present position, and we cannot, if we are honest, feel at all confident that the key notions and distinctions which we inherited from Frege have the sense and import which (in our various, conflicting ways) we attribute to them.

Still, two decades of sometimes acrimonious exchanges between the main protagonists in the field of Frege exegesis have not been entirely in vain: there is now, it seems, a growing recognition that the probable root cause of our radical disagreements lies in the fact that there is what looks like a *crucial* gap in our knowledge of Frege's work. It concerns the following. We know that sometime between 1884, when *Grundlagen* [4] was published, and 1891, the year *Funktion und Begriff* [5] appeared, Frege had what amounted to a fundamental change of heart. Because the changes in his work after 1891 are, even at a superficial glance, so striking, it has long been commonplace to distinguish his early work, comprising *Begriffsschrift* [6] and *Grundlagen*, from his later work, which - apart from *Grundgesetze* [7] - contains the famous series of articles including *Über Sinn und Bedeutung* [8]. But - and this Dummett has now himself identified as "the principal problem of Frege exegesis" [9] - we don't *really* understand how the early and later work are related. We do not understand, in other words, why Frege suddenly embarked on what by his own admission was a radical overhaul of his logic and philosophy of logic.

Now it would seem, *at least on the face of it*, that Frege did not change his main objective: in his later work he is still in single-minded pursuit of a proof that arithmetic is but a further-developed logic. Also, in *Grundgesetze* he frequently refers back to *Grundlagen* in order to remind his readers of the structure of his argument and to explain the relevance of individual steps in it. This reliance on the exposition of his project in the earlier work strongly suggests that the changes he had introduced do not amount to a radical break with his own past, but rather represent a more or less natural progression in his thinking, perhaps reflecting a broadening of his interests into, say, the semantics of natural language or a desire to present his reasoning in a more systematic way. And so, many students of Frege - Dummett included - have been inclined to see the later work in that light. This is how Dummett formulates it:

> "The principal problem of Frege exegesis is to determine the relation between the writings of Frege's early period, up to 1886, and those of his middle period, beginning in 1891. During the years 1887-1890, he published nothing, but was engaged in thinking through afresh his system of philosophical logic and redesigning, in accordance with it, the formal system he had presented in *Begriffsschrift*." [10]

Dummett apparently believes that, during the 'silent' years of 1887-1890, Frege's thinking on 'philosophical logic' underwent a development that culminated in his famous sense-reference distinction [11], and that, in order to accommodate these new insights, Frege had to 'redesign' his formal system.

However, this view of the matter doesn't quite tally with the very few words Frege himself devotes to the subject. In the Foreword to *Grundgesetze*, Frege tries to explain why the completion of that work had taken him so much longer than anticipated. The explanation takes the form of a few brief, intriguing statements. He tells us that he had nearly completed his manuscript for the book (i.e. the book that was to be *Grundgesetze*), when he discovered that what he calls "internal transformations" in his logical system were called for; he calls these transformations "far-reaching"; he also acknowledges that they will make acceptance of his ideas even more difficult - (he had just been complaining about the reception of his earlier work); he says that he can fully understand any reluctance on the part of his readers to go along with these innovations, because he had to overcome such reluctance himself; for they were not, he assures us, introduced "for novelty's sake", but because they proved to be "necessary". [12]

These few remarks do not suggest that the novel features of his later work were in any way the result of a natural development in his thinking. Quite the contrary: they strongly suggest that Frege was suddenly and unexpectedly faced with an unavoidable, because *purely formal*, problem in the execution of his project. In any case, whatever the shortcomings of Frege's undifferentiated notion of a 'judgeable content' in *Begriffsschrift*, it is hard to believe that the sense-reference distinction which replaced it somehow *compelled* him to recast his formal system in the way he did: some major features of the system of *Grundgesetze* are, by common consent, so counter-intuitive that we can all understand why Frege should be 'reluctant' to introduce them; but do we really have any reason to believe that these features follow *inexorably* from the sense-reference distinction or, indeed, from any other considerations in philosophical logic ?

On the other hand, if Frege was suddenly *forced*, for purely *formal* reasons, to abandon his nearly completed draft of the work he'd announced as just around the corner in 1884, why did he not set out the reason(s) why he was forced to do so. For that is the intriguing thing about these statements in the Foreword to *Grundgesetze*: Frege never in fact told us what exactly it was that compelled him - at a moment when he had, by his own lights, nearly completed his task - to change tack completely. And that, I think, is a very strange omission on the part of someone who, complaining about the reception of his earlier work, must have been keen to win over 'reluctant' readers of the volume in hand.

I think I know now why Frege chose to remain silent on the matter. He simply didn't wish to broadcast the fact that he had discovered that the logic of *Begriffsschrift* was seriously defective, in fact, strictly speaking, formally inconsistent. Such a discovery, we can all agree, would have presented Frege with the most compelling of reasons to introduce "internal transformations" into his system. Although it is likely, as I will explain, that the discovery was not entirely unaided, the help came in the form of what can only be described as a 'fluke', and so Frege was under no obligation - moral or otherwise - to acknowledge it. The discovery of the inconsistency of *Begriffsschrift* was, unlike the later one concerning *Grundgesetze*, strictly speaking his own. It is also well to remember that Frege had spent a great deal of effort defending the superiority of his system of *Begriffsschrift* over the contemporary Boolean systems, so his discovery must have caused acute embarrassment. In these circumstances, it is understandable that Frege chose to remain quiet and to proceed simply with setting out a new system without detailed explanations of why some of its more startling features were necessary. The idea that his work would only *really* be studied in the wider philosophical community more than eighty years later, and that his silence would then spawn a veritable 'battle of interpreters' could hardly have occurred to him.

So *where*, specifically, did Frege discover these defects, and *how* did he discover them ? On the first question, there is, as it happens, an important indication of where the formal defects of *Begriffsschrift* finally showed up, where they came to a head. One naturally assumes that the system of *Begriffsschrift* was devised so as to enable Frege to establish, to give a formal proof of, the foundational thesis expounded in *Grundlagen*. So the original, discarded, manuscript of *Grundgesetze*, we may also assume, will have contained formal proofs for all the informal proof-sketches that Frege draws in *Grundlagen*. Now when we look at the *published* version, we do indeed find faithful formal reproductions of all of these proof-sketches - bar one. The exception concerns the fundamental proof, - required in any 'logicist' project worthy of the name -, of the equivalence of the two propositions 'Nothing is *F*' and 'The number of *F*'s = 0'. In par.75 of *Grundlagen*, Frege proposed to prove that equivalence as *following* from the proofs of two lemmas, namely the proposition "All empty concepts are equal (gleichzahlig)" and the proposition "No empty concept is equal to a non-empty one". But in pars 96-99 of *Grundgesetze*, Frege proves the equivalence *directly*: that is to say, he simply proves that "If nothing is *F*, then the number of *F*'s is 0", and vice versa. The two lemmas, the two propositions from which the proof of the equivalence was said to follow in *Grundlagen*, have completely disappeared. So there is no correspondence at all between the informal proof-sketch of the equivalence in *Grundlagen*, and the formal proof we find in the *Grundgesetze* Frege finally published in 1893. Their structures are entirely different. [13]

Interestingly, the new proof-structure in *Grundgesetze* is a great deal more complicated than the one sketched in *Grundlagen*. So it cannot have been the case that the choice of the new one was prompted by considerations of economy or simplicity. But much more importantly, in fact crucially, the proof in *Grundgesetze* depends, *for its basic premiss*, on the very definition of 0 as a set (or, in Frege's parlance, a value-range). And Frege was, of course, aware that such a procedure is not a trifling matter. As he wrote in a passage on definitions in the *Nachlass*:

> "In the case in which what presents itself as a definition is really that which makes the truth of a proof possible, we do not have a pure definition, but there must be contained in it something which should either be proved as a theorem or acknowledged as an axiom." [14]

So the proof of the equivalence in *Grundgesetze* now comes to *depend* on the soundness of Axiom V, the very axiom about which, Frege tells us in the Foreword to *Grundgesetze* [15], he retained a residual doubt.

I think we have hit here on the *precise* reason why Frege had to recast his logic as a logic of sets: without the use, as a basic premiss, of the definition of 0 as a set, the equivalence of 'Nothing is *F*' and 'The number of *F*'s = 0' cannot be proved. When Frege discovered that his reasoning in the original proof of the equivalence in par. 75 of *Grundlagen* was inconsistent, he really had no choice in the matter. Let's now then take a closer look at that paragraph.

As regards the proof of the first lemma, that would seem to follow straightforwardly from the stipulations in par.71, as Frege himself makes clear. But I want to draw attention to a few interesting features of this par.71. Why, we should ask ourselves, does Frege think it at all necessary to give a *separate* definition of 'correlation', using italic letter *a*, for the special case where what used to be called the 'subject-concept' is empty ? After all, italic letters were introduced by Frege in par. 11 of *Begriffsschrift* as mere 'abbreviations' for variables tacitly bound by an initial universal quantifier. And so the correlation of *F*'s to *G*'s is trivially established in the case the concept *F* is empty, whether we present the proposition in terms of italics or of bound variables. A clue to what I think is at least *part* of the answer is Frege's use of modal language in his special definition: "..cannot, whatever be signified by *a*, both be true together" and "... must always be denied, whatever *a* may be". So let's keep that in mind. But despite the modal aspect of Frege's explication, we don't find it difficult to see that when he says that the two propositions "cannot both be true together", the incompatibility we are dealing with is the one involved in Frege's own definition of the material implication of two propositions A and B: namely the incompatibility of A and not-B. So, in this proof Frege simply relies on the principle that the falsity of the antecedent - here "*a* falls under *F*" and "*a* falls under *G*", respectively - is sufficient ground for the truth of the material implications

(1) $\vdash Fa \supset \exists x(Gx \,\&\, \Phi(a,x))$

and

(2) $\vdash Ga \supset \exists x(Fx \,\&\, \Phi(x,a))$ [16]

Turning now to the proof of the second lemma: Frege there relies on the truth of the proposition: "If there exists no object falling under *F*, then neither will there be such an object which stands to *a* in any relation whatsoever." Now, if we transcribe the latter part of this sentence into formula-language, it would *seem* to constitute the judgment

(3) $\vdash \forall x(Fx \supset \sim f(x,a))$

And this, if the concept indicated by capital F in this formula is empty, is indeed a trivial second-order truth. However, Frege insists on interpreting italic letter *a in this proof* in a way which is consistent with the proposition "*a* falls under *G*" coming out true when the concept *G* is non-empty. And there is really only one way to make sense of this: we must assume that, in the proof of the second lemma, Frege reads italic letter *a* as, in effect, a schematic letter. Frege does, in fact, signal that particular reading when, in the first sentence of the proof, he writes: "If, on the other hand, an object falls under *G*, for example *a* ...".

This interpretation of italic letter *a* would seem, on the face of it, to go against Frege's own stipulation regarding their use when he introduced them in par.11 of *Begriffsschrift*. There, to repeat, they were introduced as mere "abbreviations" for variables tacitly bound by an initial universal quantifier. There is then something very curious about Frege's use of italic letters in the proof-sketch of par.75 and in the stipulations of par.71, on which the proof is based. Fortunately for us, we can find the very same use of italic object-letters in the remarkable theorem no.59 of *Begriffsschrift*. That theorem is obtained by simple contraposition from a substitution-instance of axiom 58, which is Frege's version of the old *Dictum de Omni*. It is interesting to note that theorem 59 plays no part in the further development of *Begriffsschrift*: it is never cited in the deduction of any subsequent theorems. Nevertheless, Frege devotes half a page to an explanation of it by way of a curious example, in which he asks us to read italic letter *b* as standing for a *single member* of the family of ostriches. And indeed, we cannot read italic *b* as standing in for a bound variable, for in that case theorem 59 would clearly be invalid. Instead, Frege presents the theorem as his version of *Felapton* or *Fesapo*, two syllogistic moods which - as Frege was undoubtedly aware - are only valid on the assumption that universal propositions have existential import, and that is precisely what universal propositions in Frege's rendering of them do not have. So in theorem 59 of *Begriffsschrift* we are, again, forced to read italic object-letters as schematic letters.

Now, the point of this reading of italics becomes clearer if we look at Frege's reasoning in par.75 in the round. It appears to be essentially *meta*-logical. It has often been asserted that it is characteristic of Frege's conception of logic (as of Russell's) that it lacks - and in fact explicitly excludes the possibility of - a meta-logical perspective. While this is undoubtedly true of Frege's later logic, it is certainly not the case for the early work. Because, strictly considered, neither lemma in par. 75 is, in the present-day sense, a valid formula; what Frege brings into his argument as "assumptions" regarding the emptiness or otherwise of the concepts *F* and *G*, we would nowadays present in terms of a model. And, in the context of a meta-logical argument, Frege's wish to read italic letter *a* as a schematic letter is, of course, perfectly understandable. Now, as is well-known, model-theory was anticipated by Bolzano in his *Wissenschaftslehre* of 1837 [17]; what we now call 'simultaneously satisfiable', Bolzano called 'compatible', the very notion that Frege employs in the presentation of his second proof. I cannot prove

this, because Frege never referred to Bolzano's work, but it is my guess that Frege was familiar with at least the relevant part of the *Wissenschaftslehre*. Unfortunately, he then went on to mishandle the information he found there. Bolzano's notion of 'compatibility' applies to a *class* of propositions, defined by what we would now call two propositional *schemas*. The peculiarity of Frege's reasoning in par.75 is that he uses second-order quantification ("stands to *a* in any relation whatsoever") to express the *meta*-logical proposition that there is a model under which all the members of such a class are true.

Such a procedure is bound to come to grief. I believe it likely that Frege discovered that something was seriously amiss in the early summer of 1887. It was then that Kerry's long critical review of *Grundlagen* appeared [18]. We know, of course, that Frege read it at some point, because he wrote *Begriff und Gegenstand* [19] to rebut some of the criticisms contained in it. But we may assume that he read it the moment it came out. Now, in a footnote to p.289 of that article, Kerry inexplicably renders Frege's crucial sentence "There is no *F* which stands to *a* in any relation whatsoever" as "There is no F which is related to any *a*" [20]. The significance of this - probably freakish - mistake on Kerry's part must have been immediately obvious to Frege. For it amounts in fact to a reading of formula (3) as a *first*-order proposition, in which in effect the roles of italic *f* and italic *a* have been reversed: italic *f* is, in Kerry's version, a schematic letter and italic *a* has become a tacitly bound variable. Frege must *at once* have realized that he had no way to block Kerry's version. But if it cannot be blocked, then it is remarkably easy to derive a contradiction. For it then becomes possible to apply Frege's own reasoning in the first proof: if italic *a* in (3) is read as a tacitly bound variable, then the truth of the content expressed in that formula, *on its own*, will be sufficient ground for the truth of

(4) $\vdash Ga \supset [\forall x(Fx \supset \sim f(x,a)]$

irrespective of whether the concept *G* is empty or not; because, if we follow Frege's reasoning in the first lemma, the truth of Kerry's version of (3) will guarantee the truth of *any* material implication in which it is the consequent; so that, if both *F* and *G* are empty, we can - according to the rules of *Begriffsschrift* - prove that "No *F* stands to *G*'s in a relation", from which it follows that empty concepts are *not* equal, directly contradicting the outcome of the first proof.

One's first reaction to this is one of shock and disbelief: how could Frege, of all people, have got caught in such a terrible muddle ? how could he have been so blind to the possibility of Kerry's 'version' of (3) ? or, what amounts to the same thing, how could he have overlooked the important difference in *kind* between the *truth-functional* notion of incompatibility employed in the proof of the first lemma and the *modal* notion of compatibility in the second lemma ? For his mistake, to put it at its briefest, is that he treats them as contradictories. If we want to understand how Frege came to make this mistake, we should, I think, start with the recognition that the problem arises at a peculiar intersection of three distinctions which we now routinely make: 1) between first- and second-order logic; 2) between logic and meta-logic; and 3) between one- and two-place predicates. There is no space here to go into this in any detail, but there are features in *Begriffsschrift* regarding all three distinctions, which - when combined - make it possible to make at least some sense of the muddle Frege got himself into.

First of all, *Begriffsschrift* doesn't have a proper second-order logic, in that it doesn't have a second-order quantifier; in fact, axiom 58 governs both the first- and second-order fragments of the formula-language, so Frege's early quantification-theory is a many-sorted one, where the 'sorts' range across type-levels. Secondly, Frege appears to have used the stipulation in par 11 of *Begriffsschrift* that "if an italic letter occurs in an expression that is not preceded by a judgment-stroke, then that expression is nonsensical" to signal that an expression containing italics is asserted on sufficient grounds, which may explain his use of modal ("apodeictic") language in connection with italic letters [21]. And thirdly, in *Begriffsschrift* Frege regards a function such as 'x killed x' as a function of *two* arguments; this gives rise to a curious notion of an 'argument-place' (entirely different from the one employed in *Grundgesetze*), which he then uses in the formulation of the introduction and elimination rules for italics [22]. By paying particular attention to these features of *Begriffsschrift* it is, I believe, possible to make at least some sense of what Frege is trying to do in par. 75 of *Grundlagen*.

One would, of course, expect Frege to devise his new system in such a way that the contradiction I have unearthed is blocked. The most important part of his remedy is the introduction of the quantifiers as second-level functions, which then - and I believe that is the whole *point* of so introducing them for Frege - become possible *arguments* to his second-order quantifier (third-level function). For consider the constructional history in *Grundgesetze* of the problematic formula (3), the formula that proved open to the differing interpretations of Frege and Kerry. In *Grundgesetze*, we begin with the binary quantifier - or second-level relation -

(5) $\forall x[\phi(x) \supset \sim \psi(x)]$ [23];

we then construct what Frege calls a 'composite' (zusammengesetzte) name of a second-level function of one argument, by filling its first argument-place with the name of a concept, 'F(ξ)':

(6) $\forall x[F(x) \supset \sim \psi(x)]$ [24];

next we construct the name for the value of the second-order quantifier - third-level function - for the function named by (6) as argument (where italic *f* functions as a bound predicate-variable):

(7) $\forall x[F(x) \supset \sim f(x)]$;

and, finally, we apply rule 9 of *Grundgesetze* [25], and obtain *by substitution* from (7) our formula:

(8) $\forall x[F(x) \supset \sim f(x,a)]$.

It should be clear from this constructional history of (3) in the system of *Grundgesetze*, that Frege's interpretation of it in par.75 of *Grundlagen* and Kerry's alternative reading have *both* become impossible: the last step, sanctioned by rule 9, makes it clear that italic *f* and italic *a* in (8)/(3) cannot be considered 'separately', as it were; and the *separate* treatment of these two italics is precisely what made the two differing readings possible.

Finally, two things should be noted here. First, the choice of italic *a* in (8) is, of course, entirely arbitrary (we could have used *b* or *c*); and so, even if we were to construct a formula such as (4), there is no correspondence at all between italic *a* in the antecedent and italic *a* in the consequent: the interpetation "No *F*'s are related to *G*'s" thereby becomes, syntactically, impossible [26]. Secondly, Frege makes it clear in his explication of rule 9 that "a change of argument-sign is no change in the function-sign"; this, in fact, necessitates the sense-reference distinction for proper names and predicates; for if, by instantiation, we draw conclusions from (8) by inferring to, say, "No Martians are wiser than Solomon" and "No Martians are wiser than Socrates", then Frege is, in *Grundgesetze*, committed to saying that these particular inferences/judgments are *identical*; Frege's repeated insistence that "the argument does not belong to the function" [27] (which, on the face of it, seems so baffling because seemingly vacuous) really means that the two 'composite' function-names " - is wiser than Solomon" and " - is wiser than Socrates" name the *same* function; and, since they clearly are different *predicates*, the sense-reference distinction is, I believe, necessary for *purely formal* reasons, to do with Frege's attempt in *Grundgesetze* to avoid the problem he came up against in the early summer of 1887.

NOTES

1. Dummett 1973
2. p. XIV-XV of the second edition of Dummett 1973, published in 1981
3. Dummett 1991, p.194
4. abbreviation for Frege 1950
5. contained in Frege 1966
6. contained in Frege 1988
7. abbreviation for Frege 1962
8. contained in Frege 1966
9. Dummett 1991, p.2
10. ibidem
11. ibidem: "The pricipal changes in his philosophical logic were the introduction of the far-reaching distinction between sense and reference, and the identification of truth-values as objects and as the references of sentences."
12. Frege 1962, p. IX-XI
13. Dummett, in his 1991, is concerned to show the exact correspondence between the proof-sketches in *Grundlagen* and the formal proofs in *Grundgesetze*; it is, then, surprising that he completely overlooked this important exception: see p.123.
14. Frege 1969, p.225
15. Frege 1962, p.VII
16. It is, in this context, important to stress that, in the logic of *Begriffsschrift*, Frege thought of the old 'categorical propositions' in terms of his primary logic of truth-functions; he even calls it a "reduction": see Frege 1969, p.19.
17. Bolzano 1837
18. Kerry 1887
19. contained in Frege 1966

20. In *Grundlagen*, par.75, Frege had written: "Wenn es nämlich keinen unter F fallenden Gegenstand giebt, so giebt es auch keinen solchen, der in irgendeiner Beziehung zu a stände." Kerry renders this as follows: "... bricht sich später (S.89, §75) die richtige Erkenntnis Bahn, wonach, wenn es gar keinen *F*-Gegenstand gebe, es auch keinen solchen, der zu irgend einem *a* in Beziehung steht, geben könne."
21. cf. *Begriffsschrift*, par.3. In *Grundgesetze*, that stipulation is absent; in fact, Frege there freely uses italics in formulas that are not judgments: see, for example, par.8.
22. for Frege's treatment of 'x killed x', see the italicized sentence on p.17-18 of Frege 1988; for his explication of the notion of an 'argument-place', see the last sentence on p.16; for his use of that notion regarding italics, see p.21-22, and in particular his explanation of axiom 58 on p.51.
23. cf. Frege 1962, p.39
24. on 'composite' names of functions, see Frege 1962, par.30
25. Frege 1962, p.62-3
26. the only reading of (4) that *Grundgesetze* allows is the entirely harmless one: "If everything is *G*, then no *F*'s are related to anything".
27. first expressed in *Funktion und Begriff*, p.6 (in Frege 1966).

REFERENCES

Bolzano, B.: 1837, *Wissenschaftslehre, Versuch einer ausführlichen und grösstenteils neuen Darstellung der Logik mit steter Rücksicht auf deren bisherige Bearbeiter,* Sulzbach

Dummett, M.: 1973, *Frege: Philosophy of Language*, Duckworth, London

Dummett, M.: 1984, An unsuccessful dig, in C. Wright (ed.), *Frege: Tradition and Influence*, Blackwell, Oxford

Dummett, M.: 1991, *Frege: Philosophy of Mathematics*, Duckworth, London

Frege, G.: 1950, *Die Grundlagen der Arithmetik/The Foundations of Arithmetic* (German text, with English translation by J.L. Austin), Blackwell, Oxford

Frege, G.: 1962, *Grundgesetze der Arithmetik*, Olms, Hildesheim

Frege, G.: 1966, *Funktion, Begriff, Bedeutung*, Vandenhoeck & Ruprecht, Göttingen

Frege, G.: 1969, *Nachgelassene Schriften*, ed. H. Hermes, F. Kambartel and F. Kaulbach, Hamburg

Frege, G.: 1988, *Begriffsschrift und andere Aufsätze*, ed. I. Angelelli, Olms, Hildesheim

Kerry, B.: 1887, Ueber Anschauung und ihre psychische Verarbeitung, vierter artikel, *Vierteljahrschrift für wissenschaftliche Philosophie* XI (3), 249-307

Only Updates

On the Dynamics of the Focus Particle *only*

Gerhard Jäger
Centrum für Informations- und Sprachverarbeitung
University of Munich

1 Introduction

The Montagovian paradigm of natural language semantics relies on the two crucial assumptions that (a) the meaning of a sentence can exhaustively be described by means of its truth conditions, and (b) this meaning can be built up compositionally from its parts. Unfortunately, empirical observations force us to the conclusion that the mentioned assumptions cannot be true at the same time, as soon we shift our attention to discourse phenomena. This is exemplified by the discourses in (1):

(1) a. Peter came in. He wore a hat.
b. John came in. He wore a hat.

The second sentence in (1a) is true just in case that Peter wore a hat, while the corresponding sentence in (1b) is equivalent to *John wore a hat.* These two sentences are syntactically identical, at least as far as their surface is concerned. Hence they have the same parts, and, *ceteris paribus*, they should have the same meaning. Nevertheless their truth conditions differ.

This and related problems led several authors to the conclusion that we have to give up the principle of compositionality in the strict Montagovian sense. Most influentally, Kamp 1981 and Heim 1982 propose that there is an additional level of representation relating syntactic structure and semantic interpretation. In their systems, syntactic structure has to undergo a process of "DRS-construction" (Kamp) or "LF-construal" (Heim), and it is the output of this process that is interpreted compositionally in terms of truth conditions.

Groenendijk and Stokhof 1991a and Groenendijk and Stokhof 1991b choose the other direction. They point out that the principle of compositionality can be maintained as soon as we do without the

assumption that sentence meanings coincide with truth conditions. Instead they propose that sentences (and discourses) denote transition function over information states. These transition functions are connected to truth conditions, but in an indirect way. The advantage of this strategy is a methodological rather than an empirical one, since compositional and non-representational semantic theories are generally more restrictive in their predictive power.

It is the aim of this paper to extend the coverage of the dynamic paradigm to phenomena involving the focus sensitive operator *only*. Constructions involving this item show a dependency on linguistic context reminiscent to the behavior of anaphora. Existing approaches address this phenomenon by weakening the compositionality of interpretation in several respects. Instead we will try to outline a dynamic theory of the semantics of this constructions that preserves compositionality both on the sentence and on the text level.

2 The Problem

Consider the following contrast:

(2) Who is wise? Only $[_{+F}$ SOCRATES] is wise.

(3) Which Athenians are wise? Only $[_{+F}$ SOCRATES] is wise.

Although the answers in (2) and (3) are identical at the surface, they show up different truth conditions. The answer in (2) forms a blasphemy since it entails that Zeus is unwise, while the corresponding sentence in (3) is much weaker in the sense that only the wisdom of all Athenians except Socrates is denied. Nothing is said about other individuals like, say, gods. More generally, the respective answers are truthconditionally equivalent to the first order formulae in (4a) and (b) respectively.

(4) a. $\forall x(wise(x) \rightarrow x = s)$
b. $\forall x(athenian(x) \wedge wise(x) \rightarrow x = s)$

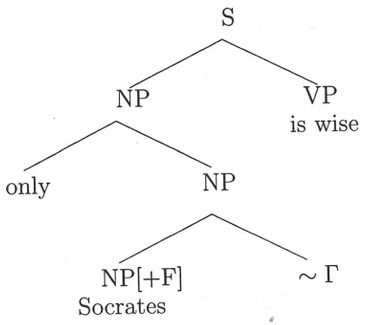
In both cases, the universal quantification is restricted by the non-focused part (the "background") of the sentence (*is wise*). Besides this, there is an additional restrictor in (4b), corresponding to the argument of the *wh*-word in the question (*Athenian*). Hence it is quite obvious that the truth conditions of a sentence involving *only* somehow depend on the question the sentence is an answer to.

We will proceed as follows. In the subsequent section, the proposal made in Rooth 1992 – which can be seen as a kind of paradigmatic approach in a static setting – is briefly presented and discussed. In section 4 we develop a dynamic system that covers both interrogative and declarative sentences. Finally, in section 5 it is shown that this system is able to account for the kind of dependency between questions and answers illustrated above.

3 Rooth 1992

In the sense of the discussion above, Rooth's proposal can be seen on a par with Heim 1982. As the feature that matters most for our purposes, he assumes that it is not surface structure that serves as input for semantic interpretation, but that there is an intermediate level of "Logical Form". The proposed LF for the sentence under discussion is roughly as in (5):

(5)



This is not the proper place to discuss the details and merits of Rooth's semantics of focus in general. It is only important that the LF contains – besides the overt material – an additional item "Γ" that is adjoined to the sister constituent of *only* by means of the operator "~". Γ is interpreted as a restriction of the domain of the universal quantification induced by *only*, such that we end up with an interpretation as it is given in (6).

(6) $\forall x(C(x) \wedge wise(x) \rightarrow x = s)$

"Γ" is considered to be a kind of anaphor that is interpreted as the free variable C above. The value of this variable is – according to Rooth 1992 – determined by a variety of factors that are external to the compositional interpretative machinery. Although he does not address comparable examples directly, it is very much in his spirit to assume that Γ should be coindexed with the *wh*-phrase of the preceding question in (2) and (3) by means of some pragmatic mechanism.

More generally, to achieve the appropriate truth conditions for constructions involving *only* in a static setup, we are forced to assume that (a) there is at least one level of representation distinct from surface structure that serves as input for the compositional interpretation, and (b) truth conditions, i.e. meaning is not completely determined by lexicon and syntax but relies on pragmatic processes. In view of this fact it strikes me a fruitful enterprise to develop a semantical analysis that avoids these stipulations, even if we have to give up the equation *meaning* = *truthconditions*. More in detail, our aim is

- to get rid of any kind of syntactic placeholders like the anaphor Γ in Rooth's proposal
- to derive the meaning of constructions involving *only* fully compositionally, i.e. without making reference to pragmatics.

4 Update Logic for Questions and Answers

4.1 A Static Approach to the Semantics of Questions: Groenendijk and Stokhof 1989

It is quite obvious that the different domain restrictions in the answers in (2) and (3) depend on the preceding question. Since we aim at an approach that is purely semantical, we have to assume that it is the semantics of the respective questions that trigger this effect. As basis for further argumentation, I adopt the framework laid down in Groenendijk and Stokhof 1984 without further argumentation. [1] It relies on the assumption that each question determines a unique proposition that constitutes the exhaustive true answer to the question. [2] This is best illustrates by an example. Take some simple yes-no question like

(7) Is it raining?

1. In Groenendijk and Stokhof 1989 it is shown that the propositional accounts to question semantics given in Hamblin 1976, Karttunen 1977, and Groenendijk and Stokhof 1984 are fundamentally equivalent, such that the choice does nor matter too much.
2. Counterexamples to this claim like free-choice questions are ignored throughout this paper.

If it is raining, the unique exhaustive true answer is the proposition *It is raining*, and in case it is not raining, this answer is constituted by *It is not raining*. This can easily be expressed in a two-sorted extensional type theory:

(8) $\lambda w.(rain\prime(w) \leftrightarrow rain\prime(w_0))$

This expression denotes a proposition in every possible world, although it is not necessarily the same proposition in different worlds. To get a fixed semantic object, we have to λ-abstract over the world of evaluation w_0. According to Groenendijk and Stokhof 1984, the meaning of a question is just this new object – the "concept of the true answer".

(9) *Is it rainig?* $\rightsquigarrow \lambda v \lambda w(rain\prime(v) \leftrightarrow rain\prime(w))$

It is obvious that this denotes an equivalence relation on the set of possible worlds. Hence a question defines an exhaustive partition on this set into a set mutually exclusive proposition. In the example above, these are just the propositions *It is raining* and *It is not raining*. Generally, the elements of the partition are those propositions that constitute exhaustive – but not necessarily true – answers to the question.

4.2 Information States and Updates

The common strategy for dynamisizing a certain static semantics runs roughly as follows: If sentences statically denote objects from some domain α, the corresponding dynamic formulae denote functions $F : \alpha \rightarrow \alpha$. Since we want to deal with question-answer sequences in a single dynamic system, we are faced with a serious problem. Declarative sentences statically denote propositions, i.e. sets of possible worlds, while interrogatives denote relations on possible worlds or – equivalently – sets of propositions. Hence there are two candidates for update-denotations: functions from propositions to propositions or functions from relations to relations. By "generalizing to the worst case", we adopt the latter option. Formulae of the dynamic logic to be defined below denote functions over information states (or simply "states"), where states are equivalence relations on possible worlds.

Let a nonempty set W of possible be given. We define:

Definition 1 (Information States)
σ is an information state iff $\sigma \subseteq W \times W$ and σ is an equivalence relation.

This immediately gives us a partial order on the set of states, corresponding to the intuitive notion of informativity. The minimal and the maximal elements of this order are called **1** (state of ignorance) and **0** (absurd state) respectively.

Definition 2 (Informativity)

$$\begin{aligned} \sigma_1 \leq \sigma_2 &\Leftrightarrow_{def} \sigma_1 \supseteq \sigma_2 \\ \mathbf{1} &=_{def} W \times W \\ \mathbf{0} &=_{def} \emptyset \end{aligned}$$

Note that these relations may be partial, i.e. their domains may be proper subsets of W. Hence each state nontrivially determines a certain proposition, which can be thought of as the factual knowledge shared by the conversants.

Definition 3 (Domain of a state)

$$D(\sigma) =_{def} \{w | w\sigma w\}$$

Since each equivalence relation uniquely defines a partition on its domain, it is worth considering the structure of the space of partitions determined by the space of states.

Definition 4 (Partitions)
Let σ, τ be information states.

$$\begin{aligned} P_\sigma &=_{def} \{\{v | v\sigma w\} | w \in W\} \setminus \{\emptyset\} \\ P_\sigma \sqsubseteq_d P_\tau &\Leftrightarrow_{def} P_\sigma \supseteq P_\tau \\ P_\sigma \sqsubseteq_i P_\tau &\Leftrightarrow_{def} \bigcup P_\sigma = \bigcup P_\tau \wedge \forall x \in P_\sigma \exists y \in P_\tau : x \supseteq y \end{aligned}$$

Equivalence classes of possible worlds can be thought of as epistemic alternatives in a certain stage of conversation. Information growth can affect these alternatives in two ways. Either some of them are eliminated – this is covered by $\sqsubseteq_d$ – or they are made more finegrained without changing the domain of the equivalence relation itself ($\sqsubseteq_i$). The latter corresponds to the effect of asking a question, while the former

is the purpose of declarative utterances. [3] The notion of informativity given above covers both ways of information growth.

Fact 1
For all states σ and τ, it holds that:

$$\begin{aligned} P_\sigma \sqsubseteq_d P_\tau &\rightarrow \sigma \leq \tau \\ P_\sigma \sqsubseteq_i P_\tau &\rightarrow \sigma \leq \tau \end{aligned}$$

Furthermore, the notion of informativity is exhausted by $\sqsubseteq_d$ and $\sqsubseteq_i$:

Fact 2
For all states σ and τ, it holds that:

$$\sigma \leq \tau \leftrightarrow \exists \upsilon : P_\sigma \sqsubseteq_i P_\upsilon \wedge P_\upsilon \sqsubseteq_d P_\tau$$

The intended interpretations of sentences/formulae are updates, i.e. transition functions over states that increase information.

Definition 5 (Updates)
Let Σ be the set of information states.

$$UP =_{def} \Sigma^\Sigma \cap POW(\leq)$$

Updates may be classified according to the way how they increase information.

Definition 6 (Interrogative and Declarative Updates)

- *An update u is called* declarative *iff*

$$\forall \sigma : P_\sigma \sqsubseteq_d P_{u(\sigma)}.$$

- *An update u is called* interrogative *iff*

$$\forall \sigma : P_\sigma \sqsubseteq_i P_{u(\sigma)}.$$

3. This is reminiscent to Dekker's EDPL (cf. Dekker 1993), where information growth either introduces new variables into the state or eliminates possible values of familiar variables.

There are only two updates that are both declarative and interrogative, namely the identity function and the constant function that always gives **0** as output. On the other hand, there are many updates that are neither declarative nor interrogative. Nevertheless, according to Fact 2, every update can be decomposed into an interrogative and a declarative one (in this order).

Since interrogative and declarative sentences denote different kinds of objects statically, there are two pairs of operators that switch between static and dynamic meanings.

Definition 7 (Up-Arrow and Down-Arrow)
Let σ be an information state, p a proposition, q a static question denotation, and u an update.

$$\begin{aligned} \downarrow_d u &=_{def} \{w | u(\{\langle w, w\rangle\}) = \{\langle w, w\rangle\}\} \\ (\uparrow_d p)(\sigma) &=_{def} \sigma \cap p \times p \\ \downarrow_i u &=_{def} u(\mathbf{1}) \\ (\uparrow_i q)(\sigma) &=_{def} \sigma \cap q \end{aligned}$$

Fact 3 For all propositions p and static question denotations q, it holds that:

$$\downarrow_d \uparrow_d p = p$$
$$\downarrow_i \uparrow_i q = q$$

Please note that the converse neither holds for $\uparrow_d \downarrow_d$ nor for $\uparrow_i \downarrow_i$. The $\uparrow_d$–operator is of particular interest since it enables us to add factual knowledge to a state without destroying the structure of the partition (of course, this holds only to the extend that the alternatives are compatible with this factual knowledge).

Fact 4 For all states σ and propositions p, it holds that:

$$D(\uparrow_d p(\sigma)) = D(\sigma) \cap p \quad (1)$$
$$\forall x(x \in P_\sigma \wedge x \cap p \neq \emptyset \rightarrow x \cap p \in P_{\uparrow_d p(\sigma)}) \quad (2)$$

4.3 The Semantics of ULQA

Having developed the necessary ontological background, we can start define a simple language that serves to reason about the described kind

of updates. The syntax of ULQA is just the syntax of first order logic without functions symbols and with identity, with the single extension that there is a oneplace propositional operator "?" that makes formulae out of formulae. For convenience, we take $\wedge, \neg, \rightarrow$, and $\forall$ as logical constants and the other connectives as abbreviations in the usual way.

A model for ULQA consists of an individual domain and a collection of classical interpretation functions (= possible worlds).

Definition 8 (Model)
A model M for ULQA is a triple $\langle E, W, F\rangle$, where E is a denumerable infinite set, W is non-empty, and F is a function that assigns a first-order interpretation function based on E to every member of W.

In the definition of the semantics of ULQA, we follow common practise in writing $\sigma[\phi]_g$ instead of $\|\phi\|_g(\sigma)$ in case ϕ denotes an update. By $=$C1$=$C1$g[e/x]$ we mean the assignment function $g\prime$ that is exactly like g except that it maps x to e.

Definition 9 (The Semantics of ULQA)

$$\begin{aligned} \|c\|_{w,g} &=_{def} F(w)(c) \text{ iff } c \text{ is an individual constant} \\ \|v\|_{w,g} &=_{def} g(v) \text{ iff } v \text{ is a variable} \\ \sigma[P(t_1, ..., t_n)]_g &=_{def} \sigma \cap \{\langle v, w\rangle |\ \forall w\prime(v\sigma w\prime \rightarrow \langle \|t_1\|_{w\prime,g}, ..., \|t_n\|_{w\prime,g}\rangle \in F(w\prime)(P))\} \\ \sigma[t_1 = t_2]_g &=_{def} \sigma \cap \{\langle v, w\rangle |\ \forall w\prime(v\sigma w\prime \rightarrow \|t_1\|_{w\prime,g} = \|t_2\|_{w\prime,g})\} \\ \sigma[\phi \wedge \psi]_g &=_{def} \sigma[\phi]_g[\psi]_g \\ \sigma[\neg\phi]_g &=_{def} \sigma \setminus \sigma[\phi]_g \\ \sigma[?\phi]_g &=_{def} \sigma \cap \{\langle v, w\rangle |\ \{\langle v, v\rangle\}[\phi]_g = \emptyset \leftrightarrow \{\langle w, w\rangle\}[\phi]_g = \emptyset\} \\ \sigma[\forall x.\phi]_g &=_{def} \bigcap_{e \in E} \sigma[\phi]_{g[e/x]} \\ \sigma[\phi \rightarrow \psi]_g &=_{def} \sigma \cap \{\langle v, w\rangle |\ v\sigma[\phi]_g w \rightarrow v\sigma[\phi \wedge \psi]_g w\} \end{aligned}$$

The interpretation of atomic formulae is based on the corresponding classical interpretation. Updating a certain state σ with a classical formula ϕ amounts saying that only those alternatives in P_σ survive that are completely included in the set of possible worlds where ϕ is true under its classical interpretation. If you consider σ as an accesibility relation in a Kripke model, the domain of the output is restricted to those worlds where ϕ is necessarily true in its static interpretation.

The clauses for dynamic conjunction, dynamic negation, and dynamic implication are familiar from other dynamic systems like Veltman's Update Semantics (cf. Veltman 1990) and do not need much

explanation. The semantics of universal quantification is a straightforward extrapolation from its classical counterpart.

The key feature of ULQA is the ?-operator. To explain its impact on a rather intuitive level, each proposition in P_σ is splitted into those worlds where the formula in the scope of "?" is true and those where it is false. To put it another way, "?ϕ" defines an equivalence relation by its own, and the output of the update is the intersection of σ with this relation.

It does not come as a surprise that atomic formulae denote declarative updates and formulae prefixed with "?" interrogative ones. Both properties are preserved under conjunction and universal quantification. Being a declarative update is also preserved under negation. Negating an interrogative update, on the other hand, returns you in all non-trivial cases a relation as output that is reflexive and symmetric but not transitive. Hence the negation of an interrogative update isn't an update at all in the general case.

The definitions of truth and entailment in ULQA are fairly standard from related dynamic calculi. A formula is called *true in a state* if updating the state with the formula does not add information. By abstracting over particular contexts, we get the notion of *truth in a model*, and by abstracting over models, we can define *logical truth*.

Definition 10 (Truth)
Let M be a model, σ be an information state and ϕ be a formula.

$$\begin{aligned} M, \sigma \models \phi &\Leftrightarrow_{def} \forall g : \sigma[\phi]_{M,g} = \sigma \\ M \models \phi &\Leftrightarrow_{def} \forall \sigma : M, \sigma \models \phi \\ \models \phi &\Leftrightarrow_{def} \forall M : M \models \phi \end{aligned}$$

The definition of the consequence relation between formulae is straightforwardly derived from this truth definitions. ψ is said to be a consequence of ϕ iff the output of ϕ is always a state where ψ is true.

Definition 11 (Entailment)
Let M be a model, σ an information state, and ϕ, ψ formulae.

$$\begin{aligned} \phi \models_{M,\sigma} \psi &\Leftrightarrow_{def} \forall g : \sigma[\phi]_{M,g} = \sigma[\phi \wedge \psi]_{M,g} \\ \phi \models_M \psi &\Leftrightarrow_{def} \forall \sigma : \phi \models_{M,\sigma} \psi \\ \phi \models \psi &\Leftrightarrow_{def} \forall M, \phi \models_M \psi \end{aligned}$$

4.4 The Relation of ULQA to First-order Logic

Syntactically speaking, ULQA is a simple extension of first-order logic, and also semantically, there is a close connection between the classical fragment of ULQA and PL1. Recall that the individual domain E of and ULQA-model M together with any possible world w from W forms a first-order model. A ULQA-formula is called *?-free* iff it does not contain any occurrence of "?". Trivially, the ?-free formulae are just the formulae of PL1.

Definition 12 *=80*
Let ϕ be a ?-free ULQA formula.

- By $\|\phi\|_M^{ULQA}$ we refer to the ULQA-interpretation of ϕ in the ULQA-model M.
- By $\|\phi\|_M^{PL1}$ we refer to the set of worlds w from W such that $\|\phi\|$ is true in $\langle E, F(w)\rangle$ under classical first-order interpretation.

Fact 5
Let ϕ be a ?-free ULQA-formula. It holds in any model M that:

$$\downarrow_d \|\phi\|_M^{ULQA} = \|\phi\|_M^{PL1}$$

Proof : By induction on the complexity of ϕ.

In a sense, the classical interpretation of some ?-free formula ϕ, i.e. $\downarrow_d \|\phi\|^{ULQA}$, can be seen as the "context-free" truth-conditional or factual impact of that formula. Hence by updating a state σ with $\uparrow_d\downarrow_d \|\phi\|$, we add the factual content of ϕ to σ without affecting the structure of P_σ.

Fact 6
Let ϕ be ?-free.

$$\uparrow_d\downarrow_d \|\phi\|^{ULQA}(\sigma) = \sigma \cap \|\phi\|^{PL1} \times \|\phi\|^{PL1}$$

Proof : Immediately from the definition of $\uparrow_d$.

Following common practise, we call $\uparrow_d\downarrow_d \|\phi\|$ the *static closure of* ϕ. Fortunately, the operation of static closure can be expressed in the object language, as far as ?-free formulae are concerned.

Fact 7
Let ϕ be ?-free.

$$\uparrow_d\downarrow_d \|\phi\| = \|?\phi \wedge \phi\|$$

Proof : By **Fact 5** and induction on the complexity of ϕ.

For convenience, we will henceforth abbreviate "$?\phi \wedge \phi$" with "$\uparrow\downarrow \phi$" and we will also refer to it as the "static closure of ϕ". Context will make clear whether we use the term in its syntactic or its semantic sense.

Remember that the union of the propositions in P_σ (the epistemic alternatives) represents the knowledge that is shared by the conversants. Static closure enables us to make statements about this state of factual knowledge in the object language.

Fact 8
Let ϕ be ?-free.

$$\sigma \models \uparrow\downarrow \phi \quad \Leftrightarrow \quad D(\sigma) \subseteq \|\phi\|^{PL1} \qquad (1)$$
$$\psi \models \uparrow\downarrow \phi \quad \Leftrightarrow \quad \forall \sigma : D(\sigma[\psi]) \subseteq \|\phi\|^{PL1} \qquad (2)$$

5 Restricted Quantification in ULQA

5.1 English $\Longrightarrow$ ULQA

Since ULQA is a first-order language, there cannot be an immediate compositional translation function from English into ULQA. But it is obvious from the definition of the semantics of ULQA that every semantic object that is the interpretation of some ULQA-formula is at the same time the interpretation of some Ty_2-formula. Hence ULQA can be seen as a convenient notation of a fragment of Ty_2. On the other hand, if the translation of a couple of English sentences into Ty_2 are given, it is a technical exercise to develop a Montague-style compositional translation from this fragment of English into Ty_2. This in mind, I will content myself with stipulating the ULQA-translations of the English sentences under debate since the described procedure would take a lot of space without illuminating anything of particular interest.

The translation of simple clauses with names in the argument positions are straightforward and do not need much explanation.

(10) Socrates is wise $\rightsquigarrow$ $wise(s)$

Prima facie, the ?-operator only enables us to form yes-no questions. Nevertheless it is possible to deal with constituent questions appropriately. We start with the observation that a *which*-question is

equivalent to a yes-no-question in the scope of a restricted universal quantifier.

(11) a. Which Athenians are wise?
b. For all Athenians: Is he or she wise?

In contrast to other approaches to the semantics of questions, the interpretations of interrogative and declarative sentences belong to the same logical type, namely updates. Hence there is no problem in quantifying into a question. Hence I assume:

(12) Which Athenians are wise? $\rightsquigarrow \forall x(athenian(x) \rightarrow ?wise(x))$

Who-questions can be handled in a similar manner. The only difference lies in the absence of a restriction to the quantifier.

(13) Who is wise $\rightsquigarrow \forall x.?wise(x)$

One of the crucial features of ULQA is the fact that universal quantification is – so to speak – automatically contextually restricted. This fact will be illustrated in some length in the subsequent paragraphs. Therefore the translation of *only*-constructions can be kept pretty simple.

(14) Only Socrates is wise $\rightsquigarrow \forall x(wise(x) \rightarrow x = s)$

This strategy is of course an oversimplification in some respects, but Krifka 1992 shows convincingly that it is possible to derive corresponding translations of more complex construction involving VP-focus and multiple focus fully compositionally.

5.2 Some Properties of ULQA

Let us start with a couple of negative results. First of all, the consequence relation defined above is not reflexive.

Fact 9 (Non-Identity)
There are formulae ϕ such that

$$\phi \not\models \phi$$

As it will turn out, this is not an accident but even a quite desirable feature. An example will be given and discussed below. It is worth

noting that identity does hold as far as ?-free formulae are concerned.

Fact 10
Let ϕ be ?-free. Then it holds that

$$\phi \models \phi$$

Sketch of proof: The semantics of ULQA can equivalently be redefined in such a way that formulae denote updates over partitions. Under this perspective, ?-free formulae denote updates that are both eliminative and distributive (cf. Groenendijk and Stokhof 1990). Hence there is a static interpretation to this fragment such that updating is just intersecting the state with the static meaning. The fact follows then from the idempotence of set intersection.⊣

For the present discussion, it is more important that we are not enabled to infer from a certain update to its static closure, even if identitiy holds for this update.

