

# The Grammar of Quantification and the Fine Structure of Interpretation Contexts

Adrian Brasoveanu,\* UC Santa Cruz, [abrsvn@gmail.com](mailto:abrsvn@gmail.com)

## Abstract

Providing a compositional interpretation procedure for discourses in which descriptions of complex dependencies between interrelated objects are incrementally built is a key challenge for formal theories of natural language interpretation. This paper examines several quantificational phenomena and argues that we need richly structured contexts of interpretation that are passed on between different parts of the same sentence and also across sentential boundaries to account for these phenomena. The main contribution of the paper is showing how we can add structure to contexts in an incremental way, starting with the basic notion of context in classical first-order logic, i.e., interpretation contexts formalized as single total variable assignments.

## 1 Introduction: Quantification in Discourse

The basic goal of formal semantics is to provide a precise characterization of natural language interpretation. The idea is to characterize the meaning of sentences and discourses like (1) and (2) below as precisely as the meaning of equations like  $2 + 3 = 5$  in elementary arithmetic.

- (1) Linus entered the room.
- (2) Linus entered the room. He sat down.

In particular, we want to associate each word *Linus*, *enter* etc. – and even parts of words, e.g., the past tense *-ed* – with a precisely defined meaning in much the same way that the symbols  $2$ ,  $+$ ,  $3$ ,  $=$  etc. are associated with precisely defined meanings. Given some familiarity with formal logic, we can fairly easily see how to start developing the foundations of such a project. It is, however, not obvious how to characterize the denotations of natural language quantifiers like *every woman*, *most rooms*, *no man* etc.

For one thing, quantifiers do not refer to any individual or set of individuals. Consider, for example, the quantifier *no man* in the sentence *Bonadea saw no man*. What kind of entity or set of entities could *no man* possibly refer to? Similarly, what kind of entity or set of entities could the adverb *never* possibly refer to in *Bonadea never smoked* or the modal verb *cannot* in *Bonadea cannot help Linus*?

Moreover, besides characterizing the meaning of quantifiers by themselves, we also want to characterize the interaction between multiple quantifiers and, also, their effects on adjectives like *same* and *different*. These adjectives can have both a deictic / sentence-external reading, exemplified in (3) and (4) below, and a sentence-internal reading, exemplified in (5).

---

\*I want to thank Johan van Benthem, Jessica Rett and an anonymous reviewer for their very helpful and detailed comments and Donka Farkas, Bill Ladusaw and the participants in the fall 2008 Seminar in Semantics and winter 2011 Semantics B course at UCSC for discussion. The usual disclaimers apply.

- (3) a. Mary recited *The Raven*.  
 b. Linus recited a different poem.  
 (deictic / sentence-external: different from *The Raven*)
- (4) a. Mary recited *The Raven*.  
 b. Then, every boy recited a different poem.  
 (deictic / sentence-external: different from *The Raven*)
- (5) Every boy recited a different poem.  
 (sentence-internal: for any two boys *a* and *b*, *a*'s poem is different from *b*'s poem)

The interpretation of *different* in (3b)/(4b) is sentence external in the sense that it is anaphoric to the variable introduced by the proper name *The Raven* in the previous sentence (3a)/(4a). Thus, in (3)/(4), *different* relates two variables and requires their values, i.e., the actual entities, to be distinct.

The sentence-internal reading in (5) seems to relate values of only one variable, introduced by the narrow-scope indefinite *a poem*. These values, i.e., the recited poems, covary with the values of the variable introduced by the universal quantifier *every boy* – and *different* requires the poems to be distinct relative to distinct boys.

We want to capture the particular way in which the interpretation of *different* is dependent on quantificational expressions – and, also, the fact that *different* is licensed by such expressions: uttering (5) out of the blue is felicitous, but uttering (3b) is not. If (3b) is discourse-initial, it begs the question: a poem different from what?

Formal semantics has made great progress with respect to the project of providing a mathematically precise characterization of the denotations of natural language quantifiers by themselves. Much less is known about how to relate multiple quantifiers and pronominal items in discourse and also, how quantifiers interact with expressions like *same* and *different*.

The goal of this paper is to overview the kinds of strategies used in the semantic literature to account for these and related phenomena and identify important decision points in the development of such accounts. The main idea is that all these phenomena provide support for a notion of natural language interpretation that involves richly structured contexts that are incrementally updated and passed on between different parts of the same sentence and across sentential boundaries.

The main contribution of the paper is showing how we can add increasingly more complex structure to interpretation contexts, starting with the basic notion of context in classical first-order logic (FOL), i.e., a single total variable assignment.

## 2 Quantification in Classical First-Order Logic (FOL)

The essence of quantification in FOL is *pointwise manipulation of variable assignments*.

This observation is almost never emphasized in the usual introductions to formal semantics. One reason is that such introductions focus on compositionality issues, i.e., on the recursive definition of truth and satisfaction and the way in which semantic clauses for FOL closely track syntactic clauses. Later, when formulas with multiple nested quantifiers are interpreted, pointwise manipulation of variable assignments could come into proper focus, but it often does not.

What we usually do at that point is provide a fairly involved formula and examine in detail how its interpretation involves multiple nested applications of the semantic clauses for universal and existential quantifiers. The main take-home lesson is that the semantic clauses can be applied recursively and some care needs to be exercised in the process, but that is all. The satisfaction of

a well-applied set of rules obscures the lack of conceptual understanding and armed with technical knowledge of FOL, we proceed to pursue some of life's worthier goals.

Interpreting formulas with multiple nested quantifiers in FOL is the simplest example in which a crucial aspect of interpretation takes main stage: a higher quantifier creates the context relative to which a lower quantifier is interpreted.

Thus, just like in everyday conversation, novels etc., an automatic side effect of interpreting an expression (the mother telling the child how to arrange her toys, what the main character did in chapter 2 etc.) is to create the context relative to which subsequent expressions are interpreted.

The formal tool that accomplishes this basic and crucial task in FOL is the notion of variable assignment and the ways in which the recursive definition of truth and satisfaction instructs us to manipulate variable assignments during interpretation.

Variable assignments are the interpretation 'glue' that is passed on between different parts of a formula and that enables us to interpret them together as a whole, in much the same way that properly sequenced pieces of conversation or text are interpreted together as a meaningful whole.

That is, variable assignments and the way in which FOL quantifiers manipulate them give us the key to building a formally explicit theory of sentence and discourse interpretation. FOL provides the simplest example in which interpretation contexts are manipulated in a syntactically local manner but are passed on in a syntactically non-local manner. This enables us to compositionally interpret a syntactically complex structure in such a way that distinct expressions – possibly separated by sentential boundaries – end up incrementally contributing to the same, richly articulated description of some part of the world.

We can see this in detail by examining the formula in (6) below, which is the FOL translation of the English sentence in (7).

$$(6) \quad \forall x(\text{WOMAN}(x) \rightarrow \exists y(\text{MAN}(y) \wedge \exists z(\text{MUSTACHE}(z) \wedge \text{HAVE}(y, z)) \wedge \text{SEE}(x, y)))$$

$$(7) \quad \text{Every}^x \text{ woman saw a}^y \text{ man who had a}^z \text{ mustache.}$$

Based on the definition of FOL syntax in (8) below, the formula in (6) is assigned the tree-like syntactic structure in Figure 1.

(8) **FOL syntax**

- a. Basic expressions:
  - i. Terms: names (individual constants) MARY, JOHN, ... and a denumerably infinite set of variables over individuals  $\mathcal{V} = \{x, y, z, \dots\}$
  - ii. Predicates: 1-place predicates MAN, WOMAN, ..., 2-place predicates SEE, HAVE, ..., 3-place predicates GIVE, SEND, ... etc.
- b. Atomic formulas:
  - i. If  $\pi$  is an  $n$ -place predicate and  $\alpha_1, \dots, \alpha_n$  are terms, then  $\pi(\alpha_1, \dots, \alpha_n)$  is a formula.
  - ii. If  $\alpha$  and  $\beta$  are terms, then  $\alpha = \beta$  is a formula.
- c. Formulas (sentential connectives):
  - i. If  $\phi$  is a formula, then  $(\neg\phi)$  is a formula.
  - ii. If  $\phi$  and  $\psi$  are formulas, then  $(\phi \wedge \psi)$  is a formula.
- d. Formulas (quantifiers):
  - i. If  $\phi$  is a formula and  $v$  is a variable, then  $(\exists v\phi)$  is a formula.
- e. Abbreviations (parentheses are sometimes omitted to improve readability):
  - i.  $(\phi \vee \psi) := (\neg(\neg\phi \wedge \neg\psi))$

- ii.  $(\phi \rightarrow \psi) := (\neg(\phi \wedge \neg\psi))$
- iii.  $(\forall v\phi) := (\neg\exists v(\neg\phi))$

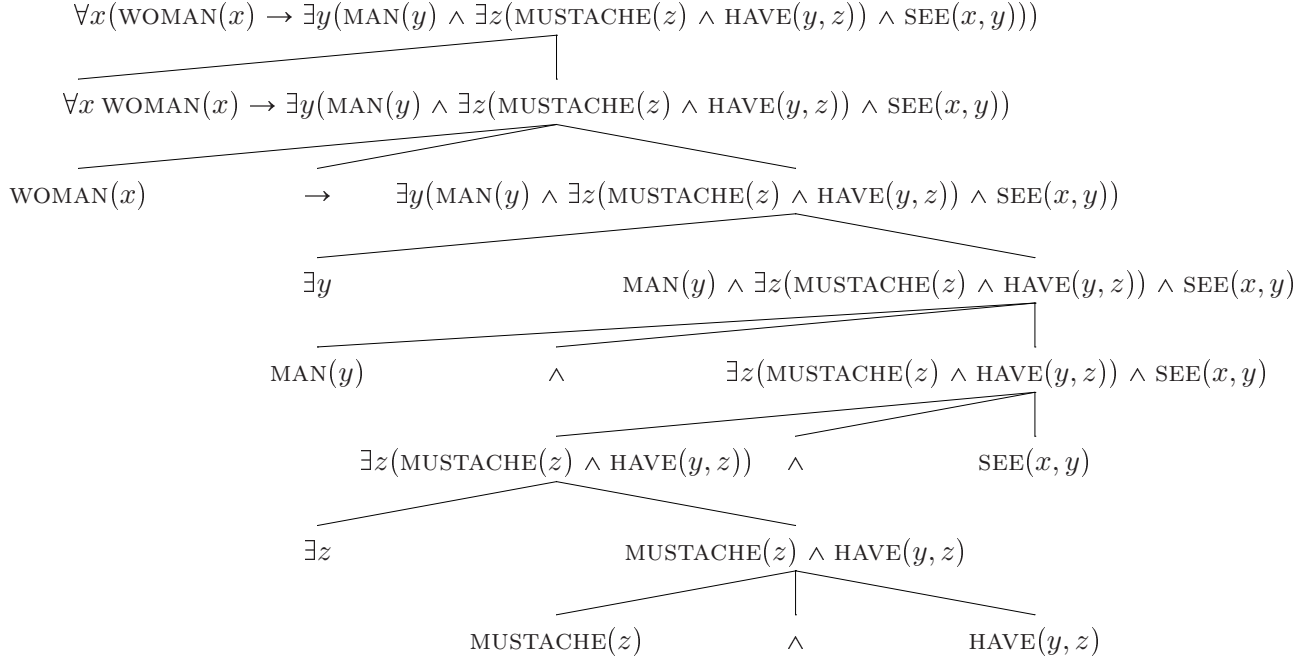


Figure 1: The syntactic structure of (6)

The definition of FOL semantics relies on the usual notions of model and variable assignment, provided below. Informally, the model is the ‘dictionary’ that provides the meaning of the basic expressions. This is not the kind of dictionary that lexicographers compile, but the mental dictionary that a child compiles as she goes about the world, understands and obeys her mother’s commands etc.: this dictionary links basic expressions of the language and non-linguistic, actual entities in the world.

In contrast, a variable assignment is a list of entities brought to salience in previous discourse. During the interpretation of a formula or during a conversation, the model (i.e., the structure of the world and the basic expressions of the language) does not change, while variable assignments are updated as new entities are mentioned: the world and the language stay the same, but the state of the conversation changes.

Variables are like pronouns, they refer to whatever entities are salient at that point in discourse: their meaning depends on the current context of interpretation, i.e., on the current variable assignment.

(9) Models and assignments:

- a. A model  $\mathfrak{M}$  for FOL is a pair  $\langle \mathbf{D}^{\mathfrak{M}}, \mathbf{I}^{\mathfrak{M}} \rangle$ , where  $\mathbf{D}$  is the domain of individuals and  $\mathbf{I}$  is a function that assigns an individual in  $\mathbf{D}$  to every name and a subset of  $\mathbf{D}^n$  to every  $n$ -place predicate (the superscript  $\mathfrak{M}$  is omitted whenever contextually retrievable).
- b. An  $\mathfrak{M}$ -assignment  $g, h, \dots$  is a total function from the set of variables  $\mathcal{V}$  to the set of individuals  $\mathbf{D}$ . Thus, the set of all variable assignments  $\mathcal{G}$  is the set of functions  $\mathbf{D}^{\mathcal{V}}$  from  $\mathcal{V}$  to  $\mathbf{D}$ .

Our formulation of FOL semantics is slightly non-standard: we rely on the abbreviation  $g[v]h$  in (10) below that makes perspicuous the *pointwise assignment manipulation* component of first-order quantification. The abbreviation  $g[v]h$  requires the variable assignments  $g$  and  $h$  to differ at most with respect to the value they assign to  $v$ . The binary relation over variable assignments induced by  $g[v]h$  (for any variable  $v$ ) is an equivalence relation.

(10)  $g[v]h := g$  differs from  $h$  at most with respect to the value assigned to the variable  $v$

That is, for any variable  $v' \neq v$ , we necessarily have that  $g(v') = h(v')$  and we also allow for the possibility that  $g(v) = h(v)$  (but this is not necessary).

Manipulating assignments in a pointwise, variablewise way by means of  $g[v]h$  is the only way in which contexts of interpretation change / are updated in FOL.

The definition of the interpretation function  $\llbracket \cdot \rrbracket^{\mathfrak{M}, g}$  – or  $\llbracket \cdot \rrbracket^g$  for short – is provided in (11) below ( $\mathbb{T}$  and  $\mathbb{F}$  stand for true and false).

(11) **FOL semantics**

a. Basic expressions:

- i. If  $\alpha$  is a name and  $\pi$  is an  $n$ -place predicate, then  $\llbracket \alpha \rrbracket^g = \mathbf{I}(\alpha)$  and  $\llbracket \pi \rrbracket^g = \mathbf{I}(\pi)$ .
- ii. If  $v$  is a variable, then  $\llbracket v \rrbracket^g = g(v)$ .

(we use the ‘dictionary’ to interpret the basic expressions of our language and the context to interpret ‘pronouns’)

b. Atomic formulas:

- i. If  $\pi$  is an  $n$ -place predicate and  $\alpha_1, \dots, \alpha_n$  are terms, then  $\llbracket \pi(\alpha_1, \dots, \alpha_n) \rrbracket^g = \mathbb{T}$  iff  $\langle \llbracket \alpha_1 \rrbracket^g, \dots, \llbracket \alpha_n \rrbracket^g \rangle \in \llbracket \pi \rrbracket^g$ .
- ii. If  $\alpha$  and  $\beta$  are terms, then  $\llbracket \alpha = \beta \rrbracket^g = \mathbb{T}$  iff  $\llbracket \alpha \rrbracket^g = \llbracket \beta \rrbracket^g$ .

(basic claims about the world are true iff the entities mentioned in those claims are actually related in the way we claim they are)

c. Formulas (sentential connectives):

- i.  $\llbracket \neg \phi \rrbracket^g = \mathbb{T}$  iff  $\llbracket \phi \rrbracket^g = \mathbb{F}$ .
- ii.  $\llbracket \phi \wedge \psi \rrbracket^g = \mathbb{T}$  iff  $\llbracket \phi \rrbracket^g = \mathbb{T}$  and  $\llbracket \psi \rrbracket^g = \mathbb{T}$ .

d. Formulas (quantifiers):

- i.  $\llbracket \exists v \phi \rrbracket^g = \mathbb{T}$  iff there is an assignment  $h$  such that (s.t.)  $g[v]h$  and  $\llbracket \phi \rrbracket^h = \mathbb{T}$ .

(existential quantification is simply bringing an entity to salience, then making a claim about it)

e. Based on the abbreviations in (8e) above, we derive the following:

- i.  $\llbracket \phi \vee \psi \rrbracket^g = \mathbb{T}$  iff  $\llbracket \phi \rrbracket^g = \mathbb{T}$  or  $\llbracket \psi \rrbracket^g = \mathbb{T}$ .
- ii.  $\llbracket \phi \rightarrow \psi \rrbracket^g = \mathbb{T}$  iff if  $\llbracket \phi \rrbracket^g = \mathbb{T}$ , then  $\llbracket \psi \rrbracket^g = \mathbb{T}$   
iff  $\llbracket \phi \rrbracket^g = \mathbb{F}$  or  $\llbracket \psi \rrbracket^g = \mathbb{T}$ .
- iii.  $\llbracket \forall v \phi \rrbracket^g = \mathbb{T}$  iff any assignment  $h$  s.t.  $g[v]h$  is s.t.  $\llbracket \phi \rrbracket^h = \mathbb{T}$ .

f. Truth:

- i. A formula  $\phi$  is true in model  $\mathfrak{M}$  iff  $\llbracket \phi \rrbracket^{\mathfrak{M}, g} = \mathbb{T}$  for any assignment  $g$ .
- ii. A formula  $\phi$  is false in model  $\mathfrak{M}$  iff  $\llbracket \phi \rrbracket^{\mathfrak{M}, g} = \mathbb{F}$  for any assignment  $g$ .

(formulas with free variables in them are neither true nor false in a model *simpliciter*; a free variable is like a pronoun that needs a context / an assignment in addition to the model to be interpreted)

What is of primary interest to us is the definition of existential quantification in (11d-i) and the definition of universal quantification in (11e-iii). These definitions indicate that first-order quantification has two components.

- (i) a component that is common to the existential and the universal quantifier: the pointwise manipulation of variable assignments encoded by  $g[v]h$ ;
- (ii) a ‘counting’ / ‘quantifying’ component that distinguishes between the two quantifiers: a particular subset of the set of manipulated assignments (different for the existential vs. the universal quantifier) needs to satisfy the formula  $\phi$  in the scope of the quantifier.

Which particular subset has to satisfy the requirement in (ii) is subject to variation. The existential merely requires at least one of the assignments  $h$  that result after the pointwise manipulation  $g[v]h$  to satisfy  $\phi$ . In contrast, the universal requires all such assignments  $h$  to satisfy  $\phi$ .

Thus, both quantifiers relate the two sets of assignments  $\{h : g[v]h\}$  (the assignments obtained as a result of pointwise manipulation) and  $\{h : \llbracket \phi \rrbracket^h = \mathbb{T}\}$  (the assignments that satisfy the formula  $\phi$  in their scope). But the existential quantifier and the universal quantifier require these sets to stand in different relations: non-empty intersection for the existential and inclusion for the universal.

This is shown in more detail below.

(12)  $\llbracket \exists v \phi \rrbracket^g = \mathbb{T}$  iff:

- there is an assignment  $h$  s.t.  $g[v]h$  and  $\llbracket \phi \rrbracket^h = \mathbb{T}$ , i.e.,
- there is an assignment  $h$  in both the set  $\{h : g[v]h\}$  and the set  $\{h : \llbracket \phi \rrbracket^h = \mathbb{T}\}$ , i.e.,
- the intersection of these sets is non-empty:  $\{h : g[v]h\} \cap \{h : \llbracket \phi \rrbracket^h = \mathbb{T}\} \neq \emptyset$

(13)  $\llbracket \forall v \phi \rrbracket^g = \mathbb{T}$  iff:

- any assignment  $h$  s.t.  $g[v]h$  is s.t.  $\llbracket \phi \rrbracket^h = \mathbb{T}$ , i.e.,
- any assignment  $h$  in the set  $\{h : g[v]h\}$  has to also be in the set  $\{h : \llbracket \phi \rrbracket^h = \mathbb{T}\}$ , i.e.,
- the first set is included in the second:  $\{h : g[v]h\} \subseteq \{h : \llbracket \phi \rrbracket^h = \mathbb{T}\}$

We will now reformulate the semantics of FOL so that the above observations about first-order quantification play a pivotal role.

