Generalized Quantifiers on Dependent Types: A System for Anaphora

Justyna Grudzińska, Marek Zawadowski,
Instytut Filozofii, Instytut Matematyki,
Uniwersytet Warszawski
j.grudzinska@uw.edu.pl
zawado@mimuw.edu.pl

January 31, 2014

Abstract

We propose a system for the interpretation of anaphoric relationships between unbound pronouns and quantifiers. The main technical contribution of our proposal consists in combining generalized quantifiers with dependent types. Empirically, our system allows a uniform treatment of all types of unbound anaphora, including the notoriously difficult cases such as quantificational subordination, cumulative and branching continuations, and 'donkey anaphora'.

Keywords: dependent type, generalized quantifier, unbound anaphora.

1 Unbound anaphora

In this paper we propose a system for the interpretation of unbound anaphora. The phenomenon of unbound anaphora refers to instances where anaphoric pronouns occur outside the syntactic scopes (i.e. the c-command domain) of their quantifier antecedents. The main kinds of unbound anaphora are:

(1) regular anaphora to quantifiers

(e.g. Most kids entered. They looked happy.)

(2) quantificational subordination

(e.g. Every man loves a woman. They kiss them.)

(3) relative clause 'donkey anaphora'

(e.g. Every farmer who owns a donkey beats it.)

(4) conditional 'donkey anaphora'

(e.g. If a farmer owns a donkey, he beats it.)

Unbound anaphoric pronouns have been dealt with in two main semantic paradigms: dynamic semantic theories ([Kamp 1981], [Kamp & Reyle 1993], [Groenendijk & Stokhof 1991], [Van den Berg 1996], [Krifka 1996], [Nouwen 2003], [Brasoveanu 2008]) and the E-type/D-type tradition ([Evans 1997], [Neale 1990], [Heim 1990], [Elbourne 2005]). In the dynamic semantic theories pronouns are taken to be (syntactically free, but semantically bound) variables, and context serves as a

medium supplying values for the variables. In the E-type/D-type tradition pronouns are treated as quantifiers (definite descriptions constructed from material in the antecedent sentences). Our system combines aspects of both families of theories. As in the E-type/ D-type tradition we treat unbound anaphoric pronouns as quantifiers; as in the systems of dynamic semantics context is used as a medium supplying (possibly dependent) types as their potential quantificational domains. Like Dekker's Predicate Logic with Anaphora and more recent multidimensional models ([Dekker 1994], [Dekker 2008a]), our system lends itself to the compositional treatment of unbound anaphora, while keeping a classical, static notion of truth.

The main novelty of our proposal consists in combining generalized quantifiers ([Mostowski 1957], [Lindström 1966], [Barwise & Cooper 1981]) with dependent types ([Martin-Löf 1972], [Ranta 1994]). Empirically, our system allows a uniform account of both regular anaphora to quantifiers and the notoriously difficult cases such as quantificational subordination, 'donkey anaphora', and also cumulative and branching continuations.

The paper is organized as follows. In Section 2 we introduce in an informal way the main features of our interpretational architecture. In Section 3 we show how to interpret a range of anaphoric data in our system. Finally, sections 4 and 5 define the syntax and semantics of the system.

2 Main features of the system

The main elements of our system are:

1. generalized quantifiers together with operations that lift quantifier phrases to chains of quantifiers (i.e. polyadic quantifiers): for capturing the readings available for (multi-) quantifier sentences;

2. context and type dependency: both (i) for the interpretation of language expressions (i.e. quantifiers, quantifier phrases, predicates, chains and sentences) and (ii) for modeling the dynamic aspects of quantification.

2.1 Context, types and dependent types

The variables of our system are always typed. We write $x : X$ to denote that the variable x is of type X and refer to this as a type specification of the variable x. Types, in this paper, are interpreted as sets. We write the interpretation of the type X as $\|X\|$.

Types can depend on variables of other types. Thus, if we already have a type specification $x : X$, then we can also have type $Y(x)$ depending on the variable x and we can declare a variable y of type Y by stating $y : Y(x)$. The fact that Y depends on X is modeled as a projection $\pi : \|Y\| \to \|X\|$. So that if the variable x of type X is interpreted as an element $a \in \|X\|$, $\|Y\|(a)$ is interpreted as the fiber of π over a (the preimage of $\{a\}$ under π), i.e.:

$$\|Y\|(a) = \{b \in \|Y\| : \pi(b) = a\}.$$

One standard natural language example of such a dependence of types is that if m is a variable of the type of months M, there is a type $D(m)$ of the days of the month m. If we interpret type M as a set $\|M\|$ of months, then we can interpret type D as a set of days of months in $\|M\|$, i.e. as a set of pairs:

$$\|D\| = \{\langle a, k \rangle \; : \; a \in \|M\|, \; k \text{ is (the number of) a day in month } a\}$$

equipped with the projection $\pi : \|D\| \to \|M\|$. The particular sets $\|D\|(a)$ of the days of the month a can be recovered as the fibers of this projection:

$$\|D\|(a) = \{d \in \|D\| : \pi(d) = a\}.$$

Such type dependencies can be nested, i.e., we can have a sequence of type specifications of the (individual) variables:

$$x : X, y : Y(x), z : Z(x, y)$$

Context for us is a partially ordered sequence of type specifications of the (individual) variables and it is interpreted as a parameter space, i.e. as a set of compatible n-tuples of elements of the sets corresponding to the types involved (compatible wrt all projections).

For the definitions of context and dependent types, see Sections 4.2 (syntax) and 5.1 (semantics).

2.2 Quantifiers, quantifier phrases and predicates

Our system defines quantifiers and predicates polymorphically. A generalized quantifier associates to every set Z a subset of the power set of Z [1]:

$$\|Q\|(Z) \subseteq \mathcal{P}(Z)$$

The interpretation $\|P\|$ of an n-ary predicate P associates to a tuple of sets $\vec{Z} = \langle Z_1, \ldots, Z_n \rangle$ a subset of the cartesian product of the sets involved [2]:

$$\|P\|(\vec{Z}) \subseteq Z_1 \times \ldots \times Z_n.$$

Quantifier phrases, e.g. *every man* or *some woman*, are interpreted as follows:

$$\|every_{m:man}\| = \{\|man\|\}$$

$$\|some_{w:woman}\| = \{X \subseteq \|woman\| : X \neq \emptyset\}$$

As an element of the denotation of a quantifier phrase like *every man* or *some woman* is homogeneous containing only men or women, we do not need to consider notions such as "live on" and "witness set" (for comparison, see [Barwise & Cooper 1981]).

The definitions of quantifiers, quantifier phrases and predicates are introduced and generalized to dependent types in Sections 4.4, 4.5 (syntax) and 5.3, 5.4 (semantics).

2.3 Chains of quantifiers and sentences

The interpretation of quantifier phrases is further extended into the interpretation of chains of quantifiers. Consider an example in (1):

(1) Two examiners marked six scripts.

Multi-quantifier sentences such as (1) have been known to be ambiguous with different readings corresponding to how various quantifiers are semantically related in the sentence. Thus a sentence like (1) admits of two scope-dependent readings where each of the two examiners marked six scripts (*two examiners* with wide scope), or where each of the

---

[1] Such an association might be required to satisfy some additional conditions (like invariance under bijections), but we shall not consider this issue here.

[2] We may allow such an association to be partial.

six scripts was marked by two examiners (*six scripts* with wide scope). There are also two further readings claimed for (1): the cumulative reading saying that each of the two examiners marked at least one of the six scripts, and each of the six scripts was marked by at least one of the two examiners, and the branching reading which says that each of the two examiners marked the same set of six scripts. To account for the readings available for such multi-quantifier sentences, we raise quantifier phrases to the front of a sentence to form (generalized) quantifier prefixes - chains of quantifiers. Chains of quantifiers are built from quantifier phrases using three chain-constructors: pack-formation rule $(?, \ldots, ?)$, sequential composition $?|?$, and parallel composition $\frac{?}{?}$. The semantical operations that correspond to the chain-constructors (known as cumulation, iteration and branching) capture in a compositional manner cumulative, scope-dependent and branching readings, respectively.

The idea of chain-constructors and the corresponding semantical operations builds on Mostowski's notion of quantifier ([Mostowski 1957]) further generalized by Lindström to a so-called polyadic quantifier ([Lindström 1966]). (See [Bellert & Zawadowski 1989], compare also [Keenan 1987], [Van Benthem 1989], [Keenan 1992], [Keenan 1993], [Westerståhl 1994]). To use a familiar example, a multi-quantifier prefix like $\forall_{m:M}\exists_{w:W}$ is thought of as a single two-place quantifier obtained by an operation on the two single quantifiers, and has as denotation:

$$\|\forall_{m:M}|\exists_{w:W}\| = \{R \subseteq \|M\| \times \|W\| : \{a \in \|M\| : \{b \in \|W\| : \langle a, b \rangle \in R\} \in \|\exists_{w:W}\|\} \in \|\forall_{m:M}\|\}.$$

The three chain-constructors and the corresponding semantical operations are introduced and generalized to (pre-) chains defined on dependent types in Sections 4.6, 4.7 (syntax) and 5.4 (semantics).

Finally, a sentence with a chain of quantifiers $Ch = Ch_{\vec{y}:\vec{Y}}$ and predicate $P = P(\vec{y})$, $Ch_{\vec{y}:\vec{Y}} \ P(\vec{y})$, is true iff the interpretation of the predicate (i.e. some set of compatible n-tuples) belongs to the interpretation of the chain (i.e. some family of sets of compatible n-tuples). For the definitions of a sentence and validity, see Sections 4.8 (syntax) and 5.5 (semantics).

3 Dynamic extensions of contexts

In this section we introduce further elements of our interpretational architecture by way of showing how to interpret a range of anaphoric data in our system: quantificational subordination (3.1), nested dependencies (3.2), regular anaphora to quantifiers (3.3), cumulative and branching continuations (3.4) and donkey anaphora of both the relative clause and conditional varieties (Section 3.5).