Fact 11
There are formulae ϕ such that

$$\phi \models \phi \quad (1)$$
$$\phi \not\models \uparrow\downarrow \phi \quad (2)$$

Proof : Suppose $\phi = \neg\psi$, let ψ be a ?-free closed atomic formula such that $\|\psi\|^{PL1} \neq \emptyset$, $\|\psi\|^{PL1} \subset W$. (1) follows immediately from Fact 10. For (2), observe that $\mathbf{1}[\neg\psi] = \mathbf{1}$, $\mathbf{1}[\uparrow\downarrow \neg\psi] = \|\neg\psi\|^{PL1}$. Hence $\mathbf{1}[\neg\psi] \neq \mathbf{1}[\neg\psi][\uparrow\downarrow \neg\psi]$ ⊣

This implies that the context-free meaning of a certain formula may be logically stronger than its context-dependent version. A quite realistic example is

(15) a. Only Socrates is wise.
b. $\forall x.(wise\prime \rightarrow x = s\prime) \not\models\uparrow\downarrow \forall x.(wise\prime \rightarrow x = s\prime =)$

The static closure of *Only Socrates is wise* means that Socrates is literally the only wise individual, and this of course cannot be inferred from the utterance of the sentence in a particular context, as (3) shows.

Neither can we infer from a universally quantified formula to the static closure of some instance.[4]

Fact 12
There are formulae ϕ and individual constants a such that

$$\forall x.\phi \models \phi(a)$$
$$\forall x.\phi \not\models \uparrow\downarrow \phi(a)$$

Proof : Let ϕ be as in the proof of Fact 11. Since ϕ is closed, the fact follows immediately from Fact 11. ⊣

This is again a desired result, since we don't want to draw the conclusion *Zeus is not wise* (⇐ *If Zeus is wise, then he is Socrates)* from *Only Socrates is wise* under all circumstances. Nevertheless there is a restricted version of the mentioned kind of universal instantiation.

Fact 13
For all ?-free formulae ϕ, ψ and individual constants a, it holds that:

$$\text{If } \psi \models \uparrow\downarrow \psi$$
$$\text{then } ?\phi(a) \wedge \forall x(\phi \rightarrow \psi) \models \uparrow\downarrow (\phi(a) \rightarrow \psi(a))$$

Sketch of proof: $P_{\sigma[?\phi(a)]}$ contains only propositions that either entail $\|\phi(a)\|^{PL1}$ or $\|\neg\phi(a)\|^{PL1}$. Among those that make $\phi(a)$ classically true, only those can survive in $P_{\sigma[?\phi(a)\wedge\forall x(\phi\rightarrow\psi)]}$ that survive under updating with $\psi(a)$. The premise ensures that these propositions are contained in $\|\psi(a)\|^{PL1}$. Hence if a proposition in $P_{\sigma[?\phi(a)\wedge\forall x(\phi\rightarrow\psi)]}$ classically entails $\phi(a)$, it also entails $\psi(a)$. The conclusion follows by Fact 8 (2).⊣.

Applied to the example, it follows that we may infer from the utterance of *Only Socrates is wise* to, say, *Plato is unwise* provided that Plato's wisdom **is under debate** in the present state of conversation.

(16) $?wise(p) \wedge \forall x(wise(x) \rightarrow x = s) \models\uparrow\downarrow (wise(p) \rightarrow p = s)$

To put it another way round, besides the syntactically present restrictor to a universal quantifier, we have the implicit restriction that

4. By $\phi(a)$ we refer to $\phi[a/x]$, provided that there are no variables besides x free in ϕ.

an individuals being an instance of the restrictor has to be under debate at the current state of conversation. Let me briefly explain why ULQA behaves in this way. A partition corresponding to a state is a collection of sets of worlds, i.e. total first-order interpretation functions. A set of interpretation functions can be identified with a partial function. If all total functions in the set agree about the value of a certain item x, the derived functions assigns this value to x, too. If x receives different values under different functions in the set, its value under the corresponding partial function is undefined. Hence an information state can be seen as a set of situations, i.e. partial first-order models. Asking a question in a certain state extends the domain of the situations in the state. The interpretation of ϕ is defined in any situation in $\sigma[?\phi]$, no matter whether it was defined in σ. Hence $\sigma \models ?\phi$ just if ϕ is completely defined in σ.

This in mind, it becomes clear why identity should not hold in ULQA. Suppose that an atomic formula ϕ is undefined in any situation in a state σ. Then $\sigma[\phi] = \mathbf{0}$ and $\sigma[\neg\phi] = \sigma$. But updating σ with $?\phi$ brings us in a state where ϕ is defined. Now suppose that ϕ expresses a proposition contingent in σ. In this situation, $\sigma[?\phi \wedge \neg\phi] \neq \sigma[?\phi]$. Hence $\neg\phi \wedge ?\phi \not\models \neg\phi \wedge ?\phi$.

A certain individual only falls in the domain of a restricted universal quantifier if it unequivocally either is an instance of the restrictor or it is not. Hence we have an implicit restriction of a quantifier domain given by definedness.

Now let us return to the example. *Only Socrates is wise* makes a statement about those objects whose wisdom is under debate. Asking the question *Which Athenians are wise?* brings you in a state where the wisdom of all individuals that are known to be Athenians is under debate. Let us make this slightly more precise. Firstly, we have a restricted version of Modus Ponens together with universal instantiation in ULQA. To give the restrictions precisely, we firstly need an auxiliary definition.

Definition 13 (Persistence)
A formula ϕ is said to be *persistent* iff for all states σ, τ:

$$\sigma \models \phi,\ \sigma \leq \tau \ \Rightarrow\ \tau \models \phi$$

Note that by the definitions, any formula prefixed with "?" is persistent.

Fact 14 =20

For all formulae ϕ, ψ and individual constants a, it holds that:

$$\text{If } \phi \models \phi,\ \psi \models \psi \text{ and } \psi(a) \text{ is persistent, then } \phi(a) \wedge \forall x(\phi \rightarrow \psi) \models \psi(a)$$

Sketch of proof: Suppose that the premises hold for ϕ and ψ. By the semantics of $\forall$, we have $\sigma[\phi(a) \wedge (\phi(a) \rightarrow \psi(a))] \leq \sigma[\phi(a) \wedge \forall x(\phi \rightarrow \psi)]$. =46rom $\phi \models \phi$, we have $\phi(a) \models \phi(a)$, and together with the definition of dynamic implication, we then have $\sigma[\phi(a) \wedge (\phi(a) \rightarrow \psi(a))] = \sigma[\phi(a) \wedge \psi(a)]$. Hence we have $\sigma[\phi(a) \wedge \psi(a)] \leq \sigma[\phi(a) \wedge \forall x(\phi \rightarrow \psi)]$. =46rom $\psi \models \psi$, we infer $\psi(a) \models \psi(a)$. Hence we have $\sigma[\phi(a) \wedge \psi(a)] \models \psi(a)$. By the persistence of $\psi(a)$, we have $\sigma[\phi(a) \wedge \forall x(\phi \rightarrow \psi)] \models \psi(a)$ $\dashv$

If we know that a person, say, Plato is an Athenian, after asking the question *Which Athenians are wise?* we are in a state where Plato's wisdom is under debate.

(17) $athenian(p) \wedge \forall x.(athenian(x) \rightarrow ?wise(x)) \models ?wise(p)$

Now from (17) and (16) we may conclude:

(18) a. Which Athenians are wise? Only Socrates is wise.
b. $athenian(p) \wedge \forall x(athenian(x) \rightarrow ?wise(x)) \wedge \forall x(wise(x) \rightarrow x = s)$
$\models \uparrow\downarrow (wise(p) \rightarrow p = s)$

Now suppose that a person, for instance Zeus, is neither known to be an Athenian nor to be wise in any of the epistemic alternatives in a certain state σ (formally: $\sigma \models \neg athenian(z)$, $\sigma \models \neg wise(z)$) In this case, there is nothing that can be inferred about Zeus' wisdom from the question-answer pair above. Hence the domain of the universal quantification introduced by *only* is in fact restricted to Athenians in this context.

To analyse *Who is wise?*, please observe that $\forall x.?wise(x)$ is synonymous to $\forall x(x = x \rightarrow ?wise(x))$. Hence after asking that question, anybody's wisdom is under debate, and therefore we do not have any domain restriction at all in the subsequent statement.

(19) $\forall x.?wise(x) \models ?wise(a)$ (where a is an arbitrary individual constant)

=46rom (19) and (16), we may conclude:

(20) a. Who is wise? Only Socrates is wise.
b. $\forall x.?wise(x) \wedge \forall x(wise(x) \rightarrow x = s) \models \uparrow\downarrow (wise(z) \rightarrow z = s)$

6 Summary

The implicit universal quantification introduced by *only*-constructions is restricted by contextual information. This fact becomes most obvious in question-answer pairs. According to Rooth 1992, these constructions nevertheless involve classical universal quantification which is restricted by a syntactically present anaphor Γ. The interpretation of Γ is governed by some pragmatic mechanism. It was the aim of this paper to achieve the same result in a purely semantic manner. Firstly, to deal with context dependency semantically, we are forced to use a dynamic setup. Secondly, we changed the semantics of universal quantification in such a way that it is restricted implicitly by the context. Suppose a universal quantifier is syntactically restricted by some predicate P. In this case, the actual domain of quantification is not the entire universe but the set of individuals whose being P is under debate in the current state of conversation, and it is the purpose of questions to bring issues into the debate. Hence the dependency between questions and focus is a rather indirect one under the present approach.

There are plenty of questions left that require further investigations, concerning both the logic developed here and its linguistic application. As I mentioned in the preceding section, the semantics of ULQA makes use of a kind of simulated partiality. It strikes me an interesting issue to see what happens if we replace sets of propositions by sets of situations. Besides this, it is worth investigating whether it is possible to incorporate ULQA in a static logic with program modalities in a similar manner as presented in van Eijck and de Fries 1993 and van Eijck 1993 for Update Logic and Dynamic Predicate Logic. Linguistically, the coverage should be extended to other phenomena like the semantics of indirect questions of *wh*-pronouns.

7 Acknowledgement

I am particularly indebted to Henk Zeevat for a very inspiring discussion during his stay in Berlin in spring 1995.

References

Dekker, P.: 1993, *Transsentential Meditations. Ups and Downs in Dynamic Semantics*, Ph.D. thesis, University of Amsterdam

Groenendijk, J. and Stokhof, M.: 1984, *Studies on the Semantics of Questions and the Pragmatics of Answers*, Ph.D. thesis, University of Amsterdam

Groenendijk, J. and Stokhof, M.: 1989, Type-shifting rules and the semantics of interrogatives', in G. Chierchia, B. Partee, and R. Turner (eds.), *Properties, Types and Meaning*, Kluwer

Groenendijk, J. and Stokhof, M.: 1990, Two theories of dynamic semantics, in J. van Eijck (ed.), *Logics in AI*, Springer, Berlin

Groenendijk, J. and Stokhof, M.: 1991a, Dynamic Montague Grammar, in J. Groenendijk, M. Stokhof, and D. I. Beaver (eds.), *Quantification and Anaphora I*, DYANA deliverable R2.2a, Amsterdam

Groenendijk, J. and Stokhof, M.: 1991b, Dynamic Predicate Logic, *Linguistics and Philosophy* 14(1)

Hamblin, C.: 1976, Questions in Montague English, in B. Partee (ed.), *Montague Grammar*, Academic Press, NY

Heim, I.: 1982, *The Semantics of Definite and Indefinite Noun Phrases*, Ph.D. thesis, University of Massachusetts, Amherst

Kamp, H.: 1981, A theorie of truth and semantic representation, in J. Groenendijk, T. Janssen, and M. Stokhof (eds.), *Formal Methods in the Study of Language*, Amsterdam

Karttunen, L.: 1977, Syntax and semantics of questions, *Linguistics and Philosophy* 1, 3–44

Krifka, M.: 1992, A compositional semantics for multiple focus constructions, in J.Jacobs (ed.), *Informationsstruktur und Grammatik*, Linguistische Berichte, Sonderheft 4

Rooth, M.: 1992, A theory of focus interpretation, *Natural Language Semantics* 1, 75–116

van Eijck, J.: 1993, Axiomatizing Dynamic Predicate Logic with Quantified Dynamic Logic, in J. van Eijck (ed.), *Course Material of the Fifth European Summerschool in Logic, Language and Information*, Lisbon

van Eijck, J. and de Fries, F.-J.: 1993, Reasoning about Update Logic, in J. van Eijck (ed.), *Course Material of the Fifth European Summerschool in Logic, Language and Information*, Lisbon

Veltman, F.: 1990, Defaults in update semantics, in H. Kamp (ed.), *Conditionals, Defaults, and Believe Revision*, Dyana deliverable R2.5.A, CCS, Edinburgh

Zeevat, H.: 1995, *Applying an Exhausification Operator in Update Semantics*, ms., University of Amsterdam

Is Compositionality of Meaning possible? *

Theo M.V. Janssen
ILLC / Fac. WINS, Univerity of Amsterdam

1 The principle of compositionality of meaning

1.1 Introduction

The principle of compositionality reads, in its best known formulation:

> The meaning of a compound expression is a function of the meanings of its parts

The principle of compositionality of meaning has immediate appeal, but at the same time it arouses many emotions. Does the principle hold for natural languages? This question cannot be answered directly, because the formulation of the principle is sufficiently vague, that anyone can put his own interpretation on the principle. One topic of investigation in this contribution is providing a more precise interpretation of the principle, and developing a mathematical model for the principle. The second topic of investigation is to discuss challenges to the principle in the literature. Methods will be presented that help to obtain compositionality. It will be argued that the principle should not be considered an empirically verifiable restriction, but a methodological principle that describes how a system for syntax and semantics should be designed.

1.2 Towards a formalization

The principle of compositionality of meaning is not a formal statement. It contains several vague words which have to be made precise in order to give formal content to the principle. In this section the first steps in this direction are made. In later sections (viz. 3, 4) mathematical formalizations are given, making it possible to prove certain consequences of the compositional approach.

Suppose that an expression E is constituted by the parts E_1 and E_2 (according to some syntactic rule). Then compositionality says that the meaning $M(E)$ of E can be found by finding the meanings $M(E_1)$ and $M(E_2)$ of respectively E_1 and E_2, and combining them (according to some semantic rule). Suppose moreover that E_1 is constituted by E_{1a} and E_{1b} (according to some syntactic rule, maybe another that the one used for E). Then the meaning $M(E_1)$ is in turn obtained from the meanings $M(E_{1a})$ and $M(E_{1b})$ (maybe according to another rule than the one combining $M(E_1)$ and $M(E_2)$). This situation is presented in figure 1.

1.3 Assumptions

The interpretation in 1.2 is a rather straightforward explication of the principle, but there are several assumptions implicit in it. Most assumptions on compositionality

*This is a shortened and adapted version of an article that will appear in J. van Benthem & A. ter Meulen (eds) *Handbook of logic and linguistics*

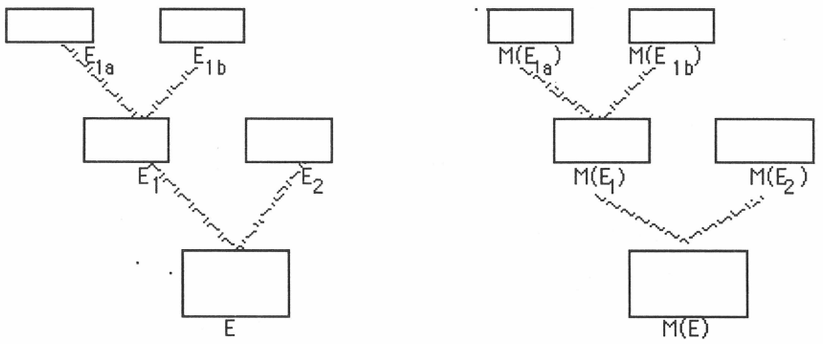


Figure 1: Compositionality: the compositional formation of expression E from its parts and the compositional formation of the meaning of E from the meanings its parts

are widely accepted, others we will return later, when the principle is discussed further. The assumptions are:

1. In a grammar the syntax and the semantics are distinguished components connected by the requirement of compositionality. This assumption excludes approaches, as in some variants of Transformational Grammar, with a series of intermediate levels between the syntax and the semantics.

2. It is assumed that the output of the syntax is the input for meaning assignment. This is for instance in contrast to the situation in Generative Semantics, where the syntactic form is projected from the meanings.

3. The rules specify how to combine the parts, i.e. they are instructions for combining expressions. So this gives a different perspective from the traditional view of a grammar as a rewriting system.

4. The grammar determines what the parts of an expression are. It depends on the rules whether *Mary does not cry* has two parts *Mary* and *does not cry*, or three *Mary, does not* and *cry*. This illustrates that **part** is a technical notion.

5. All expressions that arise as parts have meaning. This excludes systems in which only complete sentences can be assigned meaning (as in some variants of Transformational Grammar). Not only parts for which we have an intuitive meaning (as *loves* in *John loves Mary*), but also parts for which this is less intuitive (as *only* in *Only John loves Mary*). The choice what the meaning of a part is might depend on what we consider a suitable ingredient for building the meaning of the whole expression.

6. The meaning of an expression is not only determined by the parts, but also by the rule which combines those parts. From the same collection of parts several sentences can be made with different meanings (e.g. *John loves Mary* vs. *Mary loves John*). Several authors make this explicit in their formulation of the principle, e.g. Partee, ter Meulen & Wall (1990, p.318):

   > The meaning of a compound expression is a function of the meanings of its parts and of the syntactic rule by which they are combined.

7. For each syntactic rule there is a semantic rule that describes its effect. In order to obtain this correspondence, the syntactic rules should be designed appropriately. For instance, semantic considerations may influence the design of syntactic rules. This correspondence leaves open the possibility that the semantic rule is a meaning-preserving rule (no change of meanings), or that different syntactic rules have the same meaning.

8. The meaning of an expression is determined by the way in which it is formed from its parts. The syntactic production process is, therefore, the only input to the process of determining its meaning. There is no other input, so no external factors can have an effect on the meaning of a sentence. If, for instance, discourse factors should contribute to meaning, the conception of meaning has to be enriched in order to capture this.

9. The production process is the input for the meaning assignment. Ambiguous expressions must have different derivations: i.e. a derivation with different rules, and/or with different basic expressions.

1.4 Discussion

In the above discussion no position is taken as to what the nature is of the syntactic units nor what meanings are (it is left unspecified what the contents is of the boxes in figure 1). Several options are available, and the discussion whether natural language is compositional has to do with these options. If one has a definite opinion on what parts, meanings and rules should be like, then it may be doubted whether compositionality holds. But if one leaves one or more of these choices open, then the issue becomes: in which way can compositionality be obtained? These two positions will return in discussions concerning the principle of compositionality.

2 Counterexamples to compositionality

2.1 Introduction

In the present section we consider some examples from natural language that are used in the literature as arguments against compositionality. Several other examples could be given, see Partee (1984). The selection here suits to illustrate the methods available to obtain compositionality. The presentation of the examples follows closely the original argumentation; proposals for a compositional treatment are given afterwards. In the last section the methods to obtain compositional solutions are considered from a general perspective.

2.2 Counterexamples

2.2.1 Would

The need for the introduction of the NOW-operator was based upon the classical example (Kamp 1971):

(1) A child was born that will become ruler of the world.

The following more complex variants are discussed by Saarinen (1979), who argues for other new tense operators.

(2) A child was born who would become ruler of the world.

(3) Joseph said that a child had been born who would become ruler of the world.

(4) Balthazar mentioned that Joseph said that a child was born who would become ruler of the world.

Sentence (2) is not ambiguous, the moment that the child becomes ruler of the world lies in the future of its birth. Sentence (3) is twofold ambiguous: the moment of becoming ruler can be in the future of the birth, but also in Joseph's future. And in (4) the child's becoming rules can even be in Balthazars future. So the number of ambiguities increases with the length of the sentence. Therefore Hintikka (1983, pp. 276-279) presents (2)-(4) as arguments against compositionality.

2.2.2 Unless

Higginbotham (1986) presents arguments against compositionality; we discuss variants of his examples (from Pelletier (1993)). In (5) and (6) *unless* has the meaning of a (non-exclusive) disjunction.

(5) John will eat steak unless he eats lobster.

(6) Every person will eat steak unless he eats lobster.

However, in (7) the situation is different.

(7) No person will eat steak unless he eats lobster.

This sentence is to be represented as

(8) [No: person] (x eat steak $\wedge \neg$ x eats lobster).

These examples show that the meaning of *unless* depends on the context of the sentence in which it occurs. Therefore compositionality does not hold.

2.2.3 Any

Hintikka (1983, pp. 266-267) presents several interesting sentences with *any* as challenges to compositionality. Consider

(9) Chris can win any match.

In this sentence it is expressed that for all matches it holds that Chris can win them, so *any* has the impact of a universal quantification. But in (10) it has the impact of an existential quantification.

(10) Jean doesn't believe that Chris can win any match.

Analogously for the pair (11) and (12), and for the pair (13) and (14):

(11) Anyone can beat Chris.

(12) I'd be greatly surprised if anyone can beat Chris.

(13) Chris will beat any opponent

(14) Chris will not beat any opponent.

All these examples show that the meaning of the English determiner *any* depends on its environment.

The most exciting example is the one given below. As preparation, recall that Tarski required a theory of truth to result in T-schemes for all sentences:

(15) 'ϕ' is true if and only if ϕ is the case.

A classical example of this scheme is:

(16) *Snow is white* is true if and only if snow is white

The next sentence is a counterexample against one half of the Tarskian T-scheme.

(17) *Anybody can become a millionaire* is true if anybody can become a millionaire.

This sentence happens to be false.

2.3 Compositional solutions

2.3.1 Would

A compositional analysis of (18) is indeed problematic if we assume that it has to be based on (19), because (18) is ambiguous and (19) is not.

(18) Joseph said that a child had been born who would become ruler of the world.

(19) A child was born who would become ruler of the world.

However, another approach is possible: there may be two derivations for (18). In the reading that 'becoming ruler' lies in the future of Joseph's saying it may have (20) as part.

(20) say that a child was born that will become ruler of the world

The rule assigning past tense to the main clause should then deal with the 'sequence of tense' in the embedded clause, transforming *will* into *would*. The reading in which the time of becoming ruler lies in the future of the birth could then be obtained by building (18) from:

(21) say that a child was born who would become ruler of the world.

The strategy to obtain compositionality will now be clear: account for the ambiguities by using different derivations. In this way the parts of (18) are not necessarily identical to substrings of the sentences under consideration (the involved tenses may be different). Such an approach is followed for other scope phenomena with tenses in Janssen (1983).

2.3.2 Unless

Pelletier (1993) discusses the arguments of Higginbotham (1986) concerning *unless*, and presents two proposals for a compositional solution.

The first solution is to consider the meaning of *unless* to be one out of a set of two meanings. If it is combined with a positive subject (as in *every person will eat steak unless he eats lobster*) then the meaning 'disjunction' is selected, and when combined with negative subject (as in *no person eats steak unless he eats lobster*) the other meaning is selected. For details of the solution, see Pelletier (1993). So *unless* is considered as a single word, with a single meaning, offering a choice between two alternatives. This can be defined by a function from contexts to values.

The second solution is to consider *unless* a homonym. So there are two words written as *unless*. The first one is $unless_{[-neg]}$, occurring only with subjects which

bear (as is the case for *every person*) the syntactic feature [-neg], and having 'disjunction' as meaning. The second one is $unless_{[+neg]}$, which has the other meaning. Now *unless* is considered to be two words, each with its own meaning. The syntax determines which combinations are possible.

2.3.3 Any

Hintikka (1983, p.280) is explicit about the fact that his arguments concerning the non-compositionality of *any*-sentences are based upon specific ideas about their syntactic structure. In particular it is assumed that (22) is a 'component part' of (23)

(22) Anyone can beat Chris

(23) I'll be greatly surprised if anyone can beat Chris.

He claims that this analysis is in accordance with common sense, and in agreement with the best syntactic analysis. But, as he admits, other analyses cannot be excluded a priori; for instance that (24) is a component of (23).

(24) I'll be greatly surprised if — can beat Chris.

One might even be more radical in the syntax than Hintikka suggests, and introduce a rule that produces (23) from

(25) Someone can beat Chris.

Partee (1984) discusses the challenges of *any*. She shows that the situation is more complicated than suggested by the examples of Hintikka. Sentence (27) has two readings, only one of which can come from (26).

(26) Anyone can solve that problem.

(27) If anyone can solve that problem, I suppose John can.

Partee discusses the literature concerning the context-sensitivity of *any*, and concludes that here are strong arguments for two 'distinct' *any*'s: an *affective any* and a *free-choice any*. The two impose distinct (though overlapping) constraints on the contexts in which their semantic contributions 'make sense'. The constraints on affective *any* can be described in model-theoretic terms, whereas those of the free-choice *any* are less well understood. For references concerning this discussion see Partee (1984).

We conclude that the *any*-examples can be dealt with in a compositional way by distinguishing ambiguous *any*, with one or both readings eliminated when incompatible with the surrounding context.

2.4 General methods for compositionality

In this section we have encountered three methods to obtain compositionality:

1. New meanings.
   These are formed by the introduction of a new parameter, or alternatively, a function from such a parameter to old meanings. This was the first solution for *unless*

2. New basic parts
   Duplicate basic expressions, together with different meanings for the new expressions, or even new categories. This was the solution for *any*, and the second solution for *unless*.

3. New constructions
   Use unorthodox parts, together with new syntactic rules forming those parts and rules operating on those parts. This approach may result in abstract parts, new categories, and new methods to form compound expressions. This was the solution for the *would* sentences.

For most of counterexamples several of these methods are in principle possible, and a choice must be motivated. That is not an easy task because the methods are not just technical tools to obtain compositionality: they raise fundamental questions concerning the syntax and semantics interface. If meanings include a new parameter, then meanings have this parameter in the entire grammar, and it must be decided what role the parameter plays. If new basic parts are introduced, then each part should have meaning, and each part is available everywhere. If new constructions are introduced, they can be used everywhere. Other expressions may then be produced in new ways, and new ambiguities may arise. So adopting compositionality raises fundamental questions about what meanings are, what the basic building blocks are and what ways of construction are.

The real question is not whether a certain phenomenon can be analyzed compositionally, as enough methods are available, but what makes the overall theory (un)attractive or (un)acceptable.

3 A mathematical model of compositionality

3.1 Introduction

In this section a mathematical model is developed that describes the essential aspects of compositional meaning assignment. The assumptions leading to this model have been discussed in section 1.2. The model is closely related to the one presented in 'Universal Grammar' (Montague 1970). The mathematical tools used in this section are tools from *Universal Algebra*, a branch of mathematics that deals with general structures; a standard textbook is Graetzer (1979). For easy reference, the principle is repeated here:

> The meaning of a compound expression is a function of the meanings of its parts and of the syntactic rule by which they are combined.

3.2 Algebra

The first notion to be considered is *parts*. Since the information on how expressions are formed is given by the syntax of a language, the rules of the grammar determine what the parts of an expression are. The rules build new expressions from old expressions, so they are operators taking inputs and yielding an output. A syntax with this kind of rules is a specific example of what is called in mathematics an *algebra*. Informally stated, an algebra is a set with functions defined on that set. After the formal definitions some examples will be given.

Definitions 3.1. *An* **Algebra** $\mathcal{A}$*, consists of a set* A *called the* **carrier** *of the algebra, and a set* F *of functions defined on that set. So* $\mathcal{A} = \langle A, F \rangle$*. The elements of the carrier are called the* **elements** *of the algebra. Instead of the name function, often the name* **operator** *is used. If an operator is not defined on the whole carrier, it is called a* **partial operator**. *If* $F(E_1, E_2, .., E_n) = E$ *, then* $E_1, E_2, \ldots,$ *and* E_n *are called* **parts** *of* E*. If an operator takes* n *arguments, it is called an* **n-ary operator**.

The notion *set* is a very general notion, and so is the notion *algebra* which has a set as one of its basic ingredients. This abstractness that makes algebras suitable models for compositionality, because it is abstracted from the particular grammatical theory. Three examples of a completely different nature will be considered.

1. The algebra $\langle \mathbf{N}, \{+, \times\}\rangle$ of natural numbers $\{1, 2, 3, \ldots\}$, with addition and multiplication as operators.
2. The set of trees (constituent structures) and the operation of making a new tree from two old ones by giving them a common root.
3. The carrier consists of the words *boy, girl, apple, pear, likes, takes ,the* and all possible strings that can be formed from them. There are two partial defined operations. R_{def} forms from a common noun a noun-phrase by adding the article *the*. R_S forms a sentence from two noun-phrases and a verb. Examples of sentences are *The boy likes the apple* and *The pear takes the girl.*

In order to avoid the misconception that anything is an algebra, finally a *non-example*. Take the second algebra (finite strings of words with concatenation), and add an operator that counts the length of a string. This not an algebra any more, since the lengths (natural numbers) are not elements of the algebra.

3.3 Generators

Next we will define a subclass of the algebras, viz. the finitely generated algebras. To give an example, consider the subset $\{1\}$ in the algebra $\langle \mathbf{N}, \{+\}\rangle$ of natural numbers. By application of the operator $+$ to elements in this subset, that is by calculating $1 + 1$, one gets 2. Then 3 can be produced (by $2 + 1$, or $1 + 2$), and in this way the whole carrier can be obtained. Therefore the subset $\{1\}$ is called *a generating set* for this algebra. Since this algebra has a finite generating set, it is called a finitely generated algebra. If we have in the same algebra the subset $\{2\}$, then only the even numbers can be formed. Therefore the subset $\{2\}$ is *not* a generating subset of the algebra of natural numbers. On the other hand, the even numbers form an algebra, and $\{2\}$ is a generating set for that algebra. More generally, any subset is generating set for some algebra. This can be seen as follows. If one starts with some set, and adds all elements that can be produced from the given set and from already produced elements, then one gets a set that is closed under the given operators. Hence it is an algebra.

Definitions 3.2. *Let $\mathcal{A} = \langle A, F\rangle$ be an algebra, and H be a subset of A. Then $\langle [H], F\rangle$ denotes the smallest algebra containing H, and is called the by H* **generated subalgebra**. *If $\langle [H], F\rangle = \langle A, F\rangle$, then H is called a generating set for $\mathcal{A}$. The elements of H are called* **generators**. *If H is finite, then $\mathcal{A}$ is called a* **finitely generated** *algebra.*

The first example in section 3.2 is a finitely generated algebra because

$$\langle \mathbf{N}, \{+, \times\}\rangle = \langle [\{1\}], \{+, \times\}\rangle.$$

The last example (with the set of strings over a lexicon) is finitely generated: the lexicon is the generating set. An algebra that is not finitely generated is $\langle \mathbf{N}, \{\times\}\rangle$, the natural numbers with multiplication (it is generated by the set of prime numbers).

A grammar that is suitable for a compositional meaning assignment has to be a generated algebra. Furthermore, some criterion is needed to select certain elements of the algebra as the generated language. For instance the expressions that are output of certain rules, or, (if the grammar generates tree like structures) the elements with root labeled S.

Definition 3.3. *A compositional grammar is a pair $\langle \mathcal{A}, S\rangle$, where $\mathcal{A}$ is a generated algebra $\langle A, F\rangle$, and S a selection predicate that selects a subset of A, so $S(A) \subseteq A$.*

3.4 Terms

In section 1.2 it was argued that *way of production* is crucial for the purpose of meaning assignment. Therefore it is useful to have a representation for such a production process or derivational history. In section 1.2 we represented such a derivation by means of a tree. That is not the standard format. Let us first consider the linguistic example given in section 3.2. By application of the operator R_{Def} to the noun *apple*, the noun phrase *the apple* is formed, and likewise *the boy* is formed by application of R_{Def} to *boy*. Next the operator R_S is applied to the just formed noun phrases and the verb *like*, yielding the sentence *the boy likes the apple*. This process is described by the following expression (sequence of symbols):

(1) $R_S\langle R_{Def}\langle boy\rangle, R_{Def}\langle apple\rangle, like\rangle$

Such expressions are called **terms**. There is a simple relation of the terms to the elements in the original algebra. For instance, with the term $R_{Def}\langle apple\rangle$ corresponds an element which is found by evaluating the term (i.e. executing the operator on its arguments), viz. the string *the apple*. In principle, different terms may evaluate to the same element, and the evaluation of a term usually is very different from the term itself. Terms can be combined to form new terms: the term (1) above, is formed from the terms $R_{Def}\langle apple,\rangle$, $R_{Def}\langle boy\rangle$ and *like*. Thus the terms over an algebra form an algebra themselves.

Definition 3.4. *Let $\mathcal{B} = \langle [B], F\rangle$ be an algebra. The set of* **terms** *over $\mathcal{B} = \langle [B], F\rangle$, denoted as $T_{B,F}$, is defined as follows:*

1. *for each element in B a new symbol $b \in T_{B,F}$*
2. *For every operator in F there is a new symbol f. If f corresponds with a n-ary operator and $t_1, t_2, \ldots t_n \in T_{B,F}$, then $f\langle t_1, t_2, \ldots t_n\rangle \in T_{B,F}$.*

The terms over $\mathcal{B} = \langle [B], F\rangle$ form an algebra with as operators combinations of terms according to the operators of B. This algebra is called the **term algebra** *over $\langle [B], F\rangle$. This term algebra is denoted $\mathcal{T}_{B,F}$, or shortly $\mathcal{T}_B$.*

In section 1.2 it was argued that, according to the principle of compositionality of meaning, the derivation of an expression determines its meaning. Hence the meaning assignment is a function defined on the term algebra.

3.5 Homomorphisms

The principle of compositionality does not only tell us on which objects the meaning is defined (terms), but also in which way this has to be done. Suppose we have an expression obtained by application of operation f to arguments $a_1, \ldots, a_n$. Then its translation in algebra $\mathcal{B}$ should be obtained from the translations of its parts, hence by application of an operator g (corresponding with f) to the translations of $a_1, \ldots, a_n$. So, if we let Tr denote the translation function, we have

$$Tr(f\langle a_1, \ldots, a_n\rangle) = g(Tr(a_1), ..Tr(a_n))$$

Such a mapping is called a *homomorphism*. Intuitively speaking, a homomorphism h from an algebra $\mathcal{A}$ to algebra $\mathcal{B}$ is a mapping which respects the structure of $\mathcal{A}$ in the following way. If in $\mathcal{A}$ an element a is obtained by means of application of an operator f, then the image of a is obtained in $\mathcal{B}$ by application of an operator

corresponding with f. The structural difference that may arise between $\mathcal{A}$ and $\mathcal{B}$ is that two distinct elements of $\mathcal{A}$ may be mapped to the same element of $\mathcal{B}$, and that two distinct operators of $\mathcal{A}$ may correspond with the same operator in $\mathcal{B}$.

Definition 3.5. *Let $\mathcal{A} = \langle A, F\rangle$ and $\mathcal{B} = \langle B, G\rangle$ be algebras. A mapping $h : \mathcal{A} \to \mathcal{B}$ is called a* **homomorphism** *if there is a 1-1 mapping $h' : F \to G$ such that for all $f \in F$ and all $a_1, \ldots, a_n \in A$ holds $h(f(a_1, \ldots, a_n)) = h'(f)(h(a_1), \ldots, h(a_n))$*

Now that the notions 'terms' and' homomorphisms' are introduces, all ingredients are present needed to formalize 'compositional meaning assignment'.

> A compositional meaning assignment for a language A in a model B is obtained by designing an algebra $\langle [G], F\rangle$ as syntax for A, an algebra $\langle [H], F\rangle$ for B, and by letting the meaning assignment be a homomorphism from the term algebra T_A to $\langle [H], G\rangle$.

3.6 Polynomials

Usually the meaning assignment is not direct, but indirectly via a translation into a logical language. The standard way to do this is by using polynomials. Here the algebraic background of this method will be investigated.

First the definition. A polynomial is term with variables, so

Definitions 3.6. *Let $\mathcal{B} = \langle [B], F\rangle$ be an algebra. The set* $\mathbf{Pol^n_{\langle [B],F\rangle}}$ *– shortly Pol^n – of* **n-ary polynomial symbols**, *or n-ary polynomials, over the algebra $\langle [B], F\rangle$ is defined as follows:*

1. *For every element in B there is a new symbol (a constant) $b \in Pol^n$.*
2. *For every n, with $1 \leq i \leq n$, the variable $x_i \in Pol^n$*
3. *For every operator in F there is a new symbol. If f corresponds with a n-ary operator, and $p_1, p_2, \ldots, p_n \in Pol^n$ then also $f(p_1, p_2, \ldots, p_n) \in Pol^n$.*

The set $Pol_{\langle [B],F\rangle}$ of **polynomial symbols** *over algebra $\langle [B], F\rangle$ is defined as the union for all n of the n-ary polynomial symbols, shortly $Pol = \bigcup_n Pol^n$.*

A polynomial symbol $p \in Pol^n$ defines an n-ary polynomial operator; its value for n given arguments is obtained by evaluating the term that is obtained by replacing x_1 by the first argument x_2 by the second, etc.

Given an algebra $\langle [B], F\rangle$ and a set P of polynomial over A, we obtain a new algebra $\langle [B], P\rangle$ by replacing the original set of operators by the polynomial operators. An algebra obtained in this way is a **polynomially derived algebra**.

If an operation is added to a given logic, it should be an operation on meanings. In other words, whatever the interpretation of the logic is, the new operator should have a unique semantic interpretation. This is expressed in the definition below, where h is a compositional meaning assignment to the original algebra, and h' describes the interpretation of new operators.

Definition 3.7. *Let $\langle [A], F\rangle$ be an algebra. A collection operators G is called* **safe** *if for all algebras $\mathcal{B}$ and all surjective homomorphisms h from $\mathcal{A}$ onto $\mathcal{B}$ holds that there is a unique algebra $\mathcal{B}'$ such that the restriction h' of h to the elements of $\langle [A], G\rangle$ is a surjective homomorphism.*

This definition is illustrated in figure 2.

Theorem 3.8 ((Montague 1970))**.** *Polynomial operators are safe.*

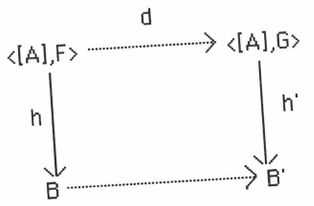


Figure 2: G is safe if for all $\mathcal{B}$ there is a unique $\mathcal{B}'$ such that h', the restriction of h, is a surjective homomorphism

Proof (sketch) Mimic the polynomial operators in the homomorphic image.
There are of course other methods to define operations on logic, but safeness is then not guaranteed. Examples are

- Replaces all occurrences of x by y
  There is no semantic interpretation for this operator because some of the new y's may become bound. So there is no algebra $\mathcal{B}'$ in the sense of the above theorem.
- Replace all existential quantifiers by universal ones
  For equivalent formulas (e.g. where one formula has $\forall$ and the other $\neg\exists\neg$) non-equivalent results are obtained.
- Recursion on the length of a formula
  In the model for logic this notion (operator) has no interpretation, hence the recursion is not well-founded in the model.

The use of polynomials is not a restriction on the expressive power, as follows from the next theorem.

Theorem 3.9. *Let $\langle A, F\rangle$ be an algebra with infinitely many generators, and G a collection of safe operators over$\langle A, F\rangle$. Then all elements of G are polynomially definable.*

Proof A proof for this theorem is given by van Benthem (1979), and for many sorted algebras by F. Wiedijk in Janssen (1986*a*).

Theorem 3.9 is important for applications since it justifies the restriction to polynomially defined operators. Suppose one introduces a new operator, then either it is safe, and polynomially definable, or it is not safe, and consequently should not be used. In applications the requirement of infinitely many generators is not a real restriction, since the logic usually has indexed variables $x_1, x_2, x_3, \ldots$ Furthermore it is claimed (Wiedijk pers. comm.) that the theorem holds for any algebra with at least two generators.

We may summarize the section by giving the formalization of the principle of compositionality of meaning.

> Let L be some language. A compositional meaning assignment to L is obtained as follows. We design for L a compositional grammar $\mathcal{A} = \langle\langle A_L, F_L\rangle, S_L\rangle$, and a compositional grammar $\mathcal{B} = \langle\langle B, G\rangle, S_B\rangle$ to represent the meanings, where B has a homomorphic interpretation in some

model M. The meaning assignment for L is defined by a homomorphism on from T_A to an algebra that is polynomially derived from $\mathcal{B}$.

3.7 Developments of the model

The algebraic framework presented here is almost the same as the one developed by Montague in Universal Grammar (Montague 1970). That article was written in a time that the mathematical theory of universal algebra was rather young (the first edition of the main textbook in the field (Graetzer 1979) originates from 1968). The notions used in this section are the notions that are standard nowadays, and differ at some cases from the ones used by Montague. For instance, he uses a 'disambiguated language', where we use a 'term algebra', notions which, although closely related, differ not only by name. The algebraic model developed by Montague turned out to be the same as the model used in computer science in the approach to semantics called *initial algebra semantics* (Adj 1978), as was noticed by Janssen & van Emde Boas (1981). Montague's framework is redesigned using many sorted algebras in Janssen (1986*a*) and Janssen (1986*b*); that framework is developed further for dealing with flexibility in Hendriks (1993).

4 The formal power of compositionality

4.1 Introduction

In the present section the power of the framework with respect to the generated language and the assigned meanings will be investigated. It will be shown that on the one hand compositionality is restrictive in the sense that, in some circumstances, a compositional analysis is impossible. On the other hand it will be shown that compositionality does not restrict the class of languages that can be analyzed, nor the meanings that can be assigned. Finally a restriction will be considered that guarantees recursiveness.

4.2 Not every grammar can be used

In the preceding sections examples are given which illustrate that not every grammar is suitable for a compositional meaning assignment. The example below gives a formal underpinning of this. A grammar for a language is given, together with the meanings for its expressions. It is proven that it is not possible to assign the given meanings in a compositional way to the given grammar.

Example 4.1. *The basic expressions are the digits:* $\{0,1,2,3,4,5,6,7,8,9\}$. *There are two operations in the algebra. The first one makes from a digit a number name and is defined by* $G_1(d) = d$. *The second one makes from a digit and a number name a new number name by writing the digit in front of the number name:* $G_2(d,n) = dn$. *So* $G_2(2, G_1(3)) = G_2(2,3) = 23$, *and* $G_2(0, G_1(6)) = G_2(0,6) = 06$. *The meaning of an expression is the natural number it denotes, so* 007 *has the same meaning as* 7. *This meaning function is denoted by M.*

Fact 4.2. *There is no function F such that* $M(G_2(a,b)) = F(M(a), M(b))$.

Proof Suppose that there was such an operation F. Since $M(7) = M(007)$, we would have

$$M(27) = M(G_2(2,7)) = F(M(2), M(7)) = F(M(2), M(007)) = M(G_2(2,007)) = M(2007)$$

This is a contradiction. Hence no such operation F can exist.
End of Proof

This result is from Janssen (1986*a*); in Zadrozny (1994) a weaker result is proved, viz. that there does not exist a polynomial F with the required property.

A compositional treatment can be obtained by changing rule G_2. The digit should be written at the end of the already obtained number: $G_3(d,n) = nd$. Then there is a corresponding semantic operation F defined by $F(d,n) = 10 \times n + d$, for instance $M(07) = M(G_3(7,0) = F(M(7), M(0)) = 10 \times M(0) + M(7)$. So a compositional assignment of the intended meaning is possible, but requires another syntax. This illustrates that compositionality becomes possible if semantic considerations influence the design of the syntactic rules.

4.3 Power from syntax

The next theme is the (generative) power of compositional grammars and of compositional meaning assignment. In this section we will consider the results of (Janssen 1986*a*), and in the next section those of (Zadrozny 1994).

In the theorem below it is proved that any recursively enumerable language can be generated by a compositional grammar. The recursively enumerable languages form the class of languages which can be generated by the most powerful kinds of grammars (unrestricted rewriting systems, transformational grammars, Turing machine languages etc.), or, more generally, by any kind of algorithm. Therefore, the theorem shows that if a language can be generated by any algorithm, it can be generated by a compositional grammar. The proof exploits the freedom of compositionality to choose some suitable grammar. The basic idea is that the rules of the grammar (operations of the algebra) can simulate a Turing Machine.

Theorem 4.3. *Any recursively enumerable language can be generated by a compositional grammar.*

Proof In order to prove the theorem, we will simulate a nondeterministic Turing machine of the following type. The machine operates on a tape that has a beginning but no end, and it starts on an empty tape with its read/write head placed on the initial blank. The machine acts on the basis of its memory state and of the symbol read by the head. It may move right (R), left (L) or print a symbol, together with a change of memory state. Two examples of instructions are

I_1 : $q_1 s q_2 R$ (= if the Turing machine reads in state q_1 an s, then its state changes in q_2 and its head moves to the right)

I_2 : $q_1 s q_2 t$ (= if the Turing machine reads in state q_1 an s, then its state changes in q_2 and it writes an t)

The machine halts when no instruction is applicable. Then the string of symbols on the tape (neglecting the blanks) is the generated string. The set of all the strings the nondeterministic machine can generate is the generated language.

A compositional grammar is of another nature than a Turing Machine. A grammar does not work with infinite tapes, and it has no memory. These features can be encoded by a finite string in the following way. In any stage of the calculations, the head of the Turing machine has passed only a finite number of positions on the tape. That finite string determines the whole tape, since the remainder is filled with blanks. The current memory state is inserted as an extra symbol in the string on a position to the left of the symbol that is currently scanned by the head. Such strings are elements of the algebra.

Each instruction of the Turing machine will be mimicked by an operation of the algebra. This will be shown below for the two examples mentioned before. Besides this, some additional operations are needed: operations that add additional blanks to the string if the head stands on the last symbol on the right and has to move to the right, and operations that remove at the end of the calculations the state symbol and the blanks from the string. These additional operations will not be described in further detail.

I_1 : The corresponding operator F_1 is defined for strings of the form w_1qsw_2 where w_1 and w_2 are arbitrary strings consisting of symbols from the alphabet and blanks. The effect of F_1 is defined by $F_1(w_1q_1sw_2) = w_1sq_2w_2$.

I_2 : The corresponding operator F_2 is defined for strings of the form $F_2(w_1q_1sw_2) = w_1q_2tw_2$.

Since the algebra imitates the Turing machine, the generated language is the same.
End of Proof The above result can be extended to meanings. The theorem below says that any meaning can be assigned to any language in a compositional way.

Theorem 4.4. *(Any language, any meaning)*
Let L be a recursively enumerable language, and $M : L \to D$ a computable function of the expressions of L into D. Then there are algebras for L and D with computable operations such that M is an homomorphism.

Proof In the proof of theorem 4.3 the existence is proven of an algebra $\mathcal{A}$ as syntax for the source language L. A variant $\mathcal{A}'$ of $\mathcal{A}$ is taken as grammar for L: the rules produce strings that end with a single #-sign, and an additional rule, say $R_\#$ removes that #. For the semantic algebra a copy of $\mathcal{A}'$ is taken, but instead of $R_\#$ there is a rule R_M that performs the meaning assignment M. Since M is computable, so is R_M. The syntactic rules of $\mathcal{A}'$ extended with $R_\#$ are in a one to one correspondence with the rules of $\mathcal{A}'$ extended with R_M. Hence the meaning assignment is an homomorphism.
End of Proof

4.4 Power from semantics

Zadrozny proves that any semantics can be dealt with in a compositional way. He takes a version of compositionality that is most intuitive: in the syntax only concatenation of strings is used. On the other hand, he exploits the freedom to use unorthodox meanings. Let us quote his theorem (Zadrozny 1994):

Theorem 4.5. *Let M be an arbitrary set. Let A be an arbitrary alphabet. Let "." be a binary operation, and let S be the set closure of A under ".". Let $m : S \to M$ be an arbitrary function. Then there is a set of functions M^* and a unique map $\mu : S \to M^*$ such that for all $s, t \in S$*

$$\mu(s.t) = \mu(s)(\mu(t)), \text{ and } \mu(s)(s) = m(s)$$

The first equality says that μ obeys compositionality, and the second equality says that from $\mu(s)$ the originally given meaning can be retrieved. The proof roughly proceeds as follows. The requirement of compositionality is formulated by an infinite set of equations concerning μ. Then a basic lemma from non-wellfounded set theory is evoked, the *solution lemma*. It guarantees that there is a unique solution for this set of equations – in non-wellfounded set theory. This non-wellfounded set theory is a recently developed model for set theory in which the axiom of foundation does not hold. Zadrozny claims that the result also holds if the involved functions are restricted to computable ones.

On the syntactic side this result is very attractive. It formalizes the intuitive version of compositionality: in the syntax there is concatenation of visible parts. However it remains to be investigated for which class of languages this result holds; with a partially defined computable concatenation operation only recursive languages can be generated.

Zadrozny claims that the result also holds if the language is not specified by a (partial) concatenation operation, but by a Turing Machine. However, then the attractiveness of the result disappears (the intuitive form of compositionality), and the same result is obtained as described in the previous section (older and with standard mathematics).