We will take the denotations of FOL formulas to be *sets* of variable assignments, with truth being the set  $\mathcal{G}$  of all assignments and falsity being  $\emptyset$ .

That is, we define an interpretation function  $\llbracket \cdot \rrbracket^{\mathfrak{M}}$  from FOL formulas to  $\wp(\mathcal{G})$  (the powerset of  $\mathcal{G}$ ) that is not parametrized by variable assignments, but only by the model  $\mathfrak{M}$ . And we reformulate the definition of truth in (11) above: if a formula is true *simpliciter* (i.e., independently of context), its denotation is the entire set of assignments  $\mathcal{G}$ ; if a formula is false *simpliciter*, its denotation is the empty set of assignments  $\emptyset$ .

A clear parallel between FOL semantics and the Kripke semantics for modal logic will emerge (see Ben-Shalom 1996, van Benthem 1997 and Marx & Venema 1997 among others): just as a formula in modal logic denotes a proposition / a sets of worlds, namely those worlds relative to which the formula is true, a formula in FOL denotes a set of assignments / a set of contexts, namely those contexts relative to which the formula is true. Moreover, existential quantification is

like the possibility operator  $\diamond$  and universal quantification is like the necessity operator  $\Box$ ; both of them, the accessibility relation is the binary relation over variable assignments induced by pointwise assignment manipulation, i.e.,  $g[v]h$ .

This modal-like definition of the classical FOL interpretation function  $\llbracket \cdot \rrbracket^{\mathfrak{M}}$  – or  $\llbracket \cdot \rrbracket$  for short – is provided in (15) below. The clauses for existential and universal quantification (see (15c-i) and (15d-iii)) clearly distinguish between the two components of first-order quantification identified above, namely pointwise assignment manipulation on one hand and relations between sets of assignments on the other.

$$(14) \quad \text{For any term } \alpha \text{ and any assignment } g, \text{ let } g/\mathbf{I}(\alpha) := \begin{cases} g(\alpha), & \text{if } \alpha \text{ is a variable.} \\ \mathbf{I}(\alpha), & \text{if } \alpha \text{ is a name.} \end{cases}$$

(15) **FOL semantics –  $\wp(\mathcal{G})$  version**

a. Atomic formulas:

i. If  $\pi$  is an  $n$ -place predicate and  $\alpha_1, \dots, \alpha_n$  are terms, then

$$\llbracket \pi(\alpha_1, \dots, \alpha_n) \rrbracket = \{g : \langle g/\mathbf{I}(\alpha_1), \dots, g/\mathbf{I}(\alpha_n) \rangle \in \mathbf{I}(\pi)\}.$$

ii. If  $\alpha$  and  $\beta$  are terms, then  $\llbracket \alpha = \beta \rrbracket = \{g : g/\mathbf{I}(\alpha) = g/\mathbf{I}(\beta)\}.$

b. Formulas (sentential connectives):

i.  $\llbracket \neg\phi \rrbracket = \{g : g \notin \llbracket \phi \rrbracket\}$   
 $= \mathcal{G} \setminus \llbracket \phi \rrbracket$

ii.  $\llbracket \phi \wedge \psi \rrbracket = \llbracket \phi \rrbracket \cap \llbracket \psi \rrbracket$

c. Formulas (quantifiers):

i.  $\llbracket \exists v\phi \rrbracket = \{g : \text{there is an } h \text{ s.t. } g[v]h \text{ and } h \in \llbracket \phi \rrbracket\}$   
 $= \{g : \{h : g[v]h\} \cap \llbracket \phi \rrbracket \neq \emptyset\}$

d. Based on the abbreviations in (8e) above, we derive the following:

i.  $\llbracket \phi \vee \psi \rrbracket = \llbracket \phi \rrbracket \cup \llbracket \psi \rrbracket$

ii.  $\llbracket \phi \rightarrow \psi \rrbracket = \mathcal{G} \setminus (\llbracket \phi \rrbracket \setminus \llbracket \psi \rrbracket)$   
 $= (\mathcal{G} \setminus \llbracket \phi \rrbracket) \cup \llbracket \psi \rrbracket$

iii.  $\llbracket \forall v\phi \rrbracket = \{g : \text{any } h \text{ s.t. } g[v]h \text{ is s.t. } h \in \llbracket \phi \rrbracket\}$   
 $= \{g : \{h : g[v]h\} \subseteq \llbracket \phi \rrbracket\}$

e. Truth:

i. A formula  $\phi$  is true in model  $\mathfrak{M}$  iff  $\llbracket \phi \rrbracket = \mathcal{G}$ .

ii. A formula  $\phi$  is false in model  $\mathfrak{M}$  iff  $\llbracket \phi \rrbracket = \emptyset$ .

The formula in (6) above is interpreted as shown in (16) and (17) below. Quantifiers are interpreted as manipulating variable assignments in a pointwise / variablewise manner and passing the resulting assignments to the subformulas in their scope. This dynamics of interpretation is common to both the standard and the  $\wp(\mathcal{G})$ -style interpretations above.

(16) Standard version:

$$\llbracket \forall x(\text{WOMAN}(x) \rightarrow \exists y(\text{MAN}(y) \wedge \exists z(\text{MUSTACHE}(z) \wedge \text{HAVE}(y, z)) \wedge \text{SEE}(x, y))) \rrbracket^g = \mathbb{T} \text{ iff}$$

- any  $h$  s.t.  $g[x]h$  and  $h(x) \in \mathbf{I}(\text{WOMAN})$  is s.t.  
   there is an  $i$  s.t.  $h[y]i$  and  $i(y) \in \mathbf{I}(\text{MAN})$  and  
   there is a  $j$  s.t.  $i[z]j$  and  $j(z) \in \mathbf{I}(\text{MUSTACHE})$  and  $\langle j(y), j(z) \rangle \in \mathbf{I}(\text{HAVE})$   
   and  $\langle i(x), i(y) \rangle \in \mathbf{I}(\text{SEE})$
- any woman  $a$  is s.t. there is a man  $b$  that she saw and that had a mustache  $c$

- (17)  $\wp(\mathcal{G})$  version:  
 $\llbracket \forall x(\text{WOMAN}(x) \rightarrow \exists y(\text{MAN}(y) \wedge \exists z(\text{MUSTACHE}(z) \wedge \text{HAVE}(y, z)) \wedge \text{SEE}(x, y))) \rrbracket =$
- $\{g : \text{any } h \text{ s.t. } g[x]h \text{ and } h(x) \in \mathbf{I}(\text{WOMAN}) \text{ is s.t.}$   
     there is an  $i$  s.t.  $h[y]i$  and  $i(y) \in \mathbf{I}(\text{MAN})$  and  
     there is a  $j$  s.t.  $i[z]j$  and  $j(z) \in \mathbf{I}(\text{MUSTACHE})$  and  $\langle j(y), j(z) \rangle \in \mathbf{I}(\text{HAVE})$   
     and  $\langle i(x), i(y) \rangle \in \mathbf{I}(\text{SEE})\}$
  - $\{g : \text{any woman } a \text{ is s.t. there is a man } b \text{ that she saw and that had a mustache } c\}$   
     (this set of assignments is either  $\mathcal{G}$  or  $\emptyset$ , depending on whether the model satisfies the  
     condition “any woman  $a$  is s.t. there is a man  $b$  that she saw and that had a mustache  $c$ ”  
     or not)

The first quantifier is the universal *every<sup>x</sup> woman*, i.e.,  $\forall x(\text{WOMAN}(x) \rightarrow \dots)$ . This quantifier updates the discourse-initial context  $g$  by bringing to salience an individual  $x$  that is required to be a woman. This is achieved by manipulating  $g$  in a pointwise way: we consider any (this is the universal component) context  $h$  that is like  $g$  except that it assigns a woman to  $x$ , i.e., any  $h$  is s.t.  $g[x]h$  and  $h(x) \in \mathbf{I}(\text{WOMAN})$ .

The second quantifier is the existential *a<sup>y</sup> man that ...*, i.e.,  $\exists y(\text{MAN}(y) \wedge \dots)$ . This quantifier takes narrow scope relative to the preceding universal quantifier, which means that it is interpreted relative to the context  $h$  created by the universal. The existential further updates this context  $h$  by bringing to salience an individual  $y$  that is required to be a man. Once again, this is achieved by pointwise assignment manipulation: we consider some (this is the existential component) context  $i$  that is like  $h$  except that it assigns a man to  $y$ , i.e., some  $i$  s.t.  $h[y]i$  and  $i(y) \in \mathbf{I}(\text{MAN})$ .

Importantly, context  $i$  preserves the manipulation contributed by the universal quantifier, i.e.,  $i(x) = h(x)$ . This follows immediately from the definition of the pointwise assignment manipulation  $h[y]i$ :  $i$  differs from  $h$  at most with respect to the value of  $y$ . Therefore, when we interpret the final subformula  $\text{SEE}(x, y)$  (which we interpret relative to  $i$ ), we add to the previous information and require the woman  $x$  brought to salience by the universal quantifier to see the man  $y$  brought to salience by the existential.

The interpretation of the third quantifier *a<sup>z</sup> mustache*, i.e.,  $\exists z(\text{MUSTACHE}(z) \wedge \dots)$ , does not bring anything new *per se*. It takes narrow scope relative to the previous existential (hence also relative to the universal), so it makes its contribution relative to context  $i$ : we bring to salience an individual  $z$  that is required to be a mustache that the previously mentioned man  $y$  has, i.e., there is some  $j$  s.t.  $i[z]j$  and  $j(z) \in \mathbf{I}(\text{MUSTACHE})$  and  $\langle j(y), j(z) \rangle \in \mathbf{I}(\text{HAVE})$ . Again, bringing to salience the mustache  $z$  preserves the previously introduced information about  $y$ :  $j(y) = i(y)$  since, by the definition of  $i[z]j$ ,  $i$  differs from  $j$  at most with respect to the value of  $z$ .

What is different about the third quantifier *a<sup>z</sup> mustache* is that its scope ends before the final subformula  $\text{SEE}(x, y)$  is interpreted. Therefore, this subformula is not interpreted relative to the ‘final’ assignment  $j$ , but relative to the ‘intermediate’ assignment  $i$ , which is the most local / recent context before the assignment manipulation contributed by *a<sup>z</sup> mustache*.

This is a very important feature of classical FOL semantics: quantifiers erase the variable assignment manipulation they contribute after the subformulas in their scope are interpreted and the context of interpretation is reset to whatever it was before the quantifier was interpreted.

Thus, relative to a particular quantifier, the interpretation of formulas *inside* its scope is fundamentally different from the interpretation of formulas *outside* its scope: the former are interpreted as expected, i.e., we take into account the information contributed by the quantifier, while the latter are interpreted as if the quantifier has been ‘unsaid’. This feature of FOL and its ‘unsaying’ consequences make classical static semantics a less than adequate formalization of natural language interpretation.



### 3 Problems with FOL Quantification: Anaphora across Conjuncts and Donkey Anaphora

The fact that FOL quantifiers erase variable-assignment manipulations after the subformulas in their scope are interpreted makes it impossible for us to define a translation procedure from English into FOL that is *compositional down to clausal level* (which is the most we can expect from first-order logic anyway).

#### 3.1 Anaphora across (Clausal) Conjuncts

In particular, we cannot compositionally translate the variations on example (7) above provided in (18) and (19) below. These examples contain an instance of anaphora between the indefinite  $a^z$  *mustache* and the pronoun  $it_z$  that are located in two conjoined verb phrases (VPs) in (18) and in two conjoined clauses in (19). Antecedents are superscripted with the variable they introduce and anaphors are subscripted with the variable they retrieve.

(18) Every <sup>$x$</sup>  woman saw a <sup>$y$</sup>  man who had a <sup>$z$</sup>  mustache and was twisting it <sub>$z$</sub> .

(19) Every <sup>$x$</sup>  woman saw a <sup>$y$</sup>  man who had a <sup>$z$</sup>  mustache and who was twisting it <sub>$z$</sub> .

Such relationships that cross conjoined VP or clausal boundaries are syntactically non-local and are subject to the so-called Coordinate Structure Constraint. This constraint bans the asymmetric syntactic displacement of an expression from only one of two conjuncts.

For example, based on the English sentence in (20) below, we cannot form the asymmetric question in (21), but we can form a symmetric question like the one in (22). The symbol ‘\*’ indicates infelicity, due to either ungrammaticality or the unavailability of a particular reading.

(20) Linus has a mustache and frequently twists it.

(21) \*What does Linus have and frequently twist(s) it?

(22) What does Linus have and frequently twist?

This constraint on syntactic structures is important for the analysis of anaphora in examples like (18) and (19) because it rules out certain syntactic structures, thereby making the task of compositionally deriving their truth conditions more difficult.

To see what the difficulty is, consider first the formula in (23) below. This formula does not derive the intuitively correct truth conditions for (18) because the variable  $z$  in the conjunct  $\text{TWIST}(y, z)$  is free and not bound by the preceding existential quantifier  $\exists z$ .

$$(23) \quad \forall x(\text{WOMAN}(x) \rightarrow \exists y(\text{MAN}(y) \wedge \exists z(\text{MUSTACHE}(z) \wedge \text{HAVE}(y, z)) \wedge \text{TWIST}(y, z) \wedge \text{SEE}(x, y))))$$

Similarly, interpreting the pronoun  $it_z$  in (18) as if it was a duplicate of the indefinite  $a^z$  *mustache*, i.e., in terms of an existential quantifier, also fails to derive the intuitively correct truth conditions: the formula in (24) below is true in a situation in which the men under consideration twist someone else’s mustache and not their own – while sentence (18) is false in such a situation.

$$(24) \quad \forall x(\text{WOMAN}(x) \rightarrow \exists y(\text{MAN}(y) \wedge \exists z(\text{MUSTACHE}(z) \wedge \text{HAVE}(y, z)) \wedge \exists z(\text{TWIST}(y, z)) \wedge \text{SEE}(x, y))))$$

What we actually need is a formula like the one in (25) below, where the existential quantifier contributed by the indefinite binds the variable contributed by the pronoun.

$$(25) \quad \forall x(\text{WOMAN}(x) \rightarrow \exists y(\text{MAN}(y) \wedge \exists z(\text{MUSTACHE}(z) \wedge \text{HAVE}(y, z) \wedge \text{TWIST}(y, z)) \wedge \text{SEE}(x, y)))$$

This formula, however, cannot be obtained by a compositional, clause-by-clause (or VP-by-VP) translation from English into FOL because the existential quantification contributed by the indefinite *a<sup>z</sup> mustache* has scope only over the first VP or clausal conjunct in examples (18) and (19) above.

Moreover, the indefinite cannot take scope over both clauses or VPs as a result of a covert syntactic manipulation of the English sentence along the lines shown in (26) below.

$$(26) \quad \text{Every}^x \text{ woman saw a}^y \text{ man who } \mathbf{A}^z \text{ MUSTACHE } \text{ had } \text{---}_z \text{ and was twisting it}_z.$$

Meant to be interpreted as:

every woman  $x$  saw a man  $y$  s.t. there is a mustache  $z$  that  $y$  had and was twisting.

Such a covert syntactic manipulation would violate the Coordinate Structure Constraint mentioned above, which bans the asymmetric displacement of an expression from only one of two conjuncts.

Another argument against the possibility of the indefinite *a<sup>z</sup> mustache* covertly taking scope over the two conjuncts in (18) or (19) above is provided by the example in (27) below. This example shows that a universal quantifier in the first relative clause cannot bind a pronoun in the second relative clause. But such binding should be possible if covert movement of the kind shown in (26) above were possible.

$$(27) \quad * \text{Every}^x \text{ girl talked to a}^y \text{ boy who had bought every}^z \text{ ‘Harry Potter’ book and (who) had read it}_z \text{ several times.}$$

Meant to be interpreted as:

every girl  $x$  talked to a boy  $y$  s.t. for every ‘Harry Potter’ book  $z$ ,  $y$  bought and repeatedly read  $z$ .

### 3.2 Donkey Anaphora

The same observations apply to the donkey sentence in (28) below, based on an example in Geach (1962).

$$(28) \quad \text{Every}^x \text{ farmer who owns a}^y \text{ donkey beats it}_y.$$

We do not obtain the intuitively correct truth conditions if we compositionally translate this sentence into FOL and interpret the pronoun either as contributing a free variable or as somehow duplicating the existential quantification contributed by the indefinite – as shown in (29) and (30) below, respectively.

$$(29) \quad \forall x(\text{FARMER}(x) \wedge \exists y(\text{DONKEY}(y) \wedge \text{OWN}(x, y)) \rightarrow \text{BEAT}(x, y))$$

$$(30) \quad \forall x(\text{FARMER}(x) \wedge \exists y(\text{DONKEY}(y) \wedge \text{OWN}(x, y)) \rightarrow \exists y(\text{BEAT}(x, y)))$$

The correct FOL translation of sentence (28) is provided in (31). This translation, however, is not compositionally obtained: there is no subformula in (31) that corresponds to the indefinite *a<sup>y</sup> donkey* in (28) or to the entire relative clause *who owns a<sup>y</sup> donkey*.

$$(31) \quad \forall x \forall y ( \text{FARMER}(x) \wedge \text{DONKEY}(y) \wedge \text{OWN}(x, y) \rightarrow \text{BEAT}(x, y) )$$

Moreover, scoping out the indefinite  $a^y$  *donkey* so that it can bind the pronoun  $it_y$  involves a covert syntactic manipulation that is not possible in English. To see this, consider the sentence in (32) below. The universal quantifier *every<sup>y</sup> ‘Harry Potter’ book* in this sentence is in the same syntactic position as the indefinite  $a^y$  *donkey* in sentence (28) above.

$$(32) \quad * \text{Every}^x \text{ boy who bought every}^y \text{ ‘Harry Potter’ book read } it_y \text{ several times.}$$

Meant to be interpreted as:

every boy  $x$  who bought every ‘Harry Potter’ book  $y$  is s.t. he repeatedly read every ‘Harry Potter’ book  $y$  he bought.

However, it is not possible for the universal quantifier to scope out of its relative clause and bind the pronoun  $it_y$ : sentence (32) does not have a reading according to which every boy who bought all the ‘Harry Potter’ books repeatedly read every one of them. Correspondingly, the indefinite  $a^y$  *donkey* in sentence (28) cannot be syntactically scoped out so that it can bind the pronoun  $it_y$ .

## 4 Dynamic Predicate Logic (DPL) and the Decomposition of FOL Quantification

Dynamic Predicate Logic (DPL, Groenendijk & Stokhof 1991) couches Discourse Representation Theory (DRT, Kamp 1981, Kamp & Reyle 1993) and File Change Semantics (FCS, Heim 1982) in the familiar language of classical FOL and it enables us to provide a compositional translation procedure for sentences (18) and (19) above, which exemplify cross-VP / cross-clausal anaphora, and for sentence (28), which exemplifies donkey anaphora.

This is a consequence of the fact that the equivalences in (33) and (34) below hold in DPL *without* the usual FOL restrictions, namely that  $x$  must not occur free in  $\psi$ .

$$(33) \quad (\exists x \phi) \wedge \psi \Leftrightarrow \exists x (\phi \wedge \psi)$$

$$(34) \quad (\exists x \phi) \rightarrow \psi \Leftrightarrow \forall x (\phi \rightarrow \psi)$$

The symbol “ $\Leftrightarrow$ ” requires identity of denotation:  $\phi \Leftrightarrow \psi$  iff  $\llbracket \phi \rrbracket = \llbracket \psi \rrbracket$ . This will apply to all the different definitions of the interpretation function  $\llbracket \cdot \rrbracket$  discussed in the paper.