3.1 Quantificational subordination

Let us begin by considering an example in (1):

(1) Every man loves a woman. They kiss them.

We build the representation of the first sentence in (1):

$$L(\forall_{m:M}, \exists_{w:W}).$$

Sentences of English, contrary to sentences of our formal language, are often ambiguous. Hence one such representation can be associated with more than one sentence in our formal language. The next step thus involves disambiguation. We take quantifier phrases

of a given representation and organize them into all possible chains of quantifiers (with some restrictions imposed on particular quantifiers). The disambiguation process will not concern us here, but see [Bellert & Zawadowski 1989] for the extensive discussion of the restrictions on particular quantifiers concerning the places in prefixes at which they can occur [3].

In our system language expressions (i.e. quantifiers, quantifier phrases, predicates, (pre-) chains, and sentences) are all defined in context. Thus the first sentence in (1) (on the most natural interpretation where *a woman* depends on *every man*) translates into a sentence with a chain of quantifiers in a context:

$$\Gamma \vdash \forall_{m:M} | \exists_{w:W} L(m, w),$$

and says that the set of pairs, a man and a woman he loves, has the following property: the set of those men that love some woman each is the set of all men. The way to understand the second sentence in (1) (i.e., the anaphoric continuation) is that every man kisses the women he loves rather than those loved by someone else. Thus the first sentence in (1) must deliver some internal relation between the types corresponding to the two quantifier phrases. This observation presents a case of quantificational subordination and is well-known from the dynamic semantics literature ([Kamp & Reyle 1993], [Van den Berg 1996], [Krifka 1996], [Nouwen 2003]).

In our system the first sentence in (1) extends the context Γ by adding new variable specifications on newly formed types for every quantifier phrase in the chain:

$$Ch = \forall_{m:M} | \exists_{w:W}.$$

For the purpose of the formation of such new types we introduce a new **type constructor** $\mathbb{T}$ (see the definition in Section 4.9). That is, the first sentence in (1) (denoted as φ) extends the context by adding:

$$t_{\varphi, \forall_m} : \mathbb{T}_{\varphi, \forall_{m:M}}; \; t_{\varphi, \exists_w} : \mathbb{T}_{\varphi, \exists_{w:W}}(t_{\varphi, \forall_m})$$

The interpretation of types (that correspond to the quantifier phrases in the chain Ch) from the extended context Γ_{φ} are defined in a two-step procedure using the inductive clauses through which we define Ch but in the reverse direction (for the formal description of the procedure, see Section 5.6).

Step 1. We define fibers of new types by inverse induction.
Basic step. For the whole chain $Ch = \forall_{m:M} | \exists_{w:W}$ we put:

$$\|\mathbb{T}_{\varphi, \forall_{m:M} | \exists_{w:W}}\| := \|L\|.$$

Inductive step.

$$\|\mathbb{T}_{\varphi, \forall_{m:M}}\| = \{a \in \|M\| \; : \; \{b \in \|W\| \; : \; \langle a, b \rangle \in \|L\|\} \in \|\exists_{w:W}\|\}$$

and for $a \in \|M\|$

$$\|\mathbb{T}_{\varphi, \exists_{w:W}}\|(a) = \{b \in \|W\| \; : \; \langle a, b \rangle \in \|L\|\}$$

Step 2. We build dependent types from fibers.

$$\|\mathbb{T}_{\varphi, \forall_{m:M}}\| = \{a \in \|M\| \; : \; \{b \in \|W\| \; : \; \langle a, b \rangle \in \|L\|\} \in \|\exists_{w:W}\|\}$$

---

[3]In connection with complex sentences, we just add one further qualification: the scope of a quantifier is always clause-bounded.

$$\|\mathbb{T}_{\varphi,\exists_{w:W}}\| = \bigcup\{\{a\} \times \|\mathbb{T}_{\varphi,\exists_{w:W}}\|(a) : a \in \|\mathbb{T}_{\varphi,\forall_{m:M}}\|\}$$

Thus the first sentence in (1) extends the context by adding the type $\mathbb{T}_{\varphi,\forall_{m:M}}$, interpreted as $\|\mathbb{T}_{\varphi,\forall_{m:M}}\|$ (i.e. the set of men who love some women, in this case this set amounts to the entire set of men), and the dependent type $\mathbb{T}_{\varphi,\exists_{w:W}}(t_{\varphi,\forall_m})$, interpreted for $a \in \|\mathbb{T}_{\varphi,\forall_{m:M}}\|$ as $\|\mathbb{T}_{\varphi,\exists_{w:W}}\|(a)$ (i.e. the set of women loved by the man a).

Unbound anaphoric pronouns are interpreted with reference to the context created by the foregoing text: they are treated as universal quantifiers and newly formed (possibly dependent) types incrementally added to the context serve as their potential quantificational domains. That is, unbound anaphoric pronouns $they_m$ and $them_w$ in the second sentence of (1) have the ability to pick up and quantify universally over the respective interpretations. We represent the anaphoric continuation in (1) as

$$K(\forall_{t_{\varphi,\forall_m}:\mathbb{T}_{\varphi,\forall_{m:M}}}, \forall_{t_{\varphi,\exists_w}:\mathbb{T}_{\varphi,\exists_{w:W}}}(t_{\varphi,\forall_m})).$$

It translates into:

$$\Gamma_\varphi \vdash \forall_{t_{\varphi,\forall_m}:\mathbb{T}_{\varphi,\forall_{m:M}}} |\forall_{t_{\varphi,\exists_w}:\mathbb{T}_{\varphi,\exists_{w:W}}}(t_{\varphi,\forall_m}) K(t_{\varphi,\forall_m}, t_{\varphi,\exists_w}),$$

where:

$$\|\forall_{t_{\varphi,\forall_m}:\mathbb{T}_{\varphi,\forall_{m:M}}} |\forall_{t_{\varphi,\exists_w}:\mathbb{T}_{\varphi,\exists_{w:W}}}(t_{\varphi,\forall_m})\| = \{R \subseteq \|\mathbb{T}_{\varphi,\exists_{w:W}}\| : \{a \in \|\mathbb{T}_{\varphi,\forall_{m:M}}\| :$$

$$\{b \in \|\mathbb{T}_{\varphi,\exists_{w:W}}\|(a) : \langle a, b \rangle \in R\} \in \|\forall_{t_{\varphi,\exists_w}:\mathbb{T}_{\varphi,\exists_{w:W}}}(t_{\varphi,\forall_m})\|(a)\} \in \|\forall_{t_{\varphi,\forall_m}:\mathbb{T}_{\varphi,\forall_{m:M}}}\|\},$$

yielding the correct truth conditions *Every man kisses every woman he loves.*

3.2 Nested dependencies

As the type dependencies can be nested, our analysis can be extended to sentences involving three and more quantifiers. Consider examples in (2a) and (2b):

(2a) Every student bought most professors a flower. They will give them to them tomorrow.

(2b) Every student bought most professors a flower. They picked them carefully.

We represent the first sentence in (2a) and (2b) as $B(\forall_{s:S}, Most_{p:P}, \exists_{f:F})$. This sentence (on the interpretation where *a flower* depends on *most professors* that depends on *every student*) translates into a sentence:

$$\Gamma \vdash \forall_{s:S} |Most_{p:P}| \exists_{f:F} B(s, p, f),$$

and by the process of dynamic extension updates the context by adding new variable specifications on newly formed types for every quantifier phrase in Ch:

$$t_{\varphi,\forall_s} : \mathbb{T}_{\varphi,\forall_{s:S}}; \ t_{\varphi,Most_p} : \mathbb{T}_{\varphi,Most_{p:P}}(t_{\varphi,\forall_s}); \ t_{\varphi,\exists_f} : \mathbb{T}_{\varphi,\exists_{f:F}}(t_{\varphi,\forall_s}, t_{\varphi,Most_p})$$

We now apply our interpretation procedure.

Step 1.

Basic step. For the whole chain $Ch = \forall_{s:S} |Most_{p:P}| \exists_{f:F}$ we put:

$$\|\mathbb{T}_{\varphi,\forall_{s:S}|Most_{p:P}|\exists_{f:F}}\| := \|B\|.$$

Inductive step.

$$\|\mathbb{T}_{\varphi,\forall_{s:S}}\| = \{a \in \|S\| : \{b \in \|P\| : \{c \in \|F\| : \langle a,b,c\rangle \in \|B\|\} \in \|\exists_{f:F}\|\} \in \|Most_{p:P}\|\}$$

and for $a \in \|M\|$

$$\|\mathbb{T}_{\varphi,Most_{p:P}}\|(a) = \{b \in \|P\| : \{c \in \|F\| : \langle a,b,c\rangle \in \|B\|\} \in \|\exists_{f:F}\|\}$$

and for $a \in \|M\|$ and $b \in \|P\|$

$$\|\mathbb{T}_{\varphi,\exists_{f:F}}\|(a,b) = \{c \in \|F\| : \langle a,b,c\rangle \in \|B\|\}$$

Step 2.

$$\|\mathbb{T}_{\varphi,\forall_{s:S}}\| = \{a \in \|S\| : \{b \in \|P\| : \{c \in \|F\| : \langle a,b,c\rangle \in \|B\|\} \in \|\exists_{f:F}\|\} \in \|Most_{p:P}\|\}$$

$$\|\mathbb{T}_{\varphi,Most_{p:P}}\| = \bigcup\{\{a\} \times \|\mathbb{T}_{\varphi,Most_{p:P}}\|(a) : a \in \|\mathbb{T}_{\varphi,\forall_{s:S}}\|\}$$

$$\|\mathbb{T}_{\varphi,\exists_{f:F}}\| = \bigcup\{\{\langle a,b\rangle\} \times \|\mathbb{T}_{\varphi,\exists_{f:F}}\|(a,b) : a \in \|\mathbb{T}_{\varphi,\forall_{s:S}}\|, b \in \|\mathbb{T}_{\varphi,Most_{p:P}}\|(a)\}$$

Thus the first sentence in (2a) and (2b) extends the context by adding the type $\mathbb{T}_{\varphi,\forall_{s:S}}$ interpreted as $\|\mathbb{T}_{\varphi,\forall_{s:S}}\|$ (i.e. the set of students who bought for most his professors a flower), the dependent type $\mathbb{T}_{\varphi,Most_{p:P}}(t_{\varphi,\forall_s})$, interpreted for $a \in \|\mathbb{T}_{\varphi,\forall_{s:S}}\|$ as $\|\mathbb{T}_{\varphi,Most_{p:P}}\|(a)$ (i.e. the set of professors for whom the student a bought flowers), and another dependent type $\mathbb{T}_{\varphi,\exists_{f:F}}(t_{\varphi,\forall_s},t_{\varphi,Most_p})$, interpreted for $a \in \|\mathbb{T}_{\varphi,\forall_{s:S}}\|$ and $b \in \|\mathbb{T}_{\varphi,Most_{p:P}}\|(a)$ as $\|\mathbb{T}_{\varphi,\exists_{f:F}}\|(a,b)$ (i.e. the set of flowers that the student a bought for the professors b).