On the semantic side some doubts can be raised. The given original meanings are encoded using non wellfounded sets. Since the original sentence is parts of this encoding, each sentence has its own, distinct meaning. It is strange, however, that synonymous sentences get different meanings. Furthermore it is unclear, given two meanings, how to define a useful entailment relation among them.

In spite of these critical comments, the result is a valuable contribution to the discussion of compositionality. It shows that if we restrict the syntax considerably, but are very liberal in the semantics, a lot more is possible than expected. In this way the result is complementary to the results in the previous section. Together the results of Janssen and Zadrozny illustrate that without constraints on syntax and semantics, there are no counterexamples to compositionality. This gives the pleasant feeling that a compositional treatment is somehow always possible.

It has been suggested that restrictions should be proposed because compositionality is now a vacuous principle. That is not the opinion of this author. The challenge of compositional semantics is not to prove the existence of such a semantics, but to obtain one. The formal results do no help in this respect because the proofs of the theorems assume that some meaning assigning function is already given, and then turn it into a compositional one. Compositionality is not vacuous, because we have no recipe to obtain one, and because several proposals are ruled out by the principle. Restrictions should therefore have another motivation. The challenge of semantics is to design a function that assigns meanings, and the present paper argues that the best method is to do so in a compositional way.

4.5 Restriction to recursiveness

In this section a restriction will be discussed that reduces the generative capacity of compositional grammar to recursive sets. The idea is to use rules that are reversible. If a rule is used to generate an expression, the reverse rule can be used to parse that expression. Let us consider an example.

Suppose that there is a rule specified by $R_1(\alpha, \beta, \gamma) = \alpha\ \beta s\ \gamma$. So:

$$R_1(\textit{every man, love, a woman}) = \textit{every man loves a woman}$$

The idea is to introduce a rule R_1^{-1} such that

$$R_1^{-1}(\textit{every man loves a woman}) = \langle \textit{every man, love, a woman} \rangle$$

In a next stage other reverse rules might investigate whether the first element of this tuple is a possible noun phrase, whether the second element is a transitive verb, and whether the third element is a noun phrase. A specification of R_1^{-1} might be: find a word ending on an s, consider the expression before the verb as the first element, the verb (without the s) as the second, and the expression after the verb as the third element. Using reverse rules, a parsing procedure can easily be designed.

The following complications may arise with R_1^{-1} or with another rule:

- **Ill-formed input**
  The input of the parsing process might be a string that is not a correct sentence, e.g *John runs Mary*. Then the given specification of R_1^{-1} is applicable. It is not attractive to make the rule so restrictive that it cannot be applied to ill-formed sentences, because then rule R_1^{-1} would be as complicated as the whole grammar.

- **Applicable on several positions**
  An application of R_1^{-1} (with the given specification) to *The man who seeks Mary loves Suzy* can be applied both to *seeks*, and to *loves*. The information that *the man who* is not a noun-phrase can only be available when the rules for noun-phrase formation are considered. As in the previous case, it is not attractive to make the formulation of R_1^{-1} that restrictive that is is only applicable to wellformed sentence.

- **Infinitely may sources**
  A rule may remove information that is crucial for the reversion. Suppose that a rule deletes all words after the first word of the sentence. Then for a given output, there is an infinite collection of strings that have to be considered as possible inputs.

The above points illustrate that the reverse rule cannot be an inverse function in the mathematical sense. In order to account for the first two points, it is allowed that the reverse rule yields a set of expressions. In order to avoid the last point, it is required that is a finite set.

Requiring that there is a reverse rule, is not sufficient to obtain a parsing algorithm. For instance, it may be the case the $y \in R_1^{-1}(y)$, and a loop arises. In order to avoid this, it is required that all the rules form expressions which are more complex (in some sense) than their inputs, and that the reverse rule yields expressions that are less complex than the input. Now there is a guarantee that the process of reversion terminates.

The above considerations lead to two restrictions on compositional grammars which together guarantee recursiveness of the generated language. The restrictions are a generalization of the ones in Landsbergen (1981), and provide the basis of the parsing algorithm of the machine translation system 'Rosetta' (see Rosetta (1994)) and of the parsing algorithm in Janssen (1989).

1. **Reversibility**
   For each rule R there is a reverse rule R^{-1} such that
   (a) for all y the set $R^{-1}(y)$ is finite
   (b) $y = R(x_1, x_2, \ldots, x_n)$ if and only if $\langle x_1, x_2, \ldots, x_n \rangle \in R^{-1}(y)$
2. **Measure condition**
   There is a computable function μ hat assigns to an expression a natural number: its measure. Furthermore
   (a) If $y = R(x_1, x_2, ...x_n)$, then $\mu(y) > \max(\mu(x_1), \mu(x_2,), \ldots, \mu(x_n))$
   (b) If $\langle x_1, x_2, \ldots, x_n \rangle \in R^{-1}(y)$ then $\mu(y) > \max(\mu(x_1), \mu(x_2,)...\mu(x_n))$

Assume a given grammar together with reverse rules and a computable measure condition. A parsing algorithm for M-grammars can be based upon the above two restrictions. Condition 1 makes it possible to find, given the output of a generative rule, potential inputs for the rule. Condition 2 guarantees termination of the recursive application of this search process. So the languages generated by grammars satisfying the requirements are decidable languages. Note that the grammar in the proof of theorem 4.3 does not satisfy the requirements, since there is no sense in which the complexity increases, if the head moves to the right or the left.

5 Conclusion

The principle of compositionality of meaning really means something. It is a restriction that rules out several proposals in the literature, and is certainly not vacuous. On the other hand it was shown that there are several methods to obtain a compositional meaning assignment; so it is not an impossible task. For counterexamples to compositionality solutions were proposed. This practical experience was supported by mathematical proofs that the sentences of any language can be assigned any meaning in a compositional way. However, the formal results do not make it any easier to obtain a compositional semantics, so these results form no reason for restrictions.

Compositionality is not a formal restriction on what can be achieved, but a methodology on how to proceed. Compositionality requires a decision on what in a given approach the basic semantic units are: if one has to build meanings from them, it has to be decided what these units are. It also requires a decision on what the basic units in syntax are, and how they are combined. If a proposal is not compositional, it is an indication that the fundamental question what the basic units are, is not answered satisfactorily. If such an answer is provided, the situation under discussion is better understood. So the main reason to follow this methodology, is that compositionality guides research in the right direction!

Acknowledgments

I am indebted to Johan van Benthem, Peter van Emde Boas, Yuri Engelhardt, J. Goguen, Willem Groeneveld, Herman Hendriks, Barbara Partee, F.J. Pelletier, the participants of the 'handbook-workshop' and especially to Ede Zimmermann for their comments on other versions of this article. For their stimulating guidance I thank Johan van Benthem and Alice ter Meulen, who also turned my extraordinary English into intelligible prose.

References

Adj (1978), An initial algebra approach to the specification, correctness and implementation of abstract data types, *in* R. Yeh, ed., 'Current Trends in Programming Methodology', Prentice Hall, pp. 80–149. Adj = {J.A. Goguen, J.W. Thatcher, E.G. Wagner}.

van Benthem, J.F.A.K (1979), Universal algebra and model theory. Two excursions on the border, Technical Report ZW-7908, Dept. of mathematics, Groningen University.

Graetzer, G (1979), *Universal Algebra. Second Edition*, Springer, New York. First edition published by van Nostrand, Princeton, 1968.

Hendriks, H. (1993), Studied Flexibility. Categories and types in Syntax and Semantics, PhD thesis, University of Amsterdam. in ILLC dissertation Series 1993-5.

Higginbotham, J. (1986), Linguistic theory and Davidson's program in semantics, *in* E. le Pore, ed., 'Truth and Interpretation. Perspectives on the Philosophy of Donald Davidson', Blasil Blackwell, Oxford, pp. 29–48.

Hintikka, J. (1983), *The Game of Language. Studies in Game-Theoretical Semantics and Its Applications*, number 22 *in* 'Synthese Language Library', Reidel, Dordrecht. In collaboration with J. Kulas.

Janssen, T.M.V. (1983), Scope ambiguities of tense, aspect and negation, *in* F. Heny & B. Richards, eds, 'Syntactic categories: Auxiliaries and related puzzles', number 20 *in* 'Synthese Language Library', Reidel, Dordrecht, pp. 55–99.

Janssen, T.M.V. (1986*a*), *Foundations and Applications of Montague Grammar: part 1, Philosophy, Framework, Computer Science*, number 19 *in* 'CWI tract', Centre for Mathematics and Computer Science, Amsterdam.

Janssen, T.M.V. (1986*b*), *Foundations and Applications of Montague Grammar: part 2, Applications to Natural Language*, number 28 *in* 'CWI tracts', Centre for Mathematics and Computer Science, Amsterdam.

Janssen, T.M.V. (1989), Towards a universal parsing algoritm for functional grammar, *in* J.H. Conally & S.C.Dik, eds, 'Functional grammar and the computer', Foris, Dordrecht, pp. 65–75.

Janssen, T.M.V. & van Emde Boas, P. (1981), Some remarks on compositional semantics, *in* D. Kozen, ed., 'Logic of Programs', number 131 *in* 'Springer Lecture Notes in Computer Science', Springer, Berlin, pp. 137–149.

Kamp, H. (1971), 'Formal properties of "now"', *Theoria* **37**, 227–273.

Landsbergen, J. (1981), Adaption of Montague grammar to the requirements of parsing, *in* Groenendijk J.A.G., T.M.V. Janssen & M.B.J. Stokhof, eds, 'Formal methods in the study of language. Proceedings of the third Amsterdam colloquium', number 135 & 136 *in* 'CWI Tracts', Centre for Mathematics and Computer Science, Amsterdam, pp. 399–420. 2 volumes.

Montague, R. (1970), 'Universal grammar', *Theoria* **36**, 373–398. Reprinted in Thomason (1974, pp. 222-246).

Partee, B. (1984), Compositionality, *in* F. Landman & F. Veltman, eds, 'Varieties of Formal Semantics', number 3 *in* 'GRASS', Foris, Dordrecht, pp. 281–311.

Partee, B., ter Meulen, A. & Wall, R.E. (1990), *Mathematical Methods in Linguistics*, number 30 *in* 'Studies in Linguistics and Philosophy', Kluwer, Dordrecht.

Pelletier, F.J. (1993), On an argument against semantic compositionality, *in* D. Westerståhl, ed., 'Logic, Methodology and Philosophy of Science', Kluwer, Dordrecht.

Rosetta, M.T. (1994), *Compositional Translation*, The Kluwer International Series in Engineering and Computer Science 230, Kluwer, Dordrecht. (M.T. Rosetta = L. Appelo, T. Janssen, F. de Jong, J. Landsbergen (eds.)).

Saarinen, E. (1979), Backwards-looking operators in tense logic and in natural language, *in* J. Hintikka, I. Niiniluotov & E. Saarinen, eds, 'Essays on mathematical and philosophical logic. Proceedings of the 4th. Scandinavian logic symposium', number 122 *in* 'Synthese Library', Kluwer, Dordrecht, pp. 341–367.

Thomason, R.H. (1974), *Formal Philosophy. Selected Papers of Richard Montague*, Yale University Press, New Haven.

Zadrozny, W. (1994), 'From compositional to systematic semantics', *Linguistics and Philosophy* **17**, 329–342.

A Programme of Modal Unification of Dynamic Theories

Jan Jaspars[1] **Emiel Krahmer**[2]

1 Introduction

Change is an important concept in such diverse research areas as computer science, artificial intelligence, cognitive science and linguistics. This has led to a great proliferation of so-called *logics of change* (a.k.a. *dynamic logics*) in these areas over the past twenty years. In all these theories different semantic parameters are 'dynamified'. For example, in many discourse logics the dynamics resides on the level of variable-assignments, while in the update logics it lives on the level of possible worlds[3]. But even within the single groups there is no homogeneity. If we take the discourse logics as example, they are mutually divergent in the way the assignments are used in the interpretation. Since in different systems different aspects of the interpretation are dynamified, comparing and/or unifying the various logics is a non-trivial yet interesting task. [MvBV95] note that developing a single logic in which all relevant aspects of semantics are combined will surely lead to 'a disaster' and that it would be more useful to have 'one common general purpose logic', in which the various approaches can work 'in tandem'.

In this paper we intend to come to a *bird's-eye view* on the gamut of dynamic theories. To arrive at this point we will take the modal perspective. Given the modal character of Pratt's original dynamic logic, the contents of this paper can be characterized as a *re-modal-ization* of dynamic semantics. This remodalization is carried through in the style of the *Dynamic Modal Logic* (DML) of [Ben91] and [Rij92], which was developed to model general reasoning about expanding and reducing information states. The model-theory of DML is based on the possible world models, called *information models* here, which Kripke introduced for Heyting's intuitionistic logic in [Kri65] (according to this perspective, intuitionistic logic is a dynamic logic *avant la lettre*). An information-model M for a certain language $\mathcal{L}$ looks as follows: $M = \langle S, \sqsubseteq, [\![.]\!] \rangle$, where S is a non-empty set of information-states, $\sqsubseteq$ a pre-order over S and $[\![.]\!]$ an interpretation-function. The extension of $\mathcal{L}$ ('the static language') with modal operators will be designated as $\mathcal{L}^*$ ('the dynamic language'). Figure 1 visualizes this perspective. The dashed region may be thought of as (part of) some well-known logic.

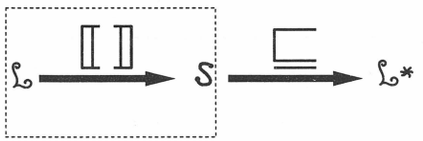


Figure 1: Dynamification

The non-dashed part allows us to *dynamify* the logic. For this purpose the semanticist has to implement his specific philosophy of the flow of information into the definition of $\sqsubseteq$.

1. CWI, Dept. of Interactive Systems, P.O. Box 94079, NL-1090 GB Amsterdam, The Netherlands, jaspars@cwi.nl, http://www.cwi.nl/~jaspars. Jan Jaspars was supported by CEC-project LRE-62-051 (FraCaS).
2. Institute for Perception Research (IPO), P.O. Box 513, 5600 MB Eindhoven, The Netherlands, krahmer@natlab.research.philips.com, http://tkiwww.kub.nl:2080/tki/Faces/Ek/ek.html.
3. Compare the distinction between Tarskian and Kripkean dynamics as made in [BC94].

To reason about it, modal operators, whose denotations are uniquely determined by this information order, are added to $\mathcal{L}$ resulting in the extended $\mathcal{L}^*$.
The modal operators can bring us from the static to the dynamic part. I.e., we can 'dynamify' a proposition ϕ by modalizing it as the argument of a dynamic modal operator. For instance, $[\phi]^+\psi$ expresses that each *expansion* of the current state with ϕ yields a ψ-state, and $\langle\phi\rangle^-\psi$ means that at least one *reduction* with ϕ brings us to a ψ-state. In this sense, it is fair to say that the static part 'generates' the dynamic part. Often the focus is on the latter part of DML. For example, Van Benthem and De Rijke discuss only the case in which the static part is classical propositional logic, i.e., the information states amount to standard truth-value assignments.
In this paper, we will show that many dynamic theories of change of information can formally be interpreted in a DML-style logic by specifying suitable static and dynamic languages, and giving suitable tunings of the three semantic parameters which together form the information models. In our setting a dynamic theory is characterized once the five parameters —$\mathcal{L}$, $\mathcal{L}^*$, S, $\sqsubseteq$ and $[\![.]\!]$— are determined. In general, the static parts of these theories are somewhat richer, while the dynamic parts are only poor fragments of the complete modal part of the original DML. This fragment is the restriction to primitive expansions and reductions and their minimal brothers and sisters, which we shall call *up-* and *downdates*.
In the end we hope to arrive at an abstract, modal unified view on dynamic semantics, which relates to cognitive science and linguistics in the same way as Pratt's original dynamic logic relates to computer science. The intention is that it will give us more than an entertaining stroll up and down the dynamic boulevard. The future aim of the enterprise started here is to come to a general and formal framework that captures the wide range of dynamic theories. Such a general minimal logic is certainly *not* meant to replace existing systems, but rather to comprehend them altogether by means of a simple language and underlying model-theory. Specific dynamic logics have been invented for treating specific dynamic phenomena of reasoning in a formal logical way. The specific formats of these logics have shown their use in practice (in the case of discourse logics the direct interpretation of natural language to logical form for example), and a meta-logic as we wish to present here is a priori not equipped for such specific utilities.
But what is our project good for? First of all, the framework is meant to give us a *single* microscope of which we hope to make observations on general logical and computational properties of different systems. Since for some of these systems, in particular the constructive ones (section 3), such meta-theoretical investigations have been going on for many decades, our hope and expectation is that this wisdom can be transferred to other parts of the spectrum in our framework. A second motivation is to connect to the current trend of congregating dynamic theories into more holistic systems covering as many dynamic phenomena in general reasoning as possible.[4] Finally we would like to stress that the present unified perspective not only provides us with a method to characterize *existing* dynamic systems in terms of possible world semantics. It also enables us to create *new* dynamic systems. For that we only need to extend an existing semantic theory by supplementing it with a 'dynamic component' (see Figure 1).[5]
The first step of our project of unification, as is worked out in this paper, consists of a survey of DML-specifications of several key dynamic systems. This means that our claim in this paper is more an empirical than a theoretical one. Of course, we will present our

4. A recent example of this tendency is the combination of dynamic predicate logic and update semantics in [GSV94]. Another combination which we find appealing is the one of first-order constructive and discourse logics. The former employs the dynamics of domains and interpretations, while the latter is concerned with the dynamics of variable assignments. Combining these two yields first-order dynamics for predicates, constants as well as for variables. This may bring us closer to the dynamics of natural language. For instance, see [BC94], [Bun90] and [MvBV95] for pleas for this kind of dynamics. In [JK96a] such dynamics is used for preferential reasoning over discourses.
5. Of course, the combination, the renovation and the innovation of dynamic logic in this modal setting is not a deterministic process. There is a lot of freedom left to the designer himself. By combination of systems we mean the mutual exchange of the individual five parameters of Figure 1.

translations in a formally precise manner, but there will be no central theorem about existing dynamic logics. By means of formal observations, we will illustrate how certain peculiarities of the translated systems, and differences between them, can be described using the dynamic modal framework.
The rest of this paper is organized as follows. In the first part (section 2) we will discuss the information models in more detail. In the second part we will show how various existing logics of change can be re-created in the resulting set-up. In section 3 we will re-survey a number of constructive logics in terms of information models. These provide the easiest examples of dynamification and allow us to clarify the definitions and terminology of section 2. In the section thereafter, 4, we will reformulate update logics in terms of information models and in section 5 will do the same for the discourse logics.

2 Basic definitions

The semantic setting of dynamic modal logic very simple. The basis of its models is given by Kripke's possible world models for intuitionistic logic, e.g., [Fit69].

Definition 2.1 *Let $\mathcal{L}$ be some language. An $\mathcal{L}$-information model M is a triple $\langle S, \sqsubseteq, [\![.]\!]\rangle$ with S a non-empty set of* information states, *$\sqsubseteq$ a pre-order over S, which is called the* information order *of M and $[\![.]\!] : \mathcal{L} \longrightarrow \wp S$. $[\![.]\!]$ is called an $\mathcal{L}$-*specification or -interpretation *over S. $\mathcal{M}_\mathcal{L}$ is the class of all $\mathcal{L}$-information models.*
If $M = \langle S, \sqsubseteq, [\![.]\!]\rangle \in \mathcal{M}_\mathcal{L}$ and $T \subseteq S$, then $\min_M T = \{t \in T \mid \forall s : s \sqsubseteq t \;\&\; s \in T \Rightarrow t \sqsubseteq s\}$ and $\max_M T = \{t \in T \mid \forall s : t \sqsubseteq s \;\&\; s \in T \Rightarrow s \sqsubseteq t\}$.[6]

The specification function interprets the language $\mathcal{L}$, which must be seen as the static language that we start with, over a given information model.

Definition 2.2 *Let $\phi \in \mathcal{L}$ and let $M = \langle S, \sqsubseteq, [\![.]\!]\rangle \in \mathcal{M}_\mathcal{L}$. The static meaning of ϕ is $[\![\phi]\!]$, and is also written as $[\![\phi]\!]_M$. The information order makes it possible to define the intended expansion ($+$) and reduction ($-$) relations. For a given proposition $\phi \in \mathcal{L}$ these relations are called the* dynamic *and the* negative dynamic meaning *of ϕ. These dynamic meanings refer to* expansion with ϕ *and* reduction with ϕ *and are defined as context sensitive interpretations with respect to a state s.*

$$\begin{aligned} [\![\phi]\!]^+_{M,s} &= \{t \in S \mid s \sqsubseteq t \;\&\; t \in [\![\phi]\!]_M\} \\ [\![\phi]\!]^-_{M,s} &= \{t \in S \mid t \sqsubseteq s \;\&\; t \notin [\![\phi]\!]_M\} \end{aligned} \tag{1}$$

The minimal expansions and reductions (the up- and downdates, *$+\mu$ and $-\mu$) are defined as follows:*

$$\begin{aligned} [\![\phi]\!]^{+\mu}_{M,s} &= \min_M [\![\phi]\!]^+_{M,s} \\ [\![\phi]\!]^{-\mu}_{M,s} &= \max_M [\![\phi]\!]^-_{M,s} \end{aligned} \tag{2}$$

Also relational variants of these interpretations are used, and often preferred in dynamic logics. They are defined as follows. Let act *be some dynamic action (e.g., $+$, $-$, $+\mu$ or $-\mu$), then:*

$$[\![\phi]\!]^{\mathsf{act}}_M = \{\langle s, t\rangle \in S \times S \mid t \in [\![\phi]\!]^{\mathsf{act}}_{M,s}\} \tag{3}$$

In the paper we will also use the abbreviation $[\![\phi_1, \ldots, \phi_n]\!]^{\mathsf{act}}_M$ for the relation composition $[\![\phi_1]\!]^{\mathsf{act}}_M \circ \ldots \circ [\![\phi_n]\!]^{\mathsf{act}}_M$. That is, $\{\langle s, t\rangle \mid \exists s_0, s_1, \ldots, s_{n-1}, s_n \in S : s_0 = s, s_n = t \;\&\; \langle s_{i-1}, s_i\rangle \in [\![\phi_i]\!]$ for all $i\}$.

6. Note that $\min_M T = \emptyset$ does not imply $T = \emptyset$. For example, the set of positive rational numbers (> 0) does not have a minimal element in models of the form $(\mathbb{Q}, \leq, [\![.]\!])$.

Having developed the dynamic relational interpretations of propositions, we can define modal operators to facilitate explicit reasoning over these relations. For example, the expression $[\phi]^+\psi$ refers to information states which necessarily turn into ψ states when they are extended with ϕ. $\langle\phi\rangle^+\psi$ means that there is at least one way to extend an information state with ϕ in such a way that also ψ holds. For negative dynamic meanings of ϕ we introduce the operators $[\phi]^-$ and $\langle\phi\rangle^-$. Minimal positive dynamic operators are $[\phi]^{+\mu}$ and $\langle\phi\rangle^{+\mu}$, and minimal negative dynamic operators are $[\phi]^{-\mu}$ and $\langle\phi\rangle^{-\mu}$. In other words, for every action act there are two corresponding modal operators, $[.]^{\mathsf{act}}$ and $\langle.\rangle^{\mathsf{act}}$ which are interpreted over the relation given by $[\![.]\!]^{\mathsf{act}}_M$. They are defined as follows:

Definition 2.3 *Let $M = \langle S, \sqsubseteq, [\![.]\!]\rangle \in \mathcal{M}_{\mathcal{L}}$. The dynamic modal operators obtain the following static interpretation once they have been supplemented with some propositional argument:*

$$\begin{aligned} [\![[\phi]^{\mathsf{act}}\psi]\!]_M &= \{s \in S \mid [\![\phi]\!]^{\mathsf{act}}_{M,s} \subseteq [\![\psi]\!]_M\} \\ [\![\langle\phi\rangle^{\mathsf{act}}\psi]\!]_M &= \{s \in S \mid [\![\phi]\!]^{\mathsf{act}}_{M,s} \cap [\![\psi]\!]_M \neq \emptyset\} \end{aligned} \tag{4}$$

Different dynamic systems employ different dynamic modalities. For example, in update semantics only update operators ($+\mu$) are used, while systems of belief revision also use downdates ($-\mu$, see section 4).
In general, we speak of $\mathcal{L}$ as the *static* or *local language* of dynamic systems. The dynamic modal extensions such as have been described here, are called *global* or *dynamic languages*, which we will also denote by $\mathcal{L}^*$.

The outline of our survey

In the following sections we will present logics which can be interpreted over a given class of information models. For each logic we give a translation function to a dynamic modal language such that the meanings are preserved, that is, the original interpretation of the propositions of a logic in terms of information models is maintained when we interpret the translations in the way we described above. For proofs we refer to the full text [JK96b].

Definition 2.4 *Let $\mathbf{L}$ be some logic with a language $\mathcal{L}_{\mathbf{L}}$ and an associated class of information models $\mathcal{M}_{\mathbf{L}}$. A* DML-specification *consists of:*

1. *A pair of languages $\langle\mathcal{L}, \mathcal{L}^*\rangle$ such that $\mathcal{L}^*$ is some* DML-*enrichment of $\mathcal{L}$ and $\mathcal{M}_{\mathbf{L}} \subseteq \mathcal{M}_{\mathcal{L}}$, and*
2. *A* DML-translation function: *$^\bullet : \mathcal{L}_{\mathbf{L}} \longrightarrow \mathcal{L}^*$ such that $^\bullet$ preserves meaning, that is to say, the original interpretation of an arbitrary $\phi \in \mathcal{L}_{\mathbf{L}}$ will be captured by one of the interpretations of $\phi^\bullet$ as given in Definition 2.2.*

Definition 2.4 elucidates our intentions in a formal way. We are aiming at an unambiguous universal dynamic modal language, – as far as the modal (relational) part is concerned – which can be used to compare dynamic logics. By comparison we mean that we can focus on the relational expressivity which is employed by different theories, as represented by the set of dynamic modal operators in the target global language, and specify individual constraints on information models by postulating corresponding axioms in the same target language.
Before we start our survey of DML-specifications of existing systems, we introduce the following conventional notations:

- $I\!P = \{p, p_1, \ldots, q, r\}$ is a set of propositional variables.
- We use two notations, $+$ and $*$, for different closure properties of languages. If C is a set of connectives and F a certain language, then

$$\begin{aligned} F + C &= \text{the smallest superset of } F \text{ closed under the connectives in } C, \\ F * C &= \text{the smallest superset of } F \text{ closed under the connectives in } C \\ &\quad \text{and the connectives which occur in } F. \end{aligned}$$

In this definition the dynamic modal operators are taken to be binary connectives. An instantiated modal operator is treated as a unary connective. For example $F+\{[\phi]^{+\mu}\}$ is the set of all formulae of the form $[\phi]^{+\mu}\ldots[\phi]^{+\mu}\psi$ with $\psi \in F$. A single formula is taken to be a 0-ary connective. This means that $F * \phi$ is the the smallest superset of $F \cup \{\phi\}$ closed under the all the connectives which appear in F.

- $\mathcal{L}_0$ is used for the language of propositional logic: $I\!P + \{\wedge, \vee, \rightarrow, \neg, \bot\}$.
- If disjunction $\vee$ and conjunction $\wedge$ are present in the target static language $\mathcal{L}$, then we consistently use the ordinary interpretations of union and intersection: $[\![\phi \vee \psi]\!] = [\![\phi]\!] \cup [\![\psi]\!]$ and $[\![\phi \wedge \psi]\!] = [\![\phi]\!] \cap [\![\psi]\!]$. Therefore, we will omit them in the specifications below.
- If the target dynamic modal language $\mathcal{L}^*$ of a specification is closed under a given n-ary connective c_n which occurs in the target local language $\mathcal{L}$ and whose interpretation is defined by an n-ary function $\overline{\mathsf{c}_n} : [\wp S]^n \longrightarrow \wp S$ over information states, i.e., $[\![\mathsf{c}_n(\phi_1, \ldots, \phi_n)]\!] = \overline{\mathsf{c}_n}([\![\phi_1]\!], \ldots, [\![\phi_n]\!])$ for each $\phi_i \in \mathcal{L}$, then in all cases we simply extend this interpretation for $\mathcal{L}^*$ ($\phi_i \in \mathcal{L}^*$).
- We will outline the specification by means of tables of the form

$\mathcal{L}_{\mathbf{L}}$	=	.. original language ..
$\mathcal{L}$	=	..target local language..
$\mathcal{L}^*$	=	..target global language..
$\mathcal{M}_{\mathbf{L}}$	=	..associated class of information models..
$\bullet$		..the DML-translation..

(5)

3 Constructive Logics

The best known logics which can be interpreted over information structures are so-called constructive logics, due to the contribution of the constructivists to the debate on the foundations of mathematics at the beginning of this century. The information state semantics for this kind of logics has been initiated by Kripke's model-theoretic treatment of *intuitionistic logic* (**IL**) [Kri65], which is one of the most influential logics among the constructive logics. In this section we will give a short illustration of the information state semantics of **IL**, and show how different constructive logics, like the *minimal logic* (**ML**) of [Joh37] and the *logic of constructible falsity* (**NL**) of [Nel49] arise from small variations of the 'local state' semantics ([.]). Furthermore, we will show more global variations of constructive logics by means of extending the dynamic part. All the constructive logics are persistent over the information order, i.e., for every ϕ if $s \in [\![\phi]\!]_M$ then also $t \in [\![\phi]\!]_M$ for all $t \sqsupseteq s$. Technically speaking, the persistence of local specifications and the restricted use of dynamic modal operators ($[\phi]^+$ only) establishes this overall persistence. In the information state analyses of programming languages, common sense reasoning and natural language one finds different non-persistent extensions of constructive logics. Non-persistence extensions of the local specifications have gained some interest in the field of logic programming. An example is adding the so-called weak negation to the underlying static language [PW90]. Non-persistence caused by extending the dynamic range most often comes down to allowing the 'possibility' variant of the expansion operator: $\langle\phi\rangle^+$. In the field of non-monotonic reasoning such extensions have been introduced by [Gab82] and [Tur84]. A linguistic theory which uses non-persistent modal operators is the so-called *data semantics* of [Vel85] and [Lan86].

3.1 Positive logics

Let us start with the simplest case: *positive implicative logic* (**PIL**). This logic coincides with the implicational fragment of intuitionistic logic. Its language only contains propositional variables $I\!P$ and an implication $\rightarrow$. Information models for this logic consists of the

monotonic subclass of $\mathcal{M}_{I\!P}$, i.e., $I\!P$-information models with a static interpretation function $[\![.]\!]$ such that:[7]

$$s \in [\![p]\!] \;\&\; s \sqsubseteq t \implies t \in [\![p]\!]. \tag{6}$$

The DML-specification does nothing more then mapping the implication $\phi \to \psi$ to $[\phi]^+\psi$. This settles our first simple translation table:

$\mathcal{L}_{\mathbf{PIL}}$	=	$I\!P + \{\to\}$
$\mathcal{L}$	=	$I\!P$
$\mathcal{L}^*$	=	$\mathcal{L} + \{[.]^+\}$
$\mathcal{M}_{\mathbf{PIL}}$	=	Monotonic subclass of $\mathcal{M}_{I\!P}$ (6)
•		$p^\bullet = p, (\phi \to \psi)^\bullet = [\phi]^+\psi$

(7)

Positive logic (**PL**) coincides with the positive fragment of intuitionistic logic. Let us be brief about this logic:

$\mathcal{L}_{\mathbf{PL}}$	=	$I\!P + \{\wedge, \vee, \to\}$
$\mathcal{L}$	=	$I\!P + \{\wedge, \vee\}$
$\mathcal{L}^*$	=	$\mathcal{L} * \{[.]^+\}$
$\mathcal{M}_{\mathbf{PL}}$	=	Monotonic subclass of $\mathcal{M}_{\mathcal{L}}$ (6)
•		$p^\bullet = p, (\phi \wedge \psi)^\bullet = \phi^\bullet \wedge \psi^\bullet,$ $(\phi \vee \psi)^\bullet = \phi^\bullet \vee \psi^\bullet, (\phi \to \psi)^\bullet = [\phi^\bullet]^+\psi^\bullet$

(8)

3.2 Constructive logics with negation(s)

The clear syntactic difference of positive logics with classical propositional logic is the lack of a negation. Negative reasoning has always been an issue of debate among the constructivists and an important point of criticism against constructivism (see e.g. [Wan93] for a survey on different constructivistic treatments of negation). The problem is that constructivists relate notions of truth and validity to the *presence* of mathematical constructions, while the truth of a negated sentence seems to refer to the absence of information, as does negation in classical logic. The solution of the intuitionists is to use an absurd proposition $\bot$ which they take to be an 'unprovable' proposition. Formally, we arrive at **IL** via extending the static input language $\mathcal{L}$ of positive logic with $\bot$ and taking $[\![\bot]\!] = \emptyset$. The negation of a proposition ϕ is then defined by means of the dynamics of constructive logic; the current situation can not be extended with a proof of ϕ, that is, $[\phi]^+\bot$ in the DML-specification. In terms of the intuitionistic (Brouwer-Heyting-Kolmogorov, BHK) interpretation (see e.g. [TvD88]), this definition means that the negation of ϕ is proved whenever a method has been found to translate every hypothetical proof of ϕ into a (the) contradiction.

$\mathcal{L}_{\mathbf{IL}}$	=	$\mathcal{L}_0$
$\mathcal{L}$	=	$I\!P + \{\wedge, \vee, \bot\}$
$\mathcal{L}^*$	=	$\mathcal{L} * \{[.]^+\}$
$\mathcal{M}_{\mathbf{IL}}$	=	as in (8) with $[\![\bot]\!] = \emptyset$
•		as in (8) with $\bot^\bullet = \bot, (\neg\phi)^\bullet = [\phi^\bullet]^+\bot$

(9)

Of course, the philosophical problems of constructive negative information shift to the presupposed existence of an unprovable proposition. [Joh37] introduced a weaker version of intuitionistic logic where the absurd proposition is taken to be possibly provable. This

7. In this paper, monotonicity should be distinguished from persistence. The former notion applies to the interpretation of atomic formulae, while the latter expresses preservation of information over the information order in general.

relatively small variant comes down to allowing non-empty specifications of $\bot$. This weakening of intuitionistic logic is known as minimal logic (**ML**). In short, $\mathcal{L}_{\mathbf{ML}} = \mathcal{L}_0$, $\mathcal{M}_{\mathbf{ML}}$ is the same as $\mathcal{M}_{\mathbf{IL}}$ minus the $\bot$-constraint, $\mathcal{L} = I\!P \cup \{p_\bot\} + \{\wedge, \vee\}$, $\mathcal{L}^* = \mathcal{L} * \{[.]^+\}$ and the translation is the same as in (8) plus $\bot^\bullet = p_\bot$ and $(\neg\phi)^\bullet = [\phi^\bullet]^+p_\bot$. In fact, this DML-translation tells us that **ML** is equivalent to **PL**, and therefore cannot be considered as a constructive contribution to the discussion on negation in constructive logic.

A more deviant position in constructive logic with respect to negation has been taken by [Nel49]. Nelson introduced an independent negative construction to the BHK interpretation. According to this view, proofs are needed to determine mathematical truth, just as in the BHK philosophy, but also, *refutations* are needed to determine mathematical *falsity*. Falsity should have a separate truth-functional interpretation, and not be defined on the basis of the absence of truth as in classical and intuitionistic logic. To implement the refutation-as-falsity in information models, we have to distinguish between the interpretation of negated an non-negated formulae. The latter formulae are then interpreted in the same way as in the case of positive logic (8), while the former have to be assigned new interpretations. In fact, Nelson opts for the falsity conditions of the well-known strong-Kleene interpretation from partial logic (e.g., see [Bla86]) where the notions of truth and falsity are detached in the same manner. Two additional structural constraints are needed to get a satisfactory class of information models: monotonicity of falsity (refutations persist!) and the mutual exclusion of proofs and refutations, that is, we do not accept a proposition to have a proof (to be true) and a refutation (false) at the same time.

$\mathcal{L}_{\mathbf{NL}}$	=	$\mathcal{L}_0$
$\mathcal{L}$	=	$I\!P + \{\wedge, \vee, \neg\}$
$\mathcal{L}^*$	=	$\mathcal{L} * \{[.]^+\}$
$\mathcal{M}_{\mathbf{NL}}$	=	as in(8) extended for formulae of the form $\neg\phi$ with $[\![\neg p]\!] \cap [\![p]\!] = \emptyset$ $[\![\neg p]\!]$ monotonic for all $p \in I\!P$ $[\![\neg\neg\phi]\!] = [\![\phi]\!]$ $[\![\neg(\phi \wedge \psi)]\!] = [\![\neg\phi]\!] \cup [\![\neg\psi]\!]$ $[\![\neg(\phi \vee \psi)]\!] = [\![\neg\phi]\!] \cap [\![\neg\psi]\!]$
•		as in (8) for non-negated formulae, and for $\neg\phi$-formulae: $(\neg p)^\bullet = \neg p, (\neg(\phi \wedge \psi))^\bullet = (\neg\phi)^\bullet \vee (\neg\psi)^\bullet,$ $(\neg\neg\phi)^\bullet = \phi^\bullet, (\neg(\phi \vee \psi))^\bullet = (\neg\phi)^\bullet \wedge (\neg\psi)^\bullet,$ $(\neg(\phi \to \psi))^\bullet = \phi^\bullet \wedge (\neg\psi)^\bullet$ (!!).

(10)

3.3 Non-persistent variants of constructive logic

Standard constructive logics are too rigid to model the 'every day' dynamics of general reasoning. In the field of applied logics one finds different non-persistent extensions of the constructive logics of the previous two sections.

3.3.1 Non-persistent statics

The most simple non-persistent variants are based on permitting non-monotonic specification functions. Somewhat more structural non-persistent static extensions of constructive logic are realized by extending their languages with an associated non-persistent connective. The most obvious example in this respect is introducing a weak negation $\sim$ in systems like **IL** and **NL**. This weak negation has the same denotation as the classical negation: $[\![\sim\phi]\!] = S \setminus [\![\phi]\!]$. In an extended BHK interpretation $\sim\phi$ would refer to a situation where no proof is present yet. It is immediately clear why such a connective should behave non-persistent.

In the field of logic programming a weak negation extension of **NL** has been proposed as a meta-logic for deductive reasoning with a negation-as-failure and explicit negation [PW90]. The weak negation is then seen as the implicit negation-as-failure,[8] while Nelson's strong negation is used for proper understanding of explicit negation as a true determination of falsity on the facts and rules in a given program.

3.3.2 Non-persistent dynamics

Illustrative examples of extensions of constructive logic by means of non-persistent dynamic supplies can be found in the fields of non-monotonic logic and conditional logic. Within the former area [Gab82] gives a straightforward non-persistent variant of **IL**, called **GL** here, by adding existential 'upward' expressivity ($\langle . \rangle^+$). The motivation for this addition is to capture the *consistency*-operator M of the original default logic of [MD80] in an explicit fashion. The statement $\mathsf{M}\phi$ means that the current state can be extended with the information ϕ. It can be defined in the dynamic modal setting by $\langle \phi \rangle^+ \phi$. In terms of the BHK-interpretation, such a sentence means that at the current state it is not yet proved that ϕ is unprovable. [9]

$\mathcal{L}_{\mathbf{GL}}$	=	$\mathcal{L}_0 * \mathsf{M}$
$\mathcal{L}$	=	The same as for **IL**
$\mathcal{L}^*$	=	$\mathcal{L} * \{[.]^+, \langle . \rangle^+\}$
$\mathcal{M}_{\mathbf{GL}}$	=	$\mathcal{M}_{\mathbf{IL}}$ (9)
$\bullet$		The same as for **IL** with $(\mathsf{M}\phi)^\bullet = \langle \phi^\bullet \rangle^+ \phi^\bullet$

(11)

In [Tur84] a similar modal approach to nonmonotonic reasoning on the basis of strong-Kleene partial logic has been described. In strong-Kleene logic the implication $\phi \rightarrow \psi$ is defined as $\neg\phi \vee \psi$. In terms of the DML-specification, it disappears into the static local language. The language of Turner's logic is the same as $\mathcal{L}_{\mathbf{GL}}$. The information models are the same as $\mathcal{M}_{\mathbf{NL}}$, the target local language $\mathcal{L}$ is also the same as (10), while the target global language is $\mathcal{L} * \{[\top]^+, \langle \top \rangle^+\}$ with $\top = [p]^+ p$, and the translation is then the same as for Nelson's logic without implication, but with the following additional clauses for M:

$$(\mathsf{M}\phi)^\bullet = \langle \top \rangle^+ \phi^\bullet \text{ and } (\neg\mathsf{M}\phi)^\bullet = [\top]^+ (\neg\phi)^\bullet. \tag{12}$$

In [Wan95] a similar logic over $\mathcal{M}_{\mathbf{NL}}$ is defined. The difference with Turner's setting is that Wansing follows the Nelson reading of the truth and falsity of implications. Hence the DML-specification for Wansing's logic boils down to the extension of $\bullet$ in (10) with the translation of M and $\neg$M sentences as given in (12).

A conditional logic which can be seen as a non-persistent extension of constructive logic is the so-called data-logic (**DaL**) of [Vel85]. The model-theoretic setting of this logic, also called *data-semantics*, is the same as for **NL** with an additional *refinability* constraint for the binary specification:

$$\forall s \in S \; \exists t \in S : s \sqsubseteq t \; \& \forall p \in I\!P : t \in [\![p]\!] \cup [\![\neg p]\!]. \tag{13}$$

[10]

In terms of Nelson's proof-theoretic terminology, this constraint means that there are no propositions which are neither provable nor refutable, a situation which Nelson does not exclude. Of course, this constraint does not affect persistence in any way. Non-persistence evolves from the falsification of implication in data-semantics. In this setting $\phi \rightarrow \psi$ is false whenever the current information state can be extended with $\phi \wedge \neg\psi$. In the dynamic modal setting, we can define this as $\langle \phi \rangle^+ \neg\psi$. The truth-conditional decomposition of all other

8. Of course, negation-as-failure is not the same as non-**NL**-derivability here. **NL** with weak negation is used by Pearce and Wagner as a meta-logic for reasoning about logic programs.
9. One can define $\mathsf{M}\phi$ when the above-mentioned weak negation is available: $\langle \phi \rangle^+ \phi$ means the same as $\sim[\phi]^+ \bot$. Intuitionistic logic with weak negation has higher expressive capacity than $\mathcal{L}_{\mathbf{GL}}$.
10. It can easily be proven that this requirement induces refinability for the full language, i.e., replace p by arbitrary formulae from $\mathcal{L}^*$.

connectives coincide with their treatment in **NL**. The definition of truth of implication is also identical to Nelson's.

$\mathcal{L}_{\mathbf{DaL}}$	=	$\mathcal{L}_0$
$\mathcal{L}$	=	As in (10)
$\mathcal{L}^*$	=	$\mathcal{L} * \{[.]^+, \langle . \rangle^+\}$
$\mathcal{M}_{\mathbf{DaL}}$	=	The refinable (13) subclass of $\mathcal{M}_{\mathbf{NL}}$ (10)
$\bullet$		As in (10) with the exception of $\neg(\phi \rightarrow \psi)$: $(\neg(\phi \rightarrow \psi))^\bullet = \langle \phi^\bullet \rangle^+ (\neg\psi)^\bullet$

(14)

Observation 3.1 *The soundness proofs of the translations in the tables (7) — (14) are straightforward. The standard interpretations can be found in the relevant references.*[11]

Observation 3.2 *The advantage of such* DML*-specifications can be illustrated when we want to compare* **IL**, **NL** *and* **DaL**. *Three systems with three different negations; the differences become clearly visible in the* DML*-specifications. From these specifications we can see that* **NL** *is in fact only an expressive enrichment of* **IL**, *while* **DaL** *is not only an expressive extension of* **NL**, *it is also logically stronger than* **NL**, *i.e., all theorems of* **NL** *are theorems of* **DaL** *when we reformulate them in the* DML*-language. An additional axiom of* **DaL** *is* $[[\phi \vee \neg\phi]^+ (\psi \wedge \neg\psi)]^+ \chi$. *This axiom corresponds directly to the refinability constraint, and is* not *valid in* **NL**.

Observation 3.3 *For all the logics discussed in this section there are also translations definable from the dynamic modal target language to the original languages. Unfortunately, this is not always the case in the following sections.*

4 Up- and Downdate logics

In this section we will focus on logical specifications of reasoning with *minimal* informational changes. Two influential representatives of this cornerstone of dynamic logic are the logic of theory change, also called the study of belief revision, as defined by [AGM85] and [Gär88] (AGM), and the update semantics (US) of [Vel91]. We will describe these two formalisms in terms of the dynamic modal setting in the following two subsections.

Besides the minimal dynamics that AGM-style belief revision logic and update semantics employ, they both use two other very important structural optimizations. Firstly, informational changes are taken to be *functional*. This means that these informational actions always succeed and uniquely determine an output state for every input. Another remarkable property of both theories is that they take only a *unique* information model into account which is *maximal* with respect to the underlying static logic. This means that $[\![\phi]\!] = S$ if and only if ϕ is a theorem of the underlying static logic. In this way we may say that the information states represent a full class of models for the underlying static logic.

Yet another striking deviant property with respect to the constructive logics of the previous section is that states are uniquely described by their static information. In our terminology:

$$[\forall \phi \in \mathcal{L} : s \in [\![\phi]\!] \Leftrightarrow t \in [\![\phi]\!]] \Longleftrightarrow s = t. \tag{15}$$

The simple reason for this equivalence is that the information orders which are employed by the logics of this section are defined on the basis of structure or content of the states. For constructive logics only a monotonicity constraint (6) is used to direct the information flow in order to guarantee that 'hard' information increases when states are enriched. Of course, this is a minimal liberal choice. It is up to the designer of a dynamic formalism to postulate additional constraints to enforce more interrelational structure between static and

11. For example, for the persistent logics, see [Wan93].

dynamic information. An example of such an additional constraint in the previous section is the refinability constraint (13) for $\mathcal{M}_{\mathbf{DaL}}$. In the initial settings of the logical formalisms of this section the complete fixation of the statics/dynamics interplay is enforced by the additional constraint which one obtains by coplementation of the monotonicity constraint (6) for atomic information:

$$\forall p : [s \in [\![p]\!] \Rightarrow t \in [\![p]\!]] \Longleftrightarrow s \sqsubseteq t \tag{16}$$

Both in AGM and US information states are meant as models or representations of the beliefs of some agent. The price of the dynamic rigidity of the monotonicity equivalence (16) is that there is no distinction made between the possible changes of this agent's beliefs and the changes which the agent himself believes to be possible. In both theories enrichments of the initial formalisms have been put forward to overcome this rigidity. In AGM the so-called *partial meet contraction* has been defined as a more cautious form of its *full meet* counterpart. In this setting the agent is allowed to select certain states when he has to give up certain information. Using this definition, as will be clear from our DML-specification of the partial meet system in AGM (**AGM-p**) below, we end up with multiple information models. In richer systems of US than the one which we will discuss here, Veltman proposes to dress up the beliefs of an agent with private *expectations* about the possible enrichments of his current beliefs. In this way default implications are defined in [Vel91].

4.1 AGM belief revision

In AGM's logical setting of theory change the information states, the set S, are sets of $\mathcal{L}_0$–formulae which represent the beliefs of an agent at a certain point in time. These *belief sets* coincide with the *theories* of classical propositional logic, i.e., sets of propositional formulae which are closed under classical deduction. The *single* information model is constructed by taking the set inclusion to be *the* information order, $\sqsubseteq = \subseteq$, and static interpretation is determined by membership.
In the original AGM belief revision theory there is no explicit definition of a logical object language. The main purpose of the AGM setting is to postulate constraints for the output belief sets after a given input state has been modified by a belief action: *expansion* $(+)$, *contraction* $(-)$ or *revision* $(*)$. The kind of expressions which are used to describe these constraints, known as AGM-postulates, are of the form $\psi \in s^{\mathsf{act}}\phi$, which means that ψ is a member of the set (state) s if it has been modified according to the action act (expansion, contraction or revision) with ϕ. We will use expressions of the form $\theta\ [\mathsf{act}\phi]\ \psi$ as to represent propositions about belief changes. The meaning of such an expression is that ψ will be believed after a state containing θ has been modified by act with ϕ. The following simple observation justifies this implementation of logical form.[12]

Observation 4.1 *Note that* $\theta\ [\mathsf{act}\phi]\ \psi$ *is the same as* $\psi \in \mathsf{Cn}(\theta)^{\mathsf{act}}\phi$ *(with* $\mathsf{Cn}(\theta)$ *denoting the classical closure of the formula* θ*) as long as the following constraint holds:*

$$\forall s, t \in S : s \subseteq t \Rightarrow s^{\mathsf{act}}\phi \subseteq t^{\mathsf{act}}\phi \text{ for all act and } \phi. \tag{17}$$

This property holds indeed for all actions in the definitions as given by [Gär88]. Furthermore, AGM-theory presupposes that every belief set s *can uniquely be determined by a single formula* θ_s *in the sense that* $s = \mathsf{Cn}(\theta_s)$.[13]

In order to obtain an appropriate DML-specification of the AGM-setting we need the minimal versions of the expansion and reduction operators: the up- and downdate operators. The

12. Compare [Rij94] and [Seg95] for such logical forms for AGM. These languages are in fact closer to the target dynamic modal language than the one which we use in (18).
13. [Gär88] postulates a compactness criterion: i.e., every belief set can be uniquely determined by a finite set of formulae. By choosing classical propositional logic to be the logic under which belief sets are closed — this assumption is not always made explicitly in AGM but frequently used in proofs about AGM-formalisms (see e.g. the appendix of [Gär88]) — such a finite set can then be replaced by the conjunction over this set.