The equivalence in (33) ensures that the *compositional* translation of sentence (18) in (23) above is equivalent to the *truth-conditionally correct* translation in (25). All these are repeated below for convenience.

$$(35) \quad \text{Every}^x \text{ woman saw } a^y \text{ man who had } a^z \text{ mustache and was twisting } it_z.$$

$$(36) \quad \forall x (\text{WOMAN}(x) \rightarrow \exists y (\text{MAN}(y) \wedge \exists z (\text{MUSTACHE}(z) \wedge \text{HAVE}(y, z)) \wedge \text{TWIST}(y, z) \wedge \text{SEE}(x, y))) \Leftrightarrow \forall x (\text{WOMAN}(x) \rightarrow \exists y (\text{MAN}(y) \wedge \exists z (\text{MUSTACHE}(z) \wedge \text{HAVE}(y, z) \wedge \text{TWIST}(y, z) \wedge \text{SEE}(x, y))))$$

Similarly, the equivalence in (34) ensures that the *compositional* translation of sentence (28) given in (29) above is equivalent to the *truth-conditionally correct* translation in (31).

(37) Every<sup>*x*</sup> farmer who owns a<sup>*y*</sup> donkey beats it<sub>*y*</sub>.

(38)  $\forall x(\text{FARMER}(x) \wedge \exists y(\text{DONKEY}(y) \wedge \text{OWN}(x, y)) \rightarrow \text{BEAT}(x, y)) \quad \Leftrightarrow$   
 $\forall x\forall y( \text{FARMER}(x) \wedge \text{DONKEY}(y) \wedge \text{OWN}(x, y) \rightarrow \text{BEAT}(x, y) )$

The DPL semantics of quantification differs from the classical static FOL semantics in only one respect. Just like in FOL, quantifiers are interpreted as manipulating variable assignments in a pointwise (i.e., variablewise) manner and passing the resulting assignments to the subformulas in their scope. But unlike FOL, DPL quantifiers do not erase the variable assignment manipulations after the subformulas in their scope are interpreted.

Thus, the DPL interpretation function records not only the input variable assignment  $g$  relative to which a formula is interpreted, but also the output variable assignment  $h$  that is the result of the manipulations / updates contributed by the quantifiers in the interpreted formula.

This single modification in the definition of the interpretation function shifts our perspective on natural language meaning in a couple of ways.

Static approaches (along FOL lines) equate the meaning of a sentence with its truth conditions, i.e., the contexts in which a sentence is true or false. For example, the set-based semantics for classical FOL in (15) above takes contexts to be variable assignments and the meaning of any formula is the set of contexts, i.e., the set of variable assignments, relative to which the formula is true. Switching from set talk to function talk, this gives us the familiar view (see Kaplan 1977/1989 among others) of sentence meaning as a function that takes in a context (or index) of evaluation  $g$  and gives back a truth-value  $\mathbb{T}$  or  $\mathbb{F}$ .

In contrast, dynamic approaches (along DPL lines) have a finer-grained conception of meaning: the meaning of a sentence is its context-change potential, i.e., the way in which the sentence changes / updates a context. The fact that natural language interpretation is context dependent, i.e., relativized to an input assignment  $g$ , is explicitly investigated in both kinds of approaches. But only dynamic approaches systematically investigate how the interpretation of a natural language expression, e.g., an indefinite, changes the context (i.e., it creates a new context out of the old one) and therefore affects how subsequent expressions, e.g., a pronoun, are interpreted.

The meaning of a formula in DPL is not a set of variable assignments (or, equivalently, a function from variable assignments to truth values), but a binary relation between assignments, i.e., a set of *pairs* of assignments  $\langle g, h \rangle$ :  $g$  is the input context relative to which the formula is interpreted and  $h$  is the output context, the context that results after the input context  $g$  is updated with the formula.

Contexts of interpretation are now finer grained: they are not simply variable assignments, but pairs of variable assignments. In function talk, the meaning of a formula is not a function from assignments to truth values, but a function from pairs of assignments to truth values.

Furthermore, we can ‘Schönfinkelize’ or ‘Curry’ such functions, i.e., re-express binary relations over assignments as functions from assignments to functions from assignments to truth values. In this Schönfinkelized version, the meaning of a DPL formula  $\phi$  is a function from an input assignment  $g$  to the set of assignments  $h, h', h'', \dots$  that are the possible output assignments when the input assignment  $g$  is changed / updated with the formula  $\phi$ .

This is parallel to the way in which English transitive verbs like *employ*, *kiss* etc. are not interpreted as binary relations between individuals (e.g., employers and employees) in (neo)Montagovian semantics. Instead, they are interpreted as functions from individuals to functions from individuals to truth values, so that they match the syntax of English better. On the syntactic side, the verb combines with the direct object first to form a VP and only afterwards with the subject to form a sentence. On the semantic side, the denotation of the verb takes the denotation of the direct object

as its first argument yielding a function from individuals to truth values; then, this function takes the denotation of the subject as its argument yielding a truth value.

Schönfinkelized / Curried binary relations over assignments provide a better format for certain purposes. But we will continue to use the notationally simpler set talk for the remainder of this paper when we define the semantics of various logics. The reader should feel free to reformulate everything in function talk if s/he finds it more intuitive; in fact, doing so might be a good way to ensure a thorough understanding of the definitions.

The DPL definition of the interpretation function  $\llbracket \cdot \rrbracket$  is provided in (41) below. As already indicated, formulas are interpreted as binary relations between variable assignments, i.e., the space of DPL denotations for formulas is  $\wp(\mathcal{G} \times \mathcal{G})$  and not simply  $\wp(\mathcal{G})$  as FOL would have it.

That is, the denotation of a formula  $\phi$  in FOL is a set  $\mathcal{S}$  of assignments of the form  $\{g, g', g'', \dots\}$ . This set  $\mathcal{S}$  is a subset of the set of all assignments  $\mathcal{G}$ :  $\mathcal{S} \subseteq \mathcal{G}$ . Equivalently,  $\mathcal{S}$  is an element of the powerset of  $\mathcal{G}$ :  $\mathcal{S} \in \wp(\mathcal{G})$ . In other words,  $\wp(\mathcal{G})$  is the space of all possible FOL denotations for formulas.

In contrast, the denotation of a formula  $\phi$  in DPL is a binary relation  $\mathcal{R}$  between assignments of the form  $\{\langle g, h \rangle, \langle g', h' \rangle, \langle g'', h'' \rangle, \dots\}$ . This relation  $\mathcal{R}$  is a subset of the Cartesian product  $\mathcal{G} \times \mathcal{G}$  over the set of all assignments:  $\mathcal{R} \subseteq \mathcal{G} \times \mathcal{G}$ . Equivalently,  $\mathcal{R}$  is an element of the powerset of this Cartesian product:  $\mathcal{R} \in \wp(\mathcal{G} \times \mathcal{G})$ . In other words,  $\wp(\mathcal{G} \times \mathcal{G})$  is the space of all possible DPL denotations for formulas.

For any binary relation  $\mathcal{R}$  over variable assignments, we will define its domain  $\mathbf{Dom}(\mathcal{R})$  as the set of all assignments that are first members of the pairs in  $\mathcal{R}$ . Correspondingly, the range  $\mathbf{Ran}(\mathcal{R})$  is the set of all assignments that are second members of the pairs in  $\mathcal{R}$ .

$$(39) \quad \mathbf{Dom}(\mathcal{R}) := \{g : \text{there exists some } h \text{ s.t. } \langle g, h \rangle \in \mathcal{R}\}$$

$$(40) \quad \mathbf{Ran}(\mathcal{R}) := \{h : \text{there exists some } g \text{ s.t. } \langle g, h \rangle \in \mathcal{R}\}$$

Given a formula  $\phi$ , its DPL denotation  $\llbracket \phi \rrbracket$  is a binary relation over assignments. The domain of this relation  $\mathbf{Dom}(\llbracket \phi \rrbracket)$  is the set of all assignments  $g$  that can be input assignments for  $\phi$ . That is,  $\mathbf{Dom}(\llbracket \phi \rrbracket)$  is the set of all assignments  $g$  relative to which  $\phi$  is true. The range of this relation  $\mathbf{Ran}(\llbracket \phi \rrbracket)$  is the set of all assignments  $h$  that are output assignments after some input assignment or other is updated with  $\phi$ . That is,  $\mathbf{Ran}(\llbracket \phi \rrbracket)$  is the set of all assignments that are the result of the update contributed by  $\phi$ ; subsequent formulas are interpreted relative to these assignments.

(41) **DPL semantics – first  $\wp(\mathcal{G} \times \mathcal{G})$  version** (Groenendijk & Stokhof 1991)

a. Atomic formulas:

i. If  $\pi$  is an  $n$ -place predicate and  $\alpha_1, \dots, \alpha_n$  are terms, then  $\llbracket \pi(\alpha_1, \dots, \alpha_n) \rrbracket = \{\langle g, g \rangle : \langle g/\mathbf{I}(\alpha_1), \dots, g/\mathbf{I}(\alpha_n) \rangle \in \mathbf{I}(\pi)\}$ .

ii. If  $\alpha$  and  $\beta$  are terms, then  $\llbracket \alpha = \beta \rrbracket = \{\langle g, g \rangle : g/\mathbf{I}(\alpha) = g/\mathbf{I}(\beta)\}$ .

b. Formulas (sentential connectives):

i.  $\llbracket \sim \phi \rrbracket = \{\langle g, g \rangle : \text{there is no } h \text{ s.t. } \langle g, h \rangle \in \llbracket \phi \rrbracket\}$   
 $= \{\langle g, g \rangle : g \notin \mathbf{Dom}(\llbracket \phi \rrbracket)\}$

ii.  $\llbracket \phi; \psi \rrbracket = \{\langle g, h \rangle : \text{there is an } i \text{ s.t. } \langle g, i \rangle \in \llbracket \phi \rrbracket \text{ and } \langle i, h \rangle \in \llbracket \psi \rrbracket\}$   
 $= \llbracket \phi \rrbracket \circ \llbracket \psi \rrbracket$

c. Formulas (quantifiers):

i.  $\llbracket \exists v \phi \rrbracket = \{\langle g, h \rangle : \text{there is an } i \text{ s.t. } g[v]i \text{ and } \langle i, h \rangle \in \llbracket \phi \rrbracket\}$

d. Based on the abbreviations in (8e) above, we derive the following:

- i.  $\llbracket \phi \vee \psi \rrbracket = \llbracket \sim (\sim \phi; \sim \psi) \rrbracket$   
 $= \{ \langle g, g \rangle : g \in \mathbf{Dom}(\llbracket \phi \rrbracket) \cup \mathbf{Dom}(\llbracket \psi \rrbracket) \}$
  - ii.  $\llbracket \phi \rightarrow \psi \rrbracket = \llbracket \sim (\phi; \sim \psi) \rrbracket$   
 $= \{ \langle g, g \rangle : \text{for all } h \text{ s.t. } \langle g, h \rangle \in \llbracket \phi \rrbracket, \text{ there is an } i \text{ s.t. } \langle h, i \rangle \in \llbracket \psi \rrbracket \}$   
 $= \{ \langle g, g \rangle : \{ h : \langle g, h \rangle \in \llbracket \phi \rrbracket \} \subseteq \mathbf{Dom}(\llbracket \psi \rrbracket) \}$
  - iii.  $\llbracket \forall v \phi \rrbracket = \llbracket \sim \exists v (\sim \phi) \rrbracket$   
 $= \{ \langle g, g \rangle : \text{for all } h \text{ s.t. } g[v]h, \text{ there is an } i \text{ s.t. } \langle h, i \rangle \in \llbracket \phi \rrbracket \}$   
 $= \{ \langle g, g \rangle : \{ h : g[v]h \} \subseteq \mathbf{Dom}(\llbracket \phi \rrbracket) \}$
- e. Truth:
- i. A formula  $\phi$  is true in model  $\mathfrak{M}$  relative to an input assignment  $g$  iff there is an output assignment  $h$  s.t.  $\langle g, h \rangle \in \llbracket \phi \rrbracket$ , i.e., iff  $g \in \mathbf{Dom}(\llbracket \phi \rrbracket)$ .

The finer-grained pair-based DPL denotations are otiose for atomic formulas and for dynamic negation, symbolized as “ $\sim$ ”. Their interpretation is basically the same as their static interpretation in (15) above except that instead of saying that an assignment  $g$  satisfies these formulas, we say that a pair  $\langle g, g \rangle$  that has the same assignment as its first and second member satisfies these formulas.

Consider, for example, the atomic formula  $\text{HUNGRY}(x)$ , which is the translation of an English sentence like *It<sub>x</sub> is hungry*. And imagine I’m pointing to an already salient duck  $x$  while saying this sentence.

The formula  $\text{HUNGRY}(x)$  takes in an input assignment  $g$ , but does not modify it in any way: it does not bring to salience any new entities. It simply refers back to the already salient duck  $x$  and claims that the duck has the property of being hungry. The output context is therefore the same as the input context, namely  $g$ , so the meaning of the formula  $\text{HUNGRY}(x)$  consists of pairs of assignments  $\langle g, g \rangle$  whose first and second members are identical.

This does not mean that the update contributed by  $\text{HUNGRY}(x)$  is vacuous. Consider the classical FOL meaning of the formula  $\text{HUNGRY}(x)$ : it is the set of assignments  $g$  s.t. the entity  $g(x)$  is hungry. If we have an assignment  $g'$  s.t.  $g'(x)$  is not the hungry duck we were just talking about, but my dog Linus that just snuck into the kitchen and ate 2 pounds of ground beef, this assignment is not in the denotation of the formula  $\text{HUNGRY}(x)$  (assuming for the sake of the argument that after eating 2 pounds of meat, Linus is not hungry anymore).

Thus, the formula  $\text{HUNGRY}(x)$  allows the assignment  $g$  (because  $g(x)$  is a hungry duck) but rules out the assignment  $g'$  (because  $g'(x)$  is an over-satiated dog). Ruling out such assignments is the non-trivial update contributed by the formula  $\text{HUNGRY}(x)$ .

In DPL terms, the pair  $\langle g, g \rangle$  is in the set  $\llbracket \text{HUNGRY}(x) \rrbracket$ , while the pair  $\langle g', g' \rangle$  is not. Atomic formulas are *tests*: they do not change input assignments, they simply test that an input assignment satisfies a certain condition and if it does, they pass it on as is. If an input assignment does not satisfy the condition, it is not a live candidate for the actual context of interpretation, so it is rejected and will not be under consideration for any subsequent update.

Let us turn now to the clause for dynamic negation, namely  $\llbracket \sim \phi \rrbracket = \{ \langle g, g \rangle : g \notin \mathbf{Dom}(\llbracket \phi \rrbracket) \}$ . By default, we expect negation to be interpreted as set complementation. This is exactly its classical FOL interpretation provided in (15) above:  $\llbracket \neg \phi \rrbracket = \mathcal{G} \setminus \llbracket \phi \rrbracket$ . That is, the denotation of the formula  $\neg \phi$  is the complement of the denotation of the formula  $\phi$ , namely the set containing all the variable assignments that are not in the set denoted by  $\phi$ .

DPL semantics is formulated in terms of sets of pairs of assignments instead of sets of assignments, so the first possibility that comes to mind is that the DPL denotation of a negative formula  $\sim \phi$  is the complement of the set of pairs denoted by  $\phi$ , i.e.,  $\{ \langle g, h \rangle : \langle g, h \rangle \notin \llbracket \phi \rrbracket \} = (\mathcal{G} \times \mathcal{G}) \setminus \llbracket \phi \rrbracket$ .

However, this denotation does not make correct predictions with respect to the interpretation of natural language negation. Take a sentence like (42) below, translated as shown in (43).

(42) It<sub>*x*</sub> is not hungry.

(43)  $\sim \text{HUNGRY}(x)$

If negation was interpreted as complementation over sets of pairs of assignments,  $\llbracket \sim \text{HUNGRY}(x) \rrbracket$  would be the set of pairs  $\{\langle g, h \rangle : \langle g, h \rangle \notin \llbracket \text{HUNGRY}(x) \rrbracket\}$ .

Now,  $\llbracket \text{HUNGRY}(x) \rrbracket$  is the set of pairs:

$$\{\langle g, g \rangle : g(x) \in \mathbf{I}(\text{HUNGRY})\} = \{\langle g, h \rangle : g = h \text{ and } g(x) \in \mathbf{I}(\text{HUNGRY})\}$$

So  $\llbracket \sim \text{HUNGRY}(x) \rrbracket$  is the complement of this set, namely:

$$\{\langle g, h \rangle : g \neq h \text{ or } g(x) \notin \mathbf{I}(\text{HUNGRY})\}$$

But this means that any pair of assignments  $\langle g, h \rangle$  is in the denotation of the negative formula  $\sim \text{HUNGRY}(x)$  as long as  $g$  and  $h$  are different.

For example, the output assignment  $h$  might differ from the input assignment  $g$  with respect to some variable  $y$  and store in  $y$  a crying baby boy. We would therefore predict that we can refer back to this crying boy and felicitously follow up the sentence in (42) above with a sentence like *He<sub>*y*</sub> misses his<sub>*y*</sub> mother*, which is meant to be interpreted as: the crying baby boy misses his mother.

However, the negative sentence in (42) above simply makes a claim about the duck  $x$ . It does not say anything about the referent of the variable  $y$  – or any other variable for that matter. It does not bring any new individuals to salience.

Thus, we do not want to interpret negation as complementation relative to sets of pairs of assignments because we would incorrectly predict that negation licenses arbitrary anaphoric relations.

We can make sure that a negative formula  $\sim \phi$  preserves the current discourse state / context by interpreting it as complementation relative to the *domain* of the denotation of  $\phi$  – which is the DPL definition in (41b-i) above.

The domain of the binary relation  $\llbracket \phi \rrbracket$  is the set of assignments that can be updated with  $\phi$ , i.e., the set of assignments that make  $\phi$  true. The complement of this set contains all the assignments that cannot be successfully updated with  $\phi$ , that is, the assignments that make  $\phi$  false – and this is precisely the truth-conditional contribution that sentential negation makes.

According to this definition,  $\llbracket \sim \text{HUNGRY}(x) \rrbracket$  is the set of pairs:

$$\{\langle g, g \rangle : g \notin \llbracket \text{HUNGRY}(x) \rrbracket\} = \{\langle g, g \rangle : g(x) \notin \mathbf{I}(\text{HUNGRY})\}$$

That is, we simply test that the entity  $g(x)$  stored by the input assignment  $g$  is not hungry and if it isn't, we pass the assignment  $g$  on. We get the truth conditions right and we do not spuriously license anaphoric relations.

It might turn out that this definition of negation is too restrictive as far as licensing anaphoric relations is concerned, but it is a good attempt to generalize the FOL definition of negation and we will henceforth stick with it.

While the relational denotations are ultimately not crucial for atomic or negated formulas, they are essential for the interpretation of dynamic conjunction, symbolized as “;”, and dynamic existential quantification.

Dynamic conjunction is interpreted as relation composition  $\circ$ , so the second conjunct is interpreted relative to the assignment that is the output of the first conjunct. Given two binary relations over assignments  $\mathcal{R}$  and  $\mathcal{R}'$ , their composition is defined as shown below.

$$(44) \quad \mathcal{R} \circ \mathcal{R}' := \{\langle g, h \rangle : \text{there exists some } i \text{ s.t. } \langle g, i \rangle \in \mathcal{R} \text{ and } \langle i, h \rangle \in \mathcal{R}'\}$$

Dynamic conjunction is the crucial dynamic sentential operator: it ensures the continuity of interpretation in discourse, i.e., it encodes the fact that later utterances are interpreted relative to the context provided by what has been said earlier.

Dynamic existential quantification preserves the variable assignment manipulation contributed by the quantifier and stores it in the output assignment. Then, the formulas in the scope of an existential quantifier are interpreted relative to this modified output assignment – *as are the formulas following the existential quantifier*.

Existential quantifiers (i.e., indefinites) and subsequent anaphora to them provide the motivation for the fact that DPL formulas denote *relations* between assignments and not functions. That is, context update is non-deterministic: in general, multiple output contexts are compatible with any given update and subsequent updates are used to further narrow down this set of output contexts.

Consider the sentence in (45a) below. There are many output assignments  $h, h', h'', \dots$  after we update a particular input assignment  $g$  with this sentence, one output assignment for every burger I had at Zachary’s:  $h(x)$  stores the burger I had in 2008 before Thanksgiving,  $h'(x)$  stores the burger I had in 2009 that one Saturday I went to the beach,  $h''(x)$  stores the burger I had in 2010 right after I moved (again) etc.

- (45) a. I had a<sup>x</sup> burger at Zachary’s once.  
 b. It<sub>x</sub> was the best burger I ever had.