In the second sentence of (2a) the three pronouns $they_s$, $them_p$, and $them_f$ quantify universally over the respective interpretations. We represent the anaphoric continuation in (2a) as:

$$G(\forall_{t_{\varphi,\forall_s}:\mathbb{T}_{\varphi,\forall_{s:S}}}, \forall_{t_{\varphi,Most_p}:\mathbb{T}_{\varphi,Most_{p:P}}(t_{\varphi,\forall_s})}, \forall_{t_{\varphi,\exists_f}:\mathbb{T}_{\varphi,\exists_{f:F}}(t_{\varphi,\forall_s},t_{\varphi,Most_p})}).$$

It translates into:

$$\Gamma_\varphi \vdash \forall_{t_{\varphi,\forall_s}:\mathbb{T}_{\varphi,\forall_{s:S}}} |\forall_{t_{\varphi,Most_p}:\mathbb{T}_{\varphi,Most_{p:P}}(t_{\varphi,\forall_s})}|\forall_{t_{\varphi,\exists_f}:\mathbb{T}_{\varphi,\exists_{f:F}}(t_{\varphi,\forall_s},t_{\varphi,Most_p})} G(t_{\varphi,\forall_s},t_{\varphi,Most_p},t_{\varphi,\exists_f}),$$

where:

$$\|\forall_{t_{\varphi,\forall_s}:\mathbb{T}_{\varphi,\forall_{s:S}}}|\forall_{t_{\varphi,Most_p}:\mathbb{T}_{\varphi,Most_{p:P}}(t_{\varphi,\forall_s})}|\forall_{t_{\varphi,\exists_f}:\mathbb{T}_{\varphi,\exists_{f:F}}(t_{\varphi,\forall_s},t_{\varphi,Most_p})}\| = \{R \subseteq \|\mathbb{T}_{\varphi,\exists_{f:F}}\| :$$

$$\{a \in \|\mathbb{T}_{\varphi,\forall_{s:S}}\| : \{b \in \|\mathbb{T}_{\varphi,Most_{p:P}}\|(a) : \{c \in \|\mathbb{T}_{\varphi,\exists_{f:F}}\|(a,b) : \langle a,b,c\rangle \in R\}$$

$$\in \|\forall_{t_{\varphi,\exists_f}:\mathbb{T}_{\varphi,\exists_{f:F}}(t_{\varphi,\forall_s},t_{\varphi,Most_p})}\|(a,b)\} \in \|\forall_{t_{\varphi,Most_p}:\mathbb{T}_{\varphi,Most_{p:P}}(t_{\varphi,\forall_s})}\|(a)\} \in \|\forall_{t_{\varphi,\forall_s}:\mathbb{T}_{\varphi,\forall_{s:S}}}\|\},$$

yielding the correct truth conditions *Every student will give the respective professors the respective flowers he bought for them.*

In the second sentence of (2b) the pronoun $them_f$ quantifies universally over the set of flowers that the student $a \in \|\mathbb{T}_{\varphi,\forall_{s:S}}\|$ bought for the professors $b \in \|\mathbb{T}_{\varphi,Most_{p:P}}\|(a)$, so in order to be able to refer to such a set we need to use a type constructor Σ (for the definition of Σ-type and its interpretation, see Sections 4.3 and 5.2):

$$\Sigma_{t_{\varphi,Most_p}:\mathbb{T}_{\varphi,Most_{p:P}}(t_{\varphi,\forall_s})} \mathbb{T}_{\varphi,\exists_{f:F}}(t_{\varphi,\forall_s},t_{\varphi,Most_p})$$

To accommodate all of the extra processes needed to obtain a new context out of the old one we introduce a **refresh operation**. The refresh operation will include: addition of variable specifications on presupposed types (where by presupposed types we understand types belonging to the relevant common ground shared by the speaker and hearer); $\sum, \prod$ of the types given in the context, etc (see Section 4.10). Thus in our example the refresh operation applies so as to update the context by adding a new variable specification on a newly formed $\sum$-type (abbrev. $T_{\varphi,\Sigma}$):

$$t_{\varphi,\forall_s} : \mathbb{T}_{\varphi,\forall_{s:S}};\ t_{\varphi,\Sigma} : T_{\varphi,\Sigma}(t_{\varphi,\forall_s})$$

We represent the anaphoric continuation in (2b) as

$$P(\forall_{t_{\varphi,\forall_s}:\mathbb{T}_{\varphi,\forall_{s:S}}}, \forall_{t_{\varphi,\Sigma}:T_{\varphi,\Sigma}(t_{\varphi,\forall_s})}).$$

It translates into:

$$\Gamma_\varphi \vdash \forall_{t_{\varphi,\forall_s}:\mathbb{T}_{\varphi,\forall_{s:S}}} |\forall_{t_{\varphi,\Sigma}:T_{\varphi,\Sigma}(t_{\varphi,\forall_s})} P(t_{\varphi,\forall_s}, t_{\varphi,\Sigma}),$$

where

$$\|\forall_{t_{\varphi,\forall_s}:\mathbb{T}_{\varphi,\forall_{s:S}}} |\forall_{t_{\varphi,\Sigma}:T_{\varphi,\Sigma}(t_{\varphi,\forall_s})}\| = \{\langle a,c \rangle : a \in \|\mathbb{T}_{\varphi,\forall_{s:S}}\|,\ c \in \|T_{\varphi,\Sigma}\| : \{a \in \|\mathbb{T}_{\varphi,\forall_{s:S}}\| :$$

$$\{c \in \|T_{\varphi,\Sigma}\|(a) : \langle a,c \rangle \in R\} \in \|\forall_{t_{\varphi,\Sigma}:T_{\varphi,\Sigma}(t_{\varphi,\forall_s})}\|(a)\} \in \|\forall_{t_{\varphi,\forall_s}:\mathbb{T}_{\varphi,\forall_{s:S}}}\|\}.$$

yielding the correct truth conditions *Every student picked every flower he bought for most his professors carefully.*

3.3 Regular anaphora to quantifiers

Consider an example in (3):

(3) Most kids entered. They looked happy.

Regarding (3), the well-known observation from the dynamic semantics literature is that the anaphoric pronoun *they* refers to the so-called "scope set", i.e. the entire set of kids who entered ([Kamp & Reyle 1993], [Nouwen 2003], [Van den Berg 1996]).

We represent the first sentence in (3) as $E(Most_{k:K})$. The representation is unambiguous. It translates into a sentence:

$$\Gamma \vdash Most_{k:K} E(k),$$

and extends the context by adding:

$$t_{\varphi,Most_k} : \mathbb{T}_{\varphi,Most_{k:K}}$$

Since in this case the chain involved contains a single quantifier phrase $Ch = Most_{k:K}$, we put

$$\|\mathbb{T}_{\varphi,Most_{k:K}}\| := \|E\|$$

The pronoun *they* in the second sentence quantifies universally over the set $\|E\|$, yielding the correct truth-conditions for the anaphoric continuation *Every kid who entered looked happy.*

3.4 Cumulative and branching continuations

Dynamic extensions of contexts and their interpretation are also defined for cumulative and branching continuations (for the definitions, see 5.6). Consider examples in (4a) and (4b):

(4a) Last year three scientists wrote (a total of) five articles (between them). They presented them at major conferences.

(4b) Last year three scientists (each) wrote (the same) five articles. They presented them at major conferences.

As discussed in [Krifka 1996], [Dekker 2008b], the dynamics of the first sentence in (4a) and (4b) can deliver some (respectively: cumulative or branching) internal relation between the types corresponding to *three scientists* and *five articles* that can be elaborated upon in the anaphoric continuation.