$\mathcal{L}_{\mathbf{AGM\text{-}f}}$	=	$\mathcal{L}_{\mathbf{AGM}} = \{\theta[\mathsf{act}\,\phi]\psi \mid \theta, \phi, \psi \in \mathcal{L}_0, \mathsf{act} \in \{+, -, *\}\}$
$\mathcal{L}$	=	$\mathsf{B} + \{\wedge, \vee, \neg, \rightarrow, \bot\}$
$\mathcal{L}^*$	=	$\mathcal{L} * \{[.]^{+\mu}, [.]^{-\mu}\}$
$\mathcal{M}_{\mathbf{AGM\text{-}f}}$	=	$\{\langle S, \sqsubseteq, [\![.]\!]\rangle\}$
		S = $\{s \subseteq \mathcal{L}_0 \mid \mathsf{Cn}(s) = s\}$
		$\sqsubseteq$ = $\subseteq$
		$[\![B\phi]\!]$ = $\{s \in S \mid \phi \in s\}$
		(The classical interpretations for the other connectives.)
$(\theta[+\phi]\psi)^\bullet$	=	$B\theta \rightarrow [B\phi]^{+\mu}B\psi$;
$(\theta[-\phi]\psi)^\bullet$	=	$B\theta \rightarrow ((\langle B\phi\rangle^{-\mu}B\top \wedge [B\phi]^{-\mu}B\psi)$ $\vee([B\phi]^{-\mu}\bot \wedge B\psi))$
$(\theta[*\phi]\psi)^\bullet$	=	$B\theta \rightarrow ((\langle B\neg\phi\rangle^{-\mu}B\top \wedge [B\neg\phi]^{-\mu}[B\phi]^{+\mu}B\psi)\vee$ $([B\neg\phi]^{-\mu}\bot \wedge [B\phi]^{+\mu}B\psi))$

(18)

target local language requires some enrichment as well. We will use the ordinary set of Boolean connectives over *belief atoms*: $\mathsf{B} = \{B\phi \mid \phi \in \mathcal{L}_0\}$, where $B\phi$ has the intended meaning that ϕ is believed. The reason for this enrichment is that during the translation we have to take care that the Booleans over belief sentences do not mess up with the beliefs over Boolean expressions. For example, a clear distinction should be made between $B\phi \vee B\psi$ and $B(\phi \vee \psi)$. The resulting target languages are adequate for specifying the AGM definition of so-called *full meet belief revision* (**AGM-f**). For the other two forms of belief revision as defined by Gärdenfors, partial meet revision (**AGM-p**) and maxi-choice revision (**AGM-m**), the target local language of the DML-specification can be dressed up with an additional epistemic operator C. Such an operator is needed to imitate the so-called selection function as defined for both partial meet revision and maxichoice revision. In the former setting a selection function selects for every formula ϕ and state s a subset of the maximal subtheories of s which do not contain ϕ (if this collection is non-empty, i.e., ϕ is not a tautology). For reasoning about such states we need some additional expressive means. In our case we have chosen for this second epistemic operator. $C\phi$ expresses a minimal positive attitude towards the proposition ϕ: the believing agents takes his beliefs to be 'close to' ϕ.

4.1.1 Full meet revision

Table (18) settles the specification for AGM's full meet definitions.

Observation 4.2 *Let* s *be a belief set, and* θ_s *a formula such that* $s = \mathsf{Cn}(\theta_s)$*, and let* $s^{\mathsf{act}}\psi$ *be defined as in the full meet definitions of [AGM85] and [Gär88], then*

$$\phi \in s^{\mathsf{act}}\psi \Leftrightarrow s \in [\![(\theta_s[\mathsf{act}\,\psi]\phi)^\bullet]\!]_M. \tag{19}$$

Here, M *is the only* **AGM-f***-model as defined in (18).*

For the proof we refer to [JK96b]. Since revision is defined by the so-called *Levi-identity*: $s^*\phi = (s^-\neg\phi)^+\phi$, it can be easily seen that the DML-specification of revision follows from the specification of contraction.

4.1.2 Partial meet revision

As said earlier, for DML-specification of partial meet revision (**AGM-p**), we need an enrichment of the local DML-language with an additional operator C. The interpretation of expressions of the form $C\phi$ (C) has to meet certain constraints to imitate the behavior of selection functions in the original setting of **AGM-p**.

$\mathcal{L}_{\textbf{AGM-p}}$	=	$\mathcal{L}_{\textbf{AGM}}$
$\mathcal{L}$	=	$\mathsf{B} \cup \mathsf{C} + \{\wedge, \vee, \neg, \rightarrow, \bot\}$
$\mathcal{L}^*$	=	$\mathcal{L} * \{[.]^{+\mu}, [.]^{-\mu}\}$
$\mathcal{M}_{\textbf{AGM-p}}$	=	All models $\langle S, \sqsubseteq, [\![.]\!]\rangle$ with $[\![.]\!]$ the same as in (18) for B and $[\![C\phi]\!] \subseteq [\![B\phi]\!]$; $[\![B\phi]\!] \subseteq [\![B\psi]\!] \Rightarrow [\![C\phi]\!] \subseteq [\![C\psi]\!]$; $[\![B\phi]\!]^{\mu}_{M,s} \cap [\![C\phi]\!] = \emptyset \Leftrightarrow [\![B\phi]\!]^{-\mu}_{M,s} = \emptyset$.
$(\theta[+\,\phi]\psi)^\bullet$	=	$B\theta \rightarrow [B\phi]^{+\mu} B\psi$;
$(\theta[-\,\phi]\psi)^\bullet$	=	$B\theta \rightarrow ((\langle B\phi\rangle^{-\mu} B\top \wedge [B\phi]^{-\mu}(C\phi \rightarrow B\psi) \vee$ $([B\phi]^{-\mu}\bot \wedge B\psi))$
$(\theta[*\,\phi]\psi)^\bullet$	=	$B\theta \rightarrow ((\langle B\neg\phi\rangle^{-\mu} B\top \wedge [B\neg\phi]^{-\mu}(C\neg\phi \rightarrow [B\phi]^{+\mu} B\psi) \vee$ $([B\neg\phi]^{-\mu}\bot \wedge [B\phi]^{+\mu} B\psi))$

(20)

As we can see from the translation of contraction expressions, the downdate operators only range over the sets which still verify $C\phi$. The Levi-identity for the definition of revision yields a similar restriction to $C\neg\phi$-states.

Observation 4.3 *Let s be a belief set, and θ_s a formula such that $s = \mathsf{Cn}(\theta_s)$ and let $s^{\mathsf{act}}\psi$ be defined as in the partial meet definitions of [AGM85] and [Gär88], then*

$$\phi \in s^{\mathsf{act}}\psi \Longleftrightarrow s \in [\![(\theta_s[\mathsf{act}\,\psi]\phi)^\bullet]\!]_M \text{ for all } M \in \mathcal{M}_{\textbf{AGM-p}}. \tag{21}$$

Maxichoice revision (**AGM-m**) can be specified by the same translations and adding the condition that $[\![B\phi]\!]^{\mu}_{M,s} \cap [\![C\phi]\!]$ is a singleton iff $[\![B\phi]\!]^{-\mu}_{M,s} \neq \emptyset$. This corresponds to the maxichoice selection functions of AGM which pick a single element from $s \perp \phi$ whenever $s \perp \phi \neq \emptyset$.

4.2 Update semantics

In the most plain system of update semantics of Veltman, the so-called *might*-system (**US-m**) information states are sets of worlds, i.e., propositional truth-value assignments. Just like in modal epistemic logics this set of worlds represents the set of *uncertainties* of a chosen agent. As we said in the introduction of this section, **US-m** makes a maximal choice in this respect: the complete collection of sets of possible worlds constitute the information space: S is the powerset of $\{0,1\}^{I\!P}$. Atomic information over such a state $s \in S$ is true if it holds (mapped to 1) with respect to (by) all the worlds in this state s, and therefore, the strong monotonicity constraint (16) yields $s \sqsubseteq t$ iff $t \subseteq s$.[14]
US-m is an extension of propositional logic with a supplementary *might*-operator. The $\mathcal{L}_0$-formula can be interpreted in the same way as atomic information. A set of worlds supports such a formula $\phi \in \mathcal{L}_0$ iff ϕ is supported by all worlds in this set (in the classical sense). A proposition of the form $might\,\phi$ is supported by a set of worlds if either one of the elements supports ϕ or this set is empty. In the latter case the agent has reached the *absurd* state, and all information is defined to be true. Below the complete DML-setting for this logic is depicted.

14. The original definition in US boils down to this definition for atomic information [Vel91].

$\mathcal{L}_{\textbf{US-m}}$	=	$\mathcal{L}_0 + \{might\}$
$\mathcal{L}$	=	$\mathcal{L}_{\textbf{US-m}}$
$\mathcal{L}^*$	=	$\mathcal{L} + \{[.]^{+\mu}\}$
$\mathcal{M}_{\textbf{US-m}}$	=	The single information model $M = \langle S, \sqsubseteq, [\![.]\!]\rangle$ with: $S = \wp(\{0,1\}^{I\!P})$ $\sqsubseteq = \supseteq$ $[\![\phi]\!] = \{s \in S \mid V \models \phi^{15} \text{ for all } V \in s\}$ and $[\![might\,\phi]\!] = \{s \in S \mid [\![\phi]\!] \cap s \neq \emptyset\} \cup \{\emptyset\}$ for all $\phi \in \mathcal{L}_0$
$\bullet$		$\phi^\bullet = \phi$ for all $\phi \in \mathcal{L}_{\textbf{US-m}}$ (identity).

(22)

The definitions of updates as given in [Vel91] are identical with the minimal dynamic meanings of the propositions of the input language $\mathcal{L}$ (Observation 4.4). In the original setting of Veltman, every proposition ϕ corresponds to an *update function* $\mathsf{up}_\phi : S \longrightarrow S$ which are defined in the following compositional manner:

$$\begin{array}{rclcrcl} \mathsf{up}_p(s) & = & \{V \in s \mid V(p) = 1\} & & \mathsf{up}_{\neg\phi}(s) & = & s \setminus \mathsf{up}_\phi(s) \\ \mathsf{up}_{might\,\phi}(s) & = & \begin{cases} s & \text{if } \mathsf{up}_\phi(s) \neq \emptyset \\ \emptyset & \text{otherwise} \end{cases} & & \mathsf{up}_{\phi\wedge\psi}(s) & = & \mathsf{up}_\phi(s) \cap \mathsf{up}_\psi(s) \end{array} \tag{23}$$

Observation 4.4 *For all $\phi \in \mathcal{L}_{\textbf{US-m}}$ and all $s \in S$: $[\![\phi]\!]^{+\mu}_{M,s} = \{\mathsf{up}_s(\phi)\}$, or equivalently, $\mathsf{up}_\phi = [\![\phi]\!]^{+\mu}_M$ for all $\phi \in \mathcal{L}_{\textbf{US-m}}$.*

4.2.1 Divergence

Differences with Veltman's functional decomposition would appear when we extend the language with Boolean combinations of *might*-sentences.[16] Extending the deterministic decomposition of (23) over the 'problematic proposition' $p \wedge might\,\neg p$ yields violation of idempotence for up: $\mathsf{up}_{p\wedge might\,\neg p} \circ \mathsf{up}_{p\wedge might\,\neg p} \neq \mathsf{up}_{p\wedge might\,\neg p}$. Suppose that $s = \{U, V\} \in S$ with $U(p) = 1$ and $V(p) = 0$, then

$$\mathsf{up}_{p\wedge might\,\neg p}(s) = \{U\} \quad \text{and } \mathsf{up}_{p\wedge might\,\neg p}(\{U\}) = \emptyset \tag{24}$$

If the static decomposition, as given in (22), is extended over such Boolean combinations, this problematic case only holds in the empty set and subsequently we obtain: $[\![p \wedge might\,\neg p]\!]^{+\mu}_{M,s} = \{\emptyset\}$ for all $s \in S$. Idempotence is restored, but we lose functionality; for example, $[\![\neg(might\,p \wedge might\,\neg p)]\!]^{+\mu}_{M,s} = \{\{V\}, \{U\}\}$ if s is the same state as in example (24). By the functional decomposition of (23) we obtain $\mathsf{up}_\phi(s) = \{\emptyset\}$ for $\phi = \neg(might\,p \wedge might\,\neg p)$.

4.2.2 Texts and entailment

Veltman's notion of truth of a proposition with respect to an information state, $s \models \phi$, is defined as the non-informativeness of ϕ with respect to s. This means that ϕ has *no effect* on s as an update: $\mathsf{up}_\phi(s) = s$. According to observation 4.4 this is the same as $\{s\} = [\![\phi]\!]^{+\mu}_{M,s}$. In the DML-specification of **US-m** no dynamic modalities are used. We only need the $+\mu$-meaning to get a corresponding soundness for the functional definitions as given in US. So, why do we need modal operators of the $[.]^{+\mu}$ in the target global language?
Besides sentences, represented by the language $\mathcal{L}_{\textbf{US-m}}$, Veltman also defines *texts* and two different notions of *entailment* over texts.[17] A text is of the form $\phi_1; \ldots; \phi_n$ where

15. $V \models \phi$ means classical verification of ϕ by the valuation V.
16. See also the 'loose ends' discussion in [Vel91] at the end of section 2.
17. Veltman's third entailment relation is defined is an entailment over sentences. It coincides with the classical notion of entailment over update definitions.

all ϕ_i are sentences from $\mathcal{L}_{\textbf{US-m}}$, and the update definition of such a text is the same as the consecutive execution of the updates corresponding to ϕ_1 through ϕ_n: $\mathsf{up}_{\phi_1;\ldots;\phi_n}(s) = \mathsf{up}_{\phi_n}(\ldots(\mathsf{up}_{\phi_1}(s))\ldots)$. This definition coincides with our definition $[\![\phi_1,\ldots,\phi_n]\!]^{+\mu}_{M,s}$ (Definition 2.2). This means we need $+\mu$-modal operators to reason about texts.

Veltman defines two types of entailment notions over text. His second definition, which is abbreviated by $\models_2$, has a nice DML counterpart:

$$[\![[\phi_1]^{+\mu}\ldots[\phi_n]^{+\mu}\psi]\!]_M = S \Leftrightarrow \phi_1,\ldots,\phi_n \models_2 \psi. \tag{25}$$

In other words, each state becomes a ψ state after it has been updated consecutively by ϕ_1 through ϕ_n: 'extending' the DML-specification to include this entailment we would get $(\phi_1,\ldots,\phi_n \models_2 \psi)^\bullet = [\phi_1]^{+\mu}\ldots[\phi_n]^{+\mu}\psi$. Thus, we can express this entailment relation by a single $\mathcal{L}^*$-formula. This is different for the first entailment relation of Veltman. The relation $\models_1$ says that $\psi \in \mathcal{L}$ follows from a text $\phi_1;\ldots;\phi_n$ whenever the *minimal* information state, i.e. the smallest information state, which coincides with the information state of all possible worlds $\{0,1\}^{I\!P}$, transforms into a ψ-state whenever it is updated consecutively with ϕ_1 up to ϕ_n. Formally,

$$\{0,1\}^{I\!P} \in [\![[\phi_1]^{+\mu},\ldots,[\phi_n]^{+\mu}\psi]\!]_M. \tag{26}$$

Unfortunately, this situation cannot be expressed by a single $\mathcal{L}^*$ expression. The underlying reason is not the shortage of dynamic modal expressivity, but merely caused by the lack of static expressivity: there is not an expression which uniquely describes the minimal state.[18] If such a state were added to $\mathcal{L}$ – let us call it $\mathbf{0}$ – then this entailment relation could be defined by putting $[\mathbf{0}]^{+\mu}$ in front of the modal sequence on the left-hand side of the equation in (25).[19]

4.2.3 AGM and update semantics

The information model of **AGM-f** is very closely related to the one for **US-m** as given in (22). This connection between worlds and theories has been made by [Gro88] where Grove relates belief revision to Lewis' possible world analysis of counterfactual sentences [Lew73]. The simple correspondence is made by two mutually invertible 1–1 mappings. Each theory s is related to the collection of its maximally consistent extension S_s. The corresponding set of worlds is $\{V_t \in \{0,1\}^{I\!P} \mid V \models \phi \text{ for all } \phi \in t, t \in S_s\}$. The inverse of this correspondence is simply defined by the propositions which are verified by all its members $s \mapsto \{\phi \in \mathcal{L}_0 \mid \forall V \in s : V \models \phi\}$.

Observation 4.5 *Let us write this correspondence between theories and worlds as $s_1 \sim s_2$, where s_1 is a theory and s_2 a set of worlds. By the* DML*-specifications we can easily relate the static part of the logics* **AGM-f** *and* **US-m**. *In fact, we used a wider language in the case of* **AGM-f**, *but the following mapping from the* **US-m** *local target language and the* **AGM-f** *static language is surely 1–1:*

$$\phi \mapsto B\phi \;\&\; \mathit{might}\,\phi \mapsto \neg B\neg\phi \vee B\bot \text{ for all } \phi \in \mathcal{L}_0. \tag{27}$$

Moreover, we get that if $\phi \mapsto \phi'$ and $s_1 \sim s_2$ then

$$s_2 \in [\![\phi']\!]_M \text{ in (22)} \iff s_1 \in [\![\phi]\!]_M \text{in (18)}. \tag{28}$$

18. From a structural point of view this shortage leads to an asymmetry in the logic. We have propositions that only hold at the maximal information state, $\emptyset$, for example, $\bot$.

19. If $I\!P$ is finite ($\{p_1,\ldots,p_n\}$), then there is a second alternative. In this case we need to allow conjunctions over the *might*-sentences and as a consequence we obtain a proposition which only holds at the minimal and at the maximal information state: the conjunction of all *might*-sentences over all conjunctions of the form $\phi_1 \wedge \ldots \wedge \phi_n$ where $\phi_i = p_i$ or $\phi_i = \neg p_i$ for all $i \in \{1,\ldots,n\}$. Let's call this proposition ϕ, then it can be shown that

$$[\![[\phi]^{+\mu}[\phi_1]^{+\mu}\ldots[\phi_n]^{+\mu}\psi]\!]_M = S \iff \phi_1,\ldots,\phi_n \models_1 \psi.$$

The structural reason for this solution is that $\emptyset$ verifies all formulas, and therefore, the left-hand side of this equation only holds if $\{0,1\}^{I\!P} \in [\![[\phi_1]^{+\mu}\ldots[\phi_n]^{+\mu}\psi]\!]_M$.

This correspondence between belief revision and dynamic epistemic formalisms has been used to check the belief revision postulates of [AGM85] on the one hand, and finding modal axiomatizations on the other. Examples of such approaches are [Bou94] and [Seg95].

4.2.4 Going down in update semantics

Our procedure of dynamification as defined in section 2 also entails downdate meanings in update semantics, and in fact, we can completely transfer the revision/contraction definition as given by the specification in (18) into the possible world setting of **US-m** by the Grove connection as explained in Observation 4.5. In this way the dramatic effect of full meet revision becomes visible. Downdating a set of ϕ-worlds s with ϕ is the same as adding all counterworlds of ϕ. Transferring the full meet contracting to **US-m** by the Grove connection entails:

$$s^-\phi = \begin{cases} s & \text{if } [\![\neg\phi]\!] = S = \{0,1\}^{I\!P}; \\ s \cup \{V \in S \mid V \not\models \phi\} & \text{otherwise.} \end{cases} \tag{29}$$

Contractions of $\neg B$-sentences are not discussed in the AGM-setting. Downdating sentences of the form $\mathit{might}\,\phi$ yields a similar separation into two cases as for $\mathit{might}\,\phi$-updates. If $\mathit{might}\,\phi$ is currently absent, then all valuations in the corresponding information state reject ϕ, which means that $\neg\phi$ must hold in this state. If $\mathit{might}\,\phi$ holds, then there exists a valuation within the current state which verifies ϕ. However, there is no possible reduction to remove this information, because going back along the information order $\supseteq$ means that only valuations can be added, and hence $\mathit{might}\,\phi$ remains true: $\neg\phi \vee [\mathit{might}\,\phi]^{-\mu}\bot$ is a theorem in this system.[20]

In [Eij96] linguistic arguments have been given against this lack of dynamic meaning of contingency modals like 'might' and 'maybe'. Van Eijck proposes that the dynamic effect of such incoming sentences have to be treated by means of downdates of the negation of the subsentence which appears in the scope of the modality. Let us call this operation mix, then

$$\begin{aligned} [\![\phi]\!]^{\mathsf{mix}}_{M,s} &= [\![\phi]\!]^{+\mu}_{M,s} \\ [\![\mathit{might}\,\phi]\!]^{\mathsf{mix}}_{M,s} &= [\![\neg\phi]\!]^{-\mu}_{M,s} \quad \text{for all } \phi \in \mathcal{L}_0. \end{aligned} \tag{30}$$

Another possibility which Van Eijck proposes in [Eij94] is that states can be defined as partial valuations rather then total ones, which also stipulates informative update effects for such sentences. The intuition is that the argument of a *might*-sentence is brought into the scope of the interpreter's awareness (after [FH88]).

5 Discourse logics

In this section we will show how various discourse logics fit into the general perspective of this paper. The basic purpose of discourse logics is to offer an analysis of context sensitive phenomena of natural language discourse of which anaphoric relations forms probably the most thoroughly investigated subclass. Logical analysis of these phenomena has led to a deviation from the standard compositional Montagovian interpretation. In the logical study of anaphors the two central examples are texts, such as (31) and the *donkey-sentences*, of which (32) is a primary example.

A farmer owns a donkey. He beats it. (31)

If a farmer owns a donkey, he beats it. (32)

The question is how to account for the link between the indefinite NPs *a farmer* and *a donkey*, represented as introductions of variables, and the pronouns *he* and *it*, represented as free occurrences of variables. It seems very difficult to enforce both an elegant compositional

20. Note that this argument is not valid for the empty set, but nevertheless, this absurd state verifies $\neg\phi$.

mapping to the logical form of standard first-order logic and stick to the ordinary Tarskian interpretation as a logical semantics.
A general answer to the model-theoretic analysis is the dynamification of variable assignments as defined in the relational setting of [Bar87]. In this chapter we will restrict ourselves to discourse logics in which the information states are variable assignments. They differ in what *kind* of (sets of) assignments and how information growth over such assignments is defined. Most often, the languages which are employed to reason about the dynamics of assignments by different discourse logics are poor enough to be highly insensitive for such model-theoretic commitments. But when the expressiveness of such logics is extended , e.g., to extend the applicability of a given discourse logic, the $S, \sqsubseteq$-choice may enlarge the distinction between theories, both in their linguistic use and their logical consequences.

5.1 Discourse representation theory

Throughout this section we will use the logical language of first-order *discourse representation theory* (DRT) of [Kam81]. The difference with the language of first-order logic is that quantifiers are replaced by special variable registers, which are meant as a representation of the *discourse referents* that are introduced in a discourse. A register V is one of the two components of a so-called *discourse representation structure* (DRS) $[V \mid C]$, of which the other component is a set of *conditions* C over which the variables of the register bind. The intended meaning is that the represented discourse both has introduced the variables V and imposed the conditions C on those variables. The attractiveness of this logical representation language is that an easy construction algorithm can be given from a natural language discourse to a DRS [KR93].
The primitive conditions are the same as the atomic sentences of first order logic. We will call this atomic language $\mathcal{L}_{\mathbf{at}}$. Let Con and Var be two non-empty, disjoint sets (for constants and variables respectively). The set of terms, Term, is the union of Con and Var. Let Pred^n be a set of n-ary predicates with $= \in \mathsf{Pred}^2$. Then the set of atoms, $\mathcal{L}_{\mathbf{at}}$, is defined as the smallest set such that $Pt_1 \dots t_n \in \mathcal{L}_{\mathbf{at}}$, for all $P \in \mathsf{Pred}^n$ and $t_1 \dots t_n \in \mathsf{Term}$. We write $t_1 = t_2$ instead of $= t_1 t_2$.
Complex conditions are constructed by connecting DRSs. Standard connectives of DRT are negation $\neg$, disjunction $\vee$ and implication $\rightarrow$. In order to distinguish between DRSs and conditions we write capital Greek letters for the former expressions and small Greek letters for conditions. The language of conditions is called $\mathcal{L}_{\mathbf{cond}}$ and the language of DRSs is written as $\mathcal{L}_{\mathbf{DRS}}$.
Our first choice of information states for DRT is the set of *partial variable assignments* over a given first-order model $\langle D, I \rangle$ where D is the domain of individuals and I the interpretation of predicates and constants: $\mathsf{Pred}^n \mapsto \wp(D^n)$, $\mathsf{Con} \mapsto D$. This system that we will discuss here, called **DRT-p**, is inspired by the semantics as given by [MvBV95] (see Observation 5.1). The associated collection of partial assignments is written as $\mathcal{A}_D$, i.e., the set of partial maps from Var to D. The domain $\mathsf{Dom}(a)$ of a partial assignment a is the set of variables which are defined by a. The information order is defined in the following way:

$$a \sqsubseteq b \Leftrightarrow (x \in \mathsf{Dom}(a) \Rightarrow b(x) = a(x)). \tag{33}$$

This means that b gives the same values to variables as a as far as they are defined by a. An expansion of an assignment can be seen as the enrichment of this assignment when new referents are introduced in the discourse.
Our target static languages $\mathcal{L}$ consists of conjunctions and disjunctions over the atoms and the absurd proposition $\bot$. Our choice for the $\mathcal{L}^*$-parameter of the DML-specification consists of an addition of the update modal operators $[.]^{+\mu}$ and $\langle . \rangle^{+\mu}$ to this language $\mathcal{L}$. The DML-specification is displayed in table (34).[21]

21. Here $\top$ abbreviates $c = c$, for some $c \in \mathsf{Con}$. The specification function $^\bullet$ will look familiar to the reader who has seen embeddings of DRT into Pratt's quantificational dynamic logic, see [Mus94] for example.

$$
\begin{array}{|lcl|}
\hline
\mathcal{L}_{\mathbf{cond}} & = & \mathcal{L}_{\mathbf{at}} \cup (\mathcal{L}_{\mathbf{DRS}} + \{\neg, \vee, \rightarrow\} \setminus \mathcal{L}_{\mathbf{DRS}}) \\
\mathcal{L}_{\mathbf{DRS}} & = & [V \mid C] \text{ with } V \text{ and } C \text{ being finite} \\
 & & \text{subsets of Var and } \mathcal{L}_{\mathbf{cond}}, \text{ respectively.} \\
\hline
\mathcal{L} & = & \mathcal{L}_{\mathbf{at}} + \{\wedge, \vee, \bot\} \\
\mathcal{L}^* & = & \mathcal{L} * \{[.]^{+\mu}, \langle . \rangle^{+\mu}\} \\
\hline
\mathcal{M}_{\mathbf{DRT\text{-}p}} & = & \text{The models } M_{\langle D,I \rangle} = \langle S, \sqsubseteq, [\![.]\!] \rangle \text{ with } \langle D, I \rangle \\
 & & \text{a given first-order model, and} \\
 & & \qquad S = \mathcal{A}_D \\
 & & \qquad a \sqsubseteq b \Leftrightarrow \forall x \in \mathsf{Dom}(a) : a(x) = b(x) \\
 & & \qquad [\![Pt_1 \dots t_n]\!] = \{a \in \mathcal{A}_D \mid \langle t_1^{I,a}, \dots, t_n^{I,a} \rangle \in I(P)\} \\
 & & \qquad [\![t_1 = t_2]\!] = \{a \in \mathcal{A}_D \mid t_1^{I,a} = t_2^{I,a}\}. \\
 & & \text{with for each } t \in \mathsf{Term}: \\
 & & \qquad t^{I,a} = \begin{cases} I(t) & \text{if } t \in \mathsf{Con} \\ a(t) & \text{if } t \in \mathsf{Var} \text{ (and } t \in \mathsf{Dom}(a)). \end{cases} \\
\hline
\multicolumn{3}{|l|}{\text{For } \phi \in \mathcal{L}_{\mathbf{at}} : \phi^\bullet = \phi.} \\
\multicolumn{3}{|l|}{\text{For complex conditions:}} \\
\multicolumn{3}{|l|}{(\neg\Phi)^\bullet = [\Phi^\bullet]^{+\mu}\bot, (\Phi \rightarrow \Psi)^\bullet = [\Phi^\bullet]^{+\mu}\langle\Psi^\bullet\rangle^{+\mu}\top, (\Phi \vee \Psi)^\bullet = \langle\Phi^\bullet\rangle^{+\mu}\top \vee \langle\Psi^\bullet\rangle^{+\mu}\top.} \\
\multicolumn{3}{|l|}{\text{For DRSs: } [x_1 \dots x_n \mid \phi_1 \dots \phi_n]^\bullet = x_1 = x_1 \wedge \dots \wedge x_n = x_n \wedge \phi_1^\bullet \wedge \dots \phi_n^\bullet.} \\
\hline
\end{array}
\tag{34}
$$

Observation 5.1 *For the sublanguage which only contains* proper *DRSs:*[22] *the $+\mu$-meaning with respect to a model $M = M_{\langle D,I \rangle}$ of a DRS Φ, i.e., $[\![\Phi^\bullet]\!]^{+\mu}_M$, is the same as the meaning of the DRS Φ over $\langle D, I \rangle$, and the static meaning $[\![\phi^\bullet]\!]_M$ is the same as the meaning of the condition ϕ, as given in e.g. [MvBV95].*

In the case of improper DRSs the translation does not entail the same interpretation as defined by Muskens *et al.* The simple reason of the soundness for proper DRSs is that if all free variables of the conditions C in a DRS $[V \mid C]$ occur in V then $[\![[V \mid C]^\bullet]\!]^{+\mu}_{M,a}$ is the same as

$$[\![\bigwedge_{x \in V} x = x]\!]^{+\mu}_{M,a} \cap [\![\bigwedge_{\phi \in C} \phi^\bullet]\!]_M. \tag{35}$$

In other words, the conditions have no dynamic *effect* as such. This is exactly the reading which is given to conditions in the ordinary partial assignment semantics of Muskens *et al.* The equation (35) does not hold if C contains free variables which are not in V, and therefore also free in $[V \mid C]$. For example, if Λ is the empty assignment ($\mathsf{Dom}(\Lambda) = \emptyset$), then $[\![[\emptyset \mid \{Px\}]^\bullet]\!]^{+\mu}_{M,\Lambda} = \{a \in \mathcal{A}_D \mid \mathsf{Dom}(a) = x, a(x) \in I(P)\}$, while the definition as given by Muskens *et al.* yields no output for Λ. Note that improper DRSs do not occur after the DRT-construction algorithm is applied to meaningful discourses [KR93].
The DRSs which can be obtained after application of this construction algorithm to the standard examples (31) and (32) look as follows:

$$[\{x, y\} \mid \{Fx, Dy, Oxy, Bxy\}] \tag{36}$$

$$[\emptyset \mid [\{x, y\} \mid \{Fx, Dy, Oxy\}] \rightarrow [\emptyset \mid \{Bxy\}]] \tag{37}$$

The DML-translations respectively yield:

$$x = x \wedge y = y \wedge Fx \wedge Dy \wedge Oxy \wedge Bxy \tag{38}$$

$$[x = x \wedge y = y \wedge Fx \wedge Dy \wedge Oxy]^{+\mu}\langle Bxy \rangle^{+\mu}\top \tag{39}$$

22. Proper DRSs are DRSs with no free variables [KR93] (def.1.4.3,p.111).

Note that (39) is equivalent to $[x = x \wedge y = y \wedge Fx \wedge Dy \wedge Oxy]^{+\mu} Bxy$. This reduction of an implication always holds for conclusions whose free variables are present in the antecedent of the implication.
The DRSs as given in (36) and (37) are not only proper, but there is no superfluous use of variables, i.e., the variables in the register are exactly those which are free in the conditions. For this type of DRSs the translation can be simplified, because:

$$[\![[V \mid C]^\bullet]\!]^{+\mu}_M = [\![\bigwedge_{\phi \in C} \phi^\bullet]\!]^{+\mu}_M. \tag{40}$$

5.2 Alternatives

In [GS91] (def.26,p.77) a relational semantics for DRT has been given in terms of total variable assignments. The difference with **DRT-p** comes with the interpretation of the DRSs. The dynamic meaning of such a DRS $[V \mid C]$, according to Groenendijk and Stokhof's definition, is the set of assignment *switches* which change (only) the variables in V in such a way that all the conditions in C will hold afterwards. This means that introduction of variables is no longer interpreted as the addition of information, but as the adjustment of values of variables. As one would expect, this causes the DML-specification to get much more complicated. In fact, downdate expressivity is needed to capture this 'reset' semantics for introducing variables.
Besides the switch from partial to total assignments, the switch from single assignments to sets of assignments is also argued for. In the theory of file change semantics of [Hei82] sets of partial assignments $\mathcal{A}_D$ are chosen as information states. The corresponding information order is the following:

$$\forall A, B \subseteq \mathcal{A}_D : A \sqsubseteq B \Longleftrightarrow \forall b \in B \exists a \in A \forall x \in \mathsf{Var} : x \in \mathsf{Dom}(a) \Rightarrow a(x) = b(x). \tag{41}$$

In [Dek93] a similar semantics has been used to define an update semantics in the style of [Vel91] for Groenendijk and Stokhof's dynamic predicate logic [GS91], that is, conditions may eliminate possible assigments. In other words, conditions may have a dynamic effect.
For precise DML-specifications of these alternatives we refer to [JK96b]. For more details on the relation between the three types of discourse semantics as discussed in this section, see e.g., [Kra95].

6 Conclusions and the future

In this paper we have presented a survey of dynamic modal interpretations of existing systems as defined in 2.4. We have shown that such specifications give a clear picture of how dynamic theories can be defined as logics of information growth and reduction, and how such a dynamic modal language can be used to describe similarities and differences of dynamic logics. As we said in the introduction, dynamic modal logic as a meta-logic cannot and should not be used to replace specific dynamic theories. It is a single representation formalism to think about different systems at the same time. We think of this paper as a first step towards such general research on dynamic theories.
By way of conclusion, let us make the following three comments about the potentials of the dynamic 'common general purpose logic' as we have introduced and 'empirically' examined above. First, it points in the direction of a 'core dynamic logic'. We can start characterizing *common actions* found in given classes of 'logics of informational change' as well as *distinguishing axioms* which separate between them. This also makes it possible to reduce proliferation in the field, which is very useful in the light of the ever increasing divergence of dynamic logics.
Second, it creates the possibility of transfer of logical knowledge between systems. For instance, the constructive logics have been around for a long time and have been studied

extensively. We may hope that this contributes to the meta-theoretical study of modern dynamic logics.
Finally, it not only provides us with a method to characterize *existing* dynamic systems in terms of possible world semantics, it also enables us to create *new* dynamic systems. For that we only need to define a 'dynamic' component on top of an existing static logic. Once a static interpretation and an information order have been specified we get all sorts of dynamic interpretations for free. One of the prospects of this 'generating capacity' is that existing semantic theories can more easily be modified when we follow the corresponding DML-specifications.
A last, technical question concerning our project of modal unification is whether the minimal systems, that is, the logical systems which correspond to the complete (or large) classes of information models, behave well. For example, can we define nice axiomatizations such as Gentzen sequent systems for such minimal systems. In [Jas95] it was shown that the relational part (modulo the static part) of the minimal system for $+$ and $-$ has such a nice Gentzen calculus. The question is whether such a result can be extended to the case of the four basic actions $+$, $-$, $+\mu$ and $-\mu$. Another question in this respect is whether the minimal system is decidable. The $+,-$ system surely is, but the complete original dynamic modal logic is not [Rij93]. What about our 'intermediate' logic?

References

[AGM85] C.E. Alchourron, P. Gärdenfors, and D. Makinson. On the logic of theory change. *Journal of Symbolic Logic*, 50:510–530, 1985.

[Bar87] J. Barwise. Noun phrases, generalized quantifiers and anaphora. In P. Gärdenfors, editor, *Generalized Quantifiers: linguistic and logical approaches*, pages 1–30. Reidel, Dordrecht, 1987.

[BC94] J. van Benthem and G. Cepparello. Tarskian variations; dynamic parameters in classical semantics. Technical Report CS-R9419, CWI, Amsterdam, March, 1994.

[Ben91] J. van Benthem. *Language in Action: categories, lambdas and dynamic logic*. Studies in Logic 130. Elsevier, Amsterdam, 1991.

[Bla86] S. Blamey. Partial logic. In D.M. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic, Vol III: Alternatives to Classical Logic*, pages 1–70. Reidel, Dordrecht, 1986.

[Bou94] C. Boutilier. Unifying default reasoning and belief revision in a modal framework. *Artificial Intelligence*, 68:33–85, 1994.

[Bun90] H.C. Bunt. DIT dynamic interpretation in text and dialogue. In L. Kálmán and L. Pólós, editors, *Papers from the Second Symposium on Logic and Language*, pages 67–104, Budapest, 1990. Akadémiai Kiadó.

[Dek93] P. Dekker. *Transsentential Meditations*. PhD thesis, ILLC, University of Amsterdam, 1993.

[Eij94] J. van Eijck. Presupposition failure — a comedy of errors. *Formal Aspects of Computing*, 6:1–22, 1994.

[Eij96] J. van Eijck. Presuppositions and information updating. In M. Kanazawa, C. Piñon, and H. de Swart, editors, *Quantifiers, Deduction, and Context*. CSLI, 1996.

[FH88] R. Fagin and J.Y. Halpern. Belief, awareness and limited reasoning. *Artificial Intelligence*, 34:39–76, 1988.

[Fit69] M.C. Fitting. *Intuitionistic Logic: Model Theory and Forcing*. Studies in Logic and the Foundations of Mathematics. North Holland, Amsterdam, 1969.

[Gab82] D.M. Gabbay. Intuitionistic basis for non-monotonic logic. In D.W. Loveland, editor, *Proceedings of the 6th Conference on Automated Deduction*, number 138 in Lecture Notes in Computer Science, pages 260–273, Heidelberg, 1982. Springer.

[Gär88] P. Gärdenfors. *Knowledge in Flux: Modelling the Dynamics of Epistemic States*. MIT Press, Cambridge Mass, 1988.

[Gro88] A. Grove. Two modellings of theory change. *Journal of Philosophical Logic*, 17:157–170, 1988.

[GS91] J. Groenendijk and M. Stokhof. Dynamic predicate logic. *Linguistics and Philosophy*, 14:39–100, 1991.

[GSV94] J. Groenendijk, M. Stokhof, and F. Veltman. This might be it. DYANA Deliverable 6852, ESPRIT Basic Research, 1994.

[Hei82] I. Heim. *The Semantics of Definite and Indefinite Noun Phrases.* PhD thesis, University of Massachusetts, Amherst, 1982.

[Jas95] J. Jaspars. Partial up and down logic. *Notre Dame Journal of Formal Logic*, 36(1):134–157, 1995.

[JK96a] J. Jaspars and M. Kameyama. Preferences in dynamic semantics. In P. Dekker and M. Stokhof, editors, *Proc. of the 10th Amsterdam Colloquium*, pages 425–444, Amsterdam, 1996. ILLC. This volume.

[JK96b] J. Jaspars and E. Krahmer. A programme of modal unification of dynamic theories: Full report. Technical report, CWI, Amsterdam, 1996. Forthcoming.

[Joh37] I. Johansson. Der minimalkalkül, ein reduzierter intuitionistischer formalismus. *Compositio Mathematicae*, 4:119–136, 1937.

[Kam81] H. Kamp. A theory of truth and semantic representation. In J. Groenendijk, Th. Jansen, and M. Stokhof, editors, *Formal Methods in the Study of Language*, pages 277–322. Mathematisch Centrum, Amsterdam, 1981.

[KR93] H. Kamp and U. Reyle. *From Discourse to Logic.* Kluwer, Dordrecht, 1993.

[Kra95] E. Krahmer. *Discourse and Presupposition.* PhD thesis, Tilburg University, November 1995.

[Kri65] S.A. Kripke. Semantical analysis of intuitionistic logic I. In J. Crossley and M. Dummett, editors, *Formal Systems and Recursive Functions*, pages 92–130. North Holland, Amsterdam, 1965.

[Lan86] F. Landman. *Towards a Theory of Information: The Status of Partial Objects in Semantics.* Number 6 in Groningen Amsterdam Studies in Semantics. Foris, Dordrecht, 1986.

[Lew73] D. Lewis. *Counterfactuals.* Basil Blackwell, Oxford, 1973.

[MD80] D. McDermott and J. Doyle. Non-monotonic logic I. *Artificial Intelligence*, 13:41–72, 1980.

[Mus94] R. Muskens. Tense and the logic of change. In U. Egli et al., editor, *Interface Aspects of Syntax, Semantics and the Lexicon*, pages 147–183. W. Benjamins, 1994.

[MvBV95] R. Muskens, J. van Benthem, and A. Visser. "Dynamics". In J. van Benthem and A. ter Meulen, editors, *Handbook of Logic and Language.* Elsevier Science, 1995. To appear.

[Nel49] D. Nelson. Constructible falsity. *Journal of Symbolic Logic*, 14:16–26, 1949.

[PW90] D. Pearce and G. Wagner. Reasoning with negative information:strong negation in logic programs. *Acta Philosophica Fennica*, 49:430–453, 1990.

[Rij92] M. de Rijke. A system of dynamic modal logic. Technical Report Research Report 92-170, CSLI, Stanford, CA, 1992. to appear in the Journal of Philosophical Logic.

[Rij93] M. de Rijke. *Extending Modal Logic.* PhD thesis, ILLC, University of Amsterdam, 1993.

[Rij94] M. de Rijke. Meeting some neighbours. In J. van Eijck and A. Visser, editors, *Logic and Information Flow*, pages 170–195. MIT Press, Cambridge, Mass., 1994.

[Seg95] K. Segerberg. Belief revision from the point of view of doxastic logic. *Bulletin of the IGPL*, 3(4):535–554, 1995.

[Tur84] R. Turner. *Logics for Artificial Intelligence.* Ellis Horwood, Chicester, 1984.

[TvD88] A.S. Troelstra and D. van Dalen. *Constructivism in Mathematics, Vol. 1.* Number 121 in Studies in Logic and the Foundations of Mathematics. Elsevier Science Publishers, Amsterdam, 1988.

[Vel85] F. Veltman. *Logics for Conditionals.* PhD thesis, University of Amsterdam, Amsterdam, 1985.

[Vel91] F. Veltman. Defaults in update semantics. Technical report, Department of Philosophy, University of Amsterdam, 1991. To appear in the Journal of Philosophical Logic.

[Wan93] H. Wansing. *The Logic of Information Structures*, volume 619 of *Lecture Notes in Computer Science.* Springer, Heidelberg, 1993.

[Wan95] H. Wansing. Semantics-based nonmonotonic inference. *Notre Dame Journal of Formal Logic*, 36(1):44–54, 1995.

Preferences in Dynamic Semantics

Jan Jaspars* **Megumi Kameyama†**

Abstract

In order to enrich dynamic semantic theories with a 'pragmatic' capacity, we combine dynamic and nonmonotonic (preferential) logics in a modal logic setting. We extend a fragment of Van Benthem and De Rijke's dynamic modal logic with additional preferential operators in the underlying static logic, which enables us to define defeasible (pragmatic) entailments over a given piece of discourse. We will show how this setting can be used for a dynamic logical analysis of preferential resolutions of ambiguous pronouns in discourse.

1 Introduction

The goal of model–theoretic semantics is to establish an interpretation function from the expressions of a given language to a class of well–understood mathematical structures (models). This enables a formal logical understanding of what an expression means and what its consequences are. For instance, natural language semantics has recently developed a relatively simple model–theoretic understanding of the dynamic interplay between indefinite descriptions and anaphoric bindings. These dynamic semantic theories of natural language give model–theoretic explanations of *possible* anaphoric bindings, assuming that additional pragmatics will address the issues of anaphora resolution. A correct dynamic semantic analysis predicts each of the possible referents available in the context, just as a classical logical analysis 'lists' all possible scoping and lexical ambiguities.

Consider the following simple discourses (1) and (2).

(1) John met Bill at the station. He_1 greeted him_1.

(2) Bill met John at the station. He_1 greeted him_1.

The two discourses are semantically equivalent. A precise dynamic semantic analysis would treat he_1 and him_1 in both examples as variables that range over the semantic values of John and Bill, with the additional constraint that the referents of he_1 and him_1 are different. This analysis predicts two sets of equally possible bindings. There is, however, a clear preferential difference between the two discourses. There is a preference for the bindings, he_1 = John and him_1 = Bill, in (1), and for the opposite bindings, he_1 = Bill and him_1 = John, in (2). Preferential effects on discourse interpretations and the entire issue of ambiguity resolution have traditionally been put outside the scope of logical semantics, into the more or less disjoint subfield of "pragmatics." This academic focus sharply contrasts with the importance placed on disambiguation and resolution issues in natural language processing (or computational linguistics), where realistic accounts of naturally occurring discourses and dialogues are constantly demanded from application systems. Computational accounts, however, often fall short of logical or model–theoretic formalizations. In artificial intelligence (AI), in contrast, logical formalization of pragmatics, or defeasible reasoning, was brought into the central focus of research at an early stage [McCarthy and Hayes, 1969], and led to the development of nonmonotonic logics.

More recently, there are proposals to incorporate defeasible reasoning within natural language semantics in order to approximate the class of realistic conclusions of a given sentence

* CWI, Dept. of Interactive Systems, P.O. Box 94079, NL-1090 GB Amsterdam, The Netherlands, jaspars@cwi.nl. Jan Jaspars was supported by CEC-project LRE-62-051 (FraCaS).

† SRI International, Artificial Intelligence Center, Ravenswood Ave. 333, CA 94025 Menlo Park, USA, megumi@ai.sri.com. Megumi Kameyama's work was in part supported by the National Science Foundation and the Advanced Research Projects Agency under Grant IRI-9314961 (Integrated Techniques for Generation and Interpretation).

or discourse, as exemplified by [Veltman, 1991] and [Lascarides and Asher, 1993]. In contrast with these specific proposals,[1] we will propose a *general* framework for preferential dynamic semantics, and illustrate how the basic properties of discourse pragmatics exhibited by ambiguous pronouns can be encoded within the framework.

The present framework combines a general model of nonmonotonic logic [Shoham, 1988] and a general model of dynamic logic [Van Benthem, 1991] [De Rijke, 1992]. In this logical set–up, we specify defeasible information and associated entailment relations over a given discourse, and classify the relative stability of conclusions made on the basis of this additional defeasible information. The paper is about a general framework of preferential dynamic semantics that abstracts away from numerous specific possibilities for how to represent utterance logical forms and discourse contexts, and how to actually compute preferences. Since logical formalization of discourse pragmatics is in an early stage of development, we believe that it benefits immensely from an attempt such as here to sort out general meta–theoretical issues from specific accounts.

The paper is organized as follows. Section 2 summarizes the preferential effects on ambiguous discourse anaphoric pronouns. Section 3 presents our basic logical framework. Section 4 illustrates formalisms at work in pronoun interpretation in a first–order discourse logic.

2 Preferences in Ambiguous Pronouns

In this section, we summarize the basic properties of preferential effects on discourse semantics. We focus on ambiguous pronouns in simple discourses, and illustrate the properties of dynamicity, defeasibility, indeterminacy, and preference classes.