The follow-up sentence in (45b) further elaborates on the burger and winnows down these output contexts by selecting that one context  $h''$  storing the best burger ever, namely  $h''(x)$ . This is possible precisely because the update contributed by sentence (45a) is non-deterministic / relational and not deterministic / functional: had it been a function, it could have yielded assignment  $h$  as its sole output, in which case the update contributed by sentence (45b) wouldn’t have gone through and we would incorrectly predict the two-sentence discourse in (45) as a whole to be infelicitous.

Dynamic conjunction “;” is associative, just like classical static conjunction “^”:

$$(46) \quad (\phi; \psi); \chi \Leftrightarrow \phi; (\psi; \chi)$$

But unlike classical static conjunction, dynamic conjunction is not commutative or idempotent, as shown by the examples below:

$$(47) \quad \llbracket (\sim \mathbf{P}(x)); \exists x \mathbf{P}(x) \rrbracket \neq \llbracket \exists x \mathbf{P}(x); (\sim \mathbf{P}(x)) \rrbracket$$

$$(48) \quad \llbracket (\sim \mathbf{P}(x)); \exists x \mathbf{P}(x) \rrbracket \neq \llbracket (\sim \mathbf{P}(x)); \exists x \mathbf{P}(x); (\sim \mathbf{P}(x)); \exists x \mathbf{P}(x) \rrbracket$$

As a final example that brings together most of the above observations, let us analyze the simple two-sentence discourse in (49) below. It is compositionally translated as shown in (50).

- (49) a. A<sup>x</sup> squirrel ran by.  
 b. It<sub>x</sub> was hungry.

$$(50) \quad \exists x(\text{SQUIRREL}(x); \text{RUN-BY}(x)); \text{HUNGRY}(x)$$

Suppose that our input context is an assignment  $g$  that assigns some arbitrary values to all variables. The conjunction of formulas in (50) above updates this input context as shown in (51) below.

The update in (51) proceeds as follows (recall that the denotations of our formulas are binary relations between sets of assignments, not functions). We first introduce  $x$  and assign it an entity



that is a squirrel and that ran by. The result: many contexts / variable assignments that differ from  $g$  at most with respect to the value of  $x$ . That is, we now have a graph with many paths – as many paths as there are squirrels that ran by:  $squirrel_1$ ,  $squirrel_2$ ,  $squirrel_3$  etc.

Then, the test  $HUNGRY(x)$  eliminates some of the paths in the graph, namely all those paths that end in a context / assignment where  $x$  is not assigned a hungry squirrel. In our case,  $squirrel_1$  is not hungry, so that particular assignment is eliminated.

$$(51) \quad g \xrightarrow{\exists x(\text{SQUIRREL}(x); \text{RUN-BY}(x))} \begin{array}{c} \dots \quad x \quad \dots \\ \boxed{\dots \quad squirrel_1 \quad \dots} \\ \dots \quad x \quad \dots \\ \boxed{\dots \quad squirrel_2 \quad \dots} \\ \dots \quad x \quad \dots \\ \boxed{\dots \quad squirrel_3 \quad \dots} \\ \dots \end{array} \xrightarrow{HUNGRY(x)} \begin{array}{c} \dots \quad x \quad \dots \\ \boxed{\dots \quad squirrel_2 \quad \dots} \\ \dots \quad x \quad \dots \\ \boxed{\dots \quad squirrel_3 \quad \dots} \\ \dots \end{array}$$

This interpretation graph is in no way different from the way interpretation proceeds in classical FOL, except for the fact that we do not erase variable assignment manipulations. Such graphs are implicit in the recursive definitions of truth and satisfaction in classical FOL.

We can depict the sequence of updates in (51) more simply by choosing a single, typical path through the graph:

$$(52) \quad g \xrightarrow{\exists x(\text{SQUIRREL}(x); \text{RUN-BY}(x))} \begin{array}{c} \dots \quad x \quad \dots \\ \boxed{\dots \quad squirrel_2 \quad \dots} \end{array} \xrightarrow{HUNGRY(x)} \begin{array}{c} \dots \quad x \quad \dots \\ \boxed{\dots \quad squirrel_2 \quad \dots} \end{array}$$

We can fairly easily check that the DPL semantics for the language of first-order logic makes the equivalences in (33) and (34) above true. This will ensure that we can give a compositional account of cross-conjunct, cross-sentential and donkey anaphora.

The crucial question is: what particular aspects of DPL semantics make the existential and universal quantifiers behave in such a way that the equivalences in (33) and (34) are true?

The answer is that DPL semantics fully decomposes first-order quantification along the lines that we already suggested for classical FOL semantics.

That is, DPL clearly separates the pointwise assignment manipulation encoded by  $g[v]h$ , which is common to the existential and the universal quantifier, from assignment ‘counting’, which distinguishes the two quantifiers.

We can see this clearly if we make pointwise assignment manipulation  $[v]$  a well-formed formula in our first-order language. This formula is interpreted as shown in (53) below.

$$(53) \quad \llbracket [v] \rrbracket = \{ \langle g, h \rangle : g[v]h \}$$

The formula  $[v]$  provides the part that is common to both existential and universal quantification.

Existential and universal quantification are distinguished by the way in which the formula  $[v]$  is related to the formula  $\phi$  that we quantify over, as shown by the equivalences in (54) and (55) below. Existential quantification relates the two formulas by dynamic conjunction, which requires non-empty intersection. Universal quantification relates the two formulas by dynamic implication, which requires inclusion.

$$(54) \quad \text{a. } \exists v \phi \Leftrightarrow [v]; \phi$$

Relative to any input assignment  $g$ , this requires that  $\{h : g[v]h\} \cap \mathbf{Dom}(\llbracket \phi \rrbracket) \neq \emptyset$ .

b. The equivalence in (33) holds because:

$$\begin{aligned} (\exists v\phi); \psi &\Leftrightarrow \\ ([v]; \phi); \psi &\Leftrightarrow \\ [v]; (\phi; \psi) &\Leftrightarrow \\ \exists v(\phi; \psi) & \end{aligned}$$

(55) a.  $\forall v\phi \Leftrightarrow [v] \rightarrow \phi$

Relative to any input assignment  $g$ , this requires that  $\{h : g[v]h\} \subseteq \mathbf{Dom}(\llbracket \phi \rrbracket)$ .

b. The equivalence in (34) holds because:

$$\begin{aligned} (\exists v\phi) \rightarrow \psi &\Leftrightarrow \\ ([v]; \phi) \rightarrow \psi &\Leftrightarrow \\ [v] \rightarrow (\phi \rightarrow \psi) &\Leftrightarrow \\ \forall v(\phi \rightarrow \psi) & \end{aligned}$$

Thus, the final definition of DPL semantics is provided in (56) below. This definition assumes that pointwise assignment manipulation  $[v]$  is a well-formed formula and assigns it an independent denotation. The definition is very brief: the only two basic sentential connectives are dynamic negation and dynamic conjunction, defined in the same way as in (41) above. The only basic “quantified” formula is pointwise assignment manipulation. The remaining sentential connectives and the existential and universal quantifiers are abbreviations of complex formulas that make use only of basic sentential connectives and pointwise assignment manipulation. Their ultimate denotations are identical to the ones provided in (41) above.

(56) **DPL semantics – final  $\wp(\mathcal{G} \times \mathcal{G})$  version**

a. Atomic formulas:

i. If  $\pi$  is an  $n$ -place predicate and  $\alpha_1, \dots, \alpha_n$  are terms, then  $\llbracket \pi(\alpha_1, \dots, \alpha_n) \rrbracket = \{\langle g, g \rangle : \langle g/\mathbf{I}(\alpha_1), \dots, g/\mathbf{I}(\alpha_n) \rangle \in \mathbf{I}(\pi)\}$ .

ii. If  $\alpha$  and  $\beta$  are terms, then  $\llbracket \alpha = \beta \rrbracket = \{\langle g, g \rangle : g/\mathbf{I}(\alpha) = g/\mathbf{I}(\beta)\}$ .

b. Formulas (sentential connectives):

i.  $\llbracket \sim \phi \rrbracket = \{\langle g, g \rangle : g \notin \mathbf{Dom}(\llbracket \phi \rrbracket)\}$

ii.  $\llbracket \phi; \psi \rrbracket = \llbracket \phi \rrbracket \circ \llbracket \psi \rrbracket$

c. Formulas (random assignment):

i.  $\llbracket [v] \rrbracket = \{\langle g, h \rangle : g[v]h\}$

d. Abbreviations:

i.  $\llbracket \phi \vee \psi \rrbracket = \llbracket \sim (\sim \phi; \sim \psi) \rrbracket$   
 $= \{\langle g, g \rangle : g \in \mathbf{Dom}(\llbracket \phi \rrbracket) \cup \mathbf{Dom}(\llbracket \psi \rrbracket)\}$

ii.  $\llbracket \phi \rightarrow \psi \rrbracket = \llbracket \sim (\phi; \sim \psi) \rrbracket$   
 $= \{\langle g, g \rangle : \{h : \langle g, h \rangle \in \llbracket \phi \rrbracket\} \subseteq \mathbf{Dom}(\llbracket \psi \rrbracket)\}$

iii.  $\llbracket \exists v\phi \rrbracket = \llbracket [v]; \phi \rrbracket$   
 $= \{\langle g, g \rangle : \{h : g[v]h\} \cap \mathbf{Dom}(\llbracket \phi \rrbracket) \neq \emptyset\}$

iv.  $\llbracket \forall v\phi \rrbracket = \llbracket [v] \rightarrow \phi \rrbracket$   
 $= \{\langle g, g \rangle : \{h : g[v]h\} \subseteq \mathbf{Dom}(\llbracket \phi \rrbracket)\}$

e. Truth:

i. A formula  $\phi$  is true in model  $\mathfrak{M}$  relative to an input assignment  $g$  iff there is an output assignment  $h$  s.t.  $\langle g, h \rangle \in \llbracket \phi \rrbracket$ , i.e., iff  $g \in \mathbf{Dom}(\llbracket \phi \rrbracket)$ .

The above definitions of universal and existential quantification explicitly capture their two components. The common component is the pointwise assignment manipulation formula  $[v]$ . The component that differs is the ‘counting’ / ‘quantifying’ component, which relates the same two sets of assignments  $\{h : g[v]h\}$  and  $\mathbf{Dom}(\llbracket\phi\rrbracket)$  in different ways:

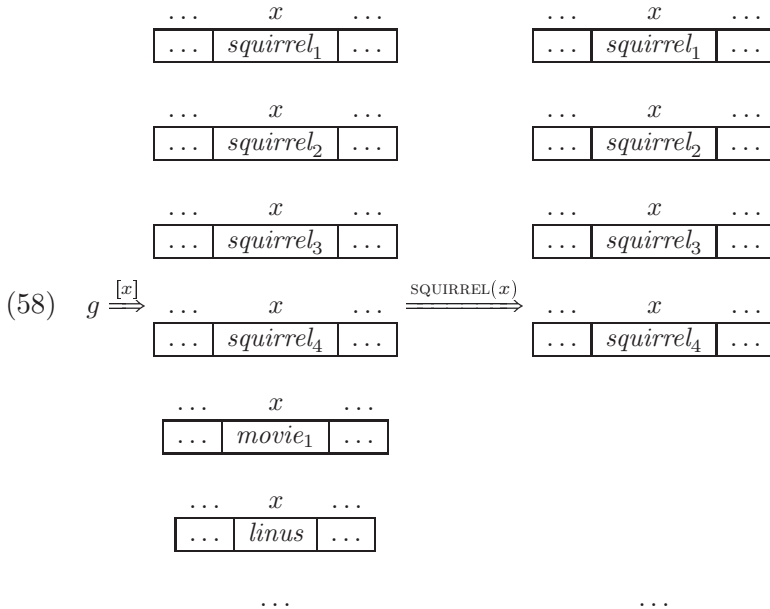
- (i) existential quantification requires non-empty intersection, formalized by means of conjunction “ $\cdot$ ”
- (ii) universal quantification requires inclusion, formalized by means of implication “ $\rightarrow$ ”

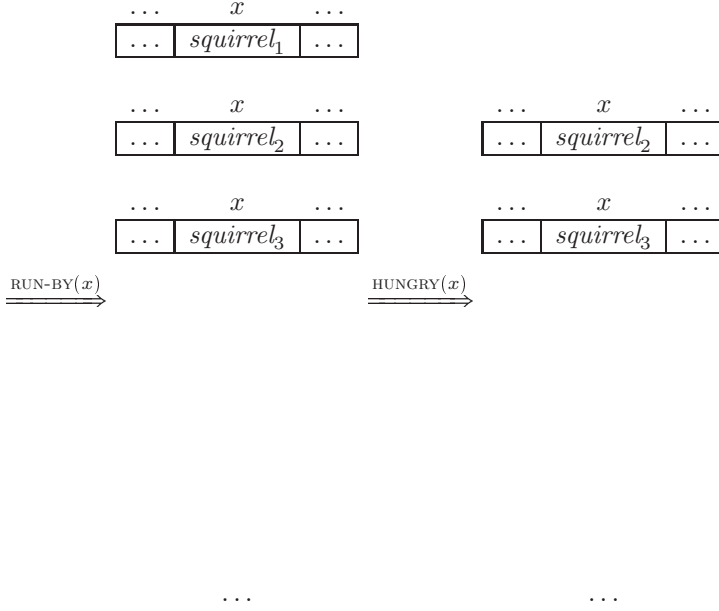
As an example, let us consider the two-sentence discourse in (49) again. The translation we provided above is repeated in (57a) below. However, according to our final version of DPL syntax and semantics, this translation is just an abbreviation of the flat conjunction in (57b).

- (57) a.  $\exists x(\text{SQUIRREL}(x); \text{RUN-BY}(x); \text{HUNGRY}(x))$   
 b.  $[x]; \text{SQUIRREL}(x); \text{RUN-BY}(x); \text{HUNGRY}(x)$

Suppose once again that our input context is an assignment  $g$  that assigns some arbitrary values to all variables. The conjunction of formulas in (57b) above updates this input context as shown in (58) below.

The update in (58) proceeds as follows. We first introduce  $x$ , i.e., assign it a random value. The result: many contexts / variable assignments that differ from  $g$  at most with respect to the value of  $x$  and that assign each individual in  $\mathfrak{D}$  to  $x$ . That is, we now have a graph with many paths. Then, the test  $\text{SQUIRREL}(x)$  eliminates some of the paths in the graph, namely all those paths that end in a context / assignment where  $x$  is not assigned a squirrel. Then, the test  $\text{RUN-BY}(x)$  eliminates further paths in the graph, namely all those in which the squirrel assigned to  $x$  didn’t run by (so  $\text{squirrel}_4$  is eliminated). Finally, the test  $\text{HUNGRY}(x)$  keeps only the contexts in which the squirrel  $x$  that ran by was hungry (so  $\text{squirrel}_1$  is eliminated).





Just as before, we can depict updates by choosing a single, typical path through the graph:

$$(59) \quad g \xrightarrow{[x]} \begin{array}{|c|c|c|} \hline \dots & x & \dots \\ \hline \dots & squirrel_2 & \dots \\ \hline \end{array} \xrightarrow{SQUIRREL(x)} \begin{array}{|c|c|c|} \hline \dots & x & \dots \\ \hline \dots & squirrel_2 & \dots \\ \hline \end{array}$$

$$\xrightarrow{RUN-BY(x)} \begin{array}{|c|c|c|} \hline \dots & x & \dots \\ \hline \dots & squirrel_2 & \dots \\ \hline \end{array} \xrightarrow{HUNGRY(x)} \begin{array}{|c|c|c|} \hline \dots & x & \dots \\ \hline \dots & squirrel_2 & \dots \\ \hline \end{array}$$

Or even more briefly:

$$(60) \quad g \xrightarrow{[x]; SQUIRREL(x); RUN-BY(x); HUNGRY(x)} \begin{array}{|c|c|c|} \hline \dots & x & \dots \\ \hline \dots & squirrel_2 & \dots \\ \hline \end{array}$$

In sum, the theoretical effort of properly understanding first-order quantification, which turns out to be an effort of decomposing first-order quantification, converges with and is empirically motivated by the need to account for anaphoric phenomena in natural language.

The resulting DPL semantics for first-order logic enables us to explicitly formalize the observation that the essence of first-order quantification is pointwise manipulation of variable assignments and show how this observation can be used to compositionally account for cross-conjunct, cross-sentential and donkey anaphora in natural language.

The decomposition of generalized quantification proposed in the remainder of this paper follows the same pattern. We will decompose generalized quantifiers by separating the component that manipulates variable assignments from the truth-conditional component that is the focus of the received account of generalized quantification (see Barwise & Cooper 1981, Keenan & Stavi 1986), namely generalized quantifiers understood as relations between sets of individuals. Just as in the case of FOL and DPL, the decomposition of generalized quantification is empirically motivated and driven by interactions between anaphora and quantification in natural language.

## 5 Adding Generalized Quantification to DPL (DPL+GQ)

As proposed in Barwise & Cooper (1981) and Keenan & Stavi (1986) (see also Westerståhl 1989 and Peters & Westerståhl 2006 for two more recent and comprehensive discussions), we need to add

generalized quantifiers to capture the meaning of a word like *most* in (61) and (62) below – since the quantification expressed by *most* (in the sense of more than half) is not first-order expressible.

(61) Most<sup>x</sup> students left the party after 5 a.m.

(62) Most<sup>x</sup> farmers who own a<sup>y</sup> donkey beat it<sub>y</sub>.

Saying that *most* is not first-order expressible means that it cannot be expressed in terms of the classical FOL existential and / or universal quantifiers. This is perfectly compatible with the fact that *most* and other generalized quantifiers quantify over first-order variables, which are variables over individuals. DPL+GQ (DPL with generalized quantifiers) is a first-order logic in the latter sense, while being strictly more expressive than classical FOL or DPL: DPL+GQ only has first-order variables; and, just like FOL or DPL, its semantics is formulated in terms of single total variable assignments.

Generalized quantifiers are interpreted as binary relations between sets of entities (again, this is compatible with the fact that they quantify over first-order variables only). For example, a quantifier like **MOST** relates two sets of individuals  $X$  and  $X'$  iff more than half of the  $X$ -entities are  $X'$ -entities, as shown in (63) below.

(63)  $\llbracket \mathbf{MOST}(X, X') \rrbracket = \mathbb{T}$  iff  $|\llbracket X \rrbracket \cap \llbracket X' \rrbracket| > |\llbracket X \rrbracket \setminus \llbracket X' \rrbracket|$

Consequently, the English sentence in (61) above is predicted to be true iff more than half of the students left the party after 5 a.m. Or equivalently: the cardinality of the set of students that left the party after 5 a.m., symbolized as  $|\llbracket X \rrbracket \cap \llbracket X' \rrbracket|$ , is strictly greater than the cardinality of the set of students that did not leave the party after 5 a.m., symbolized  $|\llbracket X \rrbracket \setminus \llbracket X' \rrbracket|$ .

The first argument of a generalized quantifier, i.e.,  $X$  in (63) above, is referred to as its *restrictor*. The second argument of a generalized quantifier, i.e.,  $X'$  in (63), is referred to as its *nuclear scope*.

In general, the entities related by generalized quantifiers do not have to be individuals. They can also be variable assignments, i.e., sequences of individuals (and once again, this is compatible with the fact that they quantify over first-order variables only). Arguably, this is what we need to account for the donkey sentence in (62) above and, also, for the donkey conditional containing the adverb of quantification *usually* in (64) below.

(64) Usually, if a<sup>x</sup> farmer owns a<sup>y</sup> donkey, he<sub>x</sub> beats it<sub>y</sub>.

The adverb *usually* is interpreted as the generalized quantifier **MOST**. But instead of two sets of individuals  $X$  and  $X'$ , the quantifier relates two sets of variable assignments  $G$  and  $G'$  satisfying the antecedent and the consequent of the conditional in (64).

## 5.1 Unselective Generalized Quantification

How do we identify the sets of assignments  $G$  and  $G'$  related by the adverb *usually*, i.e., by the generalized quantifier **MOST**( $G, G'$ ), in (64) above?