We represent the first sentence in (4a) and (4b) as $W(Three_{s:S}, Five_{a:A})$. Interpreted cumulatively, as in (4a), it translates into a sentence:

$$\Gamma \vdash (Three_{s:S}, Five_{a:A})\ W(s,a).$$

Interpreted in a branching fashion, as in (4b), it translates into a sentence:

$$\Gamma \vdash \frac{Three_{s:S}}{Five_{a:A}}\ W(s,a).$$

The anaphoric continuation in (4a) can be interpreted in what Krifka calls a "correspondence" fashion (see [Krifka 1996]). For example, Dr. Smith wrote one article, co-authored two more with Dr. Nelson, who co-authored two more with Dr. Slack, and the scientists that cooperated in writing one or more articles also cooperated in presenting these (and no other) articles at major conferences. On our analysis, the first sentence in (4a) extends the context by adding the type corresponding to $(Three_{s:S}, Five_{a:A})$:

$$t_{\varphi, (Three_s, Five_a)} : T_{\varphi, (Three_{s:S};\ Five_{a:A})},$$

interpreted as a set of tuples

$$\|T_{\varphi, (Three_{s:S}, Five_{a:A})}\| = \{\langle c, d\rangle \mid c \in \|S\| \text{ and } d \in \|A\|\ : c \text{ wrote } d\}$$

The anaphoric continuation then quantifies universally over this type (i.e. a set of pairs):

$$\Gamma_\varphi \vdash \forall_{t_{\varphi, (Three_s, Five_a)}} P(t_{\varphi, (Three_s, Five_a)}),$$

yielding the desired truth-conditions *The respective scientists cooperated in presenting at major conferences the respective articles that they cooperated in writing*

The anaphoric continuation in (4b) can be interpreted in a branching fashion. For example, Dr. Smith, Dr. Nelson and Dr. Slack all co-authored all of the five articles, and all of the scientists involved presented at major conferences all of the articles involved. On our analysis, the first sentence in (4b) extends the context by adding:

$$t_{\varphi, Three_s} : T_{\varphi, Three_{s:S}};\ t_{\varphi, Five_a} : T_{\varphi, Five_{a:A}},$$

where:

$$\|T_{\varphi, Three_{s:S}}\| \in \|Three_{s:S}\|$$

$$\|T_{\varphi,Five_{a:A}}\| \in \|Five_{a:A}\|.$$

and moreover:

$$\|T_{\varphi,\frac{Three_{s:S}}{Five_{a:A}}}\| = \|T_{\varphi,Three_{s:S}}\| \times \|T_{\varphi,Five_{a:A}}\|,$$

The anaphoric continuation then quantifies universally over the respective types:

$$\Gamma_\varphi \vdash \frac{\forall_{t_{\varphi,Three_s}}}{\forall_{t_{\varphi,Five_a}}} P(t_{\varphi,Three_s}, t_{\varphi,Five_a}),$$

yielding the desired truth-conditions *All of the three scientists cooperated in presenting at major conferences all of the five articles that they co-authored*

3.5 Donkey anaphora

Our treatment of 'donkey anaphora' does not run into the 'proportion problem' and accommodates ambiguities claimed for 'donkey sentences'. Consider examples in (5a) and (5b):

(5a) Every farmer who owns a donkey beats it.

(5b) If a farmer owns a donkey, he beats it.

On our analysis, the pronouns in (5a) and (5b) quantify over (possibly dependent) types either introduced by the clauses restricting the main determiner (as in (5a)) or provided by the antecedent clauses (as in (5b)).

We represent the sentence in (5a) as $B(\forall_{t_{\varphi,f:O(f:F,\exists_{d:D})}}, \forall_{t_{\varphi,\exists_d}})$. To handle the dynamic contribution of relative clauses we include in our system $*$-**sentences** (i.e. sentences with dummy-quantifier phrases, for the definition see Section 4.8). The process of dynamic extension applies to a restrictor clause $O(f : F, \exists_{d:D})$ with a dummy-quantifier phrase $f : F$. It gets translated into a $*$-sentence:

$$\Gamma \vdash f : F | \exists_{d:D} O(f, d)$$

and we extend the context by dropping the specifications of variables: $(f : F, d : D)$ and adding new variable specifications on newly formed types for every (dummy-) quantifier phrase in the chain Ch^*:

$$t_{\varphi,f} : \mathbb{T}_{\varphi,f:F}; \ t_{\varphi,\exists_d} : \mathbb{T}_{\varphi,\exists_{d:D}}(t_{\varphi,f}),$$

The interpretation of types (that correspond to the (dummy-) quantifier phrases in the chain Ch^*) from the extended context Γ_φ are defined in our two-step procedure. Thus the $*$-sentence in (5a) extends the context by adding the type $\mathbb{T}_{\varphi,f:F}$ interpreted as $\|\mathbb{T}_{\varphi,f:F}\|$ (i.e. the set of farmers who own some donkeys), and the dependent type $\mathbb{T}_{\varphi,\exists_{d:D}}(t_{\varphi,f})$, interpreted for $a \in \|\mathbb{T}_{\varphi,f:F}\|$ as $\|\mathbb{T}_{\varphi,\exists_{d:D}}\|(a)$ (i.e. the set of donkeys owned by the farmer a). The main clause $B(\forall_{t_{\varphi,f}:\mathbb{T}_{\varphi,f:F}}, \forall_{t_{\varphi,\exists_d}:\mathbb{T}_{\varphi,\exists_{d:D}}(t_{\varphi,f})})$ translates into:

$$\Gamma_\varphi \vdash \forall_{t_{\varphi,f}:\mathbb{T}_{\varphi,f:F}} | \forall_{t_{\varphi,\exists_d}:\mathbb{T}_{\varphi,\exists_{d:D}}(t_{\varphi,f})} B(t_{\varphi,f}, t_{\varphi,\exists_d}),$$

giving the correct truth conditions *Every farmer who owns a donkey beats every donkey he owns*.

This analysis can be extended to account for more complicated 'donkey sentences' such as *Every farmer who owns donkeys beats most of them*. Importantly, the solution does not run into the 'proportion problem'. Since we quantify over fibers (and not over

$\langle farmer, donkey \rangle$ pairs), a sentence like *Most farmers who own a donkey beat it* comes out false if there are ten farmers who own one donkey and never beat them, and one farmer who owns twenty donkeys and beats all of them. Furthermore, sentences like (5a) have been claimed to be ambiguous between the so-called (i) strong reading: *Every farmer who owns a donkey beats* EVERY *donkey he owns* and, (ii) weak reading: *Every farmer who owns a donkey beats* AT LEAST ONE *donkey he owns.* Our analysis can accommodate this observation by taking the weak reading to simply employ the quantifier *some* in place of *every.*

Finally, we propose an analysis of (5b) along the lines of (5a). We follow the literature in assuming that conditional sentences such as (5b) involve an adverb of quantification. If no such adverb is overtly present, the quantificational force is universal. We represent the sentence in (5b) as $O(\exists_{f:F}, \exists_{d:D}) \to_{QAdverb} B(\forall_{t_{\varphi, \exists_f}}, \forall_{t_{\varphi, \exists_d}})$. The process of dynamic extension applies to the antecedent clause $O(\exists_{f:F}, \exists_{d:D})$. It gets translated into a sentence:

$$\Gamma \vdash \exists f : F | \exists_{d:D} O(f, d)$$

and we extend the context by dropping the specifications of variables: $(f : F, d : D)$ and adding new variable specifications on newly formed types for every quantifier phrase in the chain:

$$t_{\varphi, \exists f} : \mathbb{T}_{\varphi, \exists f:F}; \ t_{\varphi, \exists_d} : \mathbb{T}_{\varphi, \exists_{d:D}}(t_{\varphi, \exists f}).$$

The interpretation of types (that correspond to the quantifier phrases in the chain) from the extended context Γ_φ are defined in our usual procedure. Thus the antecedent sentence in (5b) extends the context by adding the type $\mathbb{T}_{\varphi, \exists_{f:F}}$ interpreted as $\|\mathbb{T}_{\varphi, \exists_{f:F}}\|$ (i.e. the set of farmers who own some donkeys), and the dependent type $\mathbb{T}_{\varphi, \exists_{d:D}}(t_{\varphi, \exists f})$, interpreted for $a \in \|\mathbb{T}_{\varphi, \exists_{f:F}}\|$ as $\|\mathbb{T}_{\varphi, \exists_{d:D}}\|(a)$ (i.e. the set of donkeys owned by the farmer a). The consequent clause $B(\forall_{t_{\varphi, \exists_f} : \mathbb{T}_{\varphi, \exists_{f:F}}}, \forall_{t_{\varphi, \exists_d} : \mathbb{T}_{\varphi, \exists_{d:D}}(t_{\varphi, \exists f})})$ translates into:

$$\Gamma_\varphi \vdash \forall_{t_{\varphi, \exists_f} : \mathbb{T}_{\varphi, \exists_{f:F}}} | \forall_{t_{\varphi, \exists_d} : \mathbb{T}_{\varphi, \exists_{d:D}}(t_{\varphi, \exists f})} B(t_{\varphi, \exists_f}, t_{\varphi, \exists_d}),$$

giving the correct truth conditions *Every farmer who owns a donkey beats every donkey he owns.*

Sentences (5a) and (5b) have generally been deemed equivalent, and so are our associated translations. Importantly again, the solution does not run into the 'proportion problem', if the involved adverb of quantification is *usually* or *often* (the counterpart of *most*). Furthermore, some authors claim sentences like (5b) are three-way ambiguous, according to whether the counting takes into account: (i) only the farmers (who own a donkey); (ii) only the donkeys (that each farmer owns); (iii) $\langle farmer, donkey \rangle$ pairs ([Kadmon 1987], [Heim 1990]). Our analysis can accommodate this observation by correlating the three readings with three semantical relations between quantifier phrases $\exists_{f:F}$, $\exists_{d:D}$ in the antecedent clause of the conditional statement:

(i) $\exists_{f:F} | \exists_{d:D}$ - the restrictor of the $QAdverb$ extends the context by adding a dependent type: $t_f : T_F; t_d : T_D(t_f)$

(ii) $\exists_{d:D} | \exists_{f:F}$ - the restrictor of the $QAdverb$ extends the context by adding a dependent type: $t_d : T_D; t_f : T_F(t_d)$

(iii) $\exists_{f:F}$ and $\exists_{d:D}$ are in a pack $(\exists_{f:F}, \exists_{d:D})$ - the restrictor extends the context by adding a type interpreted as a set of $\langle farmer, donkey \rangle$ pairs st. the farmer owns the donkey.

4 System - syntax

This and the following section define, respectively, the syntax and the semantics of our system.

4.1 Alphabet

The alphabet consists of

1. type variables $X, Y, Z, \ldots$;

2. type constants $M, men, women, \ldots$;

3. type constructors: $\sum, \prod, \mathbb{T}$;

4. individual variables $x, y, z, \ldots$;

5. predicates $P, P', P_1, \ldots$ (with arities specified);

6. quantifier symbols $\exists, \forall, Three, Five, Q_1, Q_2, \ldots$;

7. three chain constructors: $?|?$, $\frac{?}{?}$, $(?, \ldots, ?)$.