2.1 Discourse Pragmatics as Preferential Reasoning

Most present–day linguistic theorists assume the trichotomy of syntax, semantics, and pragmatics, but there is no single agreed–upon definition of exactly what *linguistic pragmatics* is. Some equate it with "indexicality", some with "context dependence", and others with "language use" (cf. [Levinson, 1983], introduction). There is also a common pipeline view of the trichotomy, in that, pragmatics adds interpretations to the output of semantics that interprets the output of syntax. In this pipeline view, the direct link between syntax and pragmatics is lost.

We take a logic–inspired definition of pragmatics as the *nonmonotonic* subsystem characterized by *defeasible* rules. We also view all defeasible rules to be *preferences*, so the pragmatics subsystem corresponds to a subspace of preferential reasoning, which *controls* the subspace of *possible* interpretations carved out by the indefeasible linguistic rules in the "grammar" subsystem.[2] From this perspective, pragmatics is not an underdeveloped subcomponent of semantics alone, but a system that combines all the preferential aspects of phonology, morphology, syntax, semantics, and epistemics. There is evidence that these heterogeneous linguistic preferences interact with one another, and also with nonlinguistic preferences coming from the commonsense world knowledge. What we have then is a dichotomy of grammar and pragmatics subsystems rather than a trichotomy. Under this view, neither indexicality nor context dependence defines pragmatics since there are both indefeasible and defeasible indexical and context dependent rules. In fact, in a *dynamic* architecture for discourse semantics, where meaning is given to a sequence of sentences rather than to a sentence in isolation, context dependence is an inherent architectural property supporting the anaphoricity of natural language expressions.

1. Veltman [1991] defines default reasoning in terms of his update semantics. Lascarides and Asher [1993] extend Discourse Representation Theory (DRT) with the definition of commonsense entailment given in [Asher and Morreau, 1991].
2. We assume, following the theoretical linguistic tradition, that there is a linguistic rule system consisting of indefeasible rules of morphosyntax and semantics, and call it the 'grammar subsystem'. We also assume that most commonsense rules are defeasible, but leave the question open as to whether there are also indefeasible commonsense rules.

Grammatical Effects:	
A.	John hit Bill. Mary told *him* to go home.
B.	Bill was hit by John. Mary told *him* to go home.
C.	John hit Bill. Mary hit *him* too.
D.	John hit Bill. *He* doesn't like *him*.
E.	John hit Bill. *He* hit *him* back.
K.	Babar went to a bakery. He greeted the baker. *He* pointed at a blueberry pie.
L.	Babar went to a bakery. The baker greeted him. *He* pointed at a blueberry pie.
Commonsense Effects:	
F.	John hit Bill. *He* was severely injured.
G.	John hit Arnold Schwarzenegger. *He* was severely injured.
H.	John hit the Terminator. *He* was severely injured.
I.	Tommy came into the classroom. He saw Billy at the door. He hit him on the chin. *He* was severely injured.
J.	Tommy came into the classroom. He saw a group of boys at the door. He hit one of them on the chin. *He* was severely injured.

Table 1: Discourse Examples in the Survey

2.2 Basic Properties of Linguistic Preferences

We will now motivate four basic properties of linguistic preferences with examples of ambiguous discourses with ambiguous pronouns. Kameyama [1996] analyzed a survey result of pronoun interpretation preferences from the perspective of interacting preference classes in a dynamic discourse processing architecture. This analysis identified a set of basic "design features" that characterize the preferential effects on discourse meaning, and outlined how they combine to settle on preferred discourse interpretations. These basic properties can be summarized as *dynamicity*, *indeterminacy*, *defeasibility*, and *preference class interactions*. Table 1 shows those examples discussed in [Kameyama, 1996]. In a survey, speakers had to pick the preferred reference of pronouns in the last sentence of each discourse example (shown in italics).[3] Table 2 shows the survey results.[4] These and similar examples will be used in this paper.

2.2.1 Dynamicity

We are interested in discourse pragmatics — discourse semantics enriched with preferences, so it is natural to start from where discourse semantics leaves off, not losing what discourse semantics has accomplished with its dynamic architecture and the view of sentence meaning as its context change potential. We thus take *dynamicity* to be a basic architectural requirement in an integrated theory of discourse semantics and pragmatics.[5]

The discourse examples (1) and (2) repeated below demonstrate the fact that the preferred interpretation of an utterance depends on the preceding discourse context.

(1) John met Bill at the station. He_1 greeted him_1.

(2) Bill met John at the station. He_1 greeted him_1.

3. The respondents were told to read the discourses with a "neutral" intonation, for the survey was intended to investigate only *unstressed* pronouns.
4. The $\chi^2_{df=1}$ significance for each example was computed by adding an evenly divided number of the "unclear" answers to each explicitly selected answer, reflecting the assumption that an "unclear" answer shows a genuine ambiguity.
5. There are two levels of dynamicity that affect utterance interpretation in discourse. One is the utterance–by–utterance dynamicity that affects the overall discourse meaning, and the other is the word–by–word or constituent–by–constituent dynamicity that affects the meaning of the utterance being interpreted. In this paper, we will focus on the former.

	Answers			$\chi^2_{df=1}$	p
A.	John 42	Bill 0	Unclear 5	37.53	$p < .001$
B.	John 7	Bill 33	Unclear 7	14.38	$p < .001$
C.	John 0	Bill 47	Unclear 0	47	$p < .001$
D.	J. dislikes B. 42	B. dislikes J. 0	Unclear 5	37.53	$p < .001$
E.	John hit Bill 2	Bill hit John 45	Unclear 0	39.34	$p < .001$
K.	Babar 13	Baker 0	Unclear 0	13	$p < .001$
L.	Babar 3	Baker 10	Unclear 0	3.77	$.05 < p < .10$
F.	John 0	Bill 46	Unclear 1	45.02	$p < .001$
G.	John 24	Arnold 13	Unclear 10	2.57	$.10 < p < .20$
H.	John 34	Terminator 6	Unclear 7	16.68	$p < .001$
I.	Tommy 3	Billy 17	Unclear 1	9.33	$.001 < p < .01$
J.	Tommy 10	Boy 7	Unclear 3	0.45	$.50 < p < .70$

Table 2: Survey Results

The two discourses are semantically equivalent. Two male persons, "John" and "Bill", engage themselves in a symmetric action of `meeting`. Both individuals are available for anaphoric reference in the next sentence, and since the two pronouns in "He greeted him" must be disjoint in reference and each pronoun has two possible values, dynamic semantic theories predict two equally possible interpretations, `John greeted Bill` and `Bill greeted John`. However, these discourses have different *preferred values* for these pronouns. In (1), due to a *grammatical parallelism preference* (exhibited by discourse D in Table 1), the preferred interpretation is `John greeted Bill`. In (2), the same parallelism preference leads to the reverse interpretation of `Bill greeted John`.

Dynamic semantics has been motivated by examples such as *A man walks in the park. He whistles.*, where an existential scope extends beyond the syntactic sentence boundary to bind pronouns. Analogously, preferential dynamic semantics would have to account for examples such as (1) and (2), where different syntactic configurations of the same semantic content have different *extended effects* on the preferred interpretation of pronouns.

2.2.2 Indeterminacy

One notable feature of the survey results shown in Table 2 is that the resulting $\chi^2_{df=1}$ significance varies widely. We consider preference to be *significant* if $p < .05$, *weakly significant* if $.05 < p < .10$, and *insignificant* if $.10 < p$ as a straightforward application of elementary statistics. It is reasonable to assume that the statistical significance of a preference corresponds to how determinate the given preference is. Significant preferences are thus unambiguous and determinate, and insignificant preferences indicate ambiguities and indeterminacies. The preferential machinery then must allow both unambiguous and ambiguous preferences to be concluded, rather than always producing a single maximally preferred conclusion.

Preferential reasoning is supposed to resolve ambiguities, however, and unresolved preferential ambiguities make discourses incoherent. It seems reasonable to assume a discourse pragmatic meta–principle that says, *a discourse should produce a single maximally preferred interpretation.* Such a meta–principle is akin to Gricean maxims of conversation, though nothing more is said about this connection here. It seems that this kind of a meta–principle is needed to assure that speakers try to avoid indeterminate preferences precisely because the underlying preferential logical machinery does not guarantee determinate preferences.

We thus identify a basic property of preferential reasoning — Preferential conclusions are sometimes *determinate* with a single maximally preferred interpretation, and other times *indeterminate* with multiple maximally preferred interpretations. The latter results in a genuine ambiguity, or incoherence, violating the basic pragmatic felicity condition.

Let us turn to concrete examples. Both discourses (1) and (2), repeated below, have determinate preferred interpretations due to the grammatical parallelism preference. In contrast, discourse (3) leads to no clear preference because no relevant preferences converge on a single determinate choice. Discourse (3) is thus infelicitous.

(1) John met Bill at the station. He greeted him.

(2) Bill met John at the station. He greeted him.

(3) John and Bill met at the station. He greeted him.

2.2.3 Defeasibility

A conclusion is *defeasible* if it may have to be retracted when some additional facts are introduced. This property is also called *nonmonotonicity*, and is the defining property of *preferences*. This property also defines *pragmatic*, as opposed to grammatical, conclusions under the present assumption that grammatical conclusions are indefeasible.

The following continuation of (1) illustrates defeasibility.

(4) John met Bill at the station. He greeted him. John greeted him back

In (4), the third sentence, with its indefeasible semantics associated with the adverb *back* (as in discourse E in Table 1), forces a reversal of the preferred interpretation concluded after the second sentence. This on–line reversal produces a discourse–level *garden path* effect, analogous to the sentence–level phenomena as in "The horse passed the barn fell" or "The astronomer married a star."

Garden path effects are cases of *preference reversal*, which should not be confused with explicit retractions or negations of indefeasible conclusions. The former can be triggered implicitly, whereas the latter must be explicitly asserted. The latter is illustrated by the following discourse–level *repair* example, where the explicit retraction signal "No" negates the immediately preceding assertion, and opens a way for a different fact to be asserted in the next sentence.

(5) John met Bill at the station. No. He met Paul there.

2.2.4 Preference Classes

When multiple preferences simultaneously succeed, the combined effects are quite unlike the familiar patterns of grammatical rule interactions. When mutually contradictory indefeasible rules both succeed, the whole interpretation fails. For instance, "John met Mary at the station. He knows that she loves himself." leads to no interpretation. In contrast, preferences may *override* other preferences that contradict them. Ambiguities persist only when mutually contradictory preferences are equally strong. A logical model of preferential reasoning, therefore, must predict ambiguity resolutions due to overrides.

One type of override is predicted by the so–called Penguin Principle, where the conclusion based on a more specific premise wins (see [Lascarides and Asher, 1993] for a linguistic application). This principle does not explain all the override phenomena in pragmatic reasoning, however. We must posit the existence of *preference classes* in order to predict overrides among groups of preferences [Kameyama, 1996]. We thus distinguish between two kinds of conflict resolutions in pragmatics, one due to the Penguin Principle and the other due to preference class overrides.[6] In this paper, we focus on the interaction between two major preference classes — the *syntactic preferences* based on the *surface structure* of utterances[7] and the *commonsense preferences* based on the *commonsense world knowledge.*

First consider two examples (A and B) in Table 1 repeated below.

(6) John hit Bill. Mary told him to go home.

(7) Bill was hit by John. Mary told him to go home.

6. In contrast, the law of 'Lexical Impotence' in [Asher and Lascarides, 1995] (p.96), for instance, accounts for a class–overriding phenomena, where discourse inferences generally override default lexical inferences, in terms of a "meta–penguin principle" forced on rule classes.

7. This includes both the parallelism and attentional preferences discussed in [Kameyama, 1996]. It was conjectured there that these preference classes may be independent subclasses of a larger 'entity–level' preference class, which is qualitatively different from the 'propositional–level' commonsense preference class.

	Syntactic Pref.	Commonsense Pref.	Semantics	Winner
A.	John	unclear	—	Syntactic Pref.
B.	Bill	unclear	—	Syntactic Pref.
C.	John	unclear	Bill	Semantics
D.	John–Bill	unclear	—	Syntactic Pref.
E.	John–Bill	unclear	Bill–John	Semantics
K.	Babar	unclear	—	Syntactic Pref.
L.	Baker	unclear	—	Syntactic Pref.
F.	John	Bill	—	Commonsense Pref.
G.	John	John/Arnold	—	Commonsense Pref.
H.	John	John	—	Commonsense Pref.
I.	Tommy	Billy	—	Commonsense (but difficult)
J.	Tommy	Boy(/Tommy)	—	??

Table 3: Preference Interactions

Discourses (6) and (7) illustrate a syntactic preference — the preference for the main grammatical subject to be the antecedent for a pronoun in the next utterance. Henceforth, this syntactic preference is called the *subject antecedent preference*. In (6), the preferred value of the pronoun *him* is John. In (7), with passivization, the preferred value shifts to Bill. Since passivization does not affect the thematic roles (such as Agent or Theme) of these referents, we conclude that this preference shift is directly caused by the shift in grammatical functions.

Next, consider the following.[8]

(8) John hit Bill. He got injured.

(9) The wall was hit by a champagne glass. It didn't break.

Discourses (8) and (9) illustrate that the above subject antecedent preference is overridden by a stronger class of preferences having to do with commonsense causal knowledge — in these cases, about hitting causing injuring or breaking.

We thus assume that there are preference classes, or modules, that independently conclude the preferred interpretation of an utterance, and that these class–internal conclusions interact in a certain general overriding pattern to produce the final preference. Table 3 shows the survey result analyzed from this perspective of preference class interactions. Based on this analysis, we would like to model the following general patterns of preference interactions:

- Indefeasible syntax and semantics override all preferences.
- Commonsense preferences override syntactic preferences.[9]
- Syntactic preferences dominate the final interpretation only if there are no relevant commonsense preferences.

The general overriding pattern we identify here is schematically shown below, where $\geq$ represents a "can override" relation:

Indefeasible Syntax and Semantics	$\geq$	Commonsense Preferences	$\geq$	"Syntactic" Preferences

There are a number of questions about these preference classes. For instance, how do they arise, how many classes are there, and why do some classes override others?[10] In this paper,

8. (8) is a slight variation of F in Table 1. (9) is a variant of Len Schubert's (p.c.) example.

9. This overriding can be difficult when the syntactic preference is extremely strong. For instance, example I in Table 1 creates an utterance–internal garden–path effect where the first syntactically preferred choice for Tommy is retracted in favor of a more plausible interpretation supported by commonsense preferences.

10. [Kameyama, 1996] proposed that there are three preference classes that respectively concern preferred updates of three data structure components of the dynamic context. These three preference classes also seem

we simply assume the existence of multiple preference classes with predetermined override relationships, and propose a logical machinery that implements their interactions.

We will now turn to the logical machinery that will be used to model pragmatic reasoning with the requisite properties of dynamicity, indeterminacy, defeasibility, and preference class interactions.

3 Dynamic Preferential Reasoning

We hav chosen to combine dynamics and preferences in a most general logical setting in order to achieve logical transparency and theoretical independence in the following sense. We hope that the logical simplicity facilitates future meta–logical investigations on the interaction of dynamics and preferential reasoning, and that it enables applications to a wider variety of preferential (defeasible) phenomena. We have thus chosen to combine the most general dynamic logical approach and the most general logical approach to defeasible reasoning we know. The dynamic (relational) setting consists of the core of the so-called dynamic modal logic of Van Benthem [1991] and De Rijke [1992]. Our encoding of defeasibility follows Shoham's [1988] preferential modelling of nonmonotonic logics.

Subsection 3.1 will outline dynamic modal logic, following Jaspars and Krahmer's [1996] fragment of the original logic. This part encodes the dynamicity property. Subsections 3.2 and 3.3 will show how preferential reasoning can be accommodated within this fragment of dynamic modal logic. This addition encodes defeasibility, indeterminacy, and differentiation of preference classes. Finally, subsection 3.4 discusses possible pragmatic meta–constraints on preferential interpretation definable in this logical setting.

3.1 Basic Dynamic Modal Logic

Jaspars and Krahmer [1996] present specifications of current dynamic semantic theories in terms of dynamic modal logic (DML), and show how DML can be used as a universal setting in which the differences and similarities among different dynamic semantic theories can be clarified. The underlying philosophy of this unified dynamics is that dynamic theories evolve from 'dynamifying' an ordinary logic by implementing an order of information growth over the models of this logic.

To start with, one chooses a *static language* $\mathcal{L}$ to reason about the content of *information states* S by means of an *interpretation function*: $[\![.]\!] : \mathcal{L} \longrightarrow \mathbf{2}^S$. This setting most often consists of a (part of) well–known logic interpreted over a class of well–known models. These models are then taken to be the units of information, i.e., information states, within the dynamic modal framework. The second (new) step consists of a definition of an *order of information growth*, $\sqsubseteq$, over these information states. We write $s \sqsubseteq t$ whenever the state t contains more information than s according to this definition. The conclusive step is the choice of the dynamic language $\mathcal{L}^*$, which essentially comes down to selecting different dynamic modal operators for reasoning about the relation $\sqsubseteq$. The triple $\langle S, \sqsubseteq, [\![.]\!] \rangle$ is also called an *$\mathcal{L}$-information model.*

Convention. If $M = \langle S, \sqsubseteq, [\![.]\!] \rangle$ is an $\mathcal{L}$-information model, then we write $s \sqsubset t$ whenever $s \sqsubseteq t$ and not $t \sqsubseteq s$. The state t is called a *proper extension* of s. In this paper, we will assume that all the chains of proper extensions in S are countable in the order of their information size as indicated by the order $\sqsubseteq$. Formally,

$$(10) \quad \forall T \subseteq S: \ (\forall s,t \in T : s \sqsubset t \text{ or } t \sqsubset s) \Rightarrow \exists f : \mathbb{N} \longrightarrow T : f[\mathbb{N}] = T \text{ and } \forall m,n \in \mathbb{N} : m \leq n \Rightarrow f(m) \sqsubseteq f(n).$$

to correspond with the three classes of *discourse coherence relations* independently proposed by [Kehler, 1995] to account for the constraints on ellipsis and other cohesive forms. This indicates a potential integration of two apparently unrelated notions — dynamic context data structure components and coherence relations.

The set $\mathbb{N}$ denotes the set of the natural numbers, $f[\mathbb{N}] = T$ means that f is surjective. In other words, if all elements of T can be distinguished by the information order $\sqsubseteq$, then we can rank them in a discrete fashion. Most often, dynamic semantic theories can be described on the basis of information models which satisfy this constraint.

3.1.1 Static and Dynamic Meaning

On the basis of these information models, one can distinguish between static and dynamic meanings of propositions. The *static meaning* of a proposition $\varphi \in \mathcal{L}$ with respect to an $\mathcal{L}$–information model $M = \langle S, \sqsubseteq, [\![.]\!] \rangle$, written as $[\![\varphi]\!]_M$, is the same as $[\![\varphi]\!]$. The reason is that we want to define a dynamic modal extension $\mathcal{L}^*$ on top of $\mathcal{L}$, which requires static interpretation as well ($[\![.]\!]_M : \mathcal{L}^* \longrightarrow \mathbf{2}^S$).
Given the relational structure, i.e., the pre–order of information growth $\sqsubseteq$, over the information states S, we are able to define a *dynamic meaning* of a proposition. Roughly speaking, the dynamic meaning of a proposition is understood as its *effect* on a given information state $s \in S$.[11] In other words, we wish to define the meaning(s) of a proposition φ *in the context of an information state* $s \in S$: $[\![\varphi]\!]_{M,s}$.
In general, different dynamic interpretations of a proposition φ are defined according to how φ *operates* on an information state. For example, φ might be added to or retracted from an information state, or, in a somewhat more complicated case, φ may describe the content of a revision to an information state. Given such an operation o, we will define the o-meaning of a proposition φ with respect to an information state $s \in S$ (in M): $[\![\varphi]\!]^o_{M,s}$. The proposition φ is the *content* of an operation and o specifies the *type* of operation. In DML, all these operations are defined in terms of the growth relation $\sqsubseteq$.
Jaspars and Krahmer [1996] postulate that in most well–known logics of mental action or change, we need only four basic operation types: *extension* $(+)$ and *reduction* $(-)$, and their minimal counterparts, *update* $(+\mu)$ and *downdate* $(-\mu)$. Given an information order $\sqsubseteq$ for a given set of information states S, these actions are defined as follows:

$$
\begin{array}{lcl}
[\![\varphi]\!]^+_{M,s} & = & \{t \in S \mid s \sqsubseteq t,\ t \in [\![\varphi]\!]_M\} \\
[\![\varphi]\!]^-_{M,s} & = & \{t \in S \mid t \sqsubseteq s,\ t \notin [\![\varphi]\!]_M\} \\
[\![\varphi]\!]^{+\mu}_{M,s} & = & \{t \in [\![\varphi]\!]^+_{M,s} \mid \forall u \in S : u \in [\![\varphi]\!]^+_{M,s}\ \&\ u \sqsubseteq t \ \Rightarrow\ t \sqsubseteq u\} \\
[\![\varphi]\!]^{-\mu}_{M,s} & = & \{t \in [\![\varphi]\!]^-_{M,s} \mid \forall u \in S : u \in [\![\varphi]\!]^-_{M,s}\ \&\ t \sqsubseteq u \ \Rightarrow\ u \sqsubseteq t\}
\end{array} \tag{11}
$$

Furthermore, for every action type o we use $[\![\varphi]\!]^o_{M,T}$ as an abbreviation of the set $\bigcup_{s \in T} [\![\varphi]\!]^o_{M,s}$ (the o-meaning of φ with respect to T) for all $T \subseteq S$. A special instance of particular importance is the o-meaning with respect to the minimal states in M: $\min_M = \{s \in S \mid \forall t \in S : t \sqsubseteq s \Rightarrow s \sqsubseteq t\}$. We write $[\![\varphi]\!]^o_{M,\min}$ instead of $[\![\varphi]\!]^o_{M,\min_M}$, and refer to this set as the minimal o-meaning of φ in M. This is the meaning of a proposition with respect to an empty context. We will also use the notation $\min_M T$ for a given subset $T \subseteq S$ of information states. It refers to the set of minimal states in T: $\{s \in T \mid \forall t \in T : t \sqsubseteq s \Rightarrow s \sqsubseteq t\}$. The presumption (10) for information models implies that $\min_S T \neq \emptyset$ whenever $T \neq \emptyset$, and therefore, $[\![\varphi]\!]^+_{M,s} \neq \emptyset \Rightarrow [\![\varphi]\!]^{+\mu}_{M,s} \neq \emptyset$ (the same holds for $-$ with respect to $-\mu$).
Dynamic semantic theories most often describe relational meanings of propositions obtained from abstractions over the context. For every operation o, we will call the relational interpretation the *o-meaning* of φ (in M).

$$
[\![\varphi]\!]^o_M \ = \ \{\langle s, t\rangle \mid t \in [\![\varphi]\!]^o_{M,s}\} \tag{12}
$$

Finally, a dynamic modal extension $\mathcal{L}^*$ of $\mathcal{L}$ can be defined. It supplies unary dynamic modal operators of the form $[\varphi]^\circ$ and $\langle\varphi\rangle^\circ$, whose static interpretations are as follows:

$$
\begin{array}{lcl}
[\![[\varphi]^\circ \psi]\!]_M & = & \{s \in S \mid [\![\varphi]\!]^o_{M,s} \subseteq [\![\psi]\!]_M\} \\
[\![\langle\varphi\rangle^\circ \psi]\!]_M & = & \{s \in S \mid [\![\varphi]\!]^o_{M,s} \cap [\![\psi]\!]_M \neq \emptyset\}
\end{array} \tag{13}
$$

11. Note that linguistic actions most often affect the mental state of some chosen agents or interpreters, sharply contrasting with physical actions that affect physical situations, as studied in AI for analysis of so-called frame problems, e.g., [Shoham, 1988].

For example, a proposition of the form $[\varphi]^+ \psi$ means that extending the current state with φ necessarily leads to a ψ-state, while $\langle\varphi\rangle^{-\mu} \psi$ means that it is possible to retract φ from the current state in a minimal way and end up with the information ψ. In this paper, we will only discuss the extension $(+)$ and update $(+\mu)$ meanings of propositions.

Notational conventions. Let C be a set of connectives, then we write $\mathcal{L}+C$ for the smallest superset of $\mathcal{L}$ closed under the connectives in C. $\mathcal{L} * C$ denotes the smallest superset of $\mathcal{L}$ closed under the connectives appearing in $\mathcal{L}$ and the connectives in C.

3.1.2 Static and Dynamic Entailment

Entailments are defined as relations between sequences of formulae and single formulae. The former contains the *assumptions* and the latter is the *conclusion* of the entailment. In order to make concise definitions, we also define the static and dynamic meaning of a sequence $\varphi_1, \ldots, \varphi_n$, abbreviated as $\vec{\varphi}$, in a dynamic modal language $\mathcal{L}^*$. Let $M = \langle S, \sqsubseteq, [\![.]\!] \rangle \in \mathcal{M}_\mathcal{L}$, then

$$
[\![\vec{\varphi}]\!]_M = \bigcap_{i=1}^{n} [\![\varphi_i]\!]_M \quad \text{and} \quad [\![\vec{\varphi}]\!]^\circ_M = [\![\varphi_1]\!]^\circ_M \circ \ldots \circ [\![\varphi_n]\!]^\circ_M .^{12} \tag{14}
$$

The former part defines the static meaning of $\vec{\varphi}$, and the latter part defines the o-meaning of $\vec{\varphi}$. The o-meaning of $\vec{\varphi}$ is the relation of input/output pairs of consecutively o-executing (expanding, updating,...) φ_1 through φ_n.
We will subsequently write $[\![\vec{\varphi}]\!]^o_{M,s}$ for the set $\{t \in S \mid \langle s, t\rangle \in [\![\vec{\varphi}]\!]^o_M\}$ and $[\![\vec{\varphi}]\!]^o_{M,T} = \bigcup_{s \in T} [\![\vec{\varphi}]\!]^o_{M,s}$ for all $s \in S$ and $T \subseteq S$. We will write $[\![\vec{\varphi}]\!]^o_{M,\min}$ for the minimal o-meaning of the sequence $\vec{\varphi}$.

Definition 1 Let $\mathcal{M}$ be some class of $\mathcal{L}$-information models, and let $\varphi_1, \ldots, \varphi_n, \psi$ be propositions of some dynamic modal extension $\mathcal{L}^*$ of $\mathcal{L}$. We define the following entailments for discourse $\varphi_1, \ldots, \varphi_n$ $(\vec{\varphi})$:

- $\vec{\varphi}$ *statically entails* ψ with respect to $\mathcal{M}$ if $[\![\vec{\varphi}]\!]_M \subseteq [\![\psi]\!]_M$.
- $\vec{\varphi}$ *dynamically entails* ψ according to the operation o (or $\vec{\varphi}$ o-entails ψ) with respect to $\mathcal{M}$ if $[\![\vec{\varphi}]\!]^o_{M,s} \subseteq [\![\psi]\!]_M$ for all $M \in \mathcal{M}$ and s in M.
- $\vec{\varphi}$ *minimally o-entails* ψ with respect to $\mathcal{M}$ if $[\![\vec{\varphi}]\!]^o_{M,\min} \subseteq [\![\psi]\!]_M$ for all $M \in \mathcal{M}$.

We use $\vec{\varphi} \models_\mathcal{M} \psi$, $\vec{\varphi} \models^o_\mathcal{M} \psi$ and $\vec{\varphi} \models^{\min o}_\mathcal{M} \psi$ as abbreviations for these three entailment relations, respectively.

Note that if the modal operators $[\varphi]^\circ$ are present within the dynamic modal language $\mathcal{L}^*$, then the notion of o-entailment in Definition 1 boils down to the static entailment $\models_\mathcal{M}$ $[\varphi_1]^\circ \ldots [\varphi_n]^\circ \psi$.
When we think of operations as updates as in the following sections, the minimal dynamic meaning of a sequence $\varphi_1, \ldots, \varphi_n$ is the same as updating the minimal states (the initial context) consecutively with φ_1 through φ_n. This interpretation is the one we will use for *the* interpretation of a discourse or text $\vec{\varphi}$. Of course, as will be the case for most pragmatic inferences, the minimal states of an information model should not be states of complete ignorance. In order to draw the defeasible conclusions discussed in the previous section, we need to add some defeasible information. For this purpose we need the following notation. If $\Gamma \subseteq \mathcal{L}^*$, then we write $\mathcal{M}_\Gamma$ for the subclass of models in $\mathcal{M}$ which supports all the formulae in Γ: $\{M \in \mathcal{M} \mid [\![\gamma]\!]_M = S \text{ for all } \gamma \in \Gamma\}$. The entailment $\vec{\varphi} \models^{\min+\mu}_{M_\Gamma} \psi$ covers the interpretation of a discourse $\vec{\varphi}$ in the context or background knowledge of Γ.

12. The operation $\circ$ stands for relational composition. For two relations $R_1, R_2 \subseteq S^2$: $R_1 \circ R_2 = \{\langle s, t\rangle \in S^2 \mid \exists u \in S : R_1(s, u)\ \&\ R_2(u, t)\}$.

3.2 Simple Preferential Extensions

Shoham [1988] introduced preferential reasoning into nonmonotonic logics. The central idea is to add a preferential structure over the models of the logic chosen as the inference mechanism. This preferential structure is most often some partial or pre–order. A nonmonotonic inference, $\varphi_1, \ldots, \varphi_n \mid\!\sim \psi$, then says that ψ holds in all the maximally preferred $\vec{\varphi}$-models. In many nonmonotonic formalisms such as Reiter's [1980] default logic, an additional preferential structure of an assumption set $\vec{\varphi}$ is specified by explicit *default assumptions* Δ, which are defeasible. The central idea is to use as much information from Δ[13] as possible as long as it is consistent with the strict assumptions Φ. We will also encode this maximality preference in our definition. In this paper, we use a preferential operator p to specify the additional defeasible information. A proposition of the form $\mathsf{p}\,\varphi$ refers to the maximally preferred φ-states.

3.2.1 Single Preference Classes

Preferential reasoning can be accommodated within the DML framework by assigning an additional preferential structure to the space of information states. There are essentially two ways to do this. In one method, the preferential structure is added to the static structure over information states ($[\![.]\!]$), and in the other method, it is added to the dynamic structure on these states ($\sqsubseteq$). We take the first simpler option in this paper.[14]
As explained in 2.2.4, the preferential reasoning for anaphoric resolution needs to take different *preference classes* into account. In 3.3, we will give DML–style definitions for such structures, which will be a straightforward generalization of the following definition of a single preference class.

Definition 2 A *single preferential extension* $\mathcal{L}_\mathsf{p}$ of the static language $\mathcal{L}$ is the smallest superset of $\mathcal{L}$ such that $\mathsf{p}\,\varphi \in \mathcal{L}_\mathsf{p}$ for all $\varphi \in \mathcal{L}$.
A *preferential* $\mathcal{L}$-model is an information $\mathcal{L}_\mathsf{p}$-model $M = \langle S, \sqsubseteq, [\![.]\!]\rangle$, with $[\![.]\!]$ representing a pair of interpretation functions $\langle {}^0[\![.]\!], {}^1[\![.]\!]\rangle$ such that $M_0 = \langle S, \sqsubseteq, {}^0[\![.]\!]\rangle$ and $M_1 = \langle S, \sqsubseteq, {}^1[\![.]\!]\rangle$ are $\mathcal{L}$-information models, and $[\![\varphi]\!] = {}^0[\![\varphi]\!]$ and $[\![\mathsf{p}\,\varphi]\!] = {}^1[\![\varphi]\!]$ for all $\varphi \in \mathcal{L}$. If $\mathcal{M}$ is a class of $\mathcal{L}$-information models, then the class of all preferential $\mathcal{L}$ models whose nonpreferential part (0) is a member of $\mathcal{M}$ is called the single–preferential enrichment of $\mathcal{M}$.
If $\mathcal{L}^* = \mathcal{L} +(*)\ C$, then $\mathcal{L}^*_\mathsf{p}$ refers to the language $\mathcal{L}_\mathsf{p} +(*)\ C$.

The interpretation function $[\![.]\!]$ consists of an *indefeasible part* ${}^0[\![.]\!]$ and a *defeasible part* ${}^1[\![.]\!]$. Both functions are interpretation functions of the static language: ${}^{0,1}[\![.]\!] : \mathcal{L} \longrightarrow \mathbf{2}^S$. The indefeasible part replaces the ordinary interpretation function, while the additional defeasible part is the 'pragmatic' strengthening of this standard reading. Note that a preferential extension gives us a set of preferred states, allowing both determinate and indeterminate interpretations.

3.2.2 Dynamic Preferential Meaning and Preferential Entailment

(15) below defines the static and dynamic *preferential* meaning of a sentence φ analogous to the nonpreferential definitions presented in 3.1.1. The *static preferential meaning* of a sentence φ (in a model M) is written as $\langle\!\langle\varphi\rangle\!\rangle_M$, and the 'dynamic' *preferential meaning of*

13. In Reiter's default logic 'as much information from Δ' means 'as many (classical) logical conclusions from Δ'.
14. The latter, more complex, option would be a more balanced combination of dynamic and preferential reasoning because the preferential structure is represented at the same level of information order over which dynamicity is defined. From this perspective, the preferential structuring of models of a given logic that supplies a nonmonotonic component is analogous to dynamifying a logic by informational structuring as described by Jaspars and Krahmer [1996]. Such investigations are left for a future study.

φ with respect to a given information state (context) s in a model M is written as $\langle\!\langle\varphi\rangle\!\rangle_{M,s}$.

$$(15)\qquad \begin{array}{rcl} \langle\!\langle\varphi\rangle\!\rangle_M & = & [\![\mathsf{p}\,\varphi]\!](= {}^1[\![\varphi]\!]) \\ \langle\!\langle\varphi\rangle\!\rangle_{M,s} & = & \begin{cases} [\![\mathsf{p}\,\varphi]\!]^\circ_{M,s} & \text{if } [\![\mathsf{p}\,\varphi]\!]^\circ_{M,s} \neq \emptyset \\ [\![\varphi]\!]^\circ_{M,s} & \text{otherwise.} \end{cases} \end{array}$$

In line with the definitions of 3.1.1, we write $\langle\!\langle\varphi\rangle\!\rangle^\circ_M$ for the relational abstraction of $\langle\!\langle\varphi\rangle\!\rangle^\circ_{M,s}$. Our definition of the preferential dynamic meaning of a discourse $\varphi_1, \ldots, \varphi_n = \vec{\varphi}$ is written as $\langle\!\langle\vec{\varphi}\rangle\!\rangle^\circ_{M,s}$, and it deviates from the definition of $[\![\vec{\varphi}]\!]^\circ_{M,s}$ above because a simple relational composition of the preferential dynamic readings of single sentences does not give us a satisfactory definition. The failure of normal composition in this respect can be illustrated by the following simple abstract example. Suppose $\vec{\varphi} = \varphi_1, \varphi_2$ is a two–sentence discourse with

$$(16)\qquad [\![\mathsf{p}\,\varphi_1]\!]^{+\mu}_{M,0} = \{1,2\},\ [\![\varphi_2]\!]^{+\mu}_{M,1} = \{3\},\ [\![\mathsf{p}\,\varphi_2]\!]^{+\mu}_{M,1} = \emptyset \text{ and } [\![\mathsf{p}\,\varphi_2]\!]^{+\mu}_{M,2} = \{4\}.$$

We obtain both $\langle 0,3\rangle, \langle 0,4\rangle \in \langle\!\langle\varphi_1\rangle\!\rangle_M \circ \langle\!\langle\varphi_2\rangle\!\rangle_M$. The second pair ($\langle 0,4\rangle$) is composed of maximally preferred readings while the first pair ($\langle 0,3\rangle$) is not. Because these two pairs are both equal members of the composition, such a definition of the preferential meaning of a discourse is not satisfactory.
The two–sentence discourse in this example has four possible readings: (1) composing the two defeasible/preferential readings, (2) composing the indefeasible reading of one sentence and the defeasible reading of the other sentence in two possible orders, and (3) composing the two indefeasible readings. As we said earlier, it is reasonable to use as much preference as possible, which means that (1) should be the "best" composition, the two possibilities in (2) should be the next best, and (3) should be the "worst". We will encode this preferential ordering based on the amount of preferences into the entailment definition. What about then the two possible ways of mixing indefeasible and defeasible readings of the two sentences in the case of (2)? A purely amount–based comparison would not differentiate them. Are they equally preferred?
In addition to the sensitivity to the amount of overall preferences, we hypothesize that discourse's linear progression factor also gives rise to a preferential ordering. We thus distinguish between the two compositions of indefeasible and defeasible readings in (2), and assign a higher preference to the composition in which the first sentence has the defeasible/preferential reading rather than the indefeasible reading. The underlying intuition is that the defeasibility of information is inversely proportional to the flow of time. It is harder to defeat conclusions drawn earlier in the given discourse. This has to do with the fading of nonsemantic memory with time. Earlier (semantic) conclusions tend to persist while explicit sentence forms fade away as discourse continues.
We thus take the preferential context-sensitive reading of a discourse $\vec{\varphi} = \varphi_1, \ldots, \varphi_n$ to be the interpretation that results from applying preferential rules as *much* as possible and as *early* as possible. This type of interpretation can be defined on the basis of an induction on the length of discourses:

$$(17)\qquad \begin{array}{rcll} {}^{2k}[\![\vec{\varphi}]\!]^\circ_{M,s} & = & [\![\varphi_n]\!]_{M,T} & \text{and} \\ {}^{2k+1}[\![\vec{\varphi}]\!]^\circ_{M,s} & = & [\![\mathsf{p}\,\varphi_n]\!]_{M,T} & \text{with } T = {}^k[\![\varphi_1, \ldots, \varphi_{n-1}]\!]^\circ_{M,s}{}^{15}. \end{array}$$

Note that $k < 2^{n-1}$ in this inductive definition. ${}^0[\![\varphi_1]\!]_{M,s}$ and ${}^1[\![\varphi_1]\!]_{M,s}$ are given by the $\mathcal{L}$-information model M. The set of states ${}^k[\![\vec{\varphi}]\!]^\circ_{M,s}$ is called the *o*-meaning of $\vec{\varphi}$ of *priority* k with respect to s in M. In this way, we obtain 2^n readings of a given discourse. The *preferential o-meaning* of a discourse $\vec{\varphi}$ (w.r.t. s in M) is then the same as the nonempty interpretation of the highest priority larger than 0, and if all these readings are empty, then the preferential *o*-meaning coincides with the completely indefeasible reading of priority 0.

$$(18)\qquad \langle\!\langle\vec{\varphi}\rangle\!\rangle^\circ_{M,s} \;=\; {}^k[\![\vec{\varphi}]\!]^\circ_{M,s} \quad \text{with } k = \max\left(\{i \mid {}^i[\![\vec{\varphi}]\!]^\circ_{M,s} \neq \emptyset, 0 < i < 2^n\} \cup \{0\}\right).$$

Note that application of this definition to example (16) yields $\langle\!\langle \varphi_1, \varphi_2 \rangle\!\rangle^{+\mu}_{M,0} = \{4\}$. Definition (18) leads to the following succinct definition of *preferential dynamic entailment*:

(19) $$\varphi_1, \ldots, \varphi_n \mid\!\sim^{\circ}_{\mathcal{M}} \psi \quad \Leftrightarrow \quad \langle\!\langle \varphi_1, \ldots, \varphi_n \rangle\!\rangle^{\circ}_{M,s} \subseteq [\![\psi]\!]_M \text{ for all } s \text{ in } M, \text{ for all } M \in \mathcal{M}.$$

This definition says that for every input state of a discourse $\vec{\varphi}$, the maximally preferred readings of the discourse always lead to ψ-states. We write $\vec{\varphi} \mid\!\sim^{\min\circ}_{\mathcal{M}} \psi$ whenever $\langle\!\langle \vec{\varphi} \rangle\!\rangle^{\circ}_{M\,\min} \subseteq [\![\psi]\!]_M$ for all $M \in \mathcal{M}$ (minimal preferential dynamic entailment).

3.3 Multiple Preference Classes

Now we turn to information models of multiple preference classes needed for formalizing the preference interaction in pronoun resolution, as motivated in section 2. If we assume a linear priority order on these preference classes, then it is not hard to generalize Definition 2 of a single preference class given in section 3.2.1. We will assume such determinate overriding relations among preference classes here.[16]

Definition 3 A multiple (m) preferential extension $\mathcal{L}_{\mathsf{p},m}$ of $\mathcal{L}$ is the smallest superset of $\mathcal{L}$ such that $\mathsf{p}_i\,\varphi \in \mathcal{L}_{\mathsf{p},m}$ for all $\varphi \in \mathcal{L}$.
A multiple (m) preferential $\mathcal{L}$-model is a $\mathcal{L}_{\mathsf{p},m}$-information model $\langle S, \sqsubseteq, [\![.]\!]\rangle$ such that $[\![.]\!] = \langle {}^0[\![.]\!], \ldots, {}^m[\![.]\!]\rangle$ with $M_i = \langle S, \sqsubseteq, {}^i[\![.]\!]\rangle \in \mathcal{M}_{\mathcal{L}}$ for all $i \in \{0, \ldots, m\}$, and $[\![\varphi]\!] = {}^0[\![\varphi]\!]$ and $[\![\mathsf{p}_i\,\varphi]\!] = {}^i[\![\varphi]\!]$ for all $\varphi \in \mathcal{L}$ and $i \in \{1, \ldots, m\}$.
The class of *m-preferential enrichments* of a class of $\mathcal{L}$ information models $\mathcal{M}$ is the class of all preferential $\mathcal{L}$ models whose indefeasible part (0) is a member of $\mathcal{M}$.

Intuitively, $\mathsf{p}_i\,\varphi$ says that the current state is a preferred state according to the i–th preference class and the content φ. We use a simple generalization of the preferential dynamic meaning given in the previous section for the singular preference setting. For a given discourse $\vec{\varphi} = \varphi_1, \ldots, \varphi_n$ we define $(m+1)n$ readings and define their associated priority in the same manner as in (17). Let $k < (m+1)^{n-1}$ and $T = {}^k[\![\varphi_1, \ldots, \varphi_{n-1}]\!]$, then ${}^{(m+1)k}[\![\vec{\varphi}]\!] = [\![\varphi_n]\!]^{\circ}_{M,T}$ and ${}^{(m+1)k+i}[\![\vec{\varphi}]\!] = [\![\mathsf{p}_i\,\varphi_n]\!]^{\circ}_{M,T}$ for all $i \in \{1, \ldots, m\}$. The preferential o-meaning of $\vec{\varphi}$ with respect to a state s in an information model M, $\langle\!\langle \vec{\varphi} \rangle\!\rangle^{\circ}_{M,s}$, is defined in the same way as for the single preferential case (18): replace 2 with $m+1$.

3.4 Pragmatic Meta–constraints

For most applications, however, this definition is far too general, and we need to regulate the interplay of indefeasible and defeasible interpretations with additional constraints. We discuss some candidates here. Let $M = \langle S, \sqsubseteq, \langle {}^0[\![.]\!], {}^1[\![.]\!]\rangle\rangle$ be a preferential $\mathcal{L}$-model.

Principle 1 (*Realism*) Every preferential φ-state, or $\mathsf{p}\,\varphi$-state, is a φ-state itself:[17]

$$ {}^1[\![\varphi]\!] \subseteq {}^0[\![\varphi]\!].$$

This principle is perhaps too strict. In some types of defeasible reasoning, we would like to assign preferential meanings to meaningless or ill–formed input, which would give us the robustness to recover from errors. Such robustness can be expressed in terms of a restriction to nonempty indefeasible readings as follows: ${}^0[\![\varphi]\!] \neq \emptyset$ (Robust Realism).

16. Kameyama [1996] points out that this is not always the case, but in most cases, strict linearity can be enforced through 'uniting' multiple preference classes of an equal strength into a single one: $[\![(\mathsf{p} \cup \mathsf{p}')\varphi]\!] = [\![\mathsf{p}\,\varphi]\!] \cup [\![\mathsf{p}'\,\varphi]\!]$.
17. Compare the 'realism' principle of [Cohen and Levesque, 1990]: all intended or goal worlds of a rational agent should be epistemically possible. This constraint is often used to distinguish between an agent's desires and intentions.

Principle 2 (*Minimal Preference*) In minimal information states, if a proposition has an indefeasible reading, it should also have a preferential reading:

$$[\![\varphi]\!]^{\circ}_{M,\min} \neq \emptyset \Rightarrow [\![\mathsf{p}\,\varphi]\!]^{\circ}_{M,\min} \neq \emptyset.$$

The intuition here is that in a minimal state there should be no obstacles that prevent the interpreter from using his preferential expectations or prejudices. In the next section, we will discuss some variants of this principle, which are required to account for certain anaphora resolution preferences.

Principle 3 (*Preservation of Equivalence*) Two propositions with the same indefeasible content should also have the same defeasible content:

$$ {}^0[\![\varphi]\!] = {}^0[\![\psi]\!] \Rightarrow {}^1[\![\varphi]\!] = {}^1[\![\psi]\!].$$

This principle is not always desirable.[18] For example, in discourses (1) and (2) above, *John met Bill* and *Bill met John* have the same semantic/indefeasible content, but different pragmatic/defeasible readings.

Principle 4 (*Complete Determinacy*) Every preferential φ-extension of a given information state s has at most one maximal element.

$$\#({}^1[\![\varphi]\!]^{+\mu}_{M,s}) \leq 1 \text{ for all } s.$$

This excludes indeterminacy described in 2.2.2, prohibiting Nixon Diamonds. Intuitively, it says that pragmatics always enforces certainty. In other words, in cases of semantic uncertainty, pragmatics always enforces a single choice. For example, discourse (3) should always lead to a single pragmatic solution. Therefore, as argued earlier, this constraint is also unrealistic.

4 Towards a Preferential Discourse Logic

In this section, we will discuss two different instances of preferential extensions of the DML–setting of the previous section. As we have seen, such an instantiation requires a specification of static and dynamic modal languages and a class of information models. In subsection 4.1, we will discuss a simple propositional logic, and explain how simple defeasible (preferential) propositional entailments can or cannot be drawn from a set of preferential rules. Our examples will illustrate the defeasible inference patterns commonly called the Penguin Principle and the Nixon Diamond. In subsection 4.2, we will define a much richer dynamic semantics that integrates the defeasible propositional inferences explained in 4.1 into anaphora resolution preferences. Such a combination is needed to account for the preferential effects on anaphora resolution. In subsection 4.3, we define first–order variants of pragmatic meta–constraints. In subsection 4.4, we will illustrate the first–order preferential discourse logic with discourse examples with ambiguous pronouns discussed in section 2.

4.1 A Simple Propositional Preferential Dynamic Logic

Table 4 gives a DML–specification of a simple dynamic propositional logic. The single preferential extension of this logic illustrates how preferential entailments are established according to the definitions given in the previous section. The information states of this model are

18. In nonmonotonic logics this principle is often used. It implies, for example, the dominance of the default conclusions from more specific information. If *penguin*∧*bird* is equivalent to *penguin*, then Principle 3 makes all the preferential information based on *penguin* applicable while the preferential information based on *bird* may be invalid for *penguin* ∧ *bird*.

Static Language ($\mathcal{L}$):	A set of *literals*: $I\!P \cup \{\neg p \mid p \in I\!P\}$
Dynamic Language ($\mathcal{L}^*$):	$\mathcal{L} + \{[.]^{+\mu}, \langle . \rangle^{+\mu}\}$
States (S):	arbitrary non-empty set.
Order ($\sqsubseteq$):	arbitrary pre-order over S.
Interpretation ($[\![.]\!]$):	A function $\mathcal{L} \mapsto \wp(S)$ such that (i) $\forall \varphi \in \mathcal{L}, s, t \in S : s \sqsubseteq t, s \in [\![\varphi]\!] \Rightarrow t \in [\![\varphi]\!]$. (ii) $\forall p \in I\!P : [\![p]\!] \cap [\![\neg p]\!] = \emptyset$. (iii) $\forall \varphi \in \mathcal{L} : [\![\varphi]\!] \cap \min_M S = \emptyset$.