The answer is implicit in the DPL definition of dynamic implication in (56d-ii) above:

$G$ : relative to any input assignment  $g$ , the first set is the set of all output assignments that satisfy the antecedent relative to  $g$ , i.e.,  $\{h : \langle g, h \rangle \in \llbracket \phi \rrbracket\}$ , henceforth abbreviated as  $g\llbracket \phi \rrbracket$

$G'$ : the second set is the set of assignments in the domain of the consequent, i.e.,  $\mathbf{Dom}(\llbracket \psi \rrbracket)$

We introduce two new abbreviations to make the formalism more readable. First, instead of saying that the pair of assignments  $\langle g, h \rangle$  is in the denotation of the formula  $\phi$ , i.e.,  $\langle g, h \rangle \in \llbracket \phi \rrbracket$ , we will simply say that  $h$  is one of the output assignments after  $g$  is updated with  $\phi$ , symbolized as  $g\llbracket \phi \rrbracket h$ . This can be read informally as: we go from  $g$  to  $h$  by  $\phi$ .

$$(65) \quad g\llbracket \phi \rrbracket h := \langle g, h \rangle \in \llbracket \phi \rrbracket$$

The second abbreviation gives us a way to compactly indicate the set of all output assignments that we can reach from  $g$  by means of  $\phi$ .

$$(66) \quad g\llbracket \phi \rrbracket := \{h : g\llbracket \phi \rrbracket h\}$$

The resulting definition of dynamic generalized quantifiers (based on Groenendijk & Stokhof 1991: 81-82) is provided in (67) below.

$$(67) \quad \llbracket \mathbf{det}(\phi, \psi) \rrbracket = \{\langle g, g \rangle : \mathbf{DET}(g\llbracket \phi \rrbracket, \mathbf{Dom}(\llbracket \psi \rrbracket))\}$$

A dynamic generalized quantifier **det** is defined in terms of the corresponding static generalized quantifier **DET**. The static **DET** relates two sets of assignments:

- the restrictor set  $g\llbracket \phi \rrbracket$ , which is the set of all assignments that we can reach from  $g$  by means of the restrictor formula  $\phi$
- the nuclear scope set  $\mathbf{Dom}(\llbracket \psi \rrbracket)$ , which is the set of all assignments that make the nuclear scope formula  $\psi$  true

This notion of dynamic generalized quantification is *unselective* in the sense of Lewis (1975): generalized quantifiers relate sets of variable assignments, i.e., sets of cases in Lewis's terminology, and not sets of individuals, as selective quantification would have it.

Dynamic implication as defined in (56d-ii) above is equivalent to the **every**-based instantiation of this dynamic quantification schema. This is shown in (68) below.

$$(68) \quad \begin{aligned} \text{a. } \llbracket \mathbf{every}(\phi, \psi) \rrbracket &= \{\langle g, g \rangle : \mathbf{EVERY}(g\llbracket \phi \rrbracket, \mathbf{Dom}(\llbracket \psi \rrbracket))\} \\ &= \{\langle g, g \rangle : g\llbracket \phi \rrbracket \subseteq \mathbf{Dom}(\llbracket \psi \rrbracket)\} \\ &= \{\langle g, g \rangle : \{h : \langle g, h \rangle \in \llbracket \phi \rrbracket\} \subseteq \mathbf{Dom}(\llbracket \psi \rrbracket)\} \\ &= \llbracket \phi \rightarrow \psi \rrbracket \end{aligned}$$

b. Unselective **every** is equivalent to dynamic implication:

$$\begin{aligned} \mathbf{every}(\phi, \psi) &\Leftrightarrow \\ \phi \rightarrow \psi & \end{aligned}$$

We can now compositionally translate sentence (64) above as shown in (69) below. We translate indefinites, i.e., existential quantification, following the  $\exists v(\phi; \psi) \Leftrightarrow [v]; \phi; \psi$  recipe in (54) and (57) above. Unselective **most** is interpreted as shown in (70).

$$(69) \quad \mathbf{most}([x]; \text{FARMER}(x); [y]; \text{DONKEY}(y); \text{OWN}(x, y), \\ \text{BEAT}(x, y))$$

$$(70) \quad \begin{aligned} \llbracket \mathbf{most}(\phi, \psi) \rrbracket &= \{\langle g, g \rangle : \mathbf{MOST}(g\llbracket \phi \rrbracket, \mathbf{Dom}(\llbracket \psi \rrbracket))\} \\ &= \{\langle g, g \rangle : |g\llbracket \phi \rrbracket \cap \mathbf{Dom}(\llbracket \psi \rrbracket)| > |g\llbracket \phi \rrbracket \setminus \mathbf{Dom}(\llbracket \psi \rrbracket)|\} \end{aligned}$$

This translation derives the intuitively correct truth conditions, namely: most  $\langle farmer, donkey \rangle$ -pairs s.t. the farmer owns the donkey are s.t. the farmer beats the donkey.

Importantly, this notion of dynamic generalized quantification – suitable for adverbs of quantification – is completely separate and independent from the notion of variable-assignment manipulation that is the essential component of quantification in FOL / DPL.

If we bring the two notions together, we can construct a schema for the translation of natural language determiners like *most*, *no*, *few*, *every* etc. in addition to adverbs of quantification. This is shown in (71) and (72) below.

$$(71) \quad \mathbf{det}_v(\phi, \psi) := \mathbf{det}([v]; \phi, \psi)$$

$$(72) \quad \llbracket \mathbf{det}_v(\phi, \psi) \rrbracket = \llbracket \mathbf{det}([v]; \phi, \psi) \rrbracket \\ = \{ \langle g, g \rangle : \mathbf{DET}(g \llbracket [v]; \phi \rrbracket, \mathbf{Dom}(\llbracket \psi \rrbracket)) \}$$

When we instantiate this schema for **every**, we obtain a formula that is equivalent to DPL universal quantification as defined in (56d-iv) above.

$$(73) \quad \text{a. } \llbracket \mathbf{every}_v(\phi, \psi) \rrbracket = \llbracket \mathbf{every}([v]; \phi, \psi) \rrbracket \\ = \{ \langle g, g \rangle : \mathbf{EVERY}(g \llbracket [v]; \phi \rrbracket, \mathbf{Dom}(\llbracket \psi \rrbracket)) \} \\ = \{ \langle g, g \rangle : g \llbracket [v]; \phi \rrbracket \subseteq \mathbf{Dom}(\llbracket \psi \rrbracket) \}$$

b. Unselective **every**<sub>v</sub> is equivalent to DPL universal quantification:

$$\begin{aligned} \mathbf{every}_v(\phi, \psi) &\Leftrightarrow \\ \mathbf{every}([v]; \phi, \psi) &\Leftrightarrow \\ ([v]; \phi) \rightarrow \psi &\Leftrightarrow \\ [v] \rightarrow (\phi \rightarrow \psi) &\Leftrightarrow \\ \forall v(\phi \rightarrow \psi) & \end{aligned}$$

## 5.2 Problems with Unselective Quantification: Proportions and Weak / Strong Donkey Readings

We instantiate the schema in (71) above for the determiner *most* as shown below.

$$(74) \quad \llbracket \mathbf{most}_v(\phi, \psi) \rrbracket = \llbracket \mathbf{most}([v]; \phi, \psi) \rrbracket \\ = \{ \langle g, g \rangle : \mathbf{MOST}(g \llbracket [v]; \phi \rrbracket, \mathbf{Dom}(\llbracket \psi \rrbracket)) \} \\ = \{ \langle g, g \rangle : |g \llbracket [v]; \phi \rrbracket \cap \mathbf{Dom}(\llbracket \psi \rrbracket)| > \\ |g \llbracket [v]; \phi \rrbracket \setminus \mathbf{Dom}(\llbracket \psi \rrbracket)| \}$$

According to this translation for the English determiner *most*, the sentence in (62) above receives the translation in (75) below, where (75b) unpacks the abbreviated formula in (75a).

$$(75) \quad \text{a. } \mathbf{most}_x(\text{FARMER}(x); [y]; \text{DONKEY}(y); \text{OWN}(x, y), \\ \text{BEAT}(x, y)) \\ \text{b. } \mathbf{most}([x]; \text{FARMER}(x); [y]; \text{DONKEY}(y); \text{OWN}(x, y), \\ \text{BEAT}(x, y))$$

This translation is in fact identical to the translation provided in (69) above for the conditional sentence in (64). That is, we translate determiner-based sentences and conditional sentences with adverbs of quantification in the same way.

As Kadmon (1987) and Heim (1990) observe (see also references therein), this makes our translation for natural language determiners empirically inadequate because the account of donkey anaphora under the determiner *most* runs into a proportion problem.

Consider again the translation in (75b) above. We quantify over most pairs  $\langle x, y \rangle$  s.t.  $x$  is a farmer that owns the donkey  $y$ ; for most such pairs  $\langle x, y \rangle$ , the requirement in the nuclear scope, i.e.,  $x$  beats  $y$ , should be satisfied. While this is intuitively correct for donkey conditionals like (64), it is not correct for relative-clause donkey sentences like (62): we can produce a scenario in which the English sentence in (62) is intuitively false while the formula in (75) is true.

For example, imagine a village with ten farmers, nine of which own a single donkey and one of which owns twenty donkeys. The one-donkey farmers never beat their donkeys, while the twenty-donkey farmer beats all his donkeys. Sentence (62) is intuitively false in this scenario, since most farmers (nine out of ten) are s.t. they don't beat the donkeys they own. The formula in (75b), however, is true: out of twenty-nine  $\langle x, y \rangle$  pairs that satisfy the restrictor formula, twenty (hence, most) pairs also satisfy the nuclear scope formula.

Another problem for the unselective analysis of generalized quantifiers is that it fails to account for the fact that the same donkey sentence can exhibit two different readings, a strong one and a weak one.

Consider again the classical donkey sentence in (28) above. The most salient reading of this sentence is that every farmer behaves violently toward each and every one of his donkeys, i.e., the so-called strong reading. The **every** <sub>$x$</sub>  operator correctly captures this reading, as shown in (76) below: every  $\langle \text{farmer}, \text{donkey} \rangle$ -pair that satisfies the restrictor has to also satisfy the nuclear scope.

- (76) a. **every** <sub>$x$</sub> (FARMER( $x$ ); [ $y$ ]; DONKEY( $y$ ); OWN( $x, y$ ),  
           BEAT( $x, y$ ))  
 b. **every**([ $x$ ]; FARMER( $x$ ); [ $y$ ]; DONKEY( $y$ ); OWN( $x, y$ ),  
           BEAT( $x, y$ ))

However, sentence (28) can receive another, weak reading: every farmer beats some donkey that he owns, but not necessarily each and every one of them. Chierchia (1995: 64) provides a context in which the most salient reading is the weak one: imagine that the farmers under discussion are all part of an anger management program and they are encouraged by the psychotherapist in charge to channel their aggressiveness toward their donkeys (should they own any) rather than toward each other. The farmers scrupulously follow the psychotherapist's advice – in which case we can truthfully assert (28) even if the donkey-owning farmers beat only some of their donkeys.

Furthermore, there are donkey sentences for which the weak reading is the most salient one:

- (77) Every <sup>$x$</sup>  person who has a <sup>$y$</sup>  dime will put it <sub>$y$</sub>  in the meter.  
 (Pelletier & Schubert 1989)  
 (78) Yesterday, every <sup>$x$</sup>  person who had a <sup>$y$</sup>  credit card paid his <sub>$x$</sub>  bill with it <sub>$y$</sub> .  
 (R. Cooper, apud Chierchia 1995: 63, (3a))

Thus, both readings are available but our unselective notion of dynamic generalized quantification does not allow for both of them.

The proportion problem and the availability of weak donkey readings point to the fact that the unselective notion of generalized quantification is empirically inadequate and it should be supplemented with a notion of *selective* quantification that relates two sets of individuals and not two sets of assignments. On one hand, relating sets of individuals solves the proportion problem. On the other hand, we can extract the two sets of individuals based on the restrictor and nuclear scope formulas in such a way that both weak and strong donkey readings are available.



### 5.3 Selective Generalized Quantification

The notions of selective dynamic generalized quantification proposed in the literature fall into two broad classes. The first class of notions employs a dynamic framework based on single variable assignments (like DRT / FCS / DPL, see Kamp 1981, Heim 1982, Groenendijk & Stokhof 1991, Kamp & Reyle 1993) and analyzes generalized quantification as internally dynamic and externally static. The main idea is that the restrictor set of individuals is extracted based on the restrictor update, while the nuclear scope set of individuals is extracted based on both the restrictor and the nuclear scope update so that the anaphoric connections between them are captured.

The second class of notions employs a dynamic framework based on *sets* of variable assignments (like van den Berg 1996) and analyzes generalized quantification as both internally and externally dynamic. The main idea is that the restrictor set of individuals is extracted based on the restrictor update and the nuclear scope set of individuals is the maximal structured subset of the restrictor set of individuals that satisfies the nuclear scope update. This section discusses the first class of notions; the second class will be discussed in the following section.

The most common way of extending classical DRT / FCS / DPL with selective generalized quantification was first suggested by Bäuerle & Egli (1985), Root (1986) and Rooth (1987) and was first formulated in DPL-related terms by van Eijck & de Vries (1992) and Chierchia (1992, 1995). The proposal is also adopted in Heim (1990) and Kamp & Reyle (1993).

We use the same notation as above to reproduce it here: selective dynamic generalized quantification has the form  $\mathbf{det}_x(\phi, \psi)$ :

- $x$  is the bound variable
- $\phi$  is the restrictor
- $\psi$  is the nuclear scope

However, since  $\mathbf{det}_x(\phi, \psi)$  is now selective, it cannot be an abbreviation based on the unselective quantifier  $\mathbf{det}(\phi, \psi)$  anymore. We will therefore define it directly as shown in (80) below. To make the definition more readable, we define the abbreviation  $\{x : g[[\phi]]\}$ , which can be informally paraphrased as: the set of entities  $x$  that satisfy  $\phi$  given the input assignment  $g$ .<sup>1</sup>

$$(79) \quad \{x : g[[\phi]]\} := \{h(x) : g[[x]; \phi] h\}$$

$$(80) \quad \begin{aligned} \text{a. } & \llbracket \mathbf{det}_x^{\text{wk}}(\phi, \psi) \rrbracket := \{\langle g, g \rangle : \mathbf{DET}(\{x : g[[\phi]]\}, \{x : g[[\phi; \psi]]\})\} \\ \text{b. } & \llbracket \mathbf{det}_x^{\text{str}}(\phi, \psi) \rrbracket := \{\langle g, g \rangle : \mathbf{DET}(\{x : g[[\phi]]\}, \{x : g[[\phi \rightarrow \psi]]\})\} \end{aligned}$$

The fact that  $\mathbf{det}_x(\phi, \psi)$  is defined in terms of sets of individuals and not sets of assignments enables us to account for the proportion problem.

The weak / strong donkey variation is attributed to an ambiguity in the interpretation of the selective generalized quantifier, following the proposals in Rooth (1987), Heim (1990) and Kanazawa (1994). So for each generalized quantifier, we have a weak meaning  $\mathbf{det}_x^{\text{wk}}(\phi, \psi)$  and

---

<sup>1</sup>The abbreviation  $\{x : g[[\phi]]\}$  is really just static  $\lambda$ -abstraction. The definition in (79) above says that  $\{x : g[[\phi]]\}$  is the set of entities  $a$  s.t.  $\llbracket \phi \rrbracket_{\text{static}}^{g^{x/a}} = \mathbb{T}$ , where  $\llbracket \cdot \rrbracket_{\text{static}}$  is the usual static interpretation function for higher-order logic and  $g^{x/a}$  is the variable assignment exactly like  $g$  except that it assigns  $a$  to  $x$ . This set of entities is precisely the classical, static denotation of the term  $\lambda x. \phi$  relative to assignment  $g$ , i.e.,  $\llbracket \lambda x. \phi \rrbracket_{\text{static}}^g$ . The close connection between DPL-style generalized quantification and static generalized quantification shows that the proposed notion of dynamic quantification is just a refinement of the static notion, much like the DPL notion of quantification is a refinement of the classical FOL notion.

a strong meaning  $\mathbf{det}_x^{\text{str}}(\phi, \psi)$ . An English sentence containing a determiner *det* is ambiguous between the two readings.

The only difference between the weak and the strong denotation in (80) above has to do with how the nuclear scope set of individuals is obtained:

- by means of dynamic conjunction  $\{x : g[[\phi; \psi]]\}$  in the weak case
- by means of dynamic implication  $\{x : g[[\phi \rightarrow \psi]]\}$  in the strong case

Dynamic conjunction yields weak donkey readings because an existential quantifier in the restrictor will still be an existential in the nuclear scope: every farmer that owns *some* donkey beats *some* donkey he owns.

Dynamic implication yields strong donkey readings because DPL validates the equivalence  $(\exists x\phi) \rightarrow \psi \Leftrightarrow \forall x(\phi \rightarrow \psi)$ , so an indefinite in the restrictor ends up being universally quantified in the nuclear scope: every farmer that owns *some* donkey beats *every* donkey he owns.

The donkey sentences in (62) and (77) above are translated as shown in (81a) and (82a) below. It is easily checked that we derive the intuitively correct truth conditions for the two sentences based on the denotations of the two translations, provided in (81b) and (82b).

- (81) a.  $\mathbf{most}_x^{\text{str}}(\text{FARMER}(x); [y]; \text{DONKEY}(y); \text{OWN}(x, y),$   
 $\text{BEAT}(x, y))$   
 b.  $\{\langle g, g \rangle : \mathbf{MOST}(\{x : g[[\text{FARMER}(x); [y]; \text{DONKEY}(y); \text{OWN}(x, y)]]\},$   
 $\{x : g[[\text{FARMER}(x); [y]; \text{DONKEY}(y); \text{OWN}(x, y) \rightarrow \text{BEAT}(x, y)]]\})\}$
- (82) a.  $\mathbf{every}_x^{\text{wk}}(\text{PERSON}(x); [y]; \text{DIME}(y); \text{HAVE}(x, y),$   
 $\text{PUT-IN}(x, y))$   
 b.  $\{\langle g, g \rangle : \mathbf{EVERY}(\{x : g[[\text{PERSON}(x); [y]; \text{DIME}(y); \text{HAVE}(x, y)]]\},$   
 $\{x : g[[\text{PERSON}(x); [y]; \text{DIME}(y); \text{HAVE}(x, y); \text{PUT-IN}(x, y)]]\})\}$

We conclude this discussion with the observation that making judicious use of the operation of anaphoric closure ! (i.e., double dynamic negation) enables us to define selective generalized quantification in terms of unselective generalized quantification, as shown in (84) below.

- (83) a.  $!\phi := \sim (\sim \phi)$   
 b.  $[[!\phi]] = \{\langle g, g \rangle : g \in \mathbf{Dom}([[ \phi ]])\}$
- (84) a.  $[[\mathbf{det}_x^{\text{wk}}(\phi, \psi)]] := \{\langle g, g \rangle : \mathbf{DET}(g[[[x]; !\phi], g[[[x]; !(\phi; \psi)]])\}$   
 b.  $[[\mathbf{det}_x^{\text{str}}(\phi, \psi)]] := \{\langle g, g \rangle : \mathbf{DET}(g[[[x]; !\phi], g[[[x]; !(\phi \rightarrow \psi)]]))\}^2$

## 6 Interpretation Contexts and Generalized Quantification

This section discusses several natural language phenomena that pose problems for the DPL-style notion of generalized quantification introduced in the previous section. All these phenomena point to the fact that we need to decompose generalized quantification in much the same way in which DPL decomposes static first-order quantification. Most of the material in this section is taken from Brasoveanu (2008, 2010a,b).

---

<sup>2</sup>The closure operator ! is vacuous in the case of the dynamic implication  $!(\phi \rightarrow \psi)$ , so it could be omitted in this particular case.

Just as DPL separates anaphora to individuals (encoded by means of the random assignment formula  $[v]$ ) from the existential or universal ‘counting’ of assignments (encoded by means of dynamic conjunction or dynamic implication), our new decomposed notion of generalized quantification will separate anaphora to quantificational dependencies, i.e., anaphora to  $n$ -ary relations, which is common to all generalized quantifiers, from the particular way of ‘counting’ individuals that is specific to each generalized quantifier.

The notion of selective generalized quantification in (80) above fails to make this separation. Generalized quantifiers are packed together with the dynamics of anaphoric information and donkey anaphora is captured by brute force: the restrictor update is duplicated to obtain the nuclear scope set of individuals.