4.2 Contexts

A context is a list of type specifications of (individual) variables. Empty context $\emptyset$ is a context. If we have a context

$$\Gamma = x_1 : X_1, \ldots, x_k : X_k(\langle x_i \rangle_{i \in J_k}), \ldots, x_n : X_n(\langle x_i \rangle_{i \in J_n})$$

then the judgement

$$\vdash \Gamma : \mathsf{context}$$

expresses this fact. Having a context Γ as above, we can declare a type X_{n+1} in that context

$$\Gamma \vdash X_{n+1}(\langle x_i \rangle_{i \in J_{n+1}}) : \mathsf{type}$$

where $J_{n+1} \subseteq \{1, \ldots, n\}$ such that if $i \in J_{n+1}$, then $J_i \subseteq J_{n+1}$, $J_1 = \emptyset$. The type X_{n+1} depends on variables $\langle x_i \rangle_{i \in J_{n+1}}$. Now, we can declare a new variable of the type $X_{n+1}(\langle x_i \rangle_{i \in J_{n+1}})$ in the context Γ

$$\Gamma \vdash x_{n+1} : X_{n+1}(\langle x_i \rangle_{i \in J_{n+1}})$$

and extend the context Γ by adding this variable specification, i.e. we have

$$\vdash \Gamma, x_{n+1} : X_{n+1}(\langle x_i \rangle_{i \in J_{n+1}}) : \mathsf{context}$$

Γ' is a *subcontext* of Γ if Γ' is a context and a sublist of Γ. Let Δ be a list of variable specifications from a context Γ, Δ' the least subcontext of Γ containing Δ. We say that Δ is *convex* iff $\Delta' - \Delta$ is again a context.

The variables the types depend on are always explicitly written down in specifications. We can think of a context as (a linearization of) a partially ordered set of declarations such that the declaration of a variable x (of type X) precedes the declaration of the variable y (of type Y) iff the type Y depends on the variable x.

4.3 Type formation: Σ-types and Π-types

Having a type declaration

$$\Gamma, y : Y(\vec{x}) \vdash Z(\vec{y}) : \mathsf{type}$$

with y occurring in the list $\vec{y}$ we can declare Σ-type

$$\Gamma \vdash \Sigma_{y:Y(\vec{x})} Z(\vec{y}) : \mathsf{type}$$

and also Π-type

$$\Gamma \vdash \Pi_{y:Y(\vec{x})} Z(\vec{y}) : \mathsf{type}$$

So declared types do not depend on the variable y. Now we can specify new variables of those types.

4.4 Quantifier-free formulas

For our purpose we need only predicates applied to variables. So we have

$$\Gamma \vdash P(x_1, \ldots, x_n) : \mathsf{qf\text{-}formula}$$

whenever P is an n-ary predicate and the specifications of the variables $x_1, \ldots, x_n$ form a subcontext of Γ.

4.5 Quantifier phrases

If we have a context $\Gamma, y : Y(\vec{x}), \Delta$ and quantifier symbol Q, then we can form a *quantifier phrase* $Q_{y:Y(\vec{x})}$ in that context. We write

$$\Gamma, y : Y(\vec{x}), \Delta \vdash Q_{y:Y(\vec{x})} : \mathsf{QP}$$

to express this fact. In a quantifier prase $Q_{y:Y(\vec{x})}$

1. the variable y is the *binding variable* and

2. the variables $\vec{x}$ are *indexing variables.*

4.6 Packs of quantifiers

Quantifiers phrases can be grouped together to form a pack of quantifiers. The pack of quantifiers formation rule is as follows.

$$\frac{\Gamma \vdash Q_{i\ y_i:Y_i(\vec{x}_i)} : \mathsf{QP} \quad i = 1, \ldots k}{\Gamma \vdash (Q_{1\ y_1:Y_1(\vec{x}_1)}, \ldots, Q_{k\ y_k:Y_k(\vec{x}_k)}) : \mathsf{pack}}$$

where, with $\vec{y} = y_1, \ldots, y_k$ and $\vec{x} = \bigcup_{i=1}^{k} \vec{x}_i$, we have that $y_i \neq y_j$ for $i \neq j$ and $\vec{y} \cap \vec{x} = \emptyset$. In so constructed pack

1. the binding variables are $\vec{y}$ and

2. the indexing variables are $\vec{x}$.

We can denote such a pack $Pc_{\vec{y}:\vec{Y}(\vec{x})}$ to indicate the variables involved. One-element pack will be denoted and treated as a quantifier phrase. This is why we denote such a pack as $Q_{y:Y(\vec{x})}$ rather than $(Q_{y:Y(\vec{x})})$.

4.7 Pre-chains and chains of quantifiers

Chains and pre-chains of quantifiers have binding variables and indexing variables. By $Ch_{\vec{y}:\vec{Y}(\vec{x})}$ we denote a pre-chain with binding variables $\vec{y}$ and indexing variables $\vec{x}$ so that the type of the variable y_i is $Y_i(\vec{x}_i)$ with $\bigcup_i \vec{x}_i = \vec{x}$. Chains of quantifiers are pre-chains in which all indexing variables are bound. Pre-chains of quantifiers arrange quantifier phrases into N-free pre-orders, subject to some binding conditions. Mutually comparable QPs in a pre-chain sit in one pack. Thus the pre-chains are built from packs via two chain-constructors of sequential $?|?$ and parallel composition $\frac{?}{?}$.

The chain formation rules are as follows.

1. Packs of quantifiers are pre-chains of quantifiers with the same binding variable and the same indexing variables, i.e.

$$\frac{\Gamma \vdash Pc_{\vec{y}:\vec{Y}(\vec{x})} : \text{pack}}{\Gamma \vdash Pc_{\vec{y}:\vec{Y}(\vec{x})} : \text{pre-chain}}$$

2. *Sequential composition of pre-chains*

$$\frac{\Gamma \vdash Ch_{1\ \vec{y}_1:\vec{Y}_1(\vec{x}_1)} : \text{pre-chain}, \quad \Gamma \vdash Ch_{2\ \vec{y}_2:\vec{Y}_2(\vec{x}_2)} : \text{pre-chain}}{\Gamma \vdash Ch_{1\ \vec{y}_1:\vec{Y}_1(\vec{x}_1)}|Ch_{2\ \vec{y}_2:\vec{Y}_2(\vec{x}_2)} : \text{pre-chain}}$$

provided

(a) $\vec{y}_2 \cap (\vec{y}_1 \cup \vec{x}_1) = \emptyset$,

(b) the specifications of the variables $(\vec{x}_1 \cup \vec{x}_2) - (\vec{y}_1 \cup \vec{y}_2)$ form a context, a subcontext of Γ.

In so obtained pre-chain

(a) the binding variables are $\vec{y}_1 \cup \vec{y}_2$ and

(b) the indexing variables are $\vec{x}_1 \cup \vec{x}_2$.

3. *Parallel composition of pre-chains*

$$\frac{\Gamma \vdash Ch_{1\ \vec{y}_1:\vec{Y}_1(\vec{x}_1)} : \text{pre-chain}, \quad \Gamma \vdash Ch_{2\ \vec{y}_2:\vec{Y}_2(\vec{x}_2)} : \text{pre-chain}}{\Gamma \vdash \frac{Ch_{1\ \vec{y}_1:\vec{Y}_1(\vec{x}_1)}}{Ch_{2\ \vec{y}_2:\vec{Y}_2(\vec{x}_2)}} : \text{pre-chain}}$$

provided $\vec{y}_2 \cap (\vec{y}_1 \cup \vec{x}_1) = \emptyset = \vec{y}_1 \cap (\vec{y}_2 \cup \vec{x}_2)$.

As above, in so obtained pre-chain

(a) the binding variables are $\vec{y}_1 \cup \vec{y}_2$ and

(b) the indexing variables are $\vec{x}_1 \cup \vec{x}_2$.

A pre-chain of quantifiers $Ch_{\vec{y}:\vec{Y}(\vec{x})}$ is a *chain* iff $\vec{x} \subseteq \vec{y}$. The following

$$\Gamma \vdash Ch_{\vec{y}:\vec{Y}(\vec{x})} : \text{chain}$$

expresses the fact that $Ch_{\vec{y}:\vec{Y}(\vec{x})}$ is a chain of quantifiers in the context Γ.

4.8 Formulas, sentences and ∗-sentences

The formulas have binding variables, indexing variables and argument variables. We write $\varphi_{\vec{y}:Y(\vec{x})}(\vec{z})$ for a formula with binding variables $\vec{y}$, indexing variables $\vec{x}$ and argument variables $\vec{z}$. We have the following formation rule for formulas

$$\frac{\Gamma \vdash A(\vec{z}) : \mathsf{qf\text{-}formula}, \quad \Gamma \vdash Ch_{\vec{y}:\vec{Y}(\vec{x})} : \mathsf{pre\text{-}chain},}{\Gamma \vdash Ch_{\vec{y}:\vec{Y}(\vec{x})}\ A(\vec{z}) : \mathsf{formula}}$$

provided $\vec{y}$ is *final* in $\vec{z}$, i.e., $\vec{y} \subseteq \vec{z}$ and the list of variable specifications of $\vec{z} - \vec{y}$ is a subcontext of Γ. In so constructed formula

1. the binding variables are $\vec{y}$ and

2. the indexing variables are $\vec{x}$ and

3. the argument variables are $\vec{z}$.

A formula $\varphi_{\vec{y}:Y(\vec{x})}(\vec{z})$ is a *sentence* iff $\vec{z} \subseteq \vec{y}$ and $\vec{x} \subseteq \vec{y}$. So a sentence is a formula without free variables, neither individual nor indexing. The following

$$\Gamma \vdash \varphi_{\vec{y}:Y(\vec{x})}(\vec{z}) : \mathsf{sentence}$$

expresses the fact that $\varphi_{\vec{y}:Y(\vec{x})}(\vec{z})$ is a sentence formed in the context Γ.