Table 4: A Class of Propositional Information Models

partial truth value assignments for the propositional atoms: an atom is either true, false or undefined. The information order is arbitrary, while the interpretation function is (i) *monotonic*, that is, expansions contain more atomic information and (ii) *coherent*, that is, expansions contain no contradictory information, and furthermore, there is a constraint that (iii) the minimal states have empty atomic content.
Let $\mathcal{M}$ be the class of all single–preferential enrichments of this class of information models subject to both Principle 1 (Realism) and Principle 2 (Minimal Preference) defined in the previous section. Let $\{bird, penguin, can\text{–}fly\} \subseteq I\!P$, and let Γ be the following set of $\mathcal{L}^*_{\mathsf{p}}$-formulae:

(20) $\{[\mathsf{p}\ bird]^{+\mu}\ can\text{–}fly, [\mathsf{p}\ bird]^{+\mu}\ \neg penguin, [\mathsf{p}\ penguin]^{+\mu}\ \neg can\text{–}fly, [penguin]^{+\mu}\ bird\}$[19]

then:

(21) $$bird \mid\!\approx^{\min+\mu}_{\mathcal{M}_\Gamma} can\text{–}fly \quad \text{and} \quad bird, penguin \mid\!\approx^{\min+\mu}_{\mathcal{M}_\Gamma} \neg can\text{–}fly$$

This entailment is validated by the following derivation for all models $N \in \mathcal{M}_\Gamma$:

$$\begin{aligned} \langle\!\langle bird \rangle\!\rangle^{+\mu}_{N,\min} &= {}^1[\![bird]\!]^{+\mu}_{N,\min} \text{ and} \\ \langle\!\langle bird, penguin \rangle\!\rangle^{+\mu}_{N,\min} &= {}^1[\![bird, penguin]\!]^{+\mu}_{N,\min} = \\ [\![bird, \mathsf{p}\ penguin]\!]^{+\mu}_{N,\min} &= [\![\mathsf{p}\ penguin]\!]^{+\mu}_{N,\min} \subseteq [\![\neg can\text{–}fly]\!]_N \end{aligned}$$

By definition of the entailment $\mid\!\approx^{\min+\mu}_{\mathcal{M}_\Gamma}$, we obtain the results of (21).
Next, suppose that $\{republican, pacifist, quaker\} \subseteq I\!P$, and

(22) $$\Delta = \{[\mathsf{p}\ quaker]^{+\mu}\ pacifist, [\mathsf{p}\ republican]^{+\mu}\ \neg pacifist\}.$$

Here, the preferential readings of *quaker* and *republican* contradict each other. One may expect that we get $quaker, republican \mid\!\approx^{\min+\mu}_{\mathcal{M}_\Delta} pacifist$, because the preferences of the last sentence is taken to be weaker in the definition (18). This is not the case, however, because it is possible that a *republican* cannot be a normal *quaker* ($[republican]^{+\mu}\,[\mathsf{p}\ quaker]^{+\mu}\,\bot$) or vice versa ($[quaker]^{+\mu}\,[\mathsf{p}\ republican]^{+\mu}\,\bot$).
If such preferential blocks are removed, we obtain:

(23) $$\begin{aligned} quaker, republican &\mid\!\approx^{\min+\mu}_{\mathcal{M}_{\Delta'}} pacifist \quad \text{and} \\ republican, quaker &\mid\!\approx^{\min+\mu}_{\mathcal{M}_{\Delta'}} \neg pacifist \ , \end{aligned}$$

with Δ' denoting the set:

(24) $$\Delta \cup \{[quaker]^{+\mu}\,\langle \mathsf{p}\ republican\rangle^{+\mu}\,\top, [republican]^{+\mu}\,\langle \mathsf{p}\ quaker\rangle^{+\mu}\,\top\}.$$[20]

19. The set Γ prescribes that "normal birds can fly", "normal birds are not penguins", "normal penguins cannot fly" and that "penguins are birds".

Let $\mathcal{N}$ be the class of double–preferential enrichments of the model given in Table 4 subject to the realism and the minimal preferential constraints on both classes. Let Δ'' be the set

(25) $$\begin{aligned} &\{[\mathsf{p}_1\ quaker]^{+\mu}\ pacifist, [\mathsf{p}_2\ quaker]^{+\mu}\ quaker, [\mathsf{p}_2\ republican]^{+\mu}\ \neg pacifist\} \quad \bigcup \\ &\{[quaker]^{+\mu}\,\langle \mathsf{p}_i\ republican\rangle^{+\mu}\,\top, [republican]^{+\mu}\,\langle \mathsf{p}_i\ quaker\rangle^{+\mu}\,\top \mid i = 1, 2\} \quad . \end{aligned}$$

The second rule says that the p_2–reading of *quaker* does not entail any information in addition to its indefeasible reading. In this setting, the two discourses in (23) yield the same conclusion as follows:

(26) $$\begin{aligned} quaker, republican &\mid\!\approx^{\min+\mu}_{\mathcal{M}_{\Delta''}} \neg pacifist \quad \text{and} \\ republican, quaker &\mid\!\approx^{\min+\mu}_{\mathcal{M}_{\Delta''}} \neg pacifist \quad . \end{aligned}$$

4.2 A First–order Preferential Dynamic Semantics

We will now come to an analysis of the discourses with ambiguous pronouns discussed in section 2. Typical dynamic semantic analyses of discourse, such as the relational semantics for dynamic predicate logic [Groenendijk and Stokhof, 1991] or first–order DRT such as presented, for example, in [Muskens *et al.*, 1996],[21] do not yield a satisfactory preferential dynamic semantics when we integrate them with the preferential machinery of the previous section. In these types of semantic theories, dynamicity is restricted to the value assignment of variables for interpretation of possible anaphoric links, but we need a logic that supports a preferential interplay of variable assignments, predicates, names and propositions in order to account for anaphora resolution. In the terminology of Jaspars and Krahmer [1996], we need to 'dynamify' more parameters of first–order logic than just the variable assignments.[22]
In order to arrive at such extended dynamics over first–order models, we will establish a combination of the 'ordinary' dynamics–over–assignments semantics with the models of information growth used in possible world semantics for classes of constructive logics.[23]
Let us first present the class of our information models. The basic linguistic ingredients are the same as for first–order logic: Con a set of constants, Var a disjoint countably infinite set of variables, and for each natural number n a set of n-ary predicates Pred^n. The static language is the same as for first–order logic except for quantifiers and negation. The dynamic language supplies the formalism with dynamic modal operators $[.]^{+\mu}$ and $\langle . \rangle^{+\mu}$:

(27) $$\begin{aligned} \mathsf{Atoms} &= \{Pt_1 \ldots t_n \mid P \in \mathsf{Pred}^n, t_i \in \mathsf{Con} \cup \mathsf{Var}\} \\ &\quad \cup \{t_1 = t_2 \mid t_i \in \mathsf{Con} \cup \mathsf{Var}\} \\ \mathcal{L} &= \mathsf{Atoms} + \{\wedge, \vee, \bot\} \\ \mathcal{L}^* &= \mathcal{L} * \{[.]^{+\mu}, \langle . \rangle^{+\mu}\} \end{aligned}$$

Table 5 presents the intended $\mathcal{L}$-information models. The growth of the information order $\sqsubseteq$ is subject to three constraints. The first one (i) says that all the parameters of first–order logic, that is, the domains, interpretation of predicates and constants, and the variable assignments, grow with the information order. The other two constraints seem unorthodox. The second constraint (ii) ensures the freedom of variables in this setting. It tells us that in each state the range of a 'fresh' variable is unlimited, that is, it may have the value of each current or 'future' individual. This means that for every individual d in an extension t every variable x which has not been assigned a value yet, may be assigned the value d

20. Take $\top = [p]^{+\mu}\,p$.
21. For a discussion of the DML–specification of this semantics for DRT, see [Jaspars and Krahmer, 1996]. On the basis of these specifications one can transfer the definitions of preferential dynamic entailment of this paper to a range of dynamic semantics.
22. See also [Van Benthem and Cepparello, 1994] for discussion of such further dynamification. For an existing proposal for a semantic theory that combines 'propositional' and 'variable' dynamics, see [Groenendijk *et al.*, 1994]. In this paper the authors introduce a dynamic semantics over assignment–world pairs. It may be possible to obtain a suitable preferential extension of this type of semantics for our purposes as well.
23. E.g., see [Troelstra and Van Dalen, 1988] or [Fitting, 1969] for the case of intuitionistic logic.

States (S):	A collection of quadruples $s = \langle D^s, I_p^s, I_c^s, I_v^s \rangle$ with D^s a non-empty set of *individuals*, $I_p^s : \mathsf{Pred}^n \longrightarrow \wp((D^s)^n)$ the *local interpretation of predicates*, $I_c^s : \mathsf{Con} \rightharpoonup D$ a *partial local interpretation of constants*, and $I_v^s : \mathsf{Var} \rightharpoonup D$ a *partial variable assignment*.
Order ($\sqsubseteq$):	A pre-order over S such that (*i*) For all $s, t \in S$ if $s \sqsubseteq t$ then $D^s \subseteq D^t$, $I_p^s(P) \subseteq I_p^t(P)$ for all predicates P, $I_c^s(\mathsf{c}) = I_c^t(\mathsf{c})$ for all $\mathsf{c} \in \mathsf{Dom}(I_c^s)$ and $I_v^s(x) = I_v^t(x)$ for all $x \in \mathsf{Dom}(I_v^s)$. (*ii*) For all $s, t \in S$ if $s \sqsubseteq t$, $d \in D^t$ and $x \in \mathsf{Var} \setminus \mathsf{Dom}(I_v^s)$, then there exists $u \in S$ such that $s \sqsubseteq u$ and $D^t = D^u$, $I_p^t = I_p^u$, $I_c^t = I_c^u$, $\mathsf{Dom}(I_v^u) = \mathsf{Dom}(I_v^s) \cup \{x\}$ and $I_v^u(x) = d$. (*iii*) For all $s \in \min_M S$: $I_p^s(P) = \emptyset$ for all predicates P and $\mathsf{Dom}(I_c^s) = \mathsf{Dom}(I_v^s) = \emptyset$.
Interpretation ($[\![.]\!]$):	$[\![Pt_1 \dots t_n]\!] = \{s \in S \mid \langle I_t^s(t_1), \dots, I_t^s(t_n) \rangle \in I_p^s(P)\}$, $[\![t_1 = t_2]\!] = \{s \in S \mid I_t^s(t_1) = I_t^s(t_2)\}$, $[\![\varphi \wedge \psi]\!] = [\![\varphi]\!] \cap [\![\psi]\!]$, $[\![\varphi \vee \psi]\!] = [\![\varphi]\!] \cup [\![\psi]\!]$, $[\![\bot]\!] = \emptyset$.

Table 5: A Class of First–order Information Models

in a state which contains the same information as t. This constraint differentiates the roles of constants and variables in this setting. The last constraint (*iii*) says that the minimal information states do not contain atomic information. It was also used for propositional information models in 4.1.
The interpretation function is more or less standard. Verification of an atomic sentence requires determination of all the present terms, also for identities.
Quantification can be defined by means of the dynamic modal operators. For example, (28) means that the *Meet*-relation is symmetric and *Greet*-relation is irreflexive.

(28) $\qquad [\mathit{Meet}\, xy]^{+\mu}\ \mathit{Meet}\, yx$ and $[\mathit{Greet}\, xy]^{+\mu}[x = y]^{+\mu} \bot$

Ordinary universal quantification can be defined by using identity and update modality: $\forall x \varphi = [x = x]^{+\mu} \varphi$.[24] Negation can also be defined by means of a dynamic modal operator: $\neg\varphi = [\varphi]^{+\mu} \bot$.[25] A typical (singular) preferential sentence would be:

(29) $\qquad [\mathsf{p}\ \mathit{Meet}\, xy]^{+\mu}[\mathsf{p}\ \mathit{Greet}\, uv]^{+\mu}(u = x \wedge v = y)$,

which means that the concatenation of the preferential reading of a *Meet*ing and a *Greet*ing pair makes the variables match according to the grammatical parallelism preference.[26]

4.3 First–order Constraints for Preferential Dynamic Reasoning

In order to model the preferential effects on ambiguous pronouns discussed in section 2, we need to postulate several first–order variants of the pragmatic meta–constraints discussed in subsection 3.4. The first–order expressivity of the languages $\mathcal{L}$ and $\mathcal{L}^*$ given in (27) and the fine–structure of the information models presented in Table 5 enable us to calibrate these meta–constraints for preferential interpretation on first–order discourse representations.
We will adopt only Principle 1 (Realism) in its purely propositional form. Three other constraints that we will impose on preferential interpretation regulate some 'harmless' interplay of preferences and terms. Let $M = \langle S, \sqsubseteq, [\![.]\!] \rangle$ be a preferential $\mathcal{L}$-model with $[\![.]\!] = \langle {}^0[\![.]\!], {}^1[\![.]\!] \rangle$.

24. Note that to get the proper universal reading here, we need to be sure that x is a fresh variable (e.g., in the minimal states).
25. A proper definition of existential quantification seems not feasible, $\langle x = x \rangle^{+\mu} \varphi$ is not persistent. A better candidate is $\neg\forall x \neg\varphi$ which behaves persistent.
26. A general implementation of the parallelism preference would require a second–order scheme.

To begin with, fresh variables have no content, and therefore, we do not allow them to block preferential interpretation. In other words, a proposition that contains only fresh variables as terms always has a preferential $+\mu$-reading whenever it has an indefeasible $+\mu$-meaning. In fact, this is a variant of Principle 2, the principle of minimal preference.

Principle 5 (*Minimal Preference for Fresh Variables*) Let s be an information state in an information model of the type described in Table 5. If $\mathsf{Dom}(I_v^s)$ has an empty intersection with the variables occurring in a given proposition φ, and no constants occur in φ, then

$$[\![\varphi]\!]^{+\mu}_{M,s} \neq \emptyset \Rightarrow [\![\mathsf{p}\,\varphi]\!]^{+\mu}_{M,s} \neq \emptyset.$$

The two other constraints for first–order discourses are obtained by weakening Principle 3 (Preservation of Equivalence). Although this principle itself is too strong, we would like to have some innocent logical transparency of the preferential operator. We thus postulate the following principles 6 and 7:

Principle 6 (*Preservation under Renaming Fresh Variables.*) Preferential readings should be maintained when fresh variables are replaced by other fresh variables:

$$\forall x, y \in \mathsf{Var} \setminus \mathsf{Dom}(I_v^s) : s \in [\![\mathsf{p}\,\varphi]\!]_M \Leftrightarrow s \in [\![\mathsf{p}\,\varphi[x/y]]\!]_M.$$

Principle 7 (*Preservation of Identities.*) Preferential readings should be insensitive to substitutions of equals:

$$\forall t_1, t_2 \in \mathsf{Var} \cup \mathsf{Con} : s \in [\![\mathsf{p}\,\varphi \wedge t_1 = t_2]\!]_M \Leftrightarrow s \in [\![\mathsf{p}\,\varphi[t_1/t_2]]\!]_M.$$

4.4 Preferential Dynamic Disambiguation of Pronouns

We will now account for the discourse examples with ambiguous pronouns discussed in section 2 using the first–order preferential discourse logic defined in this section.

4.4.1 Single–preferential Structure

We will first examine the single–preferential structure of the 'John met Bill' sentences (1)–(4). Assume the single–preferential extensions $\mathcal{M}$ of the models presented in Table 5 subject to Principles 1, 5, 6, and 7. This model, together with the background information Γ containing (28) and (29) above, yields the intended defeasible conclusions as follows:

(30) $\qquad \begin{array}{ll} x = \mathsf{j} \wedge y = \mathsf{b}, \mathit{Meet}\, xy, \mathit{Greet}\, uv & \mid\!\sim^{\min+\mu}_{\mathcal{M}_\Gamma} \ (u = \mathsf{j} \wedge v = \mathsf{b}) \\ x = \mathsf{b} \wedge y = \mathsf{j}, \mathit{Meet}\, xy, \mathit{Greet}\, uv & \mid\!\sim^{\min+\mu}_{\mathcal{M}_\Gamma} \ (u = \mathsf{b} \wedge v = \mathsf{j}) \end{array}$

This class also entails the invalidity of this kind of a determinate resolution for the 'John and Bill met'-case (3):

(31) $\qquad x = \mathsf{j} \wedge y = \mathsf{b}, \mathit{Meet}\, xy \wedge \mathit{Meet}\, yx, \mathit{Greet}\, uv \ \not\mid\!\sim^{\min+\mu}_{\mathcal{M}_\Gamma} \ (u = \mathsf{j} \wedge v = \mathsf{b})$

The underlying reason is that the preferential meaning of $\mathit{Meet}\, xy \wedge \mathit{Meet}\, yx$ may be different form that of $\mathit{Meet}\, xy$ or $\mathit{Meet}\, yx$ though these three sentences all have the same indefeasible meaning in $\mathcal{M}_\Gamma$.
For discourse (1) extended with the sentence *John greeted back* in (4), the defeasible conclusion of the first discourse in (30) will be invalid over $\mathcal{M}_\Gamma$:

(32) $\qquad x = \mathsf{j} \wedge y = \mathsf{b}, \mathit{Meet}\, xy, \mathit{Greet}\, uv, \mathit{Greet}\, xu \ \not\mid\!\sim^{\min+\mu}_{\mathcal{M}_\Gamma} \ (u = \mathsf{j} \wedge v = \mathsf{b})$

The reason is that for every model $M \in \mathcal{M}_\Gamma$:

(33) $\qquad \forall s \in S : s \in \langle\!\langle x = \mathsf{j} \wedge y = \mathsf{b}, \mathit{Meet}\, xy, \mathit{Greet}\, uv \rangle\!\rangle^{+\mu}_{M,\min} \Rightarrow [\![\mathit{Greet}\, xu]\!]^{+\mu}_{M,s} = \emptyset.$

4.4.2 Double–preferential Structure

We will now illustrate how the overriding effects of commonsense preferences illustrated in (8) and (9) come about in a double–preferential extension of the DML–setting in Table 5. In these cases, we hypothesized that the commonsense preferences about hitting / injuring / breaking override the syntactic preferences underlying the 'John met Bill' examples (1)–(4). We postulate the following double–preferential background for the 'hitting' scene:

(34) $$[\mathsf{p}_1\ \mathit{Hit}\, xy]^{+\mu}\,[\mathsf{p}_1\ \mathit{Injured}\, v]^{+\mu}\, v = x \quad \text{and}$$
$$[\mathsf{p}_2\ \mathit{Hit}\, xy]^{+\mu}\,[\mathit{Injured}\, v]^{+\mu}\, v = y \quad .$$

The p_2-class is associated with commonsense preferences with a higher preferential rank while the p_1-class is associated with 'syntactic' preferences with a lower preferential rank. Note that we take the commonsense impact of the word *Hit* so strong that every *Injured* v-continuation — not only the preferred readings of this sentence — leads to the defeasible conclusion that the hit-tee is the one who must be injured.

The above double–preferential account also enables a formal distinction among discourses F (same as (8) involving Bill), G (involving Schwarzenegger), and H (involving the Terminator) in Table 1, whose differences are exhibited in the survey results presented in Table 2.

Let $\mathcal{N}$ be the class of double preferential enrichments of the models of Table 5 satisfying the same principles as $\mathcal{M}$ for both preference classes. When Δ represents the set containing the two preferential update rules given in (34), we obtain a determinate preference for F:

(35) $$x = \mathsf{j} \wedge y = \mathsf{b},\ \mathit{Hit}\, xy,\ \mathit{Injured}\, v \mid\!\approx^{\min+\mu}_{\mathcal{N}_\Delta} v = \mathsf{b}.$$

Let Δ' be the extension of Δ enriched with the following additional commonsense rules, where sch denotes Schwarzenegger:

(36) $$[\mathsf{p}_2\ \mathit{Injured}\, x]^{+\mu}\,[x = \mathsf{sch}]^{+\mu}\,\bot,$$

This rule says that if something is injured, then it is not to be expected that this is Schwarzenegger. We then obtain a case of indeterminacy for G:

(37) $$x = \mathsf{j} \wedge y = \mathsf{sch},\ \mathit{Hit}\, xy,\ \mathit{Injured}\, v \not\mid\!\approx^{\min+\mu}_{\mathcal{N}_{\Delta'}} v = \mathsf{sch} \text{ and}$$
$$x = \mathsf{j} \wedge y = \mathsf{sch},\ \mathit{Hit}\, xy,\ \mathit{Injured}\, v \not\mid\!\approx^{\min+\mu}_{\mathcal{N}_{\Delta'}} v = \mathsf{j}.$$

Let Δ'' be the union of Δ and the following additional rules, where the constant tm denotes the Terminator:

(38) $$[\mathsf{j} = \mathsf{tm}]^{+\mu}\,\bot \text{ and } [\mathit{Injured}\,\mathsf{tm}]^{+\mu}\,\bot$$

The second sentence says that the Terminator cannot be injured. This background information establishes the preferred meaning of H:

(39) $$x = \mathsf{j} \wedge y = \mathsf{tm},\ \mathit{Hit}\, xy,\ \mathit{Injured}\, v \mid\!\approx^{\min+\mu}_{\mathcal{N}_{\Delta''}} v = \mathsf{j}.$$

Substitution of $\Theta = \Delta' \cup \Delta''$ for Δ in (35), for Δ' in (37) and for Δ'' in (39) yields the same conclusions as above. In sum, if Θ was our background knowledge, then the discourse F predicts that Bill is injured, while G yields indeterminacy in its preferential meaning. Discourse H preferentially entails that John is injured.

5 Conclusions and Future Prospects

As a general logical basis for an integrated model of discourse semantics and pragmatics, we have combined dynamics and preferential reasoning in a dynamic modal logic setting. This logical setting encodes the basic discourse pragmatic properties of dynamicity, defeasibility, indeterminacy, and preference classes posited in an earlier linguistic analysis of the

preferential effects on ambiguous pronouns in discourse. It also provides a logical architecture in which to implement possible meta–constraints that regulate the general interplay of defeasible and indefeasible static and dynamic interpretation. We have given a number of such candidate meta–constraints here, and further empirical (perhaps psycholinguistic) investigations are needed for choosing exactly what constraints are needed.

We demonstrated how a general model theory of dynamic logic can be enriched with a preferential structure to result in a relatively simple preferential model theory. We defined the preferential dynamic entailments over given pieces of discourse, which predict that preferential information is used as much as possible and as early as possible to conclude discourse interpretations. That is, earlier defeasible conclusions are harder to defeat than more recent ones. We have also defined a logical machinery for predicting overriding relationships among preferences classes. Overriding takes place when later indefeasible information defeats earlier preferential conclusions, or when a reading corresponding to a preference class of a higher priority becomes empty and a lower preference class takes over. These preference class overrides give rise to conflict resolutions that are not predictable from straightforward applications of the Penguin Principle.

Although our focus is on pronoun resolution preferences in this paper, we hope that our logical machinery is also adequate for characterizing the conflict resolution patterns among various preferences and preference classes relevant to a wider range of linguistic phenomena. The present perspective of preference interactions assumes that preferences belong to different classes, or modules, and there are certain common conflict resolution patterns *within* each class and *across* different classes. Class–internal preference interactions yield either determinate or indeterminate preferences. Class–external preference interactions are dictated by certain pre–existing class–level overriding relations, according to which the conflicts among the respective conclusions coming from each preference class are either resolved (by class–level overrides), ending up with the preferential conclusions of the highest preference class (whether it is determinate or indeterminate), or unresolved, leading to mixed–class preferential ambiguities. We would like to investigate the applicability of this perspective to a wider range of discourse phenomena.[27]

We might also be able to extend the framework to cover on–line sentence processing pragmatics, where the word–by–word or constituent–by–constituent dynamicity affects the meaning of the utterance being interpreted, producing utterance–internal garden path and repair phenomena analogous to discourse–level counterparts discussed in this paper.

The present logical characterization of preferential dynamics may be extended and/or revised in two major areas. One is the application of actions other updates, $+\mu$. For example, discourse–level repairs as in (5) also require reductions, $-$, and/or downdates, $-\mu$. The other is the relational definition of preferences on the basis of an additional structuring of the information order $\sqsubseteq$ instead of the static interpretation function $[\![.]\!]$. Such an alternative definition would enable us to implement 'graded' preferences as in [Delgrande, 1988], that is, every state gets a certain preferential status with respect to a proposition. States were simply divided into preferential or non–preferential with respect to a proposition in the present paper. Graded preferences may be required for fine–tuning and coordinating the overall discourse pragmatics. A question which is related to this topic is whether the use of graded preferences would make the setting of multiple preference classes become superfluous.

References

[Asher and Lascarides, 1995] Asher, N. and Lascarides, A. 1995. Lexical disambiguation in a discourse context. *Journal of Semantics* 12(1):69–108. Special Issue on Lexical Semantics, Part I.

27. It is encouraging that the recent spread of Optimality Theory from phonology [Prince and Smolensky, 1993] to syntax (e.g., MIT Workshop on Optimality in Syntax, 1995) seems to indicate the descriptive adequacy of the fundamental preference interaction scheme, where potentially conflicting defeasible conclusions compete for the "maximal harmony."

[Asher and Morreau, 1991] Asher, N. and Morreau, M. 1991. Commonsense entailment: A modal theory of non-monotonic reasoning. In Van Eijck, J., editor 1991, *Logics in AI / JELIA90*, volume 478 of *Lecture Notes in Artificial Intelligence*. Springer Verlag, Heidelberg. 1–30.

[Cohen and Levesque, 1990] Cohen, P.R. and Levesque, H.J. 1990. Intention is choice with commitment. *Artificial Intelligence Journal* 42:213–261.

[De Rijke, 1992] De Rijke, M. 1992. A system of dynamic modal logic. Technical Report Research Report 92-170, CSLI, Stanford, CA. to appear in the Journal of Philosophical Logic.

[Delgrande, 1988] Delgrande, J. 1988. An approach to default reasoning based on first-order conditional logic. *Artificial Intelligence Journal* 36:63–90.

[Fitting, 1969] Fitting, M. 1969. *Intuitionistic Logic: Model Theory and Forcing*. Studies in Logic and the Foundations of Mathematics. North Holland, Amsterdam.

[Groenendijk and Stokhof, 1991] Groenendijk, J. and Stokhof, M. 1991. Dynamic predicate logic. *Linguistics and Philosophy* 14:39–100.

[Groenendijk *et al.*, 1994] Groenendijk, J.; Stokhof, M.; and Veltman, F. 1994. This might be it. DYANA Deliverable 6852, ESPRIT Basic Research.

[Jaspars and Krahmer, 1996] Jaspars, J. and Krahmer, E. 1996. A programme of modal unification of dynamic theories. In Dekker, P. and Stokhof, M., editors 1996, *Proceedings of the Tenth Amsterdam Colloquium*. ILLC, Amsterdam. 445–464. this volume.

[Kameyama, 1996] Kameyama, M. 1996. Indefeasible semantics and defeasible pragmatics. In Kanazawa, M.; Piñon, C.; and De Swart, H., editors 1996, *Quantifiers, Deduction, and Context*. CSLI, Stanford, CA. 111–138.

[Kehler, 1995] Kehler, A. 1995. *Interpreting Cohesive Forms in the Context of Discourse Inference*. Ph.D. Dissertation, Harvard University, Cambridge, MA. TR-11-95, Center for Research in Computing Technology.

[Lascarides and Asher, 1993] Lascarides, A. and Asher, N. 1993. Temporal interpretation, discourse relations, and commonsense entailment. *Linguistics and Philosophy* 16:437–493.

[Levinson, 1983] Levinson, S. C. 1983. *Pragmatics*. Cambridge Textbooks in Linguistics. Cambridge University Press, Cambridge, U.K.

[McCarthy and Hayes, 1969] McCarthy, J. and Hayes, P. 1969. Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, B. and Michie, D., editors 1969, *Machine Intelligence*, volume 4. Edinburgh University Press, Edinburgh. 463–502.

[Muskens *et al.*, 1996] Muskens, R.; Van Benthem, J.; and Visser, A. 1996. "Dynamics". In Van Benthem, J. and Ter Meulen, A., editors 1996, *Handbook of Logic and Language*. Elsevier Science. To appear.

[Prince and Smolensky, 1993] Prince, A. and Smolensky, P. 1993. Optimality theory: Constraint interaction in generative grammar. Technical Report 2, Center for Cognitive Science, Rutgers University, New Brunswick, NJ.

[Reiter, 1980] Reiter, R. 1980. A logic for default reasoning. *Artificial Intelligence Journal* 13:81–132.

[Shoham, 1988] Shoham, Y. 1988. *Reasoning About Change: Time and Causation From the Standpoint of Artificial Intelligence*. The MIT Press Series in Artificial Intelligence. MIT Press, Cambridge (MA).

[Troelstra and Van Dalen, 1988] Troelstra, A. and Van Dalen, D. 1988. *Constructivism in Mathmatics, Vol. I*. Studies in Logic and the Foundations of Mathematics. North Holland, Amsterdam.

[Van Benthem and Cepparello, 1994] Van Benthem, J. and Cepparello, G. 1994. Tarskian variations; dynamic parameters in classical logic. Technical Report CS-R9419, CWI, Amsterdam.

[Van Benthem, 1991] Van Benthem, J. 1991. *Language in Action*, volume 130 of *Studies in Logic and the Foundations of Mathematics*. North Holland, Amsterdam.

[Veltman, 1991] Veltman, F. 1991. Defaults in update semantics. Technical report, Department of Philosophy, University of Amsterdam. To appear in the Journal of Philosophical Logic.

STRONG COMPOSITIONALITY

László Kálmán
Theoretical Linguistics Programme, Budapest University

0. Introduction

This paper addresses the much-debated issue of the **principle of compositionality**, the exact content, the formulation and the validity of which have been questioned so often. In its weakest form, compositionality requires the interpretation of natural-language utterances to assign meanings in a more or less systematic manner. In Montague's (1970) classical version, it stipulates a perfect parallelism between semantic and syntactic structures. Its validity has been questioned on the basis of the interference of the (utterance-external and utterance-internal) context with interpretation as well as on the basis of the possibility of assigning more or less arbitrary syntactic structures to utterances. Finally, it also has acquired an **intuitive** meaning, which roughly corresponds to **minimizing idiomaticity**.

In this paper, I will argue that the principle of compositionality has several weak points, which make its actual content far weaker than its intuitive interpretation. As a matter of fact, the principle is almost vacuous in its usual formulations. **Section 1** introduces the principle and explains what its weak points are. I conclude that the compositionality principle must be strengthened if it is to act as a more powerful constraint on interpretation.

Section 2 is the core of the paper. I propose two additional constraints on the interpretation of natural-language expressions: The principle of **independence** says that the meanings assigned to sub-expressions in an expression must not depend on each other's **shapes** (**Section 2.1**); and the principle of **additivity** prohibits operations combining meanings from **destructively** modifying any previously assigned meanings (**Section 2.2**).

Section 3 intends to draw some consequences of strong compositionality for the semantics of natural languages. In particular, in **Section 3.1** I will propose to abandon the classical functor/operand view of composing meaning; I argue that the mathematical metaphor of 'incomplete' expressions seen as **functors** must be abandoned in terms of the principle of additivity. Instead, I will propose to abandon the concept of 'semantic incompleteness' altogether, so that the combination of meanings yields just more complete meanings from less complete ones. **Section 3.2** sketches a model-theoretic interpretation mechanism suitable for this purpose. Finally, in **Section 3.3** I speculate about the nature of the expression-internal interactions of meanings. I argue that such interactions are not excluded by the principle of independence (which only bans interactions in terms of formal properties), nor by the principle of additivity (inasmuch as the interactions are not destructive). In particular, I argue that the 'systematic ambiguities' like that of the word *coffee* (meaning 'coffee seeds', 'coffee drink', 'the act of drinking coffee' etc.) are due to an operation which intervenes between lexical lookup and meaning combination, and is similar to discourse processes that establish 'missing links' between rhetorically related utterances. Inasmuch as this operation is non-destructive, it is compatible with a strong concept of compositionality.

1. Compositionality

It is impossible to imagine a language for which no **compositional** interpretation exists under the usual definition of compositionality:[1]

(1) **Compositionality**
The meaning of a complex expression is a function of the meanings of its sub-expressions and of the way in which they are put together.

It seems obvious that this criterion can be satisfied by any complex expression whatsoever if 'the way in which its constituents are put together' can be determined at all.[2] That is, practically any interpretation may be compositional in the technical sense, even one that fails to satisfy the intuitive requirement of 'minimizing idiomaticity'. For example, languages used for giving commands to computers, such as the command language of the Unix operating system, can easily be given compositional interpretations, although people have a hard time learning such languages because of their highly idiomatic character.

The definition in (1) has at least three loose points that are jointly responsible for the the fact that it fails to capture the intuitive concept of compositionality:[3]

(2) **Weak points of compositionality**
(i) the **function** used for combining the meanings of the sub-expressions can be any function at all;
(ii) there is no *a priori* limitation on what objects **meanings** can be;
(iii) in the definition above, there are no constraints as to how the meanings should be **assigned** to the simplex sub-expressions.

Let me now elaborate on each point in (2i–iii) above.

Ad (2i): There are few things a function cannot do. In particular, we can define functions pointwise so that their behaviour cannot be considered **uniform** in any intuitive sense. That is, although the intuitive content of compositionality implies that, under normal circumstances, the same constituent in the same syntactic role will play the same semantic role, this is not guaranteed by the definition in (1). For example, in the Unix command language, there is no constraint to the effect that the various uses of a parameter like `-v` should have anything in common. The command '`grep` ⟨expr⟩' is used to select those lines of a file that match the expression ⟨expr⟩, whereas '`grep -v` ⟨expr⟩' selects the lines that do **not** match the expression. On the other hand, with other commands (such as the `tar` archiver program) the parameter `-v` results in a 'verbose' output. This property is felt idiosyncratic by users, although both the `grep` and the `tar` programs embody functions, which simply behave differently depending on whether they are given the parameter `-v`.

Similarly, under the definition in (1), a word like *yesterday* could easily mean 'yesterday' in some sentences and 'accidentally' in others. True, a word like *yesterday* could be **polysemous** in the sense that there would be **two** homonymous (homophonous) words of this shape, one of which means 'yesterday' and the other 'accidentally'. In that case, we would consider them two different expressions (the surface shapes of which incidentally coincide), and the principle of compositionality would apply to both expressions separately and independently. On the other hand,

[1] This was shown by Janssen (1983). For the variants of the definition below and their history, see Partee (1984) and Szabó (1995).

[2] Cf. Partee (1984).

[3] Partee (1984) raises the same problems.

were this to be the case, our hypothetical word *yesterday* would lead to ambiguities in most cases. That is, most sentences would be ambiguous if they contained it as a constituent. But the definition in (1) would even allow a situation in which *yesterday* could unambiguously refer to 'yesterday' in some cases, and 'accidentally' in all others, which is hard to imagine in any human language.[4]

Ad (2ii): There are few things we could not do with artificial meanings.[5] The simplest way of showing this is the following. Obviously, the intention of the principle of compositionality as formulated in (1) is that the meanings of complex expressions depend on the **meanings** rather than the **forms** of their sub-expressions. However, if what a 'meaning' is was left unconstrained, then we could assign **character strings** to certain sub-expressions as their meanings (say, character strings corresponding to their orthographic form), and have the combination function behave differently depending on what those strings are, thereby getting around an essential aspect of what (1) intends to claim. (The example of the command `grep` above can also be use here: the parameter `-v` does not have a meaning of its own; it is the **form** of the parameter that the `grep` program takes into account.) I will not have too much to say about arbitrary meanings in the following. Note, however, that a remedy for the weakness in (2i) could also help solving (2ii): For example, if the behaviour of combination functions (or a program like `grep`) simply cannot depend on character string parameters, then the above 'trick' is not feasible.

Ad (2iii): There are few things we could not do if we did not constrain the assignment of meanings to simplex expressions. Although the principle of compositionality intends to constrain the assignment of meanings to **complex** expressions, we can still do almost anything if the assignments of meanings to **simple** expressions is left unconstrained. For example, if we could determine the meaning of a simple sub-expression depending on what other expressions occur in the complex expression (and 'the way in which they are put together'), then the meanings of complex expressions could depend on the **shape** of their constituents, which would be undesirable, as pointed out in (2ii) above. For example, even if the 'meaning' of the parameter `-v` was more than just a character string (that is, if its meaning was now 'print the non-matching lines', now 'produce a verbose output'), we could still assign one or the other meaning to it depending on whether it occurs with `grep` or `tar`, and we still would not violate the principle of compositionality.

My conclusion is that new principles, constraining the compositionality principle in (1), should be developed and adopted for the syntax/semantics interface of natural languages. They should constrain both the class of functions that can be used for combining the meanings of the sub-expressions of a complex expression and the way in which simple expressions are assigned meanings. The following section will be devoted to such constraints.

2. Strong Compositionality

In this section, I develop an alternative to the traditional concept of compositionality. The alternative will consist in adding two sub-principles to the traditional definition (1) in **Section 1**, called **independence** and **additivity**. The resulting,

[4] Seemingly similar cases include the 'systematic ambiguity' of indefinite determiners (and the quantifier *any*) in English, which can have a universal meaning in certain syntactic contexts. But most linguists find it undesirable to attribute such facts to homonymy.

[5] Cf. Janssen (1983) and Partee (1984).

more restrictive principle will be called **strong compositionality**. The following sub-sections introduce these principles.

2.1. Independence

I will start with the problem of assigning meanings to the simplex sub-expressions of an expression. We have seen that the intuitive non-compositionality of Unix commands is partly due to the fact that the program invoked is free to interpret the parameters, depending on its idiosyncratic contents. The principle of compositionality (see (1) in **Section 1**) leaves it open how the simplex sub-expressions of a complex expression are assigned meanings. It is easy to see that, if one constituent of a complex expression (call it the 'functor') may determine what meaning is assigned to another constituent (the 'operand'), that is effectively the same as passing the **shape** of the 'operand' to the 'functor'. Moreover, any kind of interdependence between meaning assignments in a complex expression amounts to operating on the shapes of its constituents. Allowing such operations weakens the concept of compositionality to an intolerable extent. Using the example of the Unix command language, the meanings assigned to the operands of a command (such as the parameter `-v`) should not depend on what the command name is and what the other parameters are. That is, meanings should not be assigned in a construction-specific manner. I propose the following principle to achieve this:

(3) **Independence**
The meanings of the constituents of a complex expression are assigned independently of each other, of the way in which they are put together, and of the function that yields the meaning of the complex expression.

If a language obeys the principle of independence, then the meaning of an expression may not vary depending on what it is a constituent of. Were we not to impose such a constraint, very similar constructs (e.g., containg the same expression in the same syntactic role) could be interpreted in heterogeneous (or even unrelated) ways. Note that this principle implies that the meaning contributions of the constituents of an expression are constant, i.e., they do not vary from one construct to the other. This means a certain **context-independence**, which many would deny. I conceive of this as a price to pay for a reasonable alternative to the traditional concept of compositionality.

The role of the external context in the interpretation of complex expressions is undeniable, but it is not a challenge to either the principle of compositionality or that of independence as long as we can reduce it to an influence on the assignment of meanings to **simple** sub-expressions, and this seems perfectly feasible (cf. Partee (1984)). On the other hand, the principle of independence also does not prevent those **meanings** that the sub-expressions are assigned from interacting with each other to produce complex meanings (cf. **Section 3.3**). We only want to exclude the dependency of the meanings of complex expressions on the formal properties (shapes) of their sub-expressions.

2.2. Additivity

The independence of meaning assignments eliminates one weak aspect of the traditional concept of compositionality, but it does not eliminate the arbitrariness of meaning composition. For example, take two constituents of a complex expression again, one being the 'functor', and the other the 'operand'. Even if we assume the

principle of independence, i.e., that the 'functor' has only access to the meaning, but not the form of the 'operand', the behaviour of the 'functor' may still vary arbitrarily depending on what the operand is. One of the simplest cases would be the difference between the presence and the absence of the operand. It is clearly impossible for a natural-language verb to mean completely unrelated things depending on whether it is given an argument or not, unless there is idiomaticity involved. For example, the Unix command `sendmail` (which is used, among other things, for submitting mail messages) creates a configuration file when used with the parameter `-bz`. There is no transparent relation between the different effects that running `sendmail` can have, which we can characterize as idiomatic, intuitively non-compositional behaviour.

If we want to achieve a uniform behaviour of commands, we should stipulate that the meaning contribution of the command name cannot be radically altered by the presence or absence of parameters, and the other way round, the meaning contribution of a parameter or a type of argument must not be radically altered by the command that it is given to. To guarantee this in general, we have to stipulate that the meanings of constituents are combined in a **non-destructive** way: Whatever each constituent contributes to the meaning of the entire complex expression must be constant from one expression to the other, without being distorted by the meanings of its co-constituents. This can be formulated as a separate principle:

(4) **Additivity**
The function that combines the meanings of the sub-expressions of a complex expression must not destroy the information contained in those meanings.

The name 'additivity' is motivated by the fact that, if a function obeys this principle, then the meaning of the complex expression is a simple 'sum' of the meanings of the sub-expressions. That is, the meaning or **information content** of each constituent is simply 'summed up' in some technical sense of the word. Obviously, the definition presupposes a concept of **information content** for meanings. Usually, this kind of concept is defined by attributing an algebraic structure to the domain(s) of meanings. The structure must contain an ordering in terms of informativity, and an operation of combining pieces of information, which should not lead out of the structure. Assuming such a concept, additivity means that the combination of two members of such a domain must yield a third member that is ordered higher than both operands in the informativity hierarchy.

For example, additivity is satisfied if the domain of meanings is a powerset of some set, and the only operation that may combine meanings is intersection. Note that additivity presupposes that the meanings of the sub-expressions combined are comparable in terms of the informativity ordering. It is not easy to satisfy this requirement within the traditional models used in formal semantics. I will turn back to this problem in **Section 3**.

Additivity makes it very cumbersome to deal with **non-monotonicity**, i.e., phenomena in which so-called **default values** are involved. That is, additivity excludes those combinations of meanings in which one meaning **destroys** or **blocks** some default information associated with another. Apparent counterexamples can be found in artificial languages designed for man/computer communication, in which default mechanisms play an important role. But even in natural languages, we find cases reminiscent of destructive behaviour. For example, consider:[6]

(5) a. *Joe is eating.*
'Joe is eating food'
b. *Joe is eating sand.*

[6] I owe this example to Anna Szabolcsi.

Clearly, it looks as if the 'default' direct object of *eat* was 'some food', and as if this 'default' could be 'overridden' by an explicit direct object that does not denote food. I believe there are at least to ways to deal with this type of phenomena: (a) it could be argued that using a transitive verb intransitively corresponds to a separate syntactic construction, which specifies how the implicit direct object is to be interpreted in such cases; (b) the domain of denotations can be enriched in such a way that each denotation consists of two components, namely, a parameterized object and a specification of how the default values of parameters can be produced; It is easy to order such a domain in terms of the information content of the elements, so that this solution can be made compatible with the principle of additivity. The latter treatment certainly seems more attractive. In particular, one could argue that the phenomenon in (5) may well be universal, so there is no justification for attributing it to a separate construction of English. I will touch upon this problem in a moment (in **Section 3.3**).

2.3. Strong Compositionality

The definition of **strong compositionality** is the conjunction of compositionality, independence and additivity. I submit strong compositionality as an alternative to the principle of compositionality.

(6) **Strong Compositionality**
The meaning of a complex expression is strongly compositional if and only if it obeys
(i) the principle of compositionality (cf. (1) in **Section 1**);
(ii) the principle of independence (cf. (3) in **Section 2.1**); and
(iii) the principle of additivity (cf. (4) in **Section 2.2**).

It should be clear from the above that the three sub-principles are independent. Notice how the weaknesses of compositionality (cf. (2) in **Section 1**) are remedied by strong compositionality. Additivity is an answer to the arbitrary character of combination functions (cf. (2i)), and independence constrains the assignment of meanings to simple sub-expressions (cf. (2iii)). As I mentioned earlier, the third weakness of compositionality (i.e., the arbitrary character of meanings, cf. (2ii)) is probably harmless if the two other problems are discarded.

3. Strong Compositionality in Natural Language

This section deals with some consequences of strong compositionality for the semantics of natural language. First, in **Section 3.1**, I will examine the effect of additivity (and strong compositionality in general) on common views of semantic combination and types, and I will conclude that the traditional (Fregean) metaphor of 'incomplete' linguistic expressions as **functors** that apply to **operands** to become complete is to be abandoned.

3.1. Abandoning the Functor Metaphor

I have touched upon various consequences of strong compositionality on natural-language semantics already. In this section, I will dwell on a very particular consequence of strong compositionality, originating from the principle of additivity. If the combination of the meanings of constituents is to be additive in the technical sense explained in **Section 2.2**, then the denotations of the immediate constituents of a

complex expression must be comparable in terms of informativity. Were they not, we could not perform intersection-like operations on them. I believe the only sensible way of implementing this constraint is to assume a **type-free** semantics, which implies that the traditional (Fregean) type system cannot be used for explanatory purposes.

The question arises why we could not implement additivity in a typed semantics. For example, we could order meanings in terms of informativity by taking a propositional counterpart (a sort of 'existential closure') of every intension:

(7) **Ordering intensions**
(a) If ξ is an intension of type $\langle s, e\rangle$, then $\mathcal{E}(\xi)$, the 'closure' of ξ is

$$\{\langle w, v\rangle \in W \times 2 : v = 1\},$$

i.e., $\mathcal{E}(\xi)$ is a constant function, which assigns 'true' to every possible world (because $\exists x.a = x$ is true in every possible world);
(b) If ξ is an intension of type $\langle s, t\rangle$, then $\mathcal{E}(\xi) = \xi$;
(c) If ξ is an intension of type $\langle s, \langle \alpha, \beta\rangle\rangle$, where α and β are intensional types, then

$$\mathcal{E}(\xi) = \{\langle w, v\rangle \in W \times 2 : v = 1 \Leftrightarrow \bigvee_{\chi \in \mathcal{D}_\alpha} \mathcal{E}(\xi(w)(\chi))(w) = 1\},$$

i.e., the 'closure' of ξ assigns the value 'true' to a possible world w if there is an intension χ within the domain of the type α such that, if we apply the extension corresponding to ξ in w to χ, then the 'closure' of the resulting intension (which will be of type β) yields 'true' in w.

Given this definition of $\mathcal{E}$, an intension ξ (of any type) is more informative than an intension χ (of any type) if and only if the closure of ξ entails the closure of χ:

$$\xi \leq \chi \Leftrightarrow_{\text{def}} \{w \in W : \mathcal{E}(\xi)(w) = 1\} \subseteq \{w \in W : \mathcal{E}(\chi)(w) = 1\}.$$

Under this definition, the intension of *Joe likes Mary* is more informative than, e.g., $\lambda x.x$ *likes Mary*, $\lambda x.$*Mary likes* x or $\lambda x \lambda y.x$ *likes* y (the closure of which yield true whenever someone likes Mary, Mary likes someone, or someone likes someone, respectively).

It is easy to illustrate why this would not work with a simple example. For the ordering of denotations in terms of information content to be sound, a propositional meaning (intension) q will contain more information than another propositional meaning p just in case q entails p. Now, assuming that propositional negation is part of our language,[7] and we have a sentence S which denotes p, $\neg p$ could not correspond to the negation of S, because $\neg p$ does not entail p. So negation could not exist in a language the interpretation of which obeys additivity. This sounds like very bad news indeed. To remedy the situation, we have to assume that the meaning of S is not a proposition at all. What it should be will be clarified in

[7] This seems a reasonable assumption. Even if one denies the existence of propositional negation in natural language (saying that only predicate negation exists), the same argument can be made with predicate negation. Also, it is easy to construct analogous examples with other operators, such as various generalized quantifiers.

Section 3.2 below. For the time being, let us see what would happen if we took all expressions to be of the same type instead of ordering expressions across types.

So assume that we implement additivity by considering the meanings of the expressions that we combine to be of one and the same type, even when their **syntactic** types are considered different. In terms of this, the traditional idea that compositionality implies a perfect match of semantic and syntactic categories must be abandoned. In particular, we cannot explain the syntactic incompleteness of certain expressions with their semantic incompleteness, i.e., their being **functors** that are incomplete unless we provide them with the appropriate **operands**.[8] Under this approach, additivity seems to exclude the perfect match of syntactic and semantic types in general, and the functor/operand metaphor in particular.

Abandoning the functor/operand metaphor raises two problems:

(8) **Problems with additivity**
  (i) syntactic incompleteness cannot be expressed in terms of semantic incompleteness; and
  (ii) semantic incompleteness cannot be expressed in terms of functional types (expecting arguments).

The problem in (8i) does not seem too big a price to pay for additivity. In natural languages, syntactic incompleteness is often idiosyncratic, so it has to be stated somehow, anyway. For example, there are equivalents of the English verb *marry* in various natural languages that are obligatorily transitive or obligatorily intransitive (the English verb itself is optionally transitive), but they refer to the same concept.[9]

The problem in (8ii) looks more serious at first sight. How can we tell from the denotation of, say, a verb, if it is incomplete unless we combine it with the denotation of an argument? What is it in the denotation of a determiner that makes it so incomplete that determiners seldom occur on their own in natural languages? The functor/argument metaphor explains this type of facts very straightforwardly, to the extent that it seems almost unquestionable. For example, if a verb expresses a relation between two entities, then it is only natural that its occurrences are incomplete unless it is complemented with two other expressions, denoting entities.