While this enables us to capture donkey anaphora, it does so in an ultimately non-dynamic way. The quantificational dependency between farmers and donkeys – which is the crux of the interpretation of donkey sentences and which is intuitively passed on from the restrictor to the nuclear scope of the quantification – remains implicit because it is effectively encapsulated / built into in the definition of dynamic generalized determiners.

The logic of generating such quantificational dependencies and passing them on across updates seems to involve pointwise manipulation of *sets* of variable assignments rather than single variable assignments, as van den Berg (1996) argues (see also Krifka 1996, Nouwen 2003, Wang et al. 2006, Brasoveanu 2008, 2010a).

That is, unlike classical DRT / FCS / DPL, which follow Lewis (1975) and his notion of *case* and define updates in terms of single assignments  $g, h, \dots$ , we need to define updates in terms of sets of assignments  $G, H, \dots$ .

These sets can be represented as matrices with assignments (sequences of individuals) as rows.

| $G$   | ... | $x$              | $x'$              | ... | $w$              | $w'$              | ... |
|-------|-----|------------------|-------------------|-----|------------------|-------------------|-----|
| $g_1$ | ... | $a_1 (= g_1(x))$ | $b_1 (= g_1(x'))$ | ... | $u_1 (= g_1(w))$ | $v_1 (= g_1(w'))$ | ... |
| $g_2$ | ... | $a_2 (= g_2(x))$ | $b_2 (= g_2(x'))$ | ... | $u_2 (= g_2(w))$ | $v_2 (= g_2(w'))$ | ... |
| $g_3$ | ... | $a_3 (= g_3(x))$ | $b_3 (= g_3(x'))$ | ... | $u_3 (= g_3(w))$ | $v_3 (= g_3(w'))$ | ... |
| ...   | ... | ...              | ...               | ... |                  |                   |     |

**Values** (sets of individuals / worlds) are stored columnwise:  $\{a_1, a_2, a_3, \dots\}$ ,  $\{b_1, b_2, b_3, \dots\}$ ,  $\{u_1, u_2, u_3, \dots\}$ ,  $\{v_1, v_2, v_3, \dots\}$  etc.

**Structure** (relations between individuals and / or worlds) is stored rowwise:  $\{\langle a_1, b_1 \rangle, \langle a_2, b_2 \rangle, \langle a_3, b_3 \rangle, \dots\}$ ,  $\{\langle a_1, b_1, u_1 \rangle, \langle a_2, b_2, u_2 \rangle, \langle a_3, b_3, u_3 \rangle, \dots\}$ ,  $\{\langle u_1, v_1 \rangle, \langle u_2, v_2 \rangle, \langle u_3, v_3 \rangle, \dots\}$  etc.

A matrix / set of assignments is a two-dimensional database that stores two kinds of information:

- (i) values, i.e., sets of objects, which are stored in the columns of the matrix
- (ii) structure, i.e., correlations / dependencies between these sets of objects, which is encoded in the rows of the matrix

Formulas – and natural language sentences – denote programs incrementally updating such matrices, i.e., they denote binary relations between an input and an output *set* of assignments and not simply a binary relation between an input and an output assignment (as in DPL).

Indefinites like *a kangaroo*, modal verbs like *might* and conditional antecedents like *if Oswald hadn't killed Kennedy* are associated with variables over individuals  $x, x', y, y', \dots$  and variables over possible worlds  $w, w', \dots$  (respectively) and they non-deterministically introduce both values and structure. Pronouns, verbal moods and modal verbs are anaphoric to the sets of objects and the relations between them that are associated with these variables.

Consider, for example, the variable over individuals  $x$  and the variable over possible worlds  $w$  in matrix  $G$  above. Relative to any two assignments  $g, g' \in G$ , they denote two (possibly different) individuals  $g(x)$  and  $g'(x)$  and two (possibly different) possible worlds  $g(w)$  and  $g'(w)$ . Moreover:

- (i) each variable stores a set of values relative to matrix  $G$ , namely a set of individuals in column  $x$ , i.e.,  $\{g(x) : g \in G\}$ , and a set of worlds / a proposition in column  $w$ , i.e.,  $\{g(w) : g \in G\}$
- (ii) the correlation / dependency between these values is encoded by the rows in the matrix; this dependency is the binary relation  $\{\langle g(x), g(w) \rangle : g \in G\}$

The four-way correlation / dependency between  $x, x', w$  and  $w'$  stored in matrix  $G$  above can, for example, represent the dependency introduced by the sentence in (85) below, namely: given the current set of candidates for the actual world stored in  $w$ ,  $w'$  stores a newly introduced epistemic possibility relative to  $w$ ; this epistemically possible scenario features a student  $x$  and a book  $x'$  that the student bought.

(85) It might $_{w'}^{w'}$  be that a $^x$  student $_{w'}$  bought $_{w'}$  a $^{x'}$  book $_{w'}$ .

The set of assignments  $G$  encodes this four-way dependency as follows: for any variable assignment / row  $g \in G$ ,  $g(w)$  is the actual world for the purposes of that assignment,  $g(w')$  is some world that is epistemically accessible from the actual world and  $g(w')$  features an individual  $g(x)$  that is a student and an individual  $g(x')$  that is a book and that the student  $g(x)$  bought.

## 6.1 Quantificational Dependencies Cross-Sententially: Quantificational and Modal Subordination

One of the main arguments for the decomposition of generalized quantification is provided by quantificational subordination phenomena, exemplified in (86), (87) and (88) below. These discourses show that quantifiers like *every convention*, *most books* etc. can introduce dependencies and cross-sententially elaborate on previously introduced dependencies. Other quantificational expressions of the same kind are modals like *might* in (85) above and adverbs like *always* and *usually* in (87b) and (87c).

- (86) a. Harvey $^x$  courts a $^y$  woman at every $^z$  convention.  
b. She $_y$  is very pretty.  
(Karttunen 1976)
- (87) a. Harvey $^x$  courts a $^y$  woman at every $^z$  convention.  
b. She $_y$  always $_z$  comes to the banquet $_z$  with him $_x$ .  
c. The $_y$  girl is usually $_{z'}^{z'}$  also very pretty.  
(Karttunen 1976)
- (88) a. Most $^x$  books contain a $^y$  table of contents.  
b. In some $_{x'}^z$ , it $_y$  is at the end $_z$ .  
(Heim 1990)

But quantifiers are not the only kind of expressions that can introduce and / or elaborate on dependencies. Pronouns, e.g., *she* in (86b/87b), which are referential / non-quantificational, and indefinites, e.g., *a woman* in (86a/87a), which seem to have an ambivalent referential and quantificational nature, can also do this and interact with the dependencies introduced by quantificational expressions.

Consider, for example, the initial sentence (86a/87a) in the two discourses above. This sentence is ambiguous between two quantifier scopings: Harvey courts the same woman at every convention (the indefinite outscopes the universal:  $a^y \text{ woman} \gg \text{every}^z \text{ convention}$ ) vs. at every convention, Harvey courts a (possibly) different woman (the universal outscopes the indefinites:  $\text{every}^z \text{ convention} \gg a^y \text{ woman}$ ). However, discourse (86) as a whole allows only for the “same woman” reading, while discourse (87) allows for both readings.

Such examples of cross-sentential anaphora to quantificational dependencies suggest the decomposition of the DPL-style notion of generalized quantification into at least two components:

- (i) a *static* generalized quantifier component
- (ii) one or more components operating over matrices that regulate the *dynamics* of dependencies

Decomposing quantification along these lines enables us to account for the contrast between the interpretations of discourses (86) and (87). By using matrices, we can store and cross-sententially pass on information about both quantifier domains / values and quantificational dependencies / structure. So we can let singular pronouns like *she* constrain the cardinality of the domain of a quantifier introduced in a previous sentence, thereby affecting the scopal properties of that quantifier.

In more detail, we derive the fact that discourse (86) allows for only one of the two quantifier scopings as follows. First, sentence (86a) updates the discourse-initial context  $\emptyset$  (which contains no discourse information whatsoever) by introducing the variable  $y$  and storing in it the set of all women that Harvey courts at some convention or other and, also, introducing the variable  $z$  and storing in it all the conventions (for simplicity, I ignore the dynamic contribution made by the proper name *Harvey* <sup>$x$</sup> ). This update can happen in two ways, depending on whether the indefinite scopes over the universal quantifier or *vice versa*, as shown in (89) and (90) below.

(89)  $\emptyset \xrightarrow{\text{a}^y \text{ woman (Harvey courts at every}^z \text{ convention)}}$

|           |          |
|-----------|----------|
| $y$       | $z$      |
| $woman_1$ | $conv_1$ |
| $woman_1$ | $conv_2$ |
| $woman_1$ | $conv_3$ |

$woman_1$  is courted  
at every convention

(90)  $\emptyset \xrightarrow{\text{at every}^z \text{ convention (Harvey courts a}^y \text{ woman)}}$

|          |           |
|----------|-----------|
| $z$      | $y$       |
| $conv_1$ | $woman_1$ |
| $conv_2$ | $woman_2$ |
| $conv_3$ | $woman_3$ |

$woman_1$  is courted at  $conv_1$   
 $woman_2$  is courted at  $conv_2$   
 $woman_3$  is courted at  $conv_3$

Irrespective of which quantifier scoping we choose for sentence (86a), the singular pronoun  $she_y$  in sentence (86b) constrains the set of  $y$ -women to be a singleton set. This is automatically satisfied in (89), where the indefinite takes scope over the universal quantifier.

In the case of (90), however, the singleton requirement contributed by the singular number morphology on the pronoun  $she_y$  makes a non-trivial contribution: it requires all the cells in the  $y$ -column to store one and the same entity, as shown in (91) below.

Hence, irrespective of which quantifier scoping we choose for (86a), the only available reading for discourse (86) as a whole is the wide-scope indefinite reading.

$$(91) \quad \begin{array}{|c|c|} \hline z & y \\ \hline conv_1 & woman_1 \\ \hline conv_2 & woman_2 \\ \hline conv_3 & woman_3 \\ \hline \end{array} \xrightarrow{\text{she}_y \text{ is very pretty}}$$

$$\begin{array}{|c|c|} \hline z & y \\ \hline conv_1 & woman_{1/2/3} \\ \hline conv_2 & woman_{1/2/3} \\ \hline conv_3 & woman_{1/2/3} \\ \hline \end{array} \quad \begin{array}{l} \{woman_1, woman_2, woman_3\} \\ \text{is a singleton set, i.e.:} \\ woman_1 = woman_2 = woman_3 \end{array}$$

The fact that discourse (87) is also compatible with the narrow-scope indefinite reading is due to the fact that the quantificational adverb *always<sub>z</sub>* in (87b) can take scope over the singular pronoun *she<sub>y</sub>* and thereby neutralize the effect that singular number morphology has on the cardinality of the previously introduced set of women.

This neutralization is a consequence of the *discourse-level distributivity* operator **dist** that quantificational expressions contribute. This operator distributes over matrices in the sense that it requires the update in its scope to be interpreted relative to singleton subsets of the input matrix, as shown in (92) below. Consequently, the singleton requirement contributed by the singular pronoun *she<sub>y</sub>* is interpreted relative to single assignments / rows and is trivially satisfied.

$$(92) \quad \emptyset \xrightarrow{\text{at every}^z \text{ convention (Harvey courts a}^y \text{ woman)}}$$

$$\begin{array}{|c|c|} \hline z & y \\ \hline conv_1 & woman_1 \\ \hline conv_2 & woman_2 \\ \hline conv_3 & woman_3 \\ \hline \end{array} \xrightarrow{\text{always}_z \mathbf{dist}_z(\text{she}_y \text{ comes to the banquet with him})}$$

$$\left. \begin{array}{l} \begin{array}{|c|c|} \hline z & y \\ \hline conv_1 & woman_1 \\ \hline \end{array} \quad \{woman_1\} \text{ is a singleton set} \\ \begin{array}{|c|c|} \hline z & y \\ \hline conv_2 & woman_2 \\ \hline \end{array} \quad \{woman_2\} \text{ is a singleton set} \\ \begin{array}{|c|c|} \hline z & y \\ \hline conv_3 & woman_3 \\ \hline \end{array} \quad \{woman_3\} \text{ is a singleton set} \end{array} \right\}$$

If we assume a quantificational analysis of modal verbs and attitude report verbs, we can generalize this account of quantificational subordination to modal subordination.

In particular, the analysis of the modal subordination discourse in (93) below (based on Roberts 1989) is point-for-point parallel to the analysis of the quantificational subordination discourse in (87) above. We are therefore able to capture the systematic anaphoric and quantificational parallels between the individual and modal domains argued for in Frank (1996), van Rooy (1998), Stone (1999), Bittner (2001) and Schlenker (2005) among others (building on Partee 1973, 1984).

- (93) a. A<sup>x</sup> wolf might<sub>w</sub><sup>w'</sup> come in.  
 b. It<sub>x</sub> would<sub>w</sub><sup>w'</sup> eat Harvey first.  
 (based on Roberts 1989)

- (94) John<sup>x</sup> thinks<sub>w</sub><sup>w'</sup> that he<sub>x</sub> will<sub>w</sub> catch a<sup>y</sup> fish and he<sub>x</sub> hopes<sub>w,w'</sub><sup>w''</sup> I will<sub>w''</sub> grill it<sub>y</sub> tonight.  
 (Heim 1990)

## 6.2 Generalized Quantification in Dynamic Plural Logic (DPIL) and Plural Compositional DRT (PCDRT)

This subsection outlines a formal system within which we can compositionally account for the quantificational and modal subordination discourses discussed above. The formal system is based on the Plural Compositional DRT (PCDRT) of Brasoveanu (2008, 2010a), which is a simplified and modified version of the Dynamic Plural Logic (DPIL) introduced in van den Berg (1996).

We generalize DPL and interpret sentences / formulas as binary relations over sets of assignments, as shown in (96) below. For readability, we introduce the abbreviation  $G[[\phi]]H$ , which is parallel to the abbreviation we introduced for DPL+GQ above.

$$(95) \quad G[[\phi]]H := \langle G, H \rangle \in [[\phi]]$$

(96) **DPIL/PCDRT semantics** –  $\wp(\wp(\mathcal{G}) \times \wp(\mathcal{G}))$  **version**

a. Atomic formulas:

i. If  $\pi$  is an  $n$ -place predicate and  $\alpha_1, \dots, \alpha_n$  are terms, then  

$$[[\pi(\alpha_1, \dots, \alpha_n)]] = \{ \langle G, G \rangle : G \neq \emptyset \text{ and for any } g \in G, \\ \langle g/\mathbf{I}(\alpha_1), \dots, g/\mathbf{I}(\alpha_n) \rangle \in \mathbf{I}(\pi) \}.$$

ii. If  $\alpha$  and  $\beta$  are terms, then

$$[[\alpha = \beta]] = \{ \langle G, G \rangle : G \neq \emptyset \text{ and for any } g \in G, \\ g/\mathbf{I}(\alpha) = g/\mathbf{I}(\beta) \}.$$

b. Formulas (sentential connectives):

i.  $[[\sim \phi]] = \{ \langle G, G \rangle : G \neq \emptyset \text{ and there is no } H \subseteq G \text{ s.t.} \\ H \neq \emptyset \text{ and } H \in \mathbf{Dom}([[ \phi ]]) \}$

ii.  $[[\phi; \psi]] = \{ \langle G, H \rangle : \text{there is an } I \text{ s.t. } G[[\phi]]I \text{ and } I[[\psi]]H \} \\ = [[\phi]] \circ [[\psi]]$

c. Formulas (random assignment):

i.  $[[[v]]] = \{ \langle G, H \rangle : \text{for any } g \in G, \text{ there is an } h \in H \text{ s.t. } g[v]h \text{ and} \\ \text{for any } h \in H, \text{ there is a } g \in G \text{ s.t. } g[v]h \}$

d. Truth:

i. A formula  $\phi$  is true in model  $\mathfrak{M}$  relative to an input set of assignments  $G$  iff there is an output set of assignments  $H$  s.t.  $G[[\phi]]H$ , i.e., iff  $G \in \mathbf{Dom}([[ \phi ]])$ .

The DPIL/PCDRT interpretation of atomic formulas is just the pointwise generalization of the DPL interpretation. The interpretation of negation is also a generalization of the DPL interpretation: we require the input set of assignments  $G$  – and, in addition, all its non-empty subsets – not to be in  $\mathbf{Dom}([[ \phi ]])$ . Dynamic conjunction is still interpreted as relation composition and the definition of truth is still existential closure over outputs.

The most important differences are to be found in the definition of random assignment. Informally,  $G[v]H$  means that each input assignment  $g$  has an  $[v]$ -successor output assignment  $h$  and, vice-versa, each output assignment  $h$  has an  $[v]$ -predecessor input assignment  $g$ . This ensures that we preserve the values and structure associated with any of the variables  $x, y, \dots, w, w', \dots$  introduced before  $v$ .

The definition in (96c-i) treats the structure and value components of a set of assignments in parallel, since we non-deterministically introduce both of them, namely:

(i) some new (random) values for  $v$

- (ii) some new (random) structure associating the  $[v]$ -values and the values of any other (previously introduced) variables  $x, y, \dots, w, w', \dots$

Just as in DPL, we can define existential quantification in terms of random assignment and dynamic conjunction:

$$(97) \quad \exists v \phi := [v]; \phi$$

The definition of universal quantification is just a particular instance of the definition of dynamic generalized quantification, to which we now turn.

The definition of dynamic generalized quantification has to satisfy four desiderata, the first three of which are about anaphoric connections that can be established *internally*, within the generalized quantification, i.e., between antecedents in the restrictor and anaphors in the nuclear scope, and the last of which is about anaphora that can be established *externally*, i.e., between antecedents introduced by / within the quantification and anaphors outside the quantification.

First, we want our definition to be able to account for the fact that anaphoric connections between the restrictor and the nuclear scope of the quantification can in fact be established, i.e., we want to account for donkey anaphora.

Second, we want to account for such anaphoric connections while avoiding the proportion problem that unselective quantification runs into. That is, we need generalized determiners to relate sets of individuals and not sets of assignments.

The third desideratum is that the definition of selective generalized quantification should be compatible with both strong and weak donkey readings.

The fourth desideratum is concerned with quantification-external anaphora – and this brings us to the quantificational subordination discourses in (86) and (87) above. These discourses indicate that we need to make available for subsequent anaphora the restrictor and nuclear scope sets of individuals related by generalized determiners. Moreover, we also need to make available for anaphoric take-up the quantificational dependencies between different quantifiers and / or indefinites.

In particular, generalized quantification supports anaphora to two sets:

- (i) the maximal set of individuals satisfying the restrictor update, i.e., the restrictor set
- (ii) the maximal subset of the restrictor set that satisfies the nuclear scope update, i.e., the nuclear scope set

The discourse in (98) below exemplifies anaphora to nuclear scope sets. Sentence (98b) is interpreted as: the people that went to the beach are the students that left the party after 5 a.m. – which, in addition, formed a majority of the students at the party.

- (98) a. Most<sup>x</sup> students left the party after 5 a.m.
- b. They<sub>x</sub> went directly to the beach.

The discourses in (99) and (100) exemplify anaphora to the restrictor sets contributed by the downward monotonic quantifiers *no<sup>x</sup> student* and *very few<sup>x</sup> people* respectively. Consider (99) first: any successful update with a *no<sup>x</sup>*-quantification ensures that the nuclear scope set is empty (this is due to the fact that we need to build conservativity into our representation of generalized quantification to capture donkey anaphora) and anaphora to it is therefore infelicitous. The only possible anaphora in (99) is restrictor-set anaphora. Restrictor-set anaphora is the only possible one in (100) too, because nuclear scope anaphora yields a contradictory interpretation for (100b): most of the people with a rich uncle that inherit his fortune don't inherit his fortune.



- (99) a. No<sup>x</sup> student left the party later than 10 p.m.  
 b. They<sub>x</sub> had classes early in the morning.
- (100) a. Very few<sup>x</sup> people with a rich uncle inherit his fortune.  
 b. Most of them<sub>x</sub> don't.