We shall also consider some special formulas that we call ∗-sentences. A formula $\varphi_{\vec{y}:Y(\vec{x})}(\vec{z})$ is a *∗-sentence* if $\vec{x} \subseteq \vec{y} \cup \vec{z}$ but the set $\vec{z} - \vec{y}$ is possibly not empty and moreover the type of each variable in $\vec{z} - \vec{y}$ is *constant*, i.e., it does not depend on variables of other types. In such case we consider the set $\vec{z} - \vec{y}$ as a set of biding variables of an additional pack called a *dummy pack* that is placed in front of the whole chain Ch. The chain 'extended' by this dummy pack will be denoted by Ch^*. Clearly, if $\vec{z} - \vec{y}$ is empty there is no dummy pack and the chain Ch^* is Ch, i.e. sentences are ∗-sentences without dummy packs. We write

$$\Gamma \vdash \varphi_{\vec{y}:Y(\vec{x})}(\vec{z}) : \text{∗-sentence}$$

to express the fact that $\varphi_{\vec{y}:Y(\vec{x})}(\vec{z})$ is a ∗-sentence formed in the context Γ.

Having formed a ∗-sentence φ we can form a new context Γ_φ defined in the section 4.9.

Notation For semantics we need some notation for the variables in the ∗-sentence. Suppose we have a ∗-sentence

$$\Gamma \vdash Ch_{\vec{y}:Y(\vec{x})}\ P(\vec{z}) : \text{∗-sentence}$$

We define

1. The environment of pre-chain Ch: $Env(Ch) = Env(Ch_{\vec{y}:\vec{Y}(\vec{x})})$ - is the context defining variables $\vec{x} - \vec{y}$;

2. The binding variables of pre-chain Ch: $Bv(Ch) = Bv(Ch_{\vec{y}:\vec{Y}(\vec{x})})$ - is the convex set of declarations in Γ of the binding variables in $\vec{y}$;

3. $\mathbf{env}(Ch) = \mathbf{env}(Ch_{\vec{y}:\vec{Y}(\vec{x})})$ - the set of variables in the environment of Ch, i.e. $\vec{x} - \vec{y}$;

4. $\mathbf{bv}(Ch) = \mathbf{bv}(Ch_{\vec{y}:\vec{Y}(\vec{x})})$ - the set of biding variables $\vec{y}$;

5. The environment of a pre-chain Ch' in a $*$-sentence $\varphi = Ch_{\vec{y}:\vec{Y}(\vec{x})}\ P(\vec{z})$, denoted $Env_\varphi(Ch')$, is the set of binding variables in all the packs in Ch^* that are $<_\varphi$-smaller than all packs in Ch'. Note $Env(Ch') \subseteq Env_\varphi(Ch')$. If $Ch' = Ch_1|Ch_2$ is a sub-pre-chain of the chain $Ch_{\vec{y}:\vec{Y}(\vec{x})}$, then $Env_\varphi(Ch_2) = Env_\varphi(Ch_1) \cup Bv(Ch_1)$ and $Env_\varphi(Ch_1) = Env_\varphi(Ch')$.

4.9 Type formation $\mathbb{T}$

Suppose we have constructed a $*$-sentence in a context

$$\Gamma \vdash Ch_{\vec{y}:\vec{Y}(\vec{x})}\ A(\vec{z}) : *\text{-sentence}$$

We write φ for $Ch_{\vec{y}:\vec{Y}(\vec{x})}\ A(\vec{z})$.

We form a context Γ_φ dropping the specifications of variables $\vec{z}$ and adding one type and one variable specification for each pack in $Packs_{Ch^*}$.

Let $\check{\Gamma}$ denote the context Γ with the specifications of the variables $\vec{z}$ deleted. Suppose $\Phi \in Packs_{Ch^*}$ and Γ' is an extension of the context $\check{\Gamma}$ such that one variable specification $t_{\Phi',\varphi} : T_{\Phi',\varphi}$ was already added for each pack $\Phi' \in Packs_{Ch^*}$ such that $\Phi' <_{Ch^*} \Phi$ but not for Φ yet. Then we declare a type

$$\Gamma' \vdash T_{\Phi,\varphi}(\langle t_{\Phi',\varphi}\rangle_{\Phi' \in Packs_{Ch^*}, \Phi' <_{Ch^*} \Phi}) : \text{type}$$

and we extend the context Γ' by a specification of a variable $t_{\Phi,\varphi}$ of that type

$$\Gamma', t_{\Phi,\varphi} : T_{\Phi,\varphi}(\langle t_{\Phi',\varphi}\rangle_{\Phi' \in Packs_{Ch^*}, \Phi' <_{Ch^*} \Phi}) : \text{context}$$

The context obtained from $\check{\Gamma}$ by adding the new variables corresponding to all the packs $Packs_{Ch^*}$ as above will be denoted by

$$\Gamma_\varphi = \check{\Gamma} \cup \mathbb{T}(Ch_{\vec{y}:\vec{Y}(\vec{x})}\ A(\vec{z})).$$

At the end we add another context formation rule

$$\frac{\Gamma \vdash Ch_{\vec{y}:\vec{Y}(\vec{x})}\ A(\vec{z}) : *\text{-sentence},}{\Gamma_\varphi : \text{context}}$$

Then we can build another formula starting in the context Γ_φ. This process can be iterated. Thus in this system sentence φ in a context Γ is constructed via a *specifying sequence of formulas*, with the last formula being the sentence φ. The specifying sequence for φ is a sequence of judgements

$$\Gamma = \Gamma_0 \vdash \varphi_0 : *\text{-sentence}$$

$$\Gamma_1 = (\Gamma_0)_{\varphi_0} \vdash \varphi_1 : *\text{-sentence}$$

$$\dots$$

$$\Gamma_{n-1} = ((\Gamma_{n-2})_{\varphi_{n-2}}) \vdash \varphi_{n-1} : *\text{-sentence}$$

$$\Delta = ((\Gamma_{n-1})_{\varphi_{n-1}}) \vdash \varphi_n = \varphi : \text{sentence}$$

In order to express that such a sequence for φ exists we write

$$\Gamma \vdash \Delta \vdash \varphi : \text{sentence}$$

4.10 Discourse and pure stories

Pure story is a sequence of sentences in contexts

$$\Gamma_0 \vdash \Gamma_1 \vdash \varphi_1$$

The above means: starting in the context Γ_0 we build a sentence φ_1 in the context Γ_1.

$$\mathsf{refresh}((\Gamma_1)_{\varphi_1}) \vdash \Gamma_2 \vdash \varphi_2$$

$$\ldots$$

$$\mathsf{refresh}((\Gamma_{n-1})_{\varphi_{n-1}}) \vdash \Gamma_n \vdash \varphi_n$$

such that the next context $\mathsf{refresh}((\Gamma_{i+1})_{\varphi_{i+1}})$ is obtained from $(\Gamma_{i+1})_{\varphi_{i+1}}$ by application of the following operations

1. addition of variable specifications on presupposed types (weakening);

2. $\sum, \prod$ of these;

3. other pragmatic processes to be further studied.

5 System - semantics
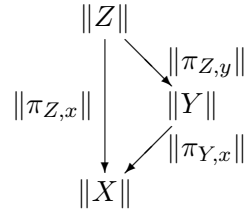
5.1 Interpretation of dependent types

The context Γ

$$\vdash x : X(\ldots), \ldots, y : Y(\ldots, x, \ldots), \ldots, z : Z(\ldots, x, y, \ldots) : \mathsf{context}$$

gives rise to a dependence graph. A *dependence graph* $DG_\Gamma = (T_\Gamma, E_\Gamma)$ for the context Γ has types of Γ as vertices and an edge $\pi_{Y,x} : Y \to X$ for every variable specification $x : X(\ldots)$ in Γ and every type $Y(\ldots, x, \ldots)$ occurring in Γ that depends on x.

The *dependence diagram* for the context Γ is an association $\| - \| : DG_\Gamma \to Set$ to every type X in T_Γ a set $\|X\|$ and every edge $\pi_{Y,x} : Y \to X$ in E_Γ a function $\|\pi_{Y,x}\| : \|Y\| \to \|X\|$, so that whenever we have a triangle of edges in E_Γ
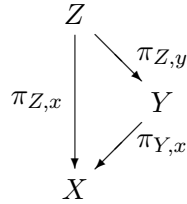
the corresponding triangle of functions

commutes, i.e.

$$\|\pi_{Z,x}\| = \|\pi_{Y,x}\| \circ \|\pi_{Z,y}\|.$$

The interpretation of the context Γ, the *parameter space* $\|\Gamma\|$, is the limit of the dependence diagram $\| - \| : DG_\Gamma \to Set$. More specifically,

$$\|\Gamma\| = \|x : X(\ldots), \ldots, y : Y(\ldots, x, \ldots), \ldots, z : Z(\ldots, x, y, \ldots)\| =$$

$$= \{\vec{a} : dom(\vec{a}) = \mathbf{var}(\Gamma), \ \ \vec{a}(z) \in \|Z\|(\vec{a}\lceil\mathbf{env}(Z)), \|\pi_{Z,x}\|(\vec{a}(z)) = \vec{a}(x),$$

$$\text{for } z : Z \text{ in } \Gamma, \ x \in \mathbf{env}Z\}$$

where $\mathbf{var}(\Gamma)$ denotes variables specified in Γ and $\mathbf{env}(Z)$ denotes indexing variables of the type Z.

5.2   Interpretation of Σ- and Π-types

As in this paper we are not going to use Π-types, we only include the interpretation of a Σ-type. For

$$\Gamma \vdash \Sigma_{y:Y(\vec{x})}Z(\vec{y}) : \mathsf{type}$$

we define

$$\|\Sigma_{y:Y(\vec{x})}Z(\vec{y})\| \ = \ \coprod_{b\in\|Y\|} (\{b\} \times \|\pi_{Z,y}\|^{-1}(b))$$

If a variable x of type X occurs in $\vec{y}$ and $x \neq y$, then we define projection

$$\|\pi_{\Sigma_{y:Y(\vec{x})}Z(\vec{y}),x}\| : \|\Sigma_{y:Y(\vec{x})}Z(\vec{y})\| \longrightarrow \|X\|$$

so that

$$\|\pi_{\Sigma_{y:Y(\vec{x})}Z(\vec{y}),x}\|(b,c) = \|\pi_{Z,x}\|(c)$$

for $b \in \|Y\|$ and $c \in \|\pi_{Z,y}\|^{-1}(b)$.