I propose a radical solution to this problem. Maybe 'semantic completeness' is not an indispensable concept at all. If we are ready to accept a model in which meanings are ordered in terms of informativity, it is not clear at all whether we have to posit the existence of 'complete' meanings on formal, ontological or linguistic grounds. In formal terms, it is certainly possible that the algebraic structure of meanings is not atomic, i.e., there need not be any meanings in the structure that can only be enriched in such a way that a contradiction arises. Although it is not ontologically implausible that certain entities in the model are 'complete', it is not at all clear whether any linguistic expression, even a large piece of discourse, can successfully denote such an entity. In sum, I see no compelling reason why the Fregean theory, in which sentences and individual names are 'complete' (whereas everything else is 'incomplete'), should be adopted.

In terms of this radical solution, a transitive verb with missing arguments or a determiner without a noun are never **semantically**, but at most **syntactically** incomplete. This need not imply a 'mismatch' between semantic and syntactic structure,

[8] The idea that syntactically incomplete expressions are to be considered functors originates from Frege (1870).

[9] The different options do not seem to correlate with the sociological differences that exist between the role of marriage across cultures.

however. Syntactic structures may be associated with types of state of affairs in a systematic way without relying on the concept of incompleteness. Uttering a transitive verb phrase may syntactically require the utterance of a transitive verb and a direct object, and denote a certain type of states of affairs. If there is any 'mismatch' at all between syntactic and semantic structures (from this perspective), it is between the 'completeness' properties of certain syntactic constructs and their semantic counterparts: 'incompleteness' may make sense for some syntactic constructs, but not for the corresponding semantic objects. If a quantifier or a determiner is syntactically incomplete as a rule, that may be due to the fact that the relation that it denotes is too abstract to be relevant to humans as such. A transitive verb like *love*, on the other hand, denotes a concept that is relevant as such, therefore, most languages possess more 'complete' expressions (nominalized versions) that refer to the same concept.

3.2. The Domain of Denotations

Abandoning the functor/operand metaphor leaves us with the problem of how the model theoretic interpretation of natural-language expressions can be implemented in a type-free model, in which every expression denotes entities comparable in terms of informativity. As a matter of course, I will have to make gross simplifications to explain the basics of such a model. In particular, I will concentrate on the domain of denotations and ignore the treatment of mechanisms related to defaults and the dynamism of interpretation in this first approximation.

The basic idea is the same as the one in Kálmán and Rádai (1995a,b), where we were mostly concerned with the compositional variants of computer command languages in the spirit of strong compositionality. In those papers, the domain of denotations for command languages are the powerset of all processes that can be run on a computer. The denotation of a command name is the set of processes that its invocations can initiate, and the denotation of a command parameter is the set of processes that can be run using the same parameter (in the same syntactic position or the same function). As a matter of course, the complexities of natural-language interpretation are enormous as compared to command languages. Let me say a few words about the most important differences.

The **semantic domains** that underlie the interpretation of natural languages are much more complex than those in the Unix command language. While the latter consist of 'machine states', the models that we need for interpreting natural languages include various **possible worlds** (hypothetical or real, actual or past/future) in which various **individuals** (and, eventually, groups of individuals, if they have properties that are not predictable from those of their members) exist, and various **relations** hold for them. These possible worlds are also **dynamic** in the sense that they may also change in time, which corresponds to the various **eventualities** that we can talk about in natural languages. (This type of dynamism is not to be confused with so-called 'dynamic interpretation', to be touched upon promptly.)

In the modern paradigm of formal semantics (called **dynamic** semantics), natural language meanings are seen as **instructions** for the hearer to **update** his/her **information state** about entities in the model. So we can think of denotations as sets of updates in the same way as Unix command lines denote sets of computations, which may change information states just like computations change machine states. But an information state is much more complex than a machine state, because it is not a complete description of a model, but some representation of what information is available about it.

Owing to all these complexities, the **deferred information** content of a natural-

language denotation (which is the specification of how default values are to be assigned to parameters if needed, if we have such a component, as I suggested in **Section 2.2**) is also much larger and much more complex than what we need for the interpretation of Unix commands.[10] It must encode a large body of linguistic and non-linguistic (scientific and cultural) knowledge that may influence the interactions of meanings (cf. **Section 3.3**).

Taking all these limitations into account, and using a simplistic static view of sentence meaning (which is the common practice in standard approaches to natural-language semantics), we will base the domain of denotations upon the powerset of all possible **situations** (I will give a precise definition of situations in a moment). For example, an individual name can be associated with the set of all the situations in which the given individual is somehow 'involved', and a relation can be associated with those situations in which some entities stand in the given relation. (This is analogous to the concept of 'closure' in (7) in **Section 3.1**.) In this way, we can deal with negation (see **Section 3.1**) in an additive way: negation is combined with polarity-less propositions, which correspond to situations in which the given proposition has any truth value at all.

Situations can be modelled with partial models, which we have called **model fragments** in Kálmán and Rádai (1995b):

(9) **Definition: Model fragments**
If $\mathcal{M} = \langle \mathcal{U}, \mathcal{I} \rangle$ is a first-order model (where the **universe** $\mathcal{U}$ is a set of individuals, and the **interpretation function** $\mathcal{I}$ assigns a set of n-tuples in $\mathcal{P}(\mathcal{U}^n)$ to every n-ary predicate constant), then the set of the **model fragments** of $\mathcal{M}$, written $\mathcal{F}_\mathcal{M}$ is a set of first-order models $f = \langle \mathcal{U}_f, \mathcal{I}_f \rangle$ such that
(i) $\mathcal{U}_f \subseteq \mathcal{U}$, and
(ii) if P is an n-ary predicate constant, then

$$\mathcal{I}_f(P) = \mathcal{I}(P) \cap \mathcal{U}_f^n.$$

That is, a model fragment has a small universe and an intrepretation function the values of which are restricted to that universe. In general, we will talk about the set $\mathcal{F}$ of **all** model fragments (of any model). So we will think of situations as model fragments.[11]

Although the powerset of $\mathcal{F}$ is itself partially ordered by the relation '$\subseteq$', we will need a more complicated domain for denotations. We need the extra complication in order to combine meanings. For example, similarly to the view explained in connection with (7) in **Section 3.1**, 'loves Mary' (i.e., $\lambda x(\text{loves}(x, \text{Mary}))$) is associated with the 'closure' set of situations in which someone loves Mary, and 'Joe' is associated with those situations in which Joe is present ($\lambda x(x = \text{Joe})$ is mapped to the 'closure' set of model fragments the universe of which contains 'Joe'). Obviously, we want the combination of these two meanings to yield the set of situations associated with 'Joe loves Mary'. But the simple intersection of the corresponding sets of situations does not yield that set: it yields those situations in which someone loves Mary and Joe is present, but Joe does not necessarily love Mary in all of them. The desired denotation is a subset of this set. In order to produce it, we need a different domain, namely, the domain of functions that map individuals to sets of situations. Then we can combine $\lambda x(\text{loves}(x, \text{Mary}))$ with $\lambda x(x = \text{Joe})$ using the operation called **coalesced intersection**:

[10] For the concept of deferred information see Kálmán (1990).

[11] It is easy to extend this concept to fragments of higher-order models if needed.

(10) **Definition: Coalesced Intersection**
If $f, g \in {}^D R$ are functions with domain D and range R such that R is closed under $\cap$, then

$$f \oplus g =_{\text{def}} \{\langle d, r \rangle \in D \times R \colon f(d) \cap g(d) = r\}.$$

For example, if f maps every individual to the set of situations in which that individual loves Mary, and g maps every individual to the set of situations in which that individual is identical to Joe, then $f \oplus g$ maps every individual to the set of situations in which the individual is Joe and loves Mary. The basic equivalence for coalesced intersection is the following:

(11) **Fact: Basic equivalence for Coalesced Intersection**
$\lambda x(\varphi) \oplus \lambda y(\psi) \equiv \lambda z(\varphi[x/z] \wedge \psi[y/z])$

whenever z does not occur free in either φ or ψ, and the interpretation of $\wedge$ is $\cap$. (As usual, $\varphi[x/z]$ is the same as φ, with the free occurrences of x replaced with z.)

The partial order that we can define over the domain defined in this way is as follows:

(12) **Definition: Partial Order Over the Domain of Denotations**
If $f, g \in {}^{\mathcal{U}}\mathcal{P}(\mathcal{F})$, then $f \sqsubseteq g$ (read: f is at least as informative as g) if and only if $f \oplus g = f$.

This definition satisfies the intuitive requirement that combining two elements of the domain using coalesced intersection is an element that is at least as informative as either one of the combined ones.[12] What function is assigned to a particular expression must be driven by syntactic information. For example, when *love Mary* is not a verb phrase (but a non-finite clause), then it should be assigned the constant function $\lambda z(\exists x.\text{love}(x, \text{Mary}))$ instead of $\lambda x(\text{love}(x, \text{Mary}))$.

Notice how this kind of interpretation excludes the undesirable consequences of the principle of compositionality as stated in (1) in **Section 1**. If we assign a set of processes to a parameter like `-v`, e.g., the set of processes which display (rather than discard) verbose output, then we can only combine this meaning with the set of processes that invoking the command `grep` can initiate in such a way that '`grep -v` ⟨expr⟩' should mean 'find the lines with ⟨expr⟩ and produce verbose output', but not 'find lines not matching ⟨expr⟩'.[13]

3.3. Interaction of Meanings

Although independence prohibits the meanings assigned to sub-expressions from depending on each other, there are clear cases when the meanings of sub-expressions **interact** in the process of interpretation. For example, consider the following expressions:

[12] As a matter of course, we would need higher-order model fragments to interpret propositional connectives, generalized quantifiers etc. in a similar way. Obviously, the domain of denotations would then be the set of functions from entities of any type to sets of situations.

[13] Even if we allow for 'adjustment' processes which bring the meanings to be combined into harmony before performing their coalesced intersection, because those processes must respect additivity as well. This will be the topic of the next section.

(13) **Uses of** *coffee*

a. *some ground coffee*
'some ground coffee (seeds)'

b. *a hot coffee*
'a hot coffee (drink)'

c. *a quick coffee*
'a coffee prepared/consumed/... quickly'

d. *after a coffee*
'after consuming a coffee'

These expressions illustrate what we might call **systematic ambiguity**:[14] it is very common for names of plants (like *coffee*) to stand for their consumable parts (like coffee beans) as well as their derivatives in various stages of preparation (like the roasted seeds and the liquid in (13a–b)). On the other hand, nouns referring to food quite often take modifiers that refer to their preparation or consumption (as *quick* in (13c)), and the nouns themselves may stand for the consumption of the food in question (as *coffee* in (13c–d)). If, however, we were to assign the meanings 'coffee seeds', 'coffee drink', 'preparation of a coffee' or 'consumption of a coffee' to *coffee* in (13a–d), we would violate independence. It is also clear that we would 'miss generalizations' if we were to treat the ambiguities in (13) as accidental surface coincidences (homonymy). On the other hand, since the formal properties of the sub-expressions play no role in the interpretation of the above examples, it must be possible to explain 'systematic ambiguities' of this sort without violating independence.

Quite obviously, such phenomena stem from the important role of **implicitness** in natural-language interpretation. The examples in (13) are compact expressions corresponding to more complex meanings, which no competent speaker would have any trouble to make explicit. On the other hand, there is some **non-determinism** in the interpretation of such compact expressions. For example, it is not absolutely excluded for *hot coffee* in (13b) to stand for 'hot coffee powder' or 'hot coffee seeds' in certain contexts. What the examples in (13) show, then, is that natural-language meanings can be combined in more than one way, and how exactly the hearer is supposed to proceed is often left implicit by the speaker. Both the fact that competent speakers can produce equivalent, more explicit paraphrases and the fact that the actual choice of the paraphrase is not entirely determined indicate that the processes involved are similar to other cases related to implicit information. For example, it is usually left implicit why two sentences are put one after the other in a piece of discourse (because they are part of the same story, they support the same argument, etc.). Similarly, definite descriptions are usually compact descriptions that can be made more explicit by attaching relative clauses to them. Establishing missing links is also essential to retrieve the antecedents of anaphoras. In sum, the process of interpreting expressions like those in (13) involves something very similar to certain **discourse processes** in which the speaker expects the hearer to establish 'missing links' such as anaphoric and rhetorical relations.

This suggests that combining the meanings of natural-language expressions may involve more than the simple 'intersection' operation that I have suggested in the preceding sections. In particular, we can assume an 'adjustment' operation which follows lexical lookup, but precedes the meaning combination proper, and is triggered by the syntactic relation in which the constituents involved stand with each other. The meanings assigned to the constituents of such expressions are

[14] For an overview of such cases and of their treatments see Pustejovsky (1991).

processed and 'brought into harmony' with each other before combining them. We can think of this 'pre-processing' as analogous to those phonological processes (e.g., assimilation) which affect the lexical phonemes when they enter into contact through affixation. Most importantly, additivity requires the pre-processing operations to be non-destructive. That is, using the phonological metaphor, phenomena analogous to genuine morphological (not phonologically driven) stem or affix alternation are excluded from semantics.

According to the above, whatever the mechanisms that **control** the 'assimilation' processes could be, their **effect** can only consist in increasing the information content of the constituents participating in these processes. That is, the effect of such processes is identical to that of 'regular' meaning composition, except that some of the constituents being composed are only virtually present. For example, the transition from 'coffee' to 'coffee drink' in (13b) is one of increasing information content. It is essentially identical to what happens when we combine the meaning of *coffee* with that of *drink*, although the latter does not occur in (13b) overtly. So we can think of the ('assimilated') meaning of *coffee* in (13b) as an **instantiation** of the more general meaning 'coffee'.

Where do the extra pieces of information involved in harmonization processes come from? As in most similar processes such as the discourse processes mentioned earlier, they originate from the speakers' knowledge of the external world, i.e., **encyclopedic** and **contextual knowledge**. For example, in the case of (13b), both the hearer's encyclopedic (generic) knowledge about coffee in general (the fact that a drink that is usually drunk hot is prepared from coffee) and his/her contextual knowledge about the situation (i.e., whether an everyday coffee-drinking situation or a different, say, roasting situation might be involved) play a role in 'harmonization'. Clearly, the process that provides the 'missing link' between *hot* and *coffee* (namely, 'drink') has something in common with the process that provides the missing argument 'food' in (5a). Although the mechanisms involved in 'default' direct object of an optionally transitive verb like *eat* are not exactly the same as the 'harmonizing' processes that we have seen in (13), they share the property that default knowledge concerning the external world gets activated (possibly biassed by the actual context). The fact that the direct object of eating is usually food is the same kind of generic knowledge as we have seen in the case of *coffee*; and the actual context may also play a role in interpreting (5a) (e.g., *Joe is eating* could in fact be interpreted as 'Joe is eating sand' in a strange situation when people take turns eating sand). Similarly, external context may play a role in interpreting expressions similar to those in (13). For example, *a quick window* does not seem to make much sense unless maybe in the context of a carpenter's workshop, in the middle of a process affecting/creating/... windows.

These considerations indicate that, hopefully, we can deal with the effects of 'harmonization' processes by complicating our meaning representations as was indicated in **Section 2.2**, i.e., by using parameterized meanings plus specifications of how the default values of the parameters can be produced: The information used for instantiating an abstract meaning can only originate from the knowledge about 'defaults' that is associated with it. But the overall properties of the calculation of complex meanings, from lexical lookup through 'harmonization' to combination, are in perfect agreement with strong compositionality.

References

Janssen, T.M.V.: 1983, *Foundations and Applications of Montague Grammar*, Mathematisch Centrum, Amsterdam

Kálmán, L.: 1990, Deferred Information: The Semantics of Commitment, in L. Kálmán and L. Pólos (eds.), *Papers from the Second Symposium on Logic and Language*, Akadémiai, Budapest, 125–157

Kálmán, L. and Rádai, G.: 1995a, Compositional interpretation of computer command languages, *Working Papers in Theoretical Linguistics 2/2*, Theoretical Linguistics Programme, Budapest University (ELTE), and Research Institute for Linguistics, Budapest

Kálmán, L. and Rádai, G.: 1995b, Dynamic Update Predicate Logic, *Working Papers in Theoretical Linguistics 2/6*, Theoretical Linguistics Programme, Budapest University (ELTE), and Research Institute for Linguistics, Budapest

Montague, R.: 1970, Universal Grammar, *Theoria* 36, 373–398. Reprinted in R. Thomason (ed.), *Formal Philosophy: Selected Papers of Richard Montague*. Yale University Press, New Haven, 1974

Partee, B.H.: 1984, Compositionality, in F. Landman and F. Veltman (eds.), *Varieties of Formal Semantics*, Foris, Dordrecht

Pustejovsky, J.: 1991, The generative lexicon, *Computational Linguistics* 17/4, 409–441

Szabó, Z.: 1995, *Issues in Compositionality*, Ph.D. thesis, MIT, Cambridge MA

A Type-Theoretic Semantics for λ-DRT[1]

Michael Kohlhase, Susanna Kuschert and Manfred Pinkal
Universität des Saarlandes

1 Introduction

In the formulation of semantic representation formalisms for NLP applications two aims have substantially shaped research: firstly, to find a compositional algorithm for semantic construction and secondly, to be able to interpret discourse. Both Montague's idea in the early seventies of using the well-known typed λ-calculus to formulate a rule system for semantic construction, and Kamp's Discourse Representation Theory (DRT), first presented in 1981, even today play a kind of paradigm role in these two aims. In particular, because of the — originally intended — lack of compositionality of Discourse Representation Theory, a number of approaches defining compositional discourse semantics have been proposed, such as, e.g. Groenendijk and Stockhof 1991 and Groenendijk and Stockhof 1990, Zeevat 1989, Eijck and Kamp 1995 and Muskens 1994.

This paper presents λ-DRT, a formalism combining these two paradigms very straightforwardly. It has been used as the semantic representation formalism of the dialogue translation project Verbmobil (Bos et al. 1994) for several years. In particular, β-reduction is used as a convenient tool for the semantic processing, just as in Montague semantics. Additionally, the formalism is able to work explicitely with the structural properties that DRT builds on, rather than eliminating its structure. In this way, λ-DRT expressions look very much like those of Muskens' language (Muskens 1994).

Though λ-DRT has been used in several applications already, its semantics and reduction system has not yet been formally described. As a result, the reduction rules, in particular the central β-reduction, have not yet been proved to be *semantically correct*. It is the main aim of this paper to provide this theoretical background, and we hope to lay appropriate grounds for extensions like a deduction calculus based on λ-DRT.

The definition of λ-DRT in this paper strongly builds on the view that it is like a marriage of λ-calculus and DRT; it may be thought of as the extension of either one of the two components by the other. The language is essentially constructed by two (parallel, i.e. equal-weighted) abstraction operators; these are the well-known λ-operator and a new δ-operator. While the first is used to construct functional, higher order expressions, the latter can be seen as declaring a set of variables as discourse referents and thereby constructing a DRS. The two concepts are kept strictly separate by not defining one by means of the other[2]. We hope through this to come to a better understanding of firstly, which of the discourse referents' properties enable the correct treatment of anaphora, and secondly, how the two binding mechanisms interact with each other. We believe that the second point is especially important to be able to define a unification algorithm for λ-DRT and thus an inference system.

The λ-DRT expressions will be given a direct semantics in this paper, using standard type-theoretic methods. The λ-calculus type system is extended by two

1. This paper developed from S. Kuschert's Diplomarbeit (Master's Thesis, Kuschert 1995). The work was partially supported by the SAMOS project -II A 2 - Pi 154/5 - 3.

2. There are at least two suggestions how to define dynamics by means of the λ-operator. Muskens 1994, for example, defines dynamics by (λ-)abstracting over two states which represent the input- and output-state of DPL Groenendijk and Stockhof 1991. Ruhrberg 1995 introduces simultaneous λ-abstraction of several variables at a time and is able to model most aspects of dynamic theory.

types. Firstly, we define a type for DRSes. The denotations for DRSes are very much alike those of Zeevat 1989, being pairs of sets of discourse referents and sets of partial assignments. Secondly, we introduce a type for individual concepts (intensional correlates to expressions of type for individual objects); the use of an intensionalizer in large parts enables the compositional treatment of discourse referents, in a similar way as in the Intensional Logic of Dynamic Montague Grammar (Groenendijk and Stockhof 1990).

Most of the above-mentioned compositional discourse representation formalisms merge their dynamic objects by using an asymmetric merge operator ';' which corresponds to the dynamic conjunction of DPL. λ-DRT adds some expressivity by allowing for a symmetric merge operator $\otimes$. Indeed, the asymmetric operator will turn out to be a specialisation of the symmetric one. Motivation of the $\otimes$-operator is in part linguistic (considering Bach-Peters-sentences in which two phrases are connected by both an anaphor and a cataphor), but mostly of methodological (e.g. to allow for incremental processing) and technical (i.e. computational/implementational) nature.

An important goal in the definition of the semantics was to achieve compositionality on denotations, since having such a semantics means that the development of inference processes will be simplified. The goal was met, together with a simple definition of the interpretation of the $\otimes$-operator, based on partial assignment functions.

2 A Syntax for λ-DRT

In this section we will define the syntax for a variant of λ-DRT (Bos et al. 1994) which will act as the basis for the whole of this paper. Let us start by fixing the type symbols of the language.

Definition 2.1 (Type Symbols) The set of **type symbols**, or **types** for short, of λ-DRT, $\mathcal{T}$, is defined inductively as follows:

1. e is a type symbol (denoting the type of individuals).
2. d is a type symbol (denoting the type of individual concepts).
3. o is a type symbol (denoting the type of truth values).
4. t is a type symbol (denoting the type of DRSes).
5. If α and β are type symbols, then so is (α, β) (denoting the type of functions with argument of type α and values of type β).

Definition 2.2 (Signature) For every type $\alpha \in \mathcal{T}$ we have a non-empty set Σ_α of constants of type α and call $\Sigma = \bigcup_{\alpha \in \mathcal{T}} \Sigma_\alpha$ the **signature**.

In particular, we include in the signature the identity relation $=_{(\alpha,\alpha,o)}$ for every type α. Furthermore, the signature must contain the following distinguished constants (the logical constants):

$\neg$	of type	(t, o)
$\vee$	of type	$(t, (t, o))$
$\rightarrow$	of type	$(t, (t, o))$
$\wedge$	of type	$(o, (o, o))$

The signature should particularly contain the standard truth constants $(\mathrm{T})_o$ and $(\mathrm{F})_o = \bot$, as well as the equivalent constants of type t, $(\mathrm{T})_t$ and $(\mathrm{F})_t$, which denote the universally valid and the unsatisfiable DRS respectively.

Remark 2.3 For every type α we have a countably infinite set $\mathcal{V}_\alpha$ of **variables** and call the set of all variables $\mathcal{V} = \bigcup_{\alpha \in \mathcal{T}} \mathcal{V}_\alpha$. Within the set $\mathcal{V}_e$ we have a distinguished

infinite set of variables $\mathcal{VD} \subset \mathcal{V}_e$. We usually denote variables of type e by X, Y, Z, variables of type d by U, V, W and variables of any other type by P, Q, R.

Note that $\wedge$ does not appear in standard DRT. Here, we use it to join conditions. We can thus write the body of a DRS as one single, possibly complex condition rather than as a set of conditions, as is common in standard DRT.

The following defines the set of well-formed formulae. This definition is a simple extension of the syntax definition of the typed intensional λ-calculus used by Montague, extended only by clauses to construct and to merge DRSes. Note that it also is an extension of Muskens 1994 in that it allows for a symmetric merging operator, the $\otimes$, which corresponds to dynamic conjunction of DPL (Groenendijk and Stockhof 1991). With respect to the definition of λ-DRT in Bos et al. 1994, we suggest a syntactic extension by the asymmetric merging operator, **;**, which is the merge operation used in Muskens 1994 and others; however, it is not more than a syntactic extension because **;** can be found to be a special case of $\otimes$.

The set of well-formed formulae will be defined in two steps here. The set of raw formulae gives the basic format of the λ-DRT syntax, only to be supplemented with an additional constraint to give the set of well-formed formulae.

Definition 2.4 (Raw Formulae) Given a distinguished set of variables of type e, $\mathcal{VD}$. For every type $\alpha \in \mathcal{T}$ we inductively define the **set of raw formulae** of type α, rf_α:

1. $\mathbf{A} \in \Sigma_\alpha \subset \mathrm{rf}_\alpha$. (**constants**)
2. $\mathbf{A} \in \mathcal{V}_\alpha \subset \mathrm{rf}_\alpha$. (**variables**)
3. If $\mathcal{X} \subset \mathcal{VD}$ and $\mathbf{B} \in \mathrm{rf}_o$, then $\mathbf{A} = \delta\mathcal{X}.\mathbf{B} \in \mathrm{rf}_t$. (construction of a **DRS**)
4. If $\mathbf{D}_1$ and $\mathbf{D}_2 \in \mathrm{rf}_t$, then $\mathbf{A} = \mathbf{D}_1 \otimes \mathbf{D}_2 \in \mathrm{rf}_t$. (**merge** of two DRSes)
5. If $\mathbf{D}_1$ and $\mathbf{D}_2 = \delta\mathcal{X}_2.\mathbf{B}_2 \in \mathrm{rf}_t$, where no $X \in \mathcal{X}_2$ occurs in $\mathbf{D}_1$, then $\mathbf{A} = \mathbf{D}_1; \mathbf{D}_2 \in \mathrm{rf}_t$. (**join** of two DRSes)
6. If $X \in \mathcal{V} \setminus \mathcal{VD}$ and $\mathbf{B} \in \mathrm{rf}_\beta$, then $\mathbf{A} = \lambda X.\mathbf{B} \in \mathrm{rf}_{(\alpha,\beta)}$. ($\lambda$-**abstraction**)
7. If $\mathbf{B}_2 \in \mathrm{rf}_\alpha$ and $\mathbf{B}_1 \in \mathrm{rf}_{(\alpha,\beta)}$, then $\mathbf{A} = \mathbf{B}_1(\mathbf{B}_2) \in \mathrm{rf}_\beta$. (**application**)
8. If $\mathbf{X} \in \mathrm{rf}_e$, then $\mathbf{A} =^\wedge \mathcal{X} \in \mathrm{rf}_d$. (**Up-operator**)
9. If $\mathbf{U} \in \mathrm{rf}_d$, then $\mathbf{A} =^\vee \mathcal{U} \in \mathrm{rf}_e$. (**Down-operator**)

Definition 2.5 (Sets of Discourse Referents of a Raw Formula A) The **set of discourse referents** of a raw formula $\mathbf{A}$, $\mathcal{DR}(\mathbf{A})$, is defined inductively in parallel to $\mathbf{A}$'s inductive definition. It collects all of $\mathbf{A}$'s discourse referents on all levels.

1./2. For $\mathbf{A} \in \Sigma_\alpha \cup \mathcal{V}_\alpha$, then $\mathcal{DR}(\mathbf{A}) = \emptyset$.
3. If $\mathbf{A} = \delta\mathcal{X}.\mathbf{B}$, then $\mathcal{DR}(\mathbf{A}) = \mathcal{X} \cup \mathcal{DR}(\mathbf{B})$.
4./5. If $\mathbf{A} = \mathbf{D}_1 \otimes \mathbf{D}_2$ or $\mathbf{A} = \mathbf{D}_1; \mathbf{D}_2$, then $\mathcal{DR}(\mathbf{A}) = \mathcal{DR}(\mathbf{D}_1) \cup \mathcal{DR}(\mathbf{D}_2)$.
6. If $\mathbf{A} = \lambda X.\mathbf{B}$, then $\mathcal{DR}(\mathbf{A}) = \mathcal{DR}(\mathbf{B})$.
7. If $\mathbf{A} = \mathbf{B}_1(\mathbf{B}_2)$, then $\mathcal{DR}(\mathbf{A}) = \mathcal{DR}(\mathbf{B}_1) \cup \mathcal{DR}(\mathbf{B}_2)$.
8./9. If $\mathbf{A} =^\wedge \mathcal{X}$ or $\mathbf{A} =^\vee \mathcal{U}$, then $\mathcal{DR}(\mathbf{A}) = \emptyset$.

Definition 2.6 (Well-Formed Formulae (wff))A raw formula $\mathbf{A}$ is called a **well-formed formula**, $\mathbf{A} \in \mathrm{wff}_\alpha$, if for every sub-expression $\mathbf{A}_1 \otimes \mathbf{A}_2$, $\mathbf{A}_1;\mathbf{A}_2$ and $\mathbf{A}_1(\mathbf{A}_2)$ we have that $\mathcal{DR}(\mathbf{A}_1) \cap \mathcal{DR}(\mathbf{A}_2) = \emptyset$.
The well-formed formulae of type t, wff_t, are called **discourse representation structures (DRSes)**.

Just a few remarks concerning these definitions.

Remark 2.7 The set of well-formed formulae will be the basis of the semantic definition below. Note that a wff obeys at least two syntactic restrictions, namely that its sets of λ- and δ-abstracted variables are disjoint (by 2.4), and that no variable

name is used more than once for δ-abstraction (by 2.6). This follows the intention that discourse referents should describe *unique* name referents and, incidentially, is in agreement with DRT's syntax itself.

Notation 2.8 Using the associativity and commutativity of the merge-operator $\otimes$, we define the operator $\bigotimes$ on sets of DRSes:

$\mathbf{D}_1 \otimes \mathbf{D}_2 \otimes \ldots \otimes \mathbf{D}_n = \bigotimes\{\mathbf{D}_1, \ldots, \mathbf{D}_n\}$

For $n = 1$ we have $\bigotimes\{\mathbf{D}_1\} = \mathbf{D}_1$, and for $n = 0$ we have $\bigotimes\{\} = (\mathrm{T})_t$.
Likewise we define $\bigwedge$ as an operator on a set of conditions, and (since ; is an associative though not commutative operator) $\bigcap$ as an operator on a list of DRSes.

We will use the convention that the group brackets (and) associate to the left, i.e. $\mathbf{ABC} = (\mathbf{AB})\mathbf{C}$

For legibility we shall in later sections use the application operator @ : $\mathrm{wff}_{(\alpha,\beta)} \times \mathrm{wff}_\alpha \longrightarrow \mathrm{wff}_\beta$ with $\mathbf{A@B} := (\mathbf{AB})$ for functional application, which, too, associates to the left.

Also, the successive λ-abstractions in $\lambda X^1.\lambda X^2.\ldots.\lambda X^n.\mathbf{A}\mathbf{E}^1\ldots\mathbf{E}^m$, which is an abbreviation of the full $(\lambda X^1(\lambda X^2\ldots(\lambda X^n.(\mathbf{A}\mathbf{E}^1)\ldots\mathbf{E}^m)\ldots)$, will be combined to give $\lambda X^1\ldots X^n.\mathbf{A}\mathbf{E}^1\ldots\mathbf{E}^m$. Furthermore, λ-abstraction binds more strongly than the merging operators.

Example 2.9 To illustrate the proposed notation for readers who are accustomed to Kamp's DRT, compare (1) and its notation in λ-DRT (2).

(1) $\lambda Q.$ $\boxed{\begin{array}{c} X \\ \mathsf{student}(X) \end{array}}$ $\otimes\, Q(X)$

(2) $\lambda Q. \otimes \{\delta\{X\}.\, \mathsf{student}(X), Q(X)\}$ with $\mathcal{DR} = [X]$.

It will be useful to be able to focus on some syntactical properties for the remainder of this work. Among these are firstly, the notion of a hierarchical ordering of DRSes in a complex DRS, secondly, a set of top level discourse referents, $\mathcal{TLDR}()$, defined to supplement the set of discourse referents by recursing also across the merging operators, and thirdly, the set of name contexts of $\mathbf{E}$ at $\mathbf{A}$, $\mathcal{NC}(\mathbf{E}, \mathbf{A})$. The latter is convenient in that it covers the accessibility relation: it is the set of discourse referents which are accessible at $\mathbf{A}$ in $\mathbf{E}$.

Definition 2.10 ((Direct) Sub-DRS)A DRS $\mathbf{D}_1$ is called **direct sub-DRS** of $\mathbf{D}_2$, $\mathbf{D}_1 \lessdot \mathbf{D}_2$, if $\mathbf{D}_2 = \bigotimes\{\mathbf{A}_i\}$ or $= \bigcap[\mathbf{A}_i]$, $i \geq 1$, and one of the $\mathbf{A}_i$ is of the form

$\delta\mathcal{X}.\neg\mathbf{B}_1 \wedge \mathbf{C}$, or
$\delta\mathcal{X}.(\mathbf{B}_1 \vee \mathbf{B}_2) \wedge \mathbf{C}$ or
$\delta\mathcal{X}.(\mathbf{B}_1 \rightarrow \mathbf{B}_2) \wedge \mathbf{C}$,

for some $\mathbf{C}$, and if some i we have $\mathbf{D}_1 = \mathbf{B}_i$.

A DRS $\mathbf{D}_1$ is called **sub-DRS** of $\mathbf{D}_2$, if $\mathbf{D}_1 \leq \mathbf{D}_2$, where the relation $\leq$ is the transitive and reflexive closure of $\lessdot$.

Remark 2.11 The arguments $\mathbf{A}_i$, $\mathbf{D} = \bigotimes\{\mathbf{A}_i\}$ or $\mathbf{D} = \bigcap[\mathbf{A}_i]$, of a merging operator are not sub-DRSes; they shall be called (well-formed) **sub-formulae** of $\mathbf{D}$.

Definition 2.12 (Top Level Discourse Referents) The **set of top level discourse referents** of $\mathbf{A}$, $\mathcal{TLDR}(\mathbf{A})$, is defined thus:

$\mathcal{TLDR}(\bigotimes\{A_i\}) = \bigcup_i \mathcal{X}_i$ for $\mathbf{A}_i = \delta\mathcal{X}_i.\mathbf{D}_i$ and $i \geq 1$.

Likewise for $\bigcap[\mathbf{A}_i]$.

We take on DRT's accessibility relation for discourse referents which says that a variable can be bound by discourse referents defined on a higher level or in the antecedent DRS if the variable occurs in a consequent DRS of an implication. Negation and disjunction, however, are barriers for accessibility. This relation is covered by the following.

Definition 2.13 (Name Contexts) Let the expression $\mathbf{A}_i$ occur in a complex expression $\mathbf{E}$. The set of all discourse referents of $\mathbf{E}$ which are accessible at the i-th occurrence of $\mathbf{A}$, called $\mathbf{A}_i$, in $\mathbf{E}$, is called the set of **name contexts**, $\mathcal{NC}(\mathbf{E}, \mathbf{A}_i)$, defined by:

- $\mathcal{NC}(\mathbf{E}, \mathbf{A}_i) = \mathcal{TLDR}(\mathbf{A}_i)$, if $\mathbf{E} = \mathbf{A}_i$.
- If $\mathbf{A}_i$ occurs in $\mathbf{B}_1$ where for some $\mathbf{C}_o$

  $\mathbf{E} = \delta\mathcal{X}.\neg\mathbf{B}_1 \wedge \mathbf{C}$
  $= \delta\mathcal{X}.(\mathbf{B}_1 \rightarrow \mathbf{D}) \wedge \mathbf{C}$
  $= \delta\mathcal{X}.(\mathbf{B}_2 \rightarrow \mathbf{B}_1) \wedge \mathbf{C}$
  $= \delta\mathcal{X}.(\mathbf{B}_1 \vee \mathbf{D}) \wedge \mathbf{C}$

  then $\mathcal{NC}(\mathbf{E}, \mathbf{A}_i) = \mathcal{X} \cup \mathcal{TLDR}(\mathbf{B}_2) \cup \mathcal{NC}(\mathbf{B}_1, \mathbf{A}_i)$.

Example 2.14 Given $\mathbf{E} = \delta\{X, Y\}.P(X) \wedge (\delta\{Z\}.\mathbf{R}(Z, X) \rightarrow \mathbf{A})$. Then we have $\mathcal{NC}(\mathbf{E}, \mathbf{A}) = \{X, Y, Z\}$, independently of $\mathbf{A}$.

We are now ready to define free and bound variables. Note that the definition for dynamically bound variables is rather complex for two reasons; firstly, it has to reflect the structure of DRSes, in particular DRT's accessiblity relation; thus the use of $\mathcal{NC}$. Secondly, we will find in section 3 that through β-reduction free variables can be caught by the δ-abstraction operator. For this reason, *dynamic boundedness*, which addresses the binding capacity of the δ will be defined across function application (cf. the second clause). *Functional boundedness*, however, is defined in a standard way.

Definition 2.15 (Bound Variables) An occurrence of the variable X in a λ-DRT-expression $\mathbf{A}$ is called **dyamically bound**, if at least one of the following conditions is met:

1. There exists an occurrence $\mathbf{B}_i \leq \mathbf{A}$, the occurrence of X is in $\mathbf{B}_i$ and $X \in \mathcal{NC}(\mathbf{A}, \mathbf{B}_i)$.
2. $\mathbf{A} = (\lambda P.\mathbf{E})\mathbf{B}$, and either of
   (a) X occurs in $\mathbf{B}$ and either X is dynamically bound in $\mathbf{B}$ or $X \in \mathcal{NC}(\mathbf{E}, P)$.
   (b) X occurs in $\mathbf{E}$ and either X is dynamically bound in $\mathbf{E}$ or X occurs in a DRS $\mathbf{B}$ such that $P \otimes \mathbf{B}$ or $P \rightarrow \mathbf{B}$ occurs in $\mathbf{E}$ and $X \in \mathcal{TLDR}(\mathbf{B})$.

An occurrence of the variable X in a λ-DRT-expression $\mathbf{A}$ is **functionally bound**, if $\mathbf{A} = \lambda X.\mathbf{D}$ and X occurs free in $\mathbf{D}$.

An occurrence of the variable X in a λ-DRT-expression $\mathbf{A}$ is **bound**, if it is either functionally or dynamically bound.

Definition 2.16 (Free Variables) An **occurrence** of the variable X in an expression of λ-DRT, $\mathbf{A}$, is called **free**, if it is not bound. A **variable** X **is free** in an expression $\mathbf{A}$, if at least one occurrence of X is free in $\mathbf{A}$.

The **set** $\mathcal{FV}(\mathbf{A})$ **of free variables** of $\mathbf{A}$ is defined by

$\mathcal{FV}(\mathbf{A}) = \{X, \mid X \text{ occurs free in } \mathbf{A}\}$

Example 2.17 In the following, P is functionally bound. X is dynamically bound in $\mathbf{A}$ but free in $\mathbf{C}$; likewise, Y is dynamically bound in $\mathbf{A}$ and free in $\mathbf{B}$:

$$\mathbf{A} = \mathbf{B}(\mathbf{C}) = (\lambda P.\ \boxed{\begin{array}{c} X \\ \ldots Y \ldots \end{array}}\ \otimes \mathrm{P})\ @\ \boxed{\begin{array}{c} Y \\ \ldots X \ldots \end{array}}$$

Since in classical λ-calculus variable capture is something to be ruled out, it is not surprising that the fact that δ-abstractions are able to capture free variables during β-reduction results in properties seem very strange from the classical point of view. In the due course of this paper, we will encounter a few more such unusual properties[3]. We find one of these oddities here: variable capture may lead to ambiguous boundedness classification of a single variable occurrence in the language defined thus far. Consider the following expression:

$$\mathbf{A}(\mathbf{B}) = \lambda P.\ \boxed{\begin{array}{c} X \\ \boxed{Y}\ \rightarrow P \\ P \vee B \end{array}}\ @\ \boxed{\ldots Y \ldots}$$

During β-reduction the argument will be duplicated and the two occurrences of the variable Y will be of different status: in the consequence of the implication Y will be bound by the discourse referent in the antecedent in the implication, while in the disjunction, Y will be free. What then is the status of the occurrence of Y in the argument of the unreduced expression?

We may leave the status of this occurrence of Y open, but we will eventually run into problems when we are to define an interpretation function such that the β-reduction is correct. In the unreduced expression the Y in the argument will be assigned only one value, whereas in the reduced expression we have two different Ys with two possibly different values.

To avoid such problems we shall concentrate on those expressions which do not lead to such problematic situations through β-reduction (that is any number of reduction steps), calling them *sensible expressions*. Let us coin this notion in a more precise way:

Definition 2.18 (Sensible Expressions)A well-formed formula $\mathbf{A}$ is called **sensible**, if

- either $\mathbf{A}$ is of base type and in any reduction step there is no boundedness ambiguity for any occurrence of any variable
- or $\mathbf{A}_{(\alpha,\beta)}$ is of functional type and $\mathbf{A}(\mathbf{B}_\alpha)$ is sensible, if $\mathbf{B}$ is sensible.

Intuitively, such situations — in which the same object is both free and bound — do not make sense both logically and linguistically. However, it turns out to be difficult to specify the syntactical restrictions which are implied by the above definition for sensible expressions. We look for the syntactical properties of the largest set of well-formed formulae such that each member is sensible, or at least of a subset of this such that the set is expressive enough for linguistic purposes. One solution, proposed by Reinhard Muskens, is to restrict well formed formulae to linear terms. However, his representation of the generalised conjunction in Muskens 1994

(3) $\lambda R_1 \ldots \lambda R_n . \lambda X_1 \ldots \lambda X_m . (R_1(X_1) \ldots R_1(X_m); \ldots ; R_n(X_1) \ldots R_n(X_m))$

3. Such as, for example, the need of restricting the denotation of the argument at functional application (cf. definition 5.7) and the need to access the *name* of a δ-abstracted variable in a denotation (cf. remark 5.13).

may (in theory) lead to a problematic situation like the above, if we choose $n = 2$, $m = 1$ and apply (3) to $\lambda P.(\delta\{Y\}.\mathrm{T} \rightarrow P)$ and $\lambda P.(\delta\{Y\}.\mathrm{T} \vee P)$ for the two Rs and $\delta\{\}.Q(Y)$ for the single X. We note that for linguistic purposes (3) will not be used with arguments of this form, so either we use a different representation or we introduce a constraint on the arguments.

The syntactical specification of sensible expression shall not be a matter of this paper. We shall be content with the conjecture, so far only based on intuition, that for linguistic purposes we can restrict ourselves to sensible expressions.

3 Substitution and reduction

As the basis for the syntactic reduction rules to be used for the semantic processing in λ-DRT let us first define substitution. To avoid the ambiguities mentioned at the end of last section, we will only consider sensible expressions (cf. end of last section) in this section. The definition of substitution presupposes that of substitutability.

Definition 3.1 (Substitutability)A sensible expression $\mathbf{B}$ is **substitutable** for Y in a sensible expression $\mathbf{A}$, if and only if:

If $X \in \mathcal{FV}(\mathbf{B})$, then Y is not free in the sub-expression of $\mathbf{A}$ which is of the form $\lambda X.\mathbf{C}$.

This definition rules out variable capture for functionally bound variables, just as in standard λ-calculus; however, as already mentioned, variable capture through the δ-abstraction operator shall be permitted. We will see that it is by variable capture that at semantic processing we will be able to establish anaphoric bindings within a sentence or phrase or between two sentences (cf. examples (4) and (6) in section 4).

Definition 3.2 (Substitution) The substitution of $\mathbf{B}_\alpha$ for free occurences of (variable) Y_α in $\mathbf{A}$, written $[\mathbf{B}/Y]\mathbf{A}$, where $\mathbf{B}$ is substitutable for Y in $\mathbf{A}$, is defined inductively on the structure of $\mathbf{A}$.

1. If $a \in \Sigma_\sigma$, then $[\mathbf{B}/Y]a = a$.
2. If $\mathbf{A} \in \mathcal{V}_\sigma$, then $[\mathbf{B}/Y]\mathbf{A} = \mathbf{B}$, if $\mathbf{A} = Y$ and $[\mathbf{B}/Y]\mathbf{A} = X$, if $\mathbf{A} = X \neq Y$.
3. If $\mathbf{A} = \delta\mathcal{X}.\mathbf{C}$, then $[\mathbf{B}/Y]\mathbf{A} = \delta\mathcal{X}.([\mathbf{B}/Y]\mathbf{C})$.
4. If $\mathbf{A} = \mathbf{D}_1 \otimes \mathbf{D}_2$, then $[\mathbf{B}/Y]\mathbf{A} = [\mathbf{B}/Y]\mathbf{D}_1 \otimes [\mathbf{B}/Y]\mathbf{D}_2$
5. If $\mathbf{A} = \mathbf{D}_1;\mathbf{D}_2$, then $[\mathbf{B}/Y]\mathbf{A} = [\mathbf{B}/Y]\mathbf{D}_1;[\mathbf{B}/Y]\mathbf{D}_2$.
6. If $\mathbf{A} = \lambda X.\mathbf{D}$, then $[\mathbf{B}/Y]\mathbf{A} = \lambda X.([\mathbf{B}/Y])\mathbf{D}$
7. If $\mathbf{A} = \mathbf{C}(\mathbf{D})$, then $[\mathbf{B}/Y]\mathbf{A} = [\mathbf{B}/Y]\mathbf{C}([\mathbf{B}/Y]\mathbf{D})$.
8. If $\mathbf{A} =^\wedge \mathbf{X}$, then $[\mathbf{B}/Y]\mathbf{A} =^\wedge \mathbf{X}$.
9. If $\mathbf{A} =^\vee \mathbf{U}$, then $[\mathbf{B}/Y]\mathbf{A} =^\vee [\mathbf{B}/Y]\mathbf{U}$.

In short, this definition is quite conservative in that simply finds all variables recursively and replaces those variables that are a member of the domain of the substitution, except that the $^\wedge$-operator acts as a barrier for substitution here.

We are now ready to give the syntactic reduction rules, the first three of which are just like in the λ-calculus. Again, the expressions to be reduced are assumed to be sensible by way of the above definition.

Definition 3.3 (α-Conversion)A λ-DRT-expression $\mathbf{B}$ results from $\mathbf{A}$ in an α-conversion, denoted ($\mathbf{A} \longrightarrow_\alpha \mathbf{B}$), by substituting a sub-expression ($\lambda X.\mathbf{C}$) in $\mathbf{A}$ by ($\lambda Y.[Y/X]\mathbf{C}$) where $Y \notin \mathcal{FV}(\mathbf{C})$.

Definition 3.4 (β**-Reduction**)A λ-DRT-expression $\mathbf{B}$ results from $\mathbf{A}$ in a β-reduction, denoted ($\mathbf{A} \longrightarrow_\beta \mathbf{B}$), by substituting a sub-expression $(\lambda X.\mathbf{D})\mathbf{C}$ in $\mathbf{A}$ by $[\mathbf{C}/X]\mathbf{D}$, if $\mathbf{C}$ is substitutable for X in $\mathbf{D}$.

Definition 3.5 (η**-Reduction**)If $X \notin \mathcal{FV}(\mathbf{C})$, then the λ-DRT-expression $\mathbf{B}$ results from $\mathbf{A}$ in an η-reduction, denoted ($\mathbf{A} \longrightarrow_\eta \mathbf{B}$), by substituting a subexpression $\lambda X.\mathbf{C}X$ in $\mathbf{A}$ by $\mathbf{C}$.

Additionally, we have a reduction rule each for the dynamic operators, the merge operators.

Definition 3.6 (δ**-Reduction**)The merge of two DRSes $\mathbf{A}_1 = \delta\mathcal{X}.\mathbf{C}_1$ and $\mathbf{A}_2 = \delta\mathcal{Y}.\mathbf{C}_2$ reduces to a DRS as follows:

$$\delta\mathcal{X}.\mathbf{C}_1 \otimes \delta\mathcal{Y}.\mathbf{C}_2 \longrightarrow_\delta \delta(\mathcal{X} \cup \mathcal{Y}).\mathbf{C}_1 \wedge \mathbf{C}_2$$

and

$$\delta\mathcal{X}.\mathbf{C}_1 \,;\, \delta\mathcal{Y}.\mathbf{C}_2 \longrightarrow_\delta \delta(\mathcal{X} \cup \mathcal{Y}).\mathbf{C}_1 \wedge \mathbf{C}_2$$

Definition 3.7 (μ**-Reduction**)Adjacent $^\vee$- and $^\wedge$-operators cancel:[4]

$$^\vee{}^\wedge\mathbf{X} \longrightarrow_\mu \mathbf{X}.$$

In connection with the definitions of reductions we have the following standard definitions and notations:

Notation 3.8 We write a sequence of θ-reductions $\mathbf{A} \longrightarrow_\theta \ldots \longrightarrow_\theta \mathbf{B}$ as $\mathbf{A} \longrightarrow_\theta^* \mathbf{B}$ (for $\theta = \alpha, \beta, \eta, \delta, \mu$).

Definition 3.9 (Redex) The subexpression of $\mathbf{A}$ to be reduced by one of the above reduction rules is called the β**-/**δ**-/etc.-redex**, or, more generally, a **redex**.