Thus, selective generalized determiners receive the translation in (101) below. This translation achieves our goal of decomposing generalized quantification: the static notion of generalized quantification, encoded by the final formula **DET**( $x, x'$ ), is separated from the dynamics of quantificational dependencies, encoded by the two previous formulas  $\mathbf{max}^x(\langle x \rangle(\phi))$  and  $\mathbf{max}^{x' \sqsubseteq x}(\langle x' \rangle(\psi))$ .

$$(101) \quad \mathbf{det}^{x, x' \sqsubseteq x}(\phi, \psi) := \mathbf{max}^x(\langle x \rangle(\phi)); \mathbf{max}^{x' \sqsubseteq x}(\langle x' \rangle(\psi)); \mathbf{DET}(x, x')$$

Dynamic generalized determiners  $\mathbf{det}^{x, x' \sqsubseteq x}$  relate a restrictor formula  $\phi$  and a nuclear scope formula  $\psi$ . A generalized determiner introduces two individual variables:  $x$  stores the restrictor set of individuals and  $x'$  stores the nuclear scope set of individuals. These two variables and the two updates  $\phi$  and  $\psi$  are the basic building blocks of the three separate updates in (101).

The first update, namely  $\mathbf{max}^x(\langle x \rangle(\phi))$ , ensures that the restrictor set  $x$  is the maximal set of individuals, i.e.,  $\mathbf{max}^x(\dots)$ , s.t., when we take each  $x$ -individual separately, i.e.,  $\langle x \rangle(\dots)$ , this individual satisfies the restrictor formula  $\phi$ .

The second update, namely  $\mathbf{max}^{x' \sqsubseteq x}(\langle x' \rangle(\psi))$ , ensures that the nuclear scope set  $x'$  is obtained in much the same way as the restrictor set  $x$ , except for the requirement that  $x'$  is the maximal structured subset of  $x$ , i.e.,  $\mathbf{max}^{x' \sqsubseteq x}(\dots)$ .

Finally, the third update, namely **DET**( $x, x'$ ), is a test: we test that the restrictor set  $x$  and the nuclear scope set  $x'$  stand in the relation denoted by the corresponding static determiner **DET**.

To formally explicate this translation schema for dynamic generalized determiners, we need:

- (i) two operators over sets of assignments, namely a selective maximization operator  $\mathbf{max}^x(\dots)$  and a selective distributivity operator  $\langle x \rangle(\dots)$
- (ii) a notion of structured inclusion  $x' \sqsubseteq x$  that requires the subset to preserve the quantificational dependencies, i.e., the structure, associated with the individuals in the superset

### 6.2.1 Structured Inclusion

Let us start with the notion of structured inclusion. Recall that sets of assignments store both values (in the columns of the matrix) and structure (in the rows of the matrix). Requiring a variable  $z$  to simply be a value-subset of another variable  $x$  relative to a set of assignments  $G$  is defined as shown in (102) below. For example, the leftmost  $z$  column in matrix  $G$  below satisfies the condition  $z \subseteq x$  because  $\{g(z) : g \in G\} = \{a_1, a_2, a_3\} \subseteq \{g(x) : g \in G\} = \{a_1, a_2, a_3, a_4\}$ .

$$(102) \quad \llbracket z \subseteq x \rrbracket = \{ \langle G, G \rangle : \{g(z) : g \in G\} \subseteq \{g(x) : g \in G\} \}$$

| $G$   | $x$   | $y$   | $z$ ( $z \subseteq x, z \not\subseteq x$ ) | $z$ ( $z \subseteq x$ ) |
|-------|-------|-------|--|-------------------------|
| $g_1$ | $a_1$ | $b_1$ | $a_1$                                      | $a_1$                   |
| $g_2$ | $a_2$ | $b_2$ | $a_3$                                      | $a_2$                   |
| $g_3$ | $a_3$ | $b_3$ | $a_1$                                      | ★                       |
| $g_4$ | $a_4$ | $b_4$ | $a_2$                                      | $a_4$                   |

Condition (102) requires only *value inclusion* and disregards structure completely. The correlation between the  $x$  and  $y$  individuals, i.e., the relation  $\{\langle a_1, b_1 \rangle, \langle a_2, b_2 \rangle, \langle a_3, b_3 \rangle, \langle a_4, b_4 \rangle\}$ , is lost in going from the  $x$ -superset to the  $z$ -subset: as far as  $z$  and  $y$  are concerned,  $a_1$  is still correlated with  $b_1$ , but it is now also correlated with  $b_3$ ,  $a_2$  is now correlated with  $b_4$  (not  $b_2$ ) and  $a_3$  with  $b_2$  (not  $b_3$ ).

If we use the notion of value-only subset in (102), we make incorrect empirical predictions.

Consider, for example, the discourse in (103) below, where  $x$  stores the set of conventions and  $y$  stores the set of corresponding women (the restrictor and nuclear scope sets of a successful *every*-quantification are identical with respect to both value and structure, so we can safely conflate them). Assume that *every<sup>x</sup> convention* takes scope over *a<sup>y</sup> woman* and that the correlation between the  $x$ -conventions and the courted  $y$ -women is the one represented in matrix  $G$  above.

- (103) a. Harvey courts a<sup>y</sup> woman at every<sup>x</sup> convention.  
 b. She<sub>y</sub> usually<sup>z $\subseteq$ x</sup> comes to the banquet with him.

Intuitively, the adverb *usually* in (103b) is anaphoric to the set of conventions introduced in (103a) and sentence (103b) is interpreted as follows: at most conventions, the woman courted by Harvey *at that convention* comes to the banquet with him. The leftmost variable  $z$  in the table above stores most  $x$ -conventions (three out of four), but it does not preserve the correlation between  $x$ -conventions and  $y$ -women established in (103a).

We obtain similarly incorrect results for donkey sentences like the one in (104) below (repeated from (62)): the restrictor of the quantification introduces a dependency between all the donkey-owning  $x$ -farmers and the  $y$ -donkeys that they own; the nuclear scope set  $z$  needs to contain most  $x$ -farmers, but in such a way that the correlated  $y$ -donkeys remain the same. That is, the nuclear scope set contains a *most*-subset of donkey-owning farmers that beat *their respective donkey(s)*. The notion of value-only inclusion in (102) is yet again inadequate.

- (104) Most<sup>x,z $\subseteq$ x</sup> farmers who own a<sup>y</sup> donkey beat it<sub>y</sub>.

Thus, to capture the intra- and cross-sentential interaction between anaphora and quantification, we need the notion of *structured inclusion* defined in (105) below where we go from a superset to a subset by discarding rows in the matrix. We are therefore guaranteed that the subset will contain *only* the dependencies associated with the superset (but not necessarily *all* dependencies – more on this below).

- (105)  $\llbracket z \subseteq x \rrbracket = \{\langle G, G \rangle : \text{any } g \in G \text{ is s.t. } g(z) = g(x) \text{ or } g(z) = \star\}$

To formalize this, we introduce a dummy / exception individual  $\star$  that is used as a tag for the cells in the matrix that should be discarded in order to obtain a structured subset  $z$  of a superset  $x$ , as shown by the rightmost  $z$  column in matrix  $G$  above. The dummy individual  $\star$  is the universal falsifier, i.e., it makes any  $n$ -place predicate  $\pi$  false. Formally, we ensure that any predicate  $\pi$  is false whenever  $\star$  is one of its arguments by requiring that  $I(\pi) \subseteq (D \setminus \{\star\})^n$ .

Thus, we need to slightly revise our interpretation of atomic formulas, which should be interpreted distributively relative to the non-dummy sub-matrix of the input matrix  $G$ , as shown in (107) below.

- (106)  $G_{\alpha_1 \neq \star, \dots, \alpha_n \neq \star} := \{g \in G : g/I(\alpha_1) \neq \star \wedge \dots \wedge g/I(\alpha_n) \neq \star\}$   
 (107)  $\llbracket \pi(\alpha_1, \dots, \alpha_n) \rrbracket = \{\langle G, G \rangle : G_{\alpha_1 \neq \star, \dots, \alpha_n \neq \star} \neq \emptyset \text{ and for any } g \in G_{\alpha_1 \neq \star, \dots, \alpha_n \neq \star}, \langle g/I(\alpha_1), \dots, g/I(\alpha_n) \rangle \in I(\pi)\}$

The notion of structured inclusion  $\subseteq$  in (105) above ensures that the subset inherits *only* the superset structure. But we also need it to inherit *all* the superset structure, which we achieve by the final definition of structured inclusion  $\sqsubseteq$  in (109) below.

$$(108) \quad G(x) := \{g(x) : g \in G_{x \neq \star}\}$$

$$(109) \quad \llbracket x' \sqsubseteq x \rrbracket = \{\langle G, G \rangle : G \llbracket x' \subseteq x \rrbracket G \text{ and for any } g \in G, \\ \text{if } g(x) \in G(x'), \text{ then } g(x) = g(x')\}$$

The additional constraint on structured inclusion in (109) is needed (among other things) to account for the donkey sentence in (104) above, which is most likely interpreted as talking about *every* donkey owned by any given farmer. That is, the nuclear scope set  $z$ , which is a *most*-subset of the restrictor set  $x$ , needs to inherit *all* the superset structure: each farmer in the nuclear scope set needs to be associated with *every* donkey  $y$  that s/he owns.

### 6.2.2 Maximization and Distributivity

We turn now to the maximization and distributivity operators  $\mathbf{max}^x$  and  $\mathbf{dist}_x$ , which enable us to extract and store the restrictor and nuclear scope sets needed to define dynamic generalized quantification.

Consider the definition of  $\mathbf{max}^x$  in (110) below first: the first conjunct  $G \llbracket [x]; \phi \rrbracket H$  introduces  $x$  as a new variable, i.e.,  $[x]$ , and makes sure that each individual in  $H(x)$  satisfies  $\phi$ , i.e., we store *only* individuals that satisfy  $\phi$ . The second conjunct enforces the maximality requirement: any other set  $I(x)$  obtained by a similar procedure, i.e., any other set of individuals that satisfies  $\phi$ , is included in  $H(x)$ ; that is,  $H(x)$  stores *all* individuals that satisfy  $\phi$ .

$$(110) \quad \llbracket \mathbf{max}^x(\phi) \rrbracket = \{\langle G, H \rangle : G \llbracket [x]; \phi \rrbracket H \text{ and} \\ \text{for any } I \text{ s.t. } G \llbracket [x]; \phi \rrbracket I, I(x) \subseteq H(x)\}$$

$$(111) \quad \mathbf{max}^{x' \sqsubseteq x}(\phi) := \mathbf{max}^{x'}(x' \sqsubseteq x; \phi)$$

$$(112) \quad G_{x=a} := \{g \in G : g(x) = a\}$$

$$(113) \quad \llbracket \mathbf{dist}_x(\phi) \rrbracket = \{\langle G, H \rangle : G_{x=\star} = H_{x=\star} \text{ and} \\ G(x) = H(x) \text{ and} \\ \text{for any } a \in G(x), G_{x=a} \llbracket \phi \rrbracket H_{x=a}\}$$

Definition (113) states that updating a set of assignments  $G$  with a formula  $\phi$  *distributively* over a variable  $x$  means:

- (i) generating the  $x$ -partition of  $G$ , i.e.,  $\{G_{x=a} : a \in G(x)\}$
- (ii) updating each cell  $G_{x=a}$  in the partition with the formula  $\phi$
- (iii) taking the union of the resulting output sets of assignments:  $H = \bigcup H_{x=a}$

The requirement  $G(x) = H(x)$  is needed to ensure that there is a bijection between the partition induced by the variable  $x$  over the input state  $G$  and the one induced over the output state  $H$ . Without this requirement, we could introduce arbitrary new values for  $x$  in the output state  $H$ , i.e., arbitrary new partition cells.<sup>3</sup>

The final conjunct (for any  $a \in G(x)$ ,  $G_{x=a} \llbracket \phi \rrbracket H_{x=a}$ ) is the one that actually defines the distributive update: the formula  $\phi$  relates every partition cell in the input state  $G$  to the corresponding partition cell in the output state  $H$ .

---

<sup>3</sup>Nouwen (2003): 87 was the first to observe that we need to add this requirement to the original definition of distributivity in van den Berg (1996): 145, (18).

### 6.2.3 Generalized Determiners, Weak / Strong Indefinites and Singular Anaphora

The translations for generalized determiners and weak / strong indefinites are provided in (117), (118) and (119) below (taken from Brasoveanu 2008, 2010a).

- (114)  $\llbracket_x(\phi)\rrbracket = \{\langle G, H \rangle : G_{x=\star} = H_{x=\star} \text{ and}$   
 $G(x) \neq \emptyset \text{ and}$   
 $G_{x \neq \star} \llbracket \mathbf{dist}_x(\phi) \rrbracket H_{x \neq \star}\}$
- (115)  $\llbracket_{\langle x \rangle}(\phi)\rrbracket = \{\langle G, H \rangle : G_{x=\star} = H_{x=\star} \text{ and}$   
 $G = H \text{ if } G(x) = \emptyset, \text{ otherwise}$   
 $G_{x \neq \star} \llbracket \mathbf{dist}_x(\phi) \rrbracket H_{x \neq \star}\}$
- (116)  $\llbracket \mathbf{DET}(x, x') \rrbracket = \{\langle G, G' \rangle : \mathbf{DET}(G(x), G(x'))\}$
- (117)  $\mathbf{det}^{x, x' \sqsubseteq x}(\phi, \psi) := \mathbf{max}^x(\llbracket_x(\phi) \rrbracket); \mathbf{max}^{x' \sqsubseteq x}(\llbracket_{x'}(\psi) \rrbracket); \mathbf{DET}(x, x')$
- (118)  $\mathbf{a}^{\text{wk}:x}(\phi, \psi) := [x]; x(\phi; \psi)$
- (119)  $\mathbf{a}^{\text{str}:x}(\phi, \psi) := \mathbf{max}^x(x(\phi; \psi))$

The last component needed for the account of discourses (86) and (87) is the translation for singular pronouns. Crucially, we need singular pronouns to contribute a **singleton**( $x$ ) requirement, so their translation will have the form provided in (121) below. The requirement is contributed by singular number morphology and requires uniqueness of the non-dummy value of the variable  $x$  relative to the current context  $G$ .

- (120)  $\llbracket \mathbf{singleton}(x) \rrbracket = \{\langle G, G \rangle : |G(x)| = 1\}$
- (121)  $\mathbf{she}_x(\phi) := \mathbf{singleton}(x); \phi$

### 6.2.4 The Analysis of Quantificational Subordination

We start with the two possible quantifier scopings for the discourse-initial sentence (86a/87a).

Note that we can safely identify the restrictor and nuclear scope variables  $x$  and  $x'$  of any  $every^{x, x' \sqsubseteq x}$ -quantification: the definition in (117) above entails that, if  $H$  is an arbitrary output state of a successful  $every^{x, x' \sqsubseteq x}$ -quantification,  $x$  and  $x'$  have to be identical with respect to both value and structure, i.e., for any  $h \in H$ ,  $h(x) = h(x')$ .

We can therefore assume that *every* contributes only one variable, as shown in (122) below. I will also assume that the restrictor set of the  $every^z$ -quantification is non-empty, so we can safely replace the distributivity operator  $\llbracket_z \rrbracket(\dots)$  with the simpler distributivity operator  $z(\dots)$ .

- (122)  $\mathbf{every}^z(\phi, \psi) := \mathbf{max}^z(z(\phi)); z(\psi)$
- (123)  $a^y \text{ woman} \gg \mathbf{every}^z \text{ convention}:$   
 $[y]; y(\mathbf{WOMAN}(y); \mathbf{max}^z(\mathbf{CONVENTION}(z)); \text{COURT-AT}(\text{HARVEY}, y, z))$
- (124)  $\mathbf{every}^z \text{ convention} \gg a^y \text{ woman}:$   
 $\mathbf{max}^z(\mathbf{CONVENTION}(z)); z([y]; \mathbf{WOMAN}(y); \text{COURT-AT}(\text{HARVEY}, y, z))$
- (125)  $\mathbf{she}_y \text{ is very pretty}:$   $\mathbf{singleton}(y); \mathbf{VERY-PRETTY}(y)$

The representations of the two quantifier scopings for sentence (86a/87a) are provided in (123) and (124) above. Redundant distributivity operators are omitted. I assume that the indefinite  $a^y \text{ woman}$  is weak because the resulting DPIL/PCDRT formulas are somewhat simpler, but a strong reading for the indefinite would also work for these examples. The dynamic contribution made by

the proper name *Harvey* is similarly ignored so that we can focus exclusively on the quantificational subordination issues.

We can now see how sentence (86b), translated in (125) above, forces the ‘wide-scope indefinite’ reading: the **singleton**( $y$ ) requirement contributed by the singular pronoun  $she_y$  effectively conflates the two scopings by requiring the set of  $y$ -women obtained after updating with (123) or (124) to be a singleton. This requirement leaves the truth conditions derived on the basis of (123) untouched, but makes the truth conditions associated with (124) strictly stronger.

In contrast, sentence (87b) contains the adverb of quantification  $always_z$ , which can take scope above or below the singular pronoun  $she_y$ . If the adverb takes scope over the pronoun, the  $y$ -uniqueness requirement is weakened (i.e., in a sense, neutralized) by being relativized to  $z$ -conventions.

As shown in (126) below, I take the translation of  $always_z$  to be a universal quantification over an anaphorically retrieved restrictor, i.e., over the nuclear scope set introduced by the quantifier  $every^z$  convention in the preceding sentence. Since  $always$  is basically interpreted as  $every$  (modulo the anaphorically retrieved restrictor), its translation is parallel to the translation for  $every$  in (122) above. The general format for the translation of quantificational expressions that anaphorically retrieve their restrictor sets is provided in (127).

$$(126) \quad \mathbf{always}_z(\phi) := {}_z(\phi)$$

$$(127) \quad \mathbf{det}_x^{x' \sqsubseteq x}(\phi) := \mathbf{max}^{x' \sqsubseteq x}(\langle x' \rangle(\phi)); \mathbf{DET}(x, x')$$

For simplicity, I take sentence (87b) to contribute a ternary relation COME-TO-BANQUET-OF, abbreviated C-T-B-OF and relating the women, their partner Harvey and the conventions under consideration. The two translations of sentence (87b) are provided in (128) and (129) below (once again, redundant distributivity operators are omitted).

$$(128) \quad she_y >> always_z: \quad \mathbf{singleton}(y); \text{C-T-B-OF}(y, \text{HARVEY}, z)$$

$$(129) \quad always_z >> she_y: \quad {}_z(\mathbf{singleton}(y); \text{C-T-B-OF}(y, \text{HARVEY}, z))$$

Thus, there are two possible translations for sentence (87a), given in (123) and (124), and two possible translations for sentence (87b), given in (128) and (129). Out of the four combinations, three end up effectively requiring the indefinite  $a^y$  woman to have wide scope relative to the universal  $every^z$  convention.

The fourth combination (124+129), provided in (130) below, encodes the ‘narrow-scope indefinite’ reading that is intuitively available for discourse (87) but not for (86).

$$(130) \quad \mathbf{max}^z(\text{CONVENTION}(z)); {}_z([y]; \text{WOMAN}(y); \text{COURT-AT}(\text{HARVEY}, y, z)); \\ {}_z(\mathbf{singleton}(y); \text{C-T-B-OF}(y, \text{HARVEY}, z))$$

In sum, decomposing generalized quantification along the lines of DPIL/PCDRT enables us to formulate a compositional dynamic account of the intra- and cross-sentential interactions between generalized quantifiers, anaphora and number morphology exhibited by the quantificational subordination discourses in (86) and (87) above.