5.3   Interpretation of predicates and quantifier symbols

Both predicates and quantifiers are interpreted polymorphically.

If we have a predicate P defined in a context Γ:

$$x_1 : X_1, \ldots, x_n : X_n(\langle x_i \rangle_{i \in J_n]}) \vdash P(x_1, \ldots, x_n) : \mathsf{qf\text{-}formula}$$

then, for any interpretation of the context $\|\Gamma\|$, it is interpreted as a subset of its parameter set, i.e. $\|P\|(\|\Gamma\|) \subseteq \|\Gamma\|$.

Quantifier symbol Q is interpreted as quantifier $\|Q\|$ i.e. an association to every[4] set Z a subset $\|Q\|(Z) \subseteq \mathcal{P}(Z)$.

---

[4]This association can be partial.

5.4   Interpretation of chains of quantifiers

We interpret QP's, packs, pre-chains, and chains in the environment of a sentence Env_φ. This is the only case that is needed. We could interpret the aforementioned syntactic objects in their natural environment Env (i.e. independently of any given sentence) but it would unnecessarily complicate some definitions. Thus having a $(*\text{-})$sentence $\varphi = Ch_{\vec{y}:Y(\vec{x})}\ P(\vec{z})$ (defined in a context Γ) and a sub-pre-chain (QP, pack) Ch', for $\vec{a} \in \|Env_\varphi(Ch')\|$ we define the meaning of

$$\|Ch'\|(\vec{a})$$

Notation Let $\varphi = Ch_{\vec{y}:\vec{Y}}\ P(\vec{y})$ be a $*$-sentence built in a context Γ, Ch' a pre-chain used in the construction of the $(*)$-chain Ch. Then $Env_\varphi(Ch')$ is a sub-context of Γ disjoint from the convex set $Bv(Ch')$ and $Env_\varphi(Ch'), Bv(Ch')$ is a sub-context of Γ. For $\vec{a} \in \|Env_\varphi(Ch')\|$ we define $\|Bv(Ch')\|(\vec{a})$ to be the largest set such that

$$\{\vec{a}\} \times \|Bv(Ch')\|(\vec{a}) \subseteq \|Env_\varphi(Ch'), Bv(Ch')\|$$

Interpretation of quantifier phrases

Quantifier phrases. If we have a quantifier phrase

$$\Gamma \vdash Q_{y:Y(\vec{x})} : \mathsf{QP}$$

and $\vec{a} \in \|Env_\varphi(Q_{y:Y(\vec{x})})\|$, then it is interpreted as $\|Q\|(\|Y\|(\vec{a})) \subseteq \mathcal{P}(\|Y\|(\vec{a}\lceil\vec{x}))$.

Interpretation of packs

If we have a pack of quantifiers in the sentence φ

$$Pc = (Q1_{y_1:Y_1(\vec{x}_1)}, \dots Qn_{y_n:Y_n(\vec{x}_n)})$$

and $\vec{a} \in \|Env_\varphi(Pc)\|$, then its interpretation with the parameter $\vec{a}$ is

$$\|Pc\|(\vec{a}) = \|(Q1_{y_1:Y_1(\vec{x}_1)}, \dots, Qn_{y_n:Y_n(\vec{x}_n)})\|(\vec{a}) =$$

$$= \{A \subseteq \prod_{i=1}^{n} \|Y_i\|(\vec{a}\lceil\vec{x}_i)\ :\ \pi_i(A) \in \|Q_i\|(\|Y_i\|(\vec{a}\lceil\vec{x}_i), \text{ for } i = 1, \dots, n\}$$

where π_i is the i-th projection from the product.

Interpretation of chain constructors

Parallel composition. For a pre-chain of quantifiers in the sentence φ

$$Ch' = \frac{Ch1_{\vec{y}_1:\vec{Y}_1(\vec{x}_1)}}{Ch2_{\vec{y}_2:\vec{Y}_2(\vec{x}_2)}}$$

and $\vec{a} \in \|Env_\varphi(Ch')\|$ we define

$$\|\frac{Ch1_{\vec{y}_1:\vec{Y}_1(\vec{x}_1)}}{Ch2_{\vec{y}_2:\vec{Y}_2(\vec{x}_2)}}\|(\vec{a}) = \{A \times B\ :\ A \in \|Ch1_{\vec{y}_1:\vec{Y}_1(\vec{x}_1)}\|(\vec{a}\lceil\vec{x}_1) \text{ and } B \in \|Ch2_{\vec{y}_2:\vec{Y}_2(\vec{x}_2)}\|(\vec{a}\lceil\vec{x}_2)\}$$

Sequential composition. For a pre-chain of quantifiers in the sentence φ

$$Ch' = Ch_{1\vec{y}_1:\vec{Y}_1(\vec{x}_1)} | Ch_{2\vec{y}_2:\vec{Y}_2(\vec{x}_2)}$$

and $\vec{a} \in \|Env_\varphi(Ch')\|$ we define

$$\|Ch_{1\vec{y}_1:\vec{Y}_1(\vec{x}_1)} | Ch_{2\vec{y}_2:\vec{Y}_2(\vec{x}_2)}\|(\vec{a}) = \{R \subseteq \|Bv(Ch')\|(\vec{a}) \ : \ \{\vec{b} \in \|Bv(Ch_1)\|(\vec{a}) \ :$$

$$\{\vec{c} \in \|Bv(Ch_2)\|(\vec{a},\vec{b}) \ : \ \langle \vec{b}, \vec{c} \rangle \in R\} \in \|Ch_{2\vec{y}_2:\vec{Y}_2(\vec{x}_2)}\|(\vec{a},\vec{b})\} \in \|Ch_{1\vec{y}_1:\vec{Y}_1(\vec{x}_1)}\|(\vec{a})\}$$

5.5 Validity

A sentence

$$\Gamma \vdash Ch_{\vec{y}:\vec{Y}} \ P(\vec{y})$$

is true under the above interpretation iff

$$\|P\|(\|\vec{y} : \vec{Y}\|) \in \|Ch_{\vec{y}:\vec{Y}}\|$$

5.6 Interpretation of dynamic extensions

Suppose we obtain a context Γ_φ from Γ by the following rule

$$\frac{\Gamma \vdash Ch_{\vec{y}:\vec{Y}(\vec{x})} \ A(\vec{z}) : *\text{-sentence},}{\Gamma_\varphi : \text{context}}$$

where φ is $Ch_{\vec{y}:\vec{Y}(\vec{x})} \ A(\vec{z})$.

Then

$$\Gamma_\varphi = \check{\Gamma} \cup \mathbb{T}(Ch_{\vec{y}:\vec{Y}(\vec{x})} \ A(\vec{z})).$$

Having the dependence diagram $\| - \|^\Gamma : DG_\Gamma \to Set$ we shall define the dependence diagram $\| - \| = \| - \|^{\Gamma_\varphi} : DG_{\Gamma_\varphi} \to Set$.

Thus, for $\Phi \in Pack_{Ch^*}$ we need to define $\|\mathbb{T}_{\Phi,\varphi}\|^{\Gamma_\varphi}$ and for $\Phi' <_{Ch^*} \Phi$ we need to define

$$\|\pi_{\mathbb{T}_{\Phi,\varphi},t_{\Phi'}}\| : \|\mathbb{T}_{\Phi,\varphi}\| \longrightarrow \|\mathbb{T}_{\Phi',\varphi}\|$$

This will be done in two steps:

Step 1. (Fibers of new types defined by inverse induction.)

We shall define for the sub-prechains Ch' of Ch^* and $\vec{a} \in \|Env_\varphi(Ch')\|$ a set

$$\|\mathbb{T}_{\varphi,Ch'}\|(\vec{a}) \subseteq \|Bv(Ch')\|(\vec{a})$$

This is done using the inductive clauses through which we have defined Ch^* but in the reverse direction.

The *basic case* is when $Ch' = Ch^*$. We put

$$\|\mathbb{T}_{\varphi,Ch}\|(\emptyset) = \|P\|$$

The *inductive step.* Now assume that the set $\|\mathbb{T}_{\varphi,Ch'}\|(\vec{a})$ is defined for $\vec{a} \in \|Env_\varphi(Ch')\|$.

Parallel decomposition. If we have

$$Ch' = \frac{Ch_{1\vec{y}_1:\vec{Y}_1(\vec{x}_1)}}{Ch_{2\vec{y}_2:\vec{Y}_2(\vec{x}_2)}}$$

then we define sets
$$\|\mathbb{T}_{\varphi,Ch_i}\|(\vec{a}) \in \|Ch_i\|(\vec{a})$$
for $i = 1, 2$ so that
$$\|\mathbb{T}_{\varphi,Ch'}\|(\vec{a}) = \|\mathbb{T}_{\varphi,Ch_1}\|(\vec{a}) \times \|\mathbb{T}_{\varphi,Ch_2}\|(\vec{a})$$
if such sets exist, and these sets ($\|\mathbb{T}_{\varphi,Ch_i}\|(\vec{a})$) are undefined[5] otherwise.