Definition 3.10 (Normal Form)A formula which does not contain a redex and therefore cannot be reduced by a β-, η-, δ or μ-reduction, is called **normal form**.

Definition 3.11 ($\beta\eta\delta\mu$**-Equality**)Two expressions $\mathbf{A}$ and $\mathbf{B}$ are called $\beta\eta\delta\mu$**-equal**, $\mathbf{A} =_{\beta\eta\delta\mu} \mathbf{B}$, iff there exists a sequence of β-,η-, δ- and μ-conversions, $\mathbf{A} \longleftrightarrow_\theta \ldots \longleftrightarrow_\vartheta \mathbf{B}$, where θ and ϑ are β, η, δ or μ.

Definition 3.6, together with definition 2.6 of well-formed formulae, defines ; as merely a specialization of $\otimes$; the only difference between the two is in definition 2.6, where the pre-conditions on ; constitute a restriction of the pre-conditions of $\otimes$[5]. So, the operator ; could in fact be left out safely without loss of expressibility. We will, however, use it for the concatenation of sentences.

4 A Fragment for English

We are now able to formulate λ-DRT representations for words of a very small English fragment and show how semantic construction works *compositionally* from the lexical entries in λ-DRT by using the tools of reduction rules defined in the precious section.

4. Just as in Montague's PTQ and Dynamic Montague Grammar.
5. The two operators will be interpreted differently in section 5. This, however, only reflects two different ways of looking at the same thing (cf. discussion of static vs. dynamic semantics in remark 5.15). What matters here is that definition 3.6 requires them to take the same reduction behaviour.

For simplicity the fragment does not consider the temporal relations of verbs in different tenses. However, it uses an event variable for the representation of verbs, similar to Davidson's account, so that modificators like `in-the-park` can be represented.

The present account does not take into account the construction of indices via alfa-conditions as proposed in the Verbmobil project Bos et al. 1994. Instead, it presupposes that the coindexation of anaphors with their antecedents has been done in the syntactical analysis and is passed on to the semantic construction. In this respect, the lexical entries really are patterns and well-formedness of the representations is guaranteed (i.e. no duplicates among the discourse referents), if every definite NP — or more generally: every antecedent — uses a new index.

The indices will be written as super- and subscripts: a superscript denotes the index of a word that acts as an antecedent, a subscript is the index of an anaphora. This fragment uses the letters i, j, k, etc. as indices for NPs, and a, b, c, etc. as indices for events.

The fragment also uses the standard functional application for simplicity instead of the generalised functional application of Bos et al. 1994; thus the representations of transitive verbs look different to those given there.

As a notational convention for the following, the variables u and v are of type d, variables x, y, x_i and x_j are of type e.

A fragment of the English Language

lexical entry	representation in λ-DRT	type of expression
a^i	$\lambda P.\lambda Q.(\delta\{x_i\}.\mathrm{T} \otimes P(^\wedge x_i) \otimes Q(^\wedge x_i))$	$((d,t),((d,t),t))$
every	$\lambda P.\lambda Q.(\delta\{\}.(\delta\{x_i\}.\mathrm{T} \otimes P(^\wedge x_i)) \rightarrow Q(^\wedge x_i))$	$((d,t),((d,t),t))$
the^i_j	$\lambda P.\lambda Q.((\delta\{x_i\}.x_i = x_j) \otimes P(^\wedge x_i) \otimes Q(^\wedge x_i))$	$((d,t),((d,t),t))$
his^i_j	$\lambda P.\lambda Q.((\delta\{x_i\}.\mathsf{poss}(x_j, x_i)) \otimes P(^\wedge x_i) \otimes Q(^\wedge x_i))$	$((d,t),((d,t),t))$
john^i	$\lambda P.(\delta\{x_i\}.x_i = j^* \otimes P(^\wedge x_i))$	$((d,t),t)$
man	$\lambda u.\delta\{\}.\mathsf{man}(^\vee u)$	(d,t)
he_i	$\lambda P.P(^\wedge x_i)$	$((d,t),t)$
walk^a	$\lambda u.\delta\{e_a\}.\mathsf{walk}(e_a, ^\vee u)$	(d,t)
whistle^a	$\lambda u.\delta\{e_a\}.\mathsf{whistle}(e_a, ^\vee u)$	(d,t)
be^a	$\lambda P.\lambda u.(P(\lambda v.{}^\vee u =^\vee v))$	$(((d,t),t),(d,t))$
black	$\lambda P.\lambda u.(\delta\{\}.\mathsf{black}(^\vee u) \otimes P(u))$	$((d,t),(d,t))$
in-the-park_a	$\lambda P.\lambda u.(P(u) \otimes \delta\{\}.\mathsf{loc}(e_a) = \mathsf{in\text{-}park})$	$((d,t),(d,t))$
.	$\lambda P_t.\lambda Q_t.(P;Q)$	$(t,(t,t))$

In the following examples note especially where free variables are caught in the course of semantic construction in examples (4) and (6). In (4), the free variable e_a in the representation of `in the park` is caught by the discourse referent in the representation of `walk`. The same happens when the representation of `He whistles` in example (5) is joined with the representation of (4) to form a text in (6).

The following presentations use $\widehat{=}$ for the equivalence to the graphical notation of an expression.

(4) (a) `John walks in the park.`

(b) John^j @ (in-the-park_a @ walk^a)

1. John^j @ ($\lambda P.\lambda u.(P(u) \otimes \delta\{\}.\text{loc}(e_a) = \text{in-park})$@ ($\lambda y.\delta\{e_a\}.\text{walk}(e_a, y)$))
   $\longrightarrow_\beta \text{John}^j$ @ ($\lambda u.(((\lambda v.\delta\{e_a\}.\text{walk}(e_a, {}^\vee v))@(u)) \otimes \delta\{\}.\text{loc}(e_a) = \text{in-park})$)
   $\longrightarrow_\beta \text{John}^j$ @ ($\lambda u.((\delta\{e_a\}.\text{walk}(e_a, {}^\vee u)) \otimes \delta\{\}.\text{loc}(e_a) = \text{in-park})$)

   $\widehat{=}$ John^j @ ($\lambda u.$ $\boxed{\begin{array}{c} e_a \\ walk(e_a, {}^\vee u) \end{array}}$ $\otimes$ $\boxed{loc(e_a) = in\text{-}park}$)

2. $= (\lambda P.(\delta\{x_j\}.\, x_j = j^* \otimes P(\hat{}x_j)))$
   @ $(\lambda u.((\delta\{e_a\}.\text{walk}(e_a, {}^\vee u)) \otimes \delta\{\}.\text{loc}(e_a) = \text{in-park}))$
   $\longrightarrow^*_\beta \delta\{x_j\}.\, x_j = j^* \otimes \delta\{e_a\}.\text{walk}(e_a, x_j)) \otimes \delta\{\}.\text{loc}(e_a) = \text{in-park}$
   $\longrightarrow_\delta \delta\{x_j, e_a\}.x_j = j^* \wedge \text{walk}(e_a, x_j) \wedge \text{loc}(e_a) = \text{in-park}$

   $\widehat{=}$ $\boxed{\begin{array}{c} x_j\ e_a \\ x_j = j^* \\ walk(e_a, x_j) \\ lox(e_a) = in\text{-}park \end{array}}$

(5) (a) `He whistles.`

(b) he_j @ whistle^b

1. $= \lambda P.P(\hat{}x_j)$ @ $\lambda u.\delta\{e_b\}.\text{whistle}(e_b, {}^\vee u)$
   $\longrightarrow_\beta (\lambda u.\delta\{e_b\}.\text{whistle}(e_b, {}^\vee u))(\hat{}x_j)$
   $\longrightarrow^*_\beta \delta\{e_b\}.\text{whistle}(e_b, x_j)$

   $\widehat{=}$ $\boxed{\begin{array}{c} e_b \\ whistle(e_b, x_j) \end{array}}$

(6) (a) `John walks in the park. He whistles.`

(b) . @ (John^j @ (in-the-park_a @ walk^a)) @ (he_j @ whistle^b)

1. $\lambda P.\lambda Q(P\,;Q)@(\delta\{x_j, e_a\}.x_j = j^* \wedge \text{walk}(e_a, x_j) \wedge \text{loc}(e_a) = \text{in-park})$
   $@(\delta\{e_b\}.\text{whistle}(e_b, x_j))$

   $\longrightarrow^*_\beta \delta\{x_j, e_a\}.x_j = j^* \wedge \text{walk}(e_a, x_j) \wedge \text{loc}(e_a) = \text{in-park}\,;\, \delta\{e_b\}.\text{whistle}(e_b, x_j)$

   $\widehat{=}$ $\boxed{\begin{array}{c} x_j\ e_a \\ x_j = j^* \\ walk(e_a, x_j) \\ loc(e_a) = in\text{-}park \end{array}}$; $\boxed{\begin{array}{c} e_b \\ whistle(e_b, x_j) \end{array}}$

   $\longrightarrow_\delta \delta\{x_j, e_a, e_b\}.x_j = j^* \wedge \text{walk}(e_a, x_j) \wedge \text{loc}(e_a) = \text{in-park} \wedge \text{whistle}(e_b, x_j)$

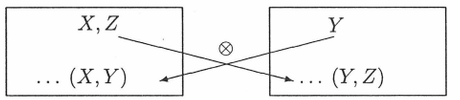
$\widehat{=}$ $\boxed{\begin{array}{c} x_j\ e_a\ e_b \\ x_j = j^* \\ walk(e_a, x_j) \\ loc(e_a) = in\text{-}park \\ whistle(e_b, x_j) \end{array}}$

5 Semantics for λ-DRT

The examples of the previous section showed that the λ-DRT language defined thus far is sufficient to compositionally construct representations for natural language texts. Now, we aim to give a direct denotational semantics for λ-DRT which should also be *compositional on denotations*. The central question for the semantics was the format of the denotations of a discourse representation structure, in particular, what amount of information the denotation of a DRS must contain in order to allow for compositionality. To trace the line of motivation, let us first look at two features which were the starting point for the semantics definition.

Firstly, consider the following situation arising due to the commutativity of the ⊗-operator. A semantically problematic situation occurs whenever discourse referents in one argument of the ⊗ bind variables in the other argument, as in the following example:

$\mathbf{A} \otimes \mathbf{B}$ =



With denotational compositionality, the subexpressions **A** and **B** must be interpreted independently first, in order to contribute to the denotation of the whole expression later. In the interpretation of **A**, however, the variable Y appears to be a free variable, and, at merging, turns out to be a bound variable. Considering that bound and free variables describe two quite different logical properties, this is a semantically very interesting situation to model.

The semantics definition presented below suggests a solution by using partial assignment functions. Just consider how total functions pose a problem for this situation: If **A** and **B** above were interpreted with respect to a total assignment function s, the two denotations would differ at the value for Z: **A**'s denotation may assign a different value to Z than s had previously done, whereas all denotations of **B** take on the value that s had assigned to Z. Thus, the two denotations cannot be merged appropriately when working in this way.

Secondly, we aimed at defining the ⊗-operator simply by means of set intersection. For this the denotations must in a way foresee the occurrence of all variables in the expressions they are about to be merged with. In case of the above example, the denotation of **B** must not only contain information about Y and Z, the variables occurring in **B**, but also about X, the variable occurring in **A**.

These considerations give rise to the following definition of a continuation of an assignment function:

Definition 5.1 (Continuation of an Assignment Function) Let s and t be partial assignment functions. We call t a **continuation of s by $\mathcal{X}$**, denoted $s[\mathcal{X}]t$, iff it is defined at least on $\mathbf{Dom}(s) \cup \mathcal{X}$ and we have $t(X) = s(X)$ for all $X \in \mathbf{Dom}(s) - \mathcal{X}$.

A continuation $s[\mathcal{X}]t$ can thus alter the value for all $X \in \mathbf{Dom}(s) \cap \mathcal{X}$ and adds values for all other $X \in \mathcal{X}$. t may also be defined on other variables $Y \notin \mathbf{Dom}(s) \cup \mathcal{X}$; these Y are assigned any values. Because the cardinality of t's domain can not be pre-determined, we speak of a *variable* continuation.

We are now ready to define the semantics itself, commencing with the definition of the carrier sets.

Definition 5.2 (Carrier Set) Let $\mathcal{U}$ be a set of individuals, and let $\mathcal{B}_e$ be the set of partial assignments for individual variables, $\mathcal{B}_e = \mathcal{F}_p(\mathcal{VD}; \mathcal{D}_e)$. We will call the members of $\mathcal{B}_e$ **states**. We define the following typed family of sets $\mathcal{D} = \{\mathcal{D}_\alpha\}$ the **carrier set** of λ-DRT:

1. $\mathcal{D}_e = \mathcal{U}$
2. $\mathcal{D}_d = \mathcal{F}_p(\mathcal{B}_e; \mathcal{D}_e)$
3. $\mathcal{D}_o = \{\mathrm{T}, \mathrm{F}\}$
4. $\mathcal{D}_t \subseteq \wp(\mathcal{V}_e) \times \wp(\mathcal{B}_e)$
5. $\mathcal{D}_{(\alpha,\beta)} \subseteq \mathcal{F}(\mathcal{D}_\alpha, \mathcal{D}_\beta)$

A λ-DRT expression $\mathbf{A}$ will be interpreted with respect to a model M and two assignment functions, s and φ. The interpretation function will be called $\mathcal{I}^s_{M,\varphi}$, or rather $\mathcal{I}^s_\varphi$ for short. The subscript assignment function, denoted φ or ψ, is responsible for the λ-bound variables, the superscript assignment function s, or t or r, will assign a value to δ-bound variables.

For legibility we define functions to retrieve the two components of a DRS denotation.

Definition 5.3 (Projection Functions for DRS Denotations) Given the denotation A of a DRS, we define the following two **projection functions for DRS denotations**:

$$\begin{array}{lll} \mathcal{VAR} := \mathcal{D}_t \longrightarrow \wp(\mathcal{V}) & \text{with} & \mathcal{VAR}(\langle \mathcal{X}, \mathcal{M} \rangle) = \mathcal{X} \\ \mathcal{FUN} := \mathcal{D}_t \longrightarrow \wp(\mathcal{B}_e) & \text{with} & \mathcal{FUN}(\langle \mathcal{X}, \mathcal{M} \rangle) = \mathcal{M} \end{array}$$

Definition 5.4 (Components of a DRS's Denotation) Likewise, given a DRS $\mathbf{D}$, we define two functions which return the respective **component of a DRS's denotation**.

$$\begin{array}{lll} \mathcal{DV}^s_\varphi := \wp(\mathrm{wff}_t) \longrightarrow \wp(\mathcal{V}) & \text{with} & \mathcal{DV}^s_\varphi(\mathbf{D}) = \mathcal{VAR}(\mathcal{I}^s_\varphi(\mathbf{D})) \\ \mathcal{IF}^s_\varphi := \wp(\mathrm{wff}_t) \longrightarrow \wp(\mathcal{B}_e) & \text{with} & \mathcal{IF}^s_\varphi(\mathbf{D}) = \mathcal{FUN}(\mathcal{I}^s_\varphi(\mathbf{D})) \end{array}$$

Since s is responsible for δ-bound variables, it is restricted to the assignment of individual variables only. However, φ, being responsible for λ-bound variables, may assign values to variables of every type, indeed to variables of type t. The values for these variables themselves contain assignments to individual variables — recall that $\mathcal{D}_t \subseteq \wp(\mathcal{V}_e) \times \wp(\mathcal{B}_e)$ — which should, of course, not be in conflict with the assignment of s and φ to any variables of type e. Take, for example,

(7) $\boxed{\begin{array}{c} X \\ \mathsf{walk}(X) \end{array}} \otimes \mathrm{P}$

with $s(X) = \mathsf{john}$ and $\varphi(P) = \langle \ldots, \{\ldots, \{X \rightarrow \mathsf{peter}, \ldots\}, \ldots\}\rangle$. What then does the variable (name), the discourse referent X, denote — john or peter? Inconsistencies like these are excluded by the following definition.

Definition 5.5 (Consistency Relation on s and φ) Let s and φ be two assignment functions for variables, i.e. $s, \varphi : \mathcal{V}_\alpha \longrightarrow \mathcal{D}_\alpha$. Then the two functions are mutually **consistent**, $\kappa(s, \varphi)$, iff[6]:

$$\forall Z_t \in \mathbf{Dom}(\varphi) : \ \forall r \in \mathcal{FUN}((s + \varphi)(Z)) : r \parallel (s + \varphi)$$

We will simplify the representation of the partial assignment functions by using the special symbol $\perp$ and implicitely extending the definition of any partial function t by $t(a) := \perp$, if $a \notin \mathbf{Dom}(t)$.

Definition 5.6 (Model Structure) Let

$\mathcal{D}$ be a carrier and

F be a typed function $F : \Sigma \longrightarrow \mathcal{D}$

We call $M = \langle \mathcal{D}, F \rangle$ a **model structure**, if all operators in F are strict, and in particular

- $F(\neg)@A = \mathrm{T}$, if $\mathcal{FUN}(A) = \emptyset$, else $= \mathrm{F}$.
- $F(\vee)@A@B = \mathrm{T}$, if $\mathcal{FUN}(A) \neq \emptyset$ or $\mathcal{FUN}(B) \neq \emptyset$, else $= \mathrm{F}$.
- $F(\rightarrow)@A@B = \mathrm{T}$, if for all $t \in \mathcal{FUN}(A)$ we have $\exists r \in \mathcal{FUN}(B) : t[\mathcal{VAR}(B)]r$, else $= \mathrm{F}$.
- $F(\wedge)@a@b = \mathrm{T}$, if $a = \mathrm{T}$ and $b = \mathrm{T}$, $= \perp$, if at least one of $a = \perp$ or $b = \perp$ $= \mathrm{F}$, else.

Definition 5.7 (Denotation $\mathcal{I}^s_\varphi(\mathbf{A})$) The interpretation of a well-formed formula $\mathbf{A}$ with respect to M and the assignment function s and φ with $\kappa(s, \varphi)$, the **denotation $\mathcal{I}^s_\varphi(\mathbf{A})$**, is defined inductively as follows:

1. $\mathcal{I}^s_\varphi(c) = F(c)$ for any $c \in \Sigma$
2. $\mathcal{I}^s_\varphi(V) = \varphi(V)$ if $V \in \mathbf{Dom}(\varphi)$ $= s(V)$ if $V \in \mathbf{Dom}(s)$ $= \perp$ else.
3. $\mathcal{I}^s_\varphi(\delta\mathcal{X}.\mathbf{B}) = \langle \mathcal{X}, \{t \mid s[\mathcal{X}]t \ , \ \kappa(t, \varphi) \ , \ \mathcal{I}^t_\varphi(\mathbf{B}) = \mathrm{T}\}\rangle$
4. $\mathcal{I}^s_\varphi(\mathbf{A} \otimes \mathbf{B}) = \langle \mathcal{DV}^s_\varphi(A) \cup \mathcal{DV}^s_\varphi(B) \ , \ \mathcal{IF}^s_\varphi(A) \cap \mathcal{IF}^s_\varphi(B) \rangle$
5. $\mathcal{I}^s_\varphi(\mathbf{A};\mathbf{B}) = \langle \mathcal{DV}^s_\varphi(A) \cup \mathcal{DV}^s_\varphi(B) \ , \ \{r \mid r \in \mathcal{IF}^s_\varphi(B) \text{ und } \exists t \in \mathcal{IF}^s_\varphi(A) : r|_{-\mathcal{DV}^s_\varphi(B)} = t\}\rangle$[7]
6. $\mathcal{I}^s_\varphi(\lambda X.\mathbf{D}) = \Lambda A.\mathcal{I}^s_{\varphi,[\varepsilon(A)/X]}(\mathbf{D})$ where $\varepsilon(A_t) = \{\langle \mathcal{VAR}(A), r\rangle \mid r \in \mathcal{FUN}(A_t) \text{ and } r \parallel (s + \varphi)\}$ $\varepsilon(A_\sigma) = A_\sigma$, if $\sigma \neq t$.
7. $\mathcal{I}^s_\varphi(\mathbf{A}(\mathbf{B})) = \mathcal{I}^s_\varphi(\mathbf{A})@\mathcal{I}^s_\varphi(\mathbf{B})$
8. $\mathcal{I}^s_\varphi(^\wedge X) = \Lambda t.t(X)$
9. $\mathcal{I}^s_\varphi(^\vee U) = U(s)$

In the remainder of this paper we shall single out a distinguished set of well-formed formulae, called contextually closed formulae, defined thus:

Definition 5.8 (Contextually Closed)A well-formed formula $\mathbf{A}$ is called **contextually closed**, denoted by $\mathbf{A} \in \mathrm{ccf}$, if for every occurrence of a not λ-abstracted variable X it holds that one of:

6. We say that two partial functions t_1 and t_2 agree, $t_1 \parallel t_2$, if they assign the same value to all members of $\mathbf{Dom}(t_1) \cap \mathbf{Dom}(t_2)$.
Furthermore, $s + \varphi$ denotes the (simple) addition of the two partial functions s and φ; note that their domains is disjunct (cf. definition of well-formed formulae, 2.6).

- The occurrence is in $\mathbf{A}$ of $\delta\mathcal{X}.\mathbf{A}$, regardless of whether $X \in \mathcal{X}$ or not. We call this occurrence of the variable X **dominated** by the δ-operator.
- If $\mathbf{A} \longrightarrow_\mu \mathbf{A}'$ and $\mathbf{A}'$ irreducible with respect to $\longrightarrow_\mu$, then if $X \in \mathcal{V}_d$, then X is not in the scope of a $^\vee$ in A', or if $X \in \mathcal{V}_e$ then it is in scope of a $^\wedge$ in A'.

The motivation of a contextually closed formula is that its definition describes the format of the representations of linguistic units, i.e. words, phrases, sentences, texts. These units may also be considered as (linguistic) context building blocks, hence the name.

We thus note that for linguistic purposes it is sufficient to concentrate on contextually closed expressions[8]. Furthermore we require all expressions to be sensible, as defined in 2.18. It turns out that to prove the reductions correct, we *need* these restrictions. We will therefore introduce

Definition 5.9 (Safe Expressions) We will call sensible, contextually closed expressions **A safe** and write $\mathbf{A} \in$ sf.

Remark 5.10 An important property of contextually closed expressions is that no undefined values may leap in, despite the use of partial assignment functions. This property can be verified by looking at the three contexts in which a variable can occur: firstly, a variable dominated by a DRS will be assigned a value on the way of making the conditions of the DRS true. Secondly, if a variable is dominated by a single $^\wedge$-operator, then the interpretation will abstract over the look-up of the variable with respect to one particular assignment function. Thirdly, a variable may be λ-abstracted; then it will eventually be assigned the value of the argument.

On the basis of what has just been said we will now define the notion of a model for λ-DRT.

Definition 5.11 (λ-DRT Model) Let $M = \langle \mathcal{D}, F \rangle$ be a model structure such that $\mathcal{I}^s_\varphi$ is defined for all safe expressions $\mathbf{A} \in$ sf, for all s and φ with $\kappa(s, \varphi)$. Then we call M a **λ-DRT model**.

Notation 5.12 We define semantic operators $\oplus$ and $;;$ for the merging operators $\otimes$ and $;$ respectively. $\oplus$ is defined by $\oplus : \mathcal{D}_t \times \mathcal{D}_t \longrightarrow \mathcal{D}_t$ with $A \oplus B := \langle\, \mathcal{VAR}(A) \cup \mathcal{VAR}(B)\,,\ \mathcal{FUN}(A) \cap \mathcal{FUN}(B)\,\rangle$, and $;;$ is defined accordingly.

Commenting on the given definitions first note that the semantics is indeed defined compositionally on the level of denotations.

Also note that those parts of definition 5.7 which originate from the λ-calculus are in principal just like their counterparts in pure typed λ-calculus. However, in clauses (2) and (6) the pure 'functional character' mixes with aspects of the dynamic part of λ-DRT: the interpretation of variables considers two sources for the values, in particular giving special attention to variables which are (potentially) δ-abstracted. The interpretation of an abstraction expression demands a filter for the argument to come to make sure that no inconsistency may creep in through the argument.

Remark 5.13 Let us take a closer look at this filter, or rather, at the consistency relation κ which lies behind it. Example (7) above illustrated the need for κ. Note that the need for this relation, too, is a feature of the dynamic part of λ-DRT, since values of DRSes contain assignments to variables themselves (and these assignments

8. This can be verified by checking that all entries of the lexicon indeed have this property (cf. section 4, and we conjecture that any lexicon can be written using such expressions), and that the reductions are closed with respect to the set of these expressions, the proof of which is easy.

can clash with other information). The important point is that these variables can be addressed by name in the denotation of the DRS. Whereas the availability of variable assignments within the assignment of a DRS variable opens the door for inconsistencies to arise, it is the accessibility of the variable *name* in the denotation of a DRS allows for a way to guarantee consistency after all. Indeed, the presence of names in the semantic object, which makes them globally accessible, is the key to discourse semantics. This feature is in contrast with pure λ-calculus, or classical logic in general, where variables are merely place holders and names can be abstracted away (cf. Bruijn 1972)— and again, as a consequence of this, variable capture is not allowed. To stress the global accessibility of variable names we suggest to call δ-abstracted variables also *dynamic declarations*.

We consider it as one of the strengths of λ-DRT as it is defined in this paper that it syntactically keeps apart its functional and dynamic parts; we hope, through this, to gain a better understanding of how the two features interfere. The definition of $\mathcal{I}^s_\varphi(\lambda X.\mathbf{D})$ is one of these points of interference. The use of the function ε in $\mathcal{I}^s_{\varphi,[\varepsilon(A)/X]}$ in the interpretation of a functional term (cf. case 6 in definition 5.7) appears unorthodox from the point of view of pure classical logic, but the question to ask here is what further consequences this obstruction through dynamics has.

It is a much debated question how actually to view discourse referents. They are neither variables nor constants in the classical sense. Through the above interpretation we have given them a status in between: the variable-like character, underlined by the fact that free variables can be bound by the δ-operator, is prominent. Yet the accessibility of the variable's name — which we found to be so crucial — makes the discourse referents resemble *constants* very much.

It is routine to prove that each step in the interpretation process, the two assignment functions do indeed stay in the consistency relation. There are only two locations in the semantics definition where an assignment function is extended: the interpretation of a DRS and the interpretation of a λ-abstraction. Consistency is by definition kept at the interpretation of a DRS and if the λ-abstracted variable is of type t. If the λ-abstracted variable is not of type t then we can use that, by definition, the λ-abstracted variables are not in the domain of a state.

The separation of functional and dynamic features is not only by syntactic means; the clear responsibilities of the two assignment functions further underlines the distinction. It is interesting to note that this is not only a design choice but it was indeed necessary to be able to prove β-reduction correct, and thus it is something the calculus itself asks for.

The two components of a DRS denotation have clear roles: the first component constitutes the DRS's *anaphoric potential* , i.e. the set of possible antecedents for phrases and sentences to follow. The second component, the set of partial assignment functions t, is in some sense the set of contexts which validate the DRS. Note that the first component is merely necessary for the definition of the implication; it cannot be derived from the second component. The second component, however, is the essential part. It is the one mainly used for the concepts of validity and consequence, to be defined below.

Let us now look at the construction of denotations in detail. We want to stress two things here: first, how the interpretation of DRSes facilitates the simple definition of the (symmetric) merge operator and second, how the intensionality operator allows correct coordination of values for variables in partially instantiated expressions.

As mentioned earlier, the key points of DRS's denotations are the partiality of the assignment function s and the concept of assignment continuation. The variable continuation ensures that no undefined values occur within a DRS denotation by being able to assign a value to all δ-abstracted *and* all free variables of the DRS (cf.

remark 5.10). The foundation to the symmetric merge, however, is that *in addition* to those variables occurring in a DRS, the functions t *may* also assign a value to any other variable $\in \mathcal{VD}$. This implies the following definition.

Definition 5.14 (Minimal Assignment Function) We will call those assignment functions t of the second component of the denotation of a DRS $\mathbf{A}$, $t \in \mathcal{IF}^s_\varphi(\mathbf{A})$, which are only defined on those variables which occur in $\mathbf{A}$ (free variables or discourse referents), **minimal assignment functions**. All other functions t in $\mathcal{IF}^s_\varphi(\mathbf{A})$ are themselves continuations of the minimal assignment functions by the empty set of variables.

The occurrence of non-minimal assignment functions in $\mathcal{IF}^s_\varphi(\mathbf{A})$ allows for using set intersection to define the merge operator. Through the intersection we get exactly those functions which also assign a value to variables occurring in other arguments of the $\oplus$. To illustrate this, look at the following example taken from example (4) in section 4; the three parts in (8) are the interpretations of the arguments of the resulting expression of step 2 there[9].

(8) (a) $\mathcal{I}^s_\varphi(\delta\{x_j\}.\, x_j = j^*) = \langle\{x_j\}, \{t_1 \mid s[\{x_j\}]t_1, \mathcal{I}^{t_1}_\varphi(x_j = j^*) = \mathrm{T}\}\rangle$

(b) $\mathcal{I}^s_\varphi(\delta\{e_a\}.\mathsf{walk}(e_a, x_j))) = \langle\{e_a\}, \{t_2 \mid s[\{e_a\}]t_2, \mathcal{I}^{t_2}_\varphi(\mathsf{walk}(e_a, x_j)) = \mathrm{T}\}\rangle$

(c) $\mathcal{I}^s_\varphi(\delta\{\}.\mathsf{loc}(e_a) = \text{in-park}) = \langle\{\}, \{t_3 \mid s[\{\}]t_3, \mathcal{I}^{t_3}_\varphi(\mathsf{loc}(e_a) = \text{in-park}) = \mathrm{T}\}\rangle$

Each of the functions t_1 in case (a) is at least defined on x_j but some of the t_1 may also be defined on any other variables. Indeed, there will be functions t_1 which are defined on e_a. In case (b), all of the functions t_2 must be defined on both x_j and e_a, in order to attain the value T in the condition; again, some of the t_2 will be defined on other variables, too. Similarly for case (c): each one of the t_3 will be defined on e_a and some of them on x_j or other variables. The set intersection of the three sets will not only match the values of the variables but also the domains of the functions; thus in this example, all functions t of the result are defined at least on both x_j and e_a. Therefore, by means of the variable continuations, DRS-interpretations are able to foresee or be ready for the assignments of yet unknown variables.

This example also shows that a free variable and a discourse referent have the same effect on an interpretation under certain conditions. In case a variable X has not been assigned a value in a certain recursion step, introducing it as a discourse referent and using it as a free variable have the same effect: X gets assigned a new value. At a merge the assignments of the discourse referent and the free variable are matched through the set intersection, fusing them in this way. Thus, through this relation between free variables and discourse referents, the logic preempts the delayed binding.

In order for this 'switch' from free to bound variables[10] to work properly (i.e. to achieve the full denotation), the assignment functions φ and s should be empty at the very beginning of the interpretation process. This reflects the fact that the interpretation is set in an empty context, or discourse. Incidentally, this also mirrows the approach of DRT; there, a DRS is valid if the empty assignment function can be extended.

Let us examine another example in some detail now. The following is a variation of example (4) of section 4, and we consider the interpretation of the expression which corresponds to step 2 there.

9. We will neglect the consistency relation here and in the following example.
10. Note that logically this is a switch between two rather contradictory concepts.

(9) (a) `A man walks in the park.`

(b) $\mathcal{I}^s_\varphi((\text{a man})\ @(\text{walk}))$

1. $= \mathcal{I}^s_\varphi(\lambda G.(\delta\{x\}.\mathsf{man}(x) \otimes G(\hat{}x))@(\lambda z_d.\delta\{e\}.\mathsf{walk}(e, \vee z)))$
2. $= \Lambda g.\mathcal{I}^s_{\varphi,[\varepsilon(g)/G]}(\delta\{x\}.\mathsf{man}(x) \otimes G(\hat{}x))@\Lambda a.\mathcal{I}^s_{\varphi,[\varepsilon(a)/z]}(\delta\{e\}.\mathsf{walk}(e, \vee z))$
3. $= \Lambda g.(\langle\{x\}, \{t \mid s[\{x\}]t, \mathcal{I}^t_{\phi,[\varepsilon(g)/G]}(\mathsf{man}(x)) = \mathrm{T}\}\rangle \oplus \varepsilon(g)@\Lambda s.s(x))$
   $@\Lambda a.\langle\{e\}, \{r \mid s[\{e\}]r, \mathcal{I}^r_{\varphi,[\varepsilon(a)/z]}(\mathsf{walk}(e, \vee z)) = \mathrm{T}\}\rangle$
4. $= \Lambda g.(\langle\{x\}, \{t \mid s[\{x\}]t, \mathsf{man}(t(x)) = \mathrm{T}\}\rangle \oplus \varepsilon(g)@\Lambda s.s(x))$
   $@\Lambda a.\langle\{e\}, \{r \mid s[\{e\}]r, \mathsf{walk}(r(e), \varepsilon(a)(r)) = \mathrm{T}\}\rangle$
5. $= \langle\{x\}, \{t \mid s[\{x\}]t, \mathsf{man}(t(x)) = \mathrm{T}\}\rangle$
   $\oplus\, \Lambda a.\langle\{e\}, \{r \mid s[\{e\}]r, \mathsf{walk}(r(e), \varepsilon(a)(r)) = \mathrm{T}\}\rangle@\Lambda s.s(x)$
6. $= \langle\{x\}, \{t \mid s[\{x\}]t, \mathsf{man}(t(x)) = \mathrm{T}\}\rangle$
   $\oplus\, \langle\{e\}, \{r \mid s[\{e\}]r, \mathsf{walk}(r(e), \Lambda s.s(x)@r) = \mathrm{T}\}\rangle$
7. $= \langle\{x\}, \{t \mid s[\{x\}]t, \mathsf{man}(t(x)) = \mathrm{T}\}\rangle$
   $\oplus\, \langle\{e\}, \{r \mid s[\{e\}]r, \mathsf{walk}(r(e), r(x)) = \mathrm{T}\}\rangle$
8. $= \langle\{x, e\}, \{q \mid s[\{x\}]q, s[\{e\}]q, \mathsf{man}(q(x)) = \mathrm{T} \wedge \mathsf{walk}(q(e), q(x)) = \mathrm{T}\}\rangle$

the latter being the intended value since $s[\mathcal{X}]q$ and $s[\mathcal{Y}]q$ implies $s[\mathcal{X} \cup \mathcal{Y}]q$.

Focus here on how the look-up of x in $G(\hat{}x)$ is delayed through the interpretation of the $\hat{}$-operator until x has moved into the dominance of the second δ-abstraction operator (i.e. the DRS that represents `walk`) and can then be assigned a value by the right states, in this case states r in clause 7. Until there, the $(\hat{}x)$ of clauses 1 and 2 becomes $\Lambda s.s(x)$, which is finally applied to r in clause 6.

Remark 5.15 In the literature (e.g. Eijck and Kamp 1995) the phrases of *static* and *dynamic semantics* have been coined. Static semantics use semantic objects rather than defining dynamics by means of pairs of input/output assignments, as was first suggested in DPL (Groenendijk and Stockhof 1991). We claim here that there is no substantial difference between the two approaches, and back this by pointing to the connection between the interpretation of ; and a dynamic semantic definition. Indeed, the definition of ; in 5.7, in distinction to $\otimes$'s definition, reflects a one-directional binding flow imitating DPL's flow. Consider the definition of the (asymmetric) merge in Eijck and Kamp 1995 which is quite similar to DPL:

$\mathcal{M}, s, r \models \mathbf{A}; \mathbf{B}$ iff there exists an assignment function t with $\mathcal{M}, s, t \models \mathbf{A}$ and $\mathcal{M}, t, r \models \mathbf{B}$.

For the comparison to ;'s definition we repeat its definition here:

$$\mathcal{I}^s_\varphi(;)@A@B = A ;_{\mathcal{VAR}} B = \langle\, \mathcal{VAR}(A) \cup \mathcal{VAR}(B)\,,\ \{r \mid r \in \mathcal{FUN}(B) \text{ and } \exists t \in \mathcal{FUN}(A) : r|_{-\mathcal{VAR}(B)} = t\}\rangle$$

In the first interpretation of $\mathbf{D} = \mathbf{A}; \mathbf{B}$ we have that s and r in $\mathcal{M}, s, r \models \mathbf{D}$ correspond to s and $r \in \mathcal{IF}^s_\varphi(\mathbf{D})$ in $\mathcal{I}^s_\varphi(\mathbf{D}) = \langle\mathcal{X}, \mathcal{IF}^s_\varphi(\mathbf{D})\rangle$ respectively. Thus, in above definitions, the two s relate to each other, as do s' and r as well as s'' and t. Since we have $t \in \mathcal{FUN}(A)$, t plays the role of the output assignment of the first DRS, in the sense of DPL, and since $r|_{-\mathcal{VAR}(B)} = t$ merely requires that r extends t by values for the $\mathcal{VAR}(B)$[11], t acts as input assignment of the second DRS.

Finally in this section we look at validity and equivalence of DRSes. The definition of validity is a straightforward check on the non-emptyness of the second component of the DRS denotation. It has, in fact, already been pre-empted in the definitions of $\neg$ and $\vee$, which include a test on the empty set.

Definition 5.16 (Validity) A DRS $\mathbf{A}$ is valid in a model M in the context of s

11. Note that we can neglect the cases in which the value for some variable in $\mathcal{VAR}(\mathbf{B})$ is *changed* since we assume that none of the $\mathcal{VAR}(\mathbf{B})$ occurs in $\mathbf{A}$.

with respect to φ, if $\mathcal{IF}^s_\varphi(\mathbf{A}) \neq \emptyset$.

In the literature it is common to distinguish two notions of equivalences for DRSes, static and dynamic equivalence. Static (s-)equivalence compares the truth conditional information extracted from the denotations, neglecting the dynamic information. In λ-DRT we may define s-equivalence thus:

Definition 5.17 (s-Equivalence) Two DRSes $\mathbf{A}$ and $\mathbf{B}$ are **s-equivalent**, denoted $\mathbf{A} \simeq_s \mathbf{B}$, if $\mathcal{IF}^s_\varphi(\mathbf{A}) \neq \emptyset \Leftrightarrow \mathcal{IF}^s_\varphi(\mathbf{B}) \neq \emptyset$, for all states s and assignments φ

By way of example, the two DRSes $\mathbf{A} = \delta\{X\}.P(X)$ and $\mathbf{B} = \delta\{Y\}.P(Y)$ are s-equivalent, even though they have a different anaphoric potential. Such dynamic information is to be captured by the dynamic equivalence. However, this notion in λ-DRT may be considered to have two variants. The first straightforwardly compares the two components of the denotations; we shall call it context (c-)equivalence, since it asks for full context (second component) equality.

Definition 5.18 (c-Equivalence) Two DRSes $\mathbf{A}$ and $\mathbf{B}$ are **c-equivalent**, denoted $\mathbf{A} \simeq_c \mathbf{B}$, if $\mathcal{DV}^s_\varphi(\mathbf{A}) = \mathcal{DV}^s_\varphi(\mathbf{A})$ and $\mathcal{IF}^s_\varphi(\mathbf{A}) = \mathcal{IF}^s_\varphi(\mathbf{B})$, for all states s and assignments φ.

The second variant will be called dynamic (d-)equivalence. It turns out that to capture the equality of dynamic behaviour (do they have the same anaphoric potential and do they behave in the same way when extended by further context?) we need not require the second components to be exactly alike. They may as well relate by $\preceq_\mathcal{Z}$, defined thus:

Definition 5.19 (Contextual Detail) A set of states ψ_A is **contextually less detailed (by a set of variables $\mathcal{Z}$)** than the set of states ψ_B, $\psi_A \preceq_\mathcal{Z} \psi_B$, if

1. Either $\emptyset \neq \psi_A \subseteq \psi_B$ or $\emptyset = \psi_A = \psi_B$.
2. For all $s \in \psi_B \setminus \psi_A$ there exists a $t \in \psi_A$ and some subset $\mathcal{Z}' \subseteq \mathcal{Z}$ with $s = t|_{-\mathcal{Z}'}$.

We may also write simply $\preceq$, if $\mathcal{Z}$ is not in focus.

The operator $\preceq$ may also be extended to be used for DRS notations: we have $A_t \preceq_\mathcal{Z} B_t$ for two DRS denotations A_t and B_t, iff $\mathcal{VAR}(A) = \mathcal{VAR}(B)$ and $\mathcal{FUN}(A) \preceq_\mathcal{Z} \mathcal{FUN}(B)$.

In words, the contextually more detailed set of states ψ_B is of larger cardinality; the states it adds to ψ_A all result from states in ψ_A by taking away the values for some or all members of $\mathcal{Z}$. By including these smaller states in addition to the broader defined states, one can think of adding some detail. Note that if one of the sets is empty, the other one must be, too.

Definition 5.20 (d-Equivalence) Two DRSes $\mathbf{A}$ and $\mathbf{B}$ are **d-equivalent**, denoted $\mathbf{A} \simeq_d \mathbf{B}$, if for all states s and assignments φ there exists some $\mathcal{Z}$ such that

$$\mathcal{I}^s_\varphi(\mathbf{A}) \preceq_\mathcal{Z} \mathcal{I}^s_\varphi(\mathbf{B}) \text{ or } \mathcal{I}^s_\varphi(\mathbf{B}) \preceq_\mathcal{Z} \mathcal{I}^s_\varphi(\mathbf{A})$$

It is instruitive to look at how two DRSes, $\mathbf{A}$ and $\mathbf{B}$, relate, if they are d-, but not c-equivalent. The first guess is that their sets of discourse referents is the same but that one of them, say $\mathbf{B}$, has at least one free variable, say $X \in \mathcal{Z}$, which does not occur in the other. Then, of course, all states in the interpretation of $\mathbf{B}$ must assign a value to X, whereas the states in the interpretation of $\mathbf{A}$ need not. The extra states in $\mathcal{IF}^s_\varphi(\mathbf{B})$ are those which do this. But such $\mathbf{A}$ and $\mathbf{B}$ are only d-equivalent, if the predication on X is a tautology, i.e. if X can be assigned any

value from the domain[12]. If, however, the extra predicate cuts down the number of possibilities of values for the X, then $\mathcal{IF}^s_\varphi(\mathbf{A})$ would include states that assign one of the impossible values to Y, and thus no longer be a subset of $\mathcal{IF}^s_\varphi(\mathbf{B})$.

A more practical situation in which the subset-relationship comes into play in connection with the $\preceq$-relation is if it is the assignment of a type-t variable in φ that keeps track of such an $X \in \mathcal{Z}$. Such a situation occurs e.g. with β-reduction and indeed it is the proof of the correctness of the functional reductions that we need the concept of $\preceq$.

It can be verified that d-equivalence indeed captures dynamic equivalence as described above. For this, we first need to check that reducing the amount of contextual detail (by the same set of variables) in denotations does not destroy the truth-conditions if joining them by logical operators. Secondly, we note that two DRSes $\mathbf{A} \simeq_d \mathbf{B}$ have the same anaphoric potential, since they equal in their first components. Lastly, we must verify that they behave equally if extended by the same further context, i.e. whether merging (using either $\oplus$ or ;) their interpretations with $\mathcal{I}^s_\varphi(\mathbf{C})$, for any $\mathbf{C}$[13], results in denotations which imply the same truth values. The proofs for this are a bit tedious, and they can be found in Kuschert 1996.

6 Metalogical Results

In this paper we looked at two aspects of the semantics of λ-DRT: we formally specified the operational semantics by defining a reduction system, and we proposed a denotational semantics by defining the function $\mathcal{I}^s_\varphi()$. The question to ask now is whether the two descriptions of semantics define the same notion of meaning. Kuschert 1996 presents proofs that the reductions are correct, and that the reduction system is complete with respect to the denotational semantics. It also shows that we get unique normal forms, if we restrict ourselves to safe expressions (cf. definition 5.9), which is an important result computationally. There is no space to present the proofs within this paper, but let us briefly describe some general ideas and interesting features of these proofs.

The substitution-value lemma, being the basis for the correctness proofs for α-, β- and η-reductions (collectively called the λ-reductions here), uses the notion of contextual detail as defined in definition 5.19; it is modified to conjecture that $\mathcal{I}^s_\varphi([\mathbf{B}/Y]\mathbf{A}_t) \preceq_\mathcal{Z} \mathcal{I}^s_{\varphi,[\mathcal{I}^s_\varphi(\mathbf{B})/Y]}(\mathbf{A}_t)$ for any $\mathbf{A} \in \text{sf}$, $\mathbf{B} \in \text{sf}_{\tau(Y)}$ and a Y not bound in $\mathbf{A}$. $\mathcal{Z}$ is the set of variables occurring in $\mathbf{B}$ but not in $\mathbf{A}$. Using this result in the correctness proofs we can then conclude that the interpretation of different λ-reduction steps reflect expressions which are d-equivalent. Thus, since we have shown that d-equivalent expressions behave identically both in truth-conditional and dynamic terms, we may consider α-, β-, and η-reductions correct.

For the completeness proof we just mention here that we need a decision procedure for the equality of type-o expressions.

Proving λ-DRT's reduction system confluent and terminating is straightforward, if we reuse proof techniques from standard λ-calculus. Since the λ-, the δ- and the μ-reductions address different operators (the λ, δ and $\wedge$-operator respectively), we can look at these three groups of reductions seperately.

7 Conclusion

λ-DRT was born by the idea to straightforwardly combine standard DRT and β-reduction of typed λ-calculus for the semantic processing core of actual NL projects.

12. For an example, take $\mathbf{B} = \delta\{X\}.P(X) \wedge (Q(Y) \vee \neg Q(Y))$ vs. $\mathbf{A} = \delta\{X\}.P(X)$.
13. Obeying the syntactic restrictions when using ;, of course.

The operational semantics immediately suggested itself from this idea of λ-DRT. The work presented here gives the formal background for systems using λ-DRT as their semantic formalism. The most important result is the definition of a denotational semantics which can be proved to be consistent with the operational semantics.

The interaction of λs and δs, which looks so simple in the reduction system, turned out to be not quite so trivial in the denotational semantics. We are faced with the fact that functionality and dynamics are based on quite contrary ideas and principles. Essentially, the interaction was enabled by the consistency relation and the declarative character of δ-abstracted variables, reflected by the possibility of accessing their variable names.

Another focus of the denotational semantics was the symmetry of the $\otimes$-operator. To interpret this operator correctly, we had to make the dynamics bi-directional. Besides the original motivation for the $\otimes$-operator, mentioned in the introduction, one might investigate whether bi-directional dynamics is cognitively adequate, and if so, how do humans perform the 'backward binding'?

This work lays the ground for a number of directions of further research. For one, λ-DRT has been and will be further extended. In section 4 we mentioned the assumption that coindexation of anaphors with their antecedents is provided by the syntactical analysis. It will be interesting to explore the alternatives of this approach in λ-DRT. We also already mentioned the goal of building an inference system. This, indeed, has been one of the major motivations for our enterprise of formalising λ-DRT.

References

Bos, J., Mastenbroek, E., McGlashan, S., Millies, S., and Pinkal, M.: 1994, A compositional DRS-based formalism for NLP-applications, *Proceedings of the International Workshop on Computational Semantics, Tilburg* pp 21–31

Bruijn, N. D.: 1972, Lambda calculus notation with nameless dummies, *Indag Math* 34, 381–392

Eijck, J. v. and Kamp, H.: 1995, Representing discourse in context, in L. S. van Benthem and A. ter Meulen (eds.), *Handbook of Logic and Language*, Elsevier Science B.V.

Groenendijk, J. and Stockhof, M.: 1990, Dynamic montague grammar, in L. Kalman and L. Polos (eds.), *Papers from the Second Symposium on Logic and Language*, pp 3 – 48, Budapest, Akademiai Kiadoo

Groenendijk, J. and Stockhof, M.: 1991, Dynamic predicate logic, *Linguistics and Philosophy* 14(1)

Kuschert, S.: 1995, *Eine Erweiterung des λ-Kalküls um Diskursrepräsentationsstrukturen*, Diplomarbeit, Universität des Saarlandes

Kuschert, S.: 1996, *Higher Order Dynamics: Relating operational and denotational semantics of λ-DRT*, CLAUS-Report 72, Universität des Saarlandes

Muskens, R.: 1994, A compositional discourse representation theory, in P. Dekker and M. Stockhof (eds.), *Proceedings of the 9th Amsterdam Colloquium*, pp 467 – 486, ILLC, Amsterdam

Ruhrberg, P.: 1995, *Simultaneous Abstraction and Semantic Theories*, Ph.D. thesis, University of Edinburgh

Zeevat, H.: 1989, A compositional approach to DRT, *Linguistics and Philosophy* 12, 95–131