### 6.2.5 The Analysis of Modal Subordination

The analysis of modal subordination is point-for-point parallel to the analysis of quantificational subordination. We will make use of a dummy / exception world  $\star$ , which is the universal falsifier in the world-domain. We can take the dummy world to be one in which no individual whatsoever

exists, hence all relations are false because a relation between certain individuals obtains at a particular world only if those individuals exist in that world.

$$(131) \quad \llbracket \pi(w, \alpha_1, \dots, \alpha_n) \rrbracket = \{ \langle G, G \rangle : G_{w \neq \star, \alpha_1 \neq \star, \dots, \alpha_n \neq \star} \neq \emptyset \text{ and} \\ \text{for any } g \in G_{w \neq \star, \alpha_1 \neq \star, \dots, \alpha_n \neq \star}, \\ \langle g/\mathbf{I}(w), g/\mathbf{I}(\alpha_1), \dots, g/\mathbf{I}(\alpha_n) \rangle \in \mathbf{I}(\pi) \}$$

$$(132) \quad \llbracket w' \subseteq w \rrbracket = \{ \langle G, G \rangle : \text{any } g \in G \text{ is s.t. } g(w') = g(w) \text{ or } g(w') = \star \}$$

$$(133) \quad \llbracket w' \sqsubseteq w \rrbracket = \{ \langle G, G \rangle : G \llbracket w' \subseteq w \rrbracket G \text{ and for any } g \in G, \\ \text{if } g(w) \in G(w'), \text{ then } g(w) = g(w') \}$$

The translation for modalized conditionals in (134) below is parallel to the translation for generalized determiners. Following Kratzer (1981),  $\mathcal{B}$  and  $\mathcal{O}$  are variables for a modal base and an ordering source respectively (with contextually-supplied values) that parametrize the interpretation of modal quantifiers.

$$(134) \quad \mathbf{if}^w(\phi) + \mathbf{modal}_{\mathcal{B}, \mathcal{O}}^{w' \sqsubseteq w}(\psi) := \mathbf{max}^w(\langle w \rangle(\phi)); \mathbf{max}^{w' \sqsubseteq w}(\langle w' \rangle(\psi)); \mathbf{MODAL}_{\mathcal{B}, \mathcal{O}}(w, w')$$

Moreover, just as a pronoun anaphorically retrieves an individual variable, the indicative mood anaphorically retrieves  $\mathcal{A}$ , which is the designated variable for the actual world – as shown in (135) below.

$$(135) \quad \mathbf{indicative}_{\mathcal{A}}(\phi) := \mathbf{singleton}(\mathcal{A}); \phi$$

Finally, the modal quantifier *would* in (93b) is anaphoric to the nuclear scope set introduced by *might* in (93a). The general format for the translation of anaphoric modal quantifiers is provided in (136).

$$(136) \quad \mathbf{modal}_{w, \mathcal{B}, \mathcal{O}}^{w' \sqsubseteq w}(\phi) := \mathbf{max}^{w' \sqsubseteq w}(\langle w' \rangle(\phi)); \mathbf{MODAL}_{\mathcal{B}, \mathcal{O}}(w, w')$$

Brasoveanu (2010b) provides additional motivation for this account of modal subordination and an extensive discussion of the technical details.

## 7 A Research Program: The Fine Structure of Interpretation Contexts

We have seen that the interactions between quantification and anaphora in natural language support the idea of decomposing first-order and generalized quantification in such a way that the ‘counting’ / ‘quantifying’ component specific to each quantifier is separated from the general dynamics of quantifier-dependency interpretation.

Let us briefly recapitulate the formal steps we have taken to explicitly encode the decomposition of natural language quantification.

First, we identified the process of pointwise variable-assignment manipulation as the crucial component that relates and ‘glues together’ quantifiers and anaphoric items. While there are alternative (e.g., variable-free) ways to capture quantifier-anaphora interactions in natural language, the Tarskian definition of truth and satisfaction for first-order logic is a particularly good first attempt at formalizing them because storing and passing on quantifier-anaphora dependencies is an automatic by-product of quantifier and anaphora interpretation.

The second step was to generalize the Tarskian idea of passing on dependencies via variable assignments from quantifier-internal to quantifier-external contexts. This generalization from exclusively quantifier-internal to quantifier-internal-*and*-external dynamics required us to explicitly

acknowledge the relational nature of formula / sentence denotations. DPL semantics was a minimal extension of classical FOL semantics that captured this relational nature – minimal because it acknowledges the quantifier-external dynamics of interpretation for existential quantification only.

The third and final step was to acknowledge the essentially dynamic nature of quantifier interpretation (both inside and outside the syntactic scope of quantification) for all generalized quantifiers. The fact that generalized quantification is inherently relational over *sets* of individuals – unlike existential quantification, where single witnesses are sufficient – emphasizes the essentially relational nature of quantificational dependencies and requires us to generalize formula / sentence denotations even further: they are binary relations between *sets* of assignments.

The resulting DPIL/PCDRT semantics preserves the appealing feature of Tarskian single-assignment based semantics: storing and passing on quantifier-anaphora dependencies is an automatic by-product of quantifier and anaphora interpretation.

The trajectory we followed from FOL through DPL to DPIL/PCDRT incrementally added structure to the quantifier-internal and quantifier-external contexts of interpretation. Thus, this trajectory points to a broader research program focused on the fine structure of such quantificational contexts of interpretation.

I will conclude the paper by mentioning two natural-language phenomena that this research program promises to shed light on: (i) the interpretation of *same* / *different* inside and outside quantificational contexts and (ii) the fact that indefinites can take exceptionally wide scope.

## 7.1 Quantificational Dependencies Intra-Sententially: *Same* / *Different*

Recall the two readings of the adjectives *same* and *different* (and their counterparts in many other languages) that we introduced in (3)/(4) and (5) above.

The interpretation of *different* in (3b)/(4b) is sentence external in the sense that it is anaphoric to the variable introduced by the proper name *The Raven* in the previous sentence (3a)/(4a). Thus, in (3)/(4), *different* relates two variables and requires their values, i.e., the actual entities, to be distinct.

The sentence-internal reading in (5) seems to relate values of only one variable, introduced by the narrow-scope indefinite *a poem*. These values, i.e., the recited poems, covary with the values of the variable introduced by the universal quantifier *every boy* and *different* requires the poems to be distinct relative to distinct boys.

These two readings have been known to exist at least since Carlson (1987), but no unified account has been proposed (see Barker 2007 for a recent discussion). Such a unified account can be provided if quantifiers like *every boy* are taken to distribute over sets of assignments / matrices in such a way that the very process of distributively evaluating their nuclear scope temporarily constructs the kind of contexts that license anaphoric, sentence-external readings for *same* and *different*. The basic idea would be that quantifier-internal contexts should be even more structured than sets of assignments: they should involve something like sets of *pairs* of assignments and the members of each pair should be simultaneously updated.

For example, the universal quantifier *every boy* in sentence (5) above would look at any pair of two boys and simultaneously ‘construct’ two variable assignments, one for each boy, that would store their corresponding poems. Relative to such richly structured quantifier-internal contexts, *same* and *different* can be interpreted very much like they are interpreted in sentence-external contexts.

The formal details of the analysis are not important here (see Brasoveanu 2011 for a detailed presentation of the analysis). What is important is that the unified account of sentence-internal

and sentence-external readings that we are after promises to fall into place if pursued as part of the more general project of investigating the fine structure of quantifier-internal contexts.

## 7.2 Quantificational Dependencies Intra-Sententially: Exceptional Scope

A similar observation can be made with respect to the phenomenon of exceptional scope exhibited by indefinites in natural language, first noticed in Farkas (1981) and Fodor & Sag (1982) and exemplified by the widest and the intermediate scope readings of sentence (137) below.

(137) Every<sup>*x*</sup> student read every<sup>*y*</sup> poem that a<sup>*z*</sup> French poet wrote.

(138) Narrowest scope indefinite:

- for every student *x*,
- for every poem *y* such that
- ▶ there is a French poet *z* that wrote *y*,
- x* read *y*.

(139) Intermediate scope indefinite:

- for every student *x*,
- ▶ there is a French poet *z* such that,
- for every poem *y* that *z* wrote,
- x* read *y*.

(140) Widest scope indefinite:

- ▶ there is a French poet *z* such that,
- for every student *x*,
- for every poem *y* that *z* wrote,
- x* read *y*.

The readings of sentence (137) differ with respect to whether the indefinite *a<sup>z</sup> French poet* covaries with another quantifier or not. If it does, they differ depending on which one of the two *every*-quantifiers the indefinite covaries with.

The intermediate and widest scope readings are instances of exceptional scope because quantifiers cannot in general scope out of a relative clause like the indefinite *a<sup>z</sup> French poet* is able to.

The (in)dependence-friendly logics of Hintikka (1973), Sandu (1993), Hintikka & Sandu (1996), Hodges (1997), and Väänänen (2007) seem particularly well-suited to express this kind of ‘selective’ covariation (a.k.a. imperfect information). But the connections between (in)dependence logics and exceptional scope phenomena in natural language have not been systematically investigated despite the fact that the exceptional scope of indefinites has been extensively discussed in the natural language semantics literature – see Abusch (1994), Farkas (1997), Reinhart (1997), Winter (1997), Kratzer (1998), Matthewson (1999), Chierchia (2001) among others.

Such an investigation promises to be particularly fruitful at this juncture, both on the logical and on the linguistic side. On the logical side, Hodges (1997) and Väänänen (2007) provide a compositional semantics for (in)dependence logic that crucially relies on sets of variable assignments to encode dependencies.

On the linguistic side, Brasoveanu & Farkas (2011) propose a novel account of exceptional scope that crucially relies on sets of assignments and a mechanism of keeping track of the variables introduced by higher / previous quantifiers that is very close to the slashed quantifiers of independence logic. Thus, the account in Brasoveanu & Farkas (2011) follows the same general strategy of adding



more structure to quantifier-internal contexts so that we can keep track of what quantifiers syntactically scope over the indefinite while allowing the indefinite to choose (independence-logic style) which ones of these quantifiers end up *semantically* scoping over the indefinite.

## 8 Conclusion

The paper introduced a variety of natural language semantics puzzles involving both intra-sentential and cross-sentential quantificational dependencies and argued that they can receive a unified, compositional account if their analysis is seen as part of a more general research program of investigating the fine structure of quantifier-internal and quantifier-external contexts of interpretation.

The examination of these puzzles indicated that the logic of quantificational dependencies integrates and unifies core intuitions and formal features of both dynamic logic and (in)dependence logic. Thus, the paper is an argument for the pursuit of such a unified logic, which should help us better understand and answer some of the open questions in the formal semantics of natural language quantification and anaphora.

## References

- Abusch, D. (1994). The Scope of Indefinites. In *Natural Language Semantics* 2.2, 83-135.
- Barker, C. (2007). Parasitic Scope. In *Linguistics and Philosophy* 30, 407-444.
- Barwise, J. & R. Cooper (1981). Generalized Quantifiers and Natural Language. In *Linguistics and Philosophy* 4, 159-219.
- Bäuerle, R. & U. Egli (1985). Anapher, Nominalphrase und Eselssätze, Papier 105 des Sonderforschungsbereichs 99, Universität Konstanz.
- Ben-Shalom, D. (1996). *Semantic Trees*. PhD dissertation, UCLA.
- van Benthem, J. (1996). *Exploring Logical Dynamics*. CSLI.
- van Benthem, J. (1997). Modal Foundations for Predicate Logic. In *Logic Journal of IGPL* 5, 259-286.
- van den Berg, M. (1996). *Some Aspects of the Internal Structure of Discourse*. PhD dissertation, University of Amsterdam.
- Bittner, M. (2001). Topical Referents for Individuals and Possibilities. In *Proceedings of SALT XI*, R. Hastings et al (eds.), Ithaca: CLC, 36-55.
- Bittner, M. (2007). Online Update: Temporal, Modal and *De Se* Anaphora in Polysynthetic Discourse. In *Direct Compositionality*, C. Barker & P. Jacobson (eds.), Oxford University Press, 363-404.
- Brasoveanu, A. (2008). Donkey Pluralities: Plural Information States versus Non-atomic Individuals. In *Linguistics and Philosophy* 31, 129-209.
- Brasoveanu, A. (2010a). Structured Anaphora to Quantifier Domains. In *Information and Computation* 208, 450-473.
- Brasoveanu, A. (2010b). Decomposing Modal Quantification. In *Journal of Semantics* 27: 437-527.
- Brasoveanu, A. (2011). Sentence-internal *Different* as Quantifier-internal Anaphora. Accepted in *Linguistics and Philosophy* (with minor revisions).

- Brasoveanu, A. & D. Farkas (2011). Scope and the Grammar of Choice. Accepted in *Linguistics and Philosophy* (with minor revisions).
- Carlson, G. (1987). *Same and Different: Some Consequences for Syntax and Semantics*. In *Linguistics and Philosophy* 10, 531-565.
- Chierchia, G. (1992). Anaphora and Dynamic Binding. In *Linguistics and Philosophy* 15, 111-183.
- Chierchia, G. (1995). *The Dynamics of Meaning: Anaphora, Presupposition and the Theory of Grammar*. University of Chicago Press.
- Chierchia, G. (2001). A puzzle about Indefinites. In *Semantic Interfaces: Reference, Anaphora and Aspect*, C. Cecchetto, G. Chierchia & M.T. Guasti (eds.), Stanford: CSLI, 51-89.
- van Eijck, J. & F.-J. de Vries (1992). Dynamic Interpretation and Hoare Deduction. in *Journal of Logic, Language and Information* 1, 1-44.
- Farkas, D.F. (1981). Quantifier Scope and Syntactic Islands. In the *Proceedings of CLS 7*, R. Hendrik et al (eds.), CLC, Cornell University, 59-66.
- Farkas, D.F. (1997a). Evaluation Indices and Scope. In *Ways of Scope Taking*, A. Szabolcsi (ed.), Dordrecht: Kluwer, 183-215.
- Fodor, J. D. & I. Sag (1982). Referential and Quantificational Indefinites. In *Linguistics and Philosophy* 5, 355-398.
- Frank, A. (1996). *Context Dependence in Modal Constructions*. PhD dissertation, University of Stuttgart.
- Gallin, D. (1975). *Intensional and Higher-Order Modal Logic with Applications to Montague Semantics*. North-Holland Mathematics Studies.
- Geach, P. (1962). *Reference and Generality*. Ithaca: Cornell University Press.
- Groenendijk, J. & M. Stokhof (1991). Dynamic Predicate Logic. In *Linguistics and Philosophy* 14, 39-100.
- Harel, D., D. Kozen & J. Tiuryn (1984). Dynamic Logic. In *Handbook of Philosophical Logic II: Extensions of Classical Logic*, D. Gabbay & F. Guenther (eds.), Dordrecht: Reidel, 497-604.
- Harel, D., D. Kozen & J. Tiuryn (2000). *Dynamic Logic*. MIT Press.
- Heim, I. (1982). *The Semantics of Definite and Indefinite Noun Phrases*. PhD dissertation, UMass Amherst.
- Heim, I. (1990). E-Type Pronouns and Donkey Anaphora. In *Linguistics and Philosophy* 13, 137-177.
- Hintikka, J. (1973). *Logic, Language Games and Information*. Oxford: Clarendon Press.
- Hintikka, J. & G. Sandu (1996). Game-Theoretical Semantics. In *Handbook of Logic and Language*, J. van Benthem & A. ter Meulen (eds.), Elsevier.
- Hodges, W. (1997). Compositional Semantics for a Language of Imperfect Information. In *Logic Journal of the IGPL* 5, 539-563.
- Janssen, T. (1986). *Foundations and Applications of Montague Grammar*. CWI Tract 19, CWI, Amsterdam.
- Kadmon, N. (1987). *On Unique and Non-unique Reference and Asymmetric Quantification*. PhD dissertation, UMass Amherst.
- Kamp, H. (1981). A theory of truth and semantic representation. In *Formal Methods in the Study of Language. Part 1*, Groenendijk, J., T. Janssen & M. Stokhof (eds.), Mathematical Center, Amsterdam, 277-322.

- Kamp, H. & U. Reyle (1993). *From Discourse to Logic. Introduction to Model-Theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*, Dordrecht: Kluwer.
- Kanazawa, M. (1994). Weak vs. Strong Readings of Donkey Sentences and Monotonicity Inference in a Dynamic Setting. In *Linguistics and Philosophy* 17, 109-158.
- Kaplan, D. (1977/1989). Demonstratives. In *Themes From Kaplan*, J. Almog, J. Perry & H. Wettstein (eds.), Oxford University Press.
- Karttunen, L. (1976). Discourse Referents. In *Syntax and Semantics Vol. 7*, J.D. McCawley (ed.), New York: Academic Press, 363-385.
- Keenan, E.L. & J. Stavi (1986). A Semantic Characterization of Natural Language Determiners. In *Linguistics and Philosophy* 9, 253-326.
- Kratzer, A. (1981). The Notional Category of Modality. In *Words, Worlds and Contexts. New Approaches in Word Semantics*, Eikmeyer, H.J. & H. Rieser (eds.), Berlin: Walter de Gruyter, 38-74.
- Krifka, M. (1996). Parametric Sum Individuals for Plural Anaphora. In *Linguistics and Philosophy* 19, 555-598.
- Kratzer, A. (1998). Scope or Pseudo-Scope: Are There Wide-Scope Indefinites?. In *Events in Grammar*, S. Rothstein (ed.), Dordrecht: Kluwer, 163-196.
- Lewis, D. (1975). Adverbs of Quantification. In *Formal Semantics of Natural Language*, E. Keenan (ed.), Cambridge: Cambridge University Press, 3-15.
- Marx, M. & Y. Venema (1997). *Multi-Dimensional Modal Logic*. Applied Logic Series Vol. 4, Dordrecht: Kluwer.
- Matthewson, L. (1999). On the Interpretation of Wide-Scope Indefinites. In *Natural Language Semantics* 7.1, 79-134.
- Muskens, R. (1996). Combining Montague Semantics and Discourse Representation. In *Linguistics and Philosophy* 19, 143-186.
- Nouwen, R. (2003). *Plural Pronominal Anaphora in Context*. PhD dissertation, University of Utrecht.
- Partee, B. (1973). Some Structural Analogies between Tenses and Pronouns in English. In *Journal of Philosophy* 70, 601-609.
- Partee, B. (1984). Nominal and Temporal Anaphora. In *Linguistics and Philosophy* 7, 243-286.
- Pelletier, F.J. & L.K. Schubert (1989). Generically Speaking or Using Discourse Representation Theory to Interpret Generics. In *Properties, Types and Meanings*, Vol. 2, G. Chierchia, B.H. Partee & R. Turner (eds.), Dordrecht: Kluwer, 193-268.
- Peters, S. & D. Westerståhl (2006). *Quantifiers in Language and Logic*. Oxford: Clarendon Press.
- Pratt, V.R. (1976). Semantical Considerations on Floyd-Hoare Logic. In the *Proceedings of the 17<sup>th</sup> Annual IEEE Symposium on Foundations of Computer Science*, 109-121.
- Reinhart, T. (1997). Quantifier Scope: How Labor is Divided between QR and Choice Functions. In *Linguistics and Philosophy* 20, 335-397.
- Roberts, C. (1989). Modal Subordination and Pronominal Anaphora in Discourse. In *Linguistics and Philosophy* 12, 683-721.
- Root, R. (1986). *The Semantics of Anaphora in Discourse*. PhD dissertation, University of Texas at Austin.

- Rooth, M. (1987). Noun Phrase Interpretation in Montague Grammar, File Change Semantics and Situation Semantics. In *Generalized Quantifiers: Linguistic and Logical Approaches*, P. Gärdenfors (ed.), Dordrecht: Kluwer, 237-268.
- van Rooy, R. (1998). Modal Subordination in Questions. In the *Proceedings of Twendial 1998*, J. Hulstijn & A. Nijholt (eds.), 237-248.
- Sandu, G. (1993). On the Logic of Informational Independence and Its Applications. In *Journal of Philosophical Logic* 22, 29-60.
- Schlenker, P. (2005). Ontological Symmetry in Language: A Brief Manifesto. In *Mind and Language* 21, 504-539.
- Schwarzschild, R. (2002). Singleton Indefinites. In *Journal of Semantics* 19.3, 289-314.
- Stone, M. (1999). Reference to Possible Worlds. RuCCS Report 49, Rutgers University, New Brunswick.
- Väänänen, J. (2007). *Dependence Logic: A New Approach to Independence Friendly Logic*. Cambridge University Press.
- Wang, L., E. McCready & N. Asher 2006. Information Dependency in Quantificational Subordination. In *Where Semantics Meets Pragmatics*, K. Turner & K. von Heusinger (eds.), Elsevier (forthcoming).
- Westerståhl, D. (1989). Quantifiers in Formal and Natural Languages. In *Handbook of Philosophical Logic*, Vol. IV, D. Gabbay & F. Guentner (eds.), Dordrecht: Reidel, 1-131.
- Winter, Y. (1997). Choice Functions and the Scopal Semantics of Indefinites. In *Linguistics and Philosophy* 20, 399-467.