Sequential decomposition. If we have
$$Ch' = Ch_{1\,\vec{y}_1:\vec{Y}_1(\vec{x}_1)}|Ch_{2\,\vec{y}_2:\vec{Y}_2(\vec{x}_2)}$$
then we put

$$\|\mathbb{T}_{\varphi,Ch_1}\|(\vec{a}) = \{\vec{b} \in \|Bv(Ch_1)\|(\vec{a}) : \{\vec{c} \in \|Bv(Ch_2)\|(\vec{a},\vec{b}) : \langle \vec{b}, \vec{c} \rangle \in \|\mathbb{T}_{\varphi,Ch'}\|(\vec{a})\} \in \|Ch_2\|(\vec{a},\vec{b})\}$$

For $\vec{b} \in \|Bv(Ch_1)\|$ we put

$$\|\mathbb{T}_{\varphi,Ch_2}\|(\vec{a},\vec{b}) = \{\vec{c} \in \|Bv(Ch_2)\|(\vec{a},\vec{b}) : \langle \vec{b}, \vec{c} \rangle \in \|\mathbb{T}_{\varphi,Ch'}\|(\vec{a})\}$$

Step 2. (Building dependent types from fibers.)
If Φ is a pack in Ch^*, $\vec{a} \in \|Env_\varphi(\Phi)\|$ then we put

$$\|\mathbb{T}_{\varphi,\Phi}\| = \bigcup\{\{\vec{a}\} \times \|\mathbb{T}_{\phi,\Phi}\|(\vec{a}) : \vec{a} \in \|Env_\varphi(\Phi)\|, \ \forall_{\Phi' <_{Ch^*}\Phi}, \ (\vec{a}\lceil\mathbf{env}_\varphi(\Phi')) \in \|\mathbb{T}_{\varphi,\Phi'}\|\}$$

It remains to define the projections between dependent types. If $\Phi' <_\varphi \Phi$ we define

$$\pi_{\mathbb{T}_{\varphi,\Phi},t_{\varphi,\Phi'}} : \|\mathbb{T}_{\varphi,\Phi}\| \longrightarrow \|\mathbb{T}_{\varphi,\Phi'}\|$$

so that

$$\vec{a} \mapsto \vec{a}\lceil(\mathbf{env}_\varphi(\Phi') \cup \mathbf{bv}\Phi')$$

6 Conclusion

It was our intention in this paper to show that adopting a typed approach to generalized quantification allows a natural and elegant treatment of a wide array of anaphoric data involving natural language quantification. The main technical contribution of our paper consists in combining generalized quantifiers with dependent types. Empirically, our system allows a uniform account of both regular anaphora to quantifiers and the notoriously difficult cases such as quantificational subordination, cumulative and branching continuations, and 'donkey anaphora'. Our treatment of unbound anaphora does not run into the 'proportion problem' and easily accommodates a whole range of ambiguities claimed for 'donkey sentences'.

---

[5]Such sets might not be determined uniquely if one of them is empty.

References

[Barwise & Cooper 1981] Barwise, John & Robin Cooper (1981), Generalized Quantifiers and Natural Language. *Linguistics & Philosophy* 4: 159-219.

[Van Benthem 1989] Van Benthem, Johan (1989), Polyadic quantifiers. *Linguistics & Philosophy* 12: 437-464.

[Bellert & Zawadowski 1989] Bellert, Irena & Marek Zawadowski (1989), Formalization of the feature system in terms of pre-orders. In Irena Bellert *Feature System for Quantification Structures in Natural Language*. Foris Dordrecht.

[Brasoveanu 2008] Brasoveanu, Adrian (2008), Donkey Pluralities: Plural Information States Versus Non-Atomic Individuals. *Linguistics & Philosophy* 31 (2): 129-209.

[Brasoveanu 2011] Brasoveanu, Adrian & Donka F. Farkas (2011), How indefinites choose their scope. *Linguistics & Philosophy* 34: 1-55

[Brennan 2012] Brennan, Jonathan (2012), Indefinites, Choice Functions, and Discourse Anaphora. In S. Lima, K. Mullin, & B. Smith (eds.), *Proceedings of NELS 39*, Amherst, MA: GLSA, 151-163.

[Chierchia 1992] Chierchia, Gennaro (1992), Anaphora and dynamic binding. *Linguistics & Philosophy* 15(2): 111-83.

[Dekker 1994] Dekker, Paul (1994), Predicate logic with anaphora. In Lynn Santelmann and Mandy Harvey (eds.), *Proceedings SALT IX*. Ithaca, NY: DMLL Publications, Cornell University. 79-95.

[Dekker 2008a] Dekker, Paul (2008), A multi-dimensional treatment of quantification in extraordinary English. *Linguistics & Philosophy* (1): 101-127.

[Dekker 2008b] Dekker, Paul (2008), A guide to dynamic semantics. ILLC Prepublications PP-2008-42, University of Amsterdam.

[Elworthy 1995] Elworthy, David A. H. (1995), A theory of anaphoric information. *Linguistics & Philosophy* 18: 297-332.

[Egli & von Heusinger 1995] Egli, Urs & Klaus von Heusinger (1995), The Epsilon Operator and E-Type Pronouns. In Urs Egli, Peter E. Pause, Christoph Schwarze, Arnim von Stechow, & G¨tz Wienold (eds), *Lexical Knowledge in the Organization of Language*, 114 in Current Issues in Linguistic Theory. Benjamins, Amsterdam. 121-141.

[Elbourne 2005] Elbourne, Paul D. (2005), *Situations and Individuals*, Cambridge, MA: MIT Press.

[Evans 1997] Evans, Gareth (1977), Pronouns, Quantifiers, and Relative Clauses (I), *Canadian Journal of Philosophy* 7: 467-536.

[Heim 1990] Heim, Irene (1990), E-type pronouns and donkey anaphora. *Linguistics & Philosophy* 13(2):137-78.

[Henkin 1959] Henkin, Leon (1959), Some Remarks on Infinitely Long Formulas, *Proceedings of the Symposium on Foundations of Mathematics*, Warsaw.

[Geurts & van der Sandt 1991] Geurts, Bart & Rob van der Sandt (1999), Domain Restriction. In P. Bosch & R. A. van der Sandt (eds.), *Focus: Linguistic, Cognitive, and Computational Perspectives*, Cambridge UP, Cambridge. 268-292.

[Groenendijk & Stokhof 1991] Groenendijk, Jeroen & Martin Stokhof (1991), Dynamic Predicate Logic. *Linguistics & Philosophy* 14:39-100.

[Kadmon 1987] Kadmon, Nirit (1987), On unique and Non-unique reference and Asymmetric Quantification. Ph.D. diss, University of Massachusetts, Amherst.

[Kamp 1981] Kamp, Hans (1981), A theory of truth and semantic representation. In J. Groenendijk, T. Janssen & M. Stokhof (eds.), *Truth, Interpretation and Information*, Foris Dordrecht. 1-41.

[Kamp & Reyle 1993] Kamp, Hans & Uwe Reyle (1993), *From Discourse to Logic*. Kluwer Academic Publishers, Dordrecht.

[Keenan 1987] Keenan, Edward, L. (1987), Unreducible n-ary quantifiers in natural language. In P. Gärdenfors (ed.), *Generalized Quantifiers: Linguistic and Logical Approaches*. Reidel Dordrecht. 109-150.

[Keenan 1992] Keenan Edward, L. (1992), Beyond the Frege Boundary. *Linguistics & Philosophy* 15: 199-221.

[Keenan 1993] Keenan Edward, L. (1993), Natural language, sortal reducibility and generalized quantifiers. *Journal of Symbolic Logic* 58:314-325.

[Krifka 1996] Krifka, Manfred (1996), Parametrized sum individuals for plural reference and partitive quantification. *Linguistics & Philosophy* 19: 555-598.

[Lindström 1966] Lindström, Per (1966), First-order predicate logic with generalized quantifiers, *Theoria* 32: 186-95.

[Luo 2010] Luo, Zhaohui (2010), Type-Theoretical Semantics with Coercive Subtyping. *Semantics and Linguistic Theory*, Vol. 20 (SALT20), 38-56.

[Luo 2012a] Luo, Zhaohui (2012), Formal Semantics in Modern Type Theories with Coercive Subtyping. *Linguistics & Philosophy* 35, 491-513.

[Luo 2012b] Luo, Zhaohui (2012), Common nouns as types. *LACL'12*, *LNCS* 7351, 173-185.

[Martin-Löf 1972] Martin-Löf, Per (1972), *An intuitionstic theory of types*, Technical Report, University of Stockholm.

[Martin-Löf 1984] Martin-Löf, Per (1984), *Intuitionistic Type Theory*, Bibliopolis.

[Montague 1974] Montague, Richard (1974), *Formal Philosophy*. Yale University Press.

[Mostowski 1957] Mostowski, Andrzej (1957), On a generalization of quantifiers, *Fundamenta Mathematicae* 44: 12-36.

[Neale 1990] Neale, Stephen (1990), *Descriptions*. MIT Press Books, Cambridge.

[Nordström, Petterson & Smith 1990] Nordström, Bengt, Kent Petterson, Jan M. Smith (1990), *Programming in Martin-Löf's Type Theory. An Introduction*, Oxford University Press, Oxford.

[Nouwen 2003] Nouwen, Rick (2003), Plural pronominal anaphora in context: dynamic aspects of quantification, Ph.D. thesis, UiL-OTS, Utrecht, LOT dissertation series, No. 84.

[Stanley & Westerståhl 2006] Peters, Stanley & Dag Westerståhl (2006), *Quantifiers in Language and Logic*. Oxford University Press, Oxford.

[Ranta 1994] Ranta, Aarne (1994), *Type-Theoretical Grammar*, Oxford University Press, Oxford.

[Schlenker 2006] Schlenker, Philippe (2006), Scopal Independence: A Note on Branching & Wide Scope Readings of Indefinites & Disjunctions. Journal of Semantics 23(3): 281-314.

[Sundholm 1989] Sundholm, Göran (1989), Constructive generalized quantifiers. *Synthese* 79: 1-12.

[Szabolcsi 2010] Szabolcsi, Anna (2010), *Quantification*. Cambridge University Press, Cambridge.

[Steedman 2007] Steedman, Mark (2007), Surface-compositional scope-alternation without existential quantifiers. Ms, University of Edinburgh.

[Van den Berg 1996] Van den Berg, Martin H. (1996), The Internal Structure of Discourse, Ph.D. thesis, Universiteit van Amsterdam, Amsterdam.

[Van der Does 1996] Van der Does, Jaap (1996), Quantification and Nominal Anaphora. In: K. von Heusinger & U. Egli (eds.). P*roceedings of the Konstanz Workshop Reference and Anaphoric Relations*. Arbeitspapier 79. 27-56.

[Westerståhl 1994] Westerståhl, Dag (1994), Iterated quantifiers. In M. Kanazawa M & Ch. Pinon (eds.), *Dynamics, Polarity, and Quantification*. 173-209.