

Indefinite donkeys on islands *

Edgar Onea
University of Göttingen

Abstract In this paper I present a theory of indefinites which captures two of their natural properties: indefinites license donkey anaphora (Geach 1962) and they exhibit ‘specific’ readings in which they appear to scope out of scope islands. In various flavours of dynamic semantics (Kamp 1981, Heim 1982, Dekker 2004), these properties can be captured to the detriment of compositionality. Other theories have employed more involved technical machinery like choice functions (Kratzer 1998, von Heusinger 2002), Hamblin-semantics (Kratzer & Shimoyama 2002) or independence friendly logics (Brasoveanu & Farkas 2011) to derive exceptional scope readings, but ignored donkey anaphora. Theories of E-type anaphora, on the other hand, generally do not consider exceptional scope readings (Heim 1990, Elbourne 2001). My own analysis combines insights from dynamic semantics with referential indexing in LF-semantics, resulting in a fully compositional, static system.

Keywords: Indefinite scope, DRT, scope islands, quantification, donkey anaphora

1 Introduction

It is a well known fact that a) indefinites can scope out of islands and that b) indefinites license donkey anaphora. a) is illustrated by the contrast in (1).¹ One can easily see that *some*, located in a scope island, allows two scope configurations which *every* lacks. b) is illustrated by (2), in which the indefinite *a donkey* is surprisingly translated as a universal quantifier instead of an existential quantifier in order to bind the object-pronoun *it*. As opposed to this, a more natural translation as an existential quantifier ends up having too narrow scope to bind the pronoun, as in (2b).

* This research has been supported by the German Initiative of Excellence funded by the German Science Foundation at the University of Göttingen as part of the Courant Research Centre “Text structures,” which I thankfully acknowledge. I am grateful to Regine Eckardt for helpful comments on an earlier version of this paper. I would also like to thank the participants of SALT 23 for insightful comments of my poster. All errors are my own.

¹ In this paper I can only give a very cursory overview of the phenomena and the literature. An extensive and comprehensive overview can be found in Bende-Farkas & Kamp 2001, Endriss 2009 and von Heusinger 2011.

- (1) a. Exactly five boys read most books that were recommended by some professor.
five > most > some | *five > some > most* | *some > five > most*
- b. Exactly five boys read most books that were recommended by every professor.
five > most > every | **five > every > most* | **every > five > most*
- (2) Every farmer who owns a donkey beats it.
- a. $\forall xy. \mathbf{F}(x) \wedge \mathbf{D}(y) \wedge \mathbf{O}(x, y) \rightarrow \mathbf{B}(x, y)$
- b. $\forall x. \mathbf{F}(x) \wedge (\exists y. \mathbf{D}(y) \wedge \mathbf{O}(x, y)) \rightarrow \mathbf{B}(x, y)$

The scope issue is most easily explained as follows: a scope island is a local environment from which quantifiers cannot escape. Bona fide quantifiers (like *most*, *every* or *less than half*) never escape scope islands. The problem of indefinites is that they can either out-scope islands or not. There seems to be a certain general unwillingness to assume that scope islands simply do not care about indefinites. One could, of course, postulate that indefinites can be interpreted in any environment, which would trivialize the puzzle, as argued in Heim 2011, however, a more natural explanation why they ignore islands seems preferable.

One way to put the the donkey problem is that in cases in which an indefinite is interpreted in a local environment, it still may bind a pronoun outside that environment. However, as can be seen in (2a), if we were to assume that indefinites can be universal quantifiers at times, the donkey readings would no longer be a problem, since then, actually, the indefinite is interpreted higher, outside its local environment, and can bind the desired pronouns without problems. But translating indefinites as universal quantifiers seems to really be a desperate last resort decision.

If we were to assume that (2a) is the kind of representation semantic theory ought to derive, this nurtures the impression that we can extract an indefinite from a scope island either as an existential or as a universal quantifier: the ‘narrow’ scope reading appears as a universal wide scope reading, as suggested in (3). And if this were not enough of an embarrassment, it is obvious that there must be heavy and quite opaque constraints on when an indefinite may become a universal quantifier and when it may not. Heim (1982) discusses at length some proposals that capture the respective properties of indefinites based on such conversions, e.g. those proposed by Egli (1979), and shows a whole range of problems.

- (3) Every farmer who owns a donkey beats it.
- a. wide scope: $\exists y. \mathbf{D}(y) \wedge \forall x. \mathbf{F}(x) \wedge \mathbf{O}(x, y) \rightarrow \mathbf{B}(x, y)$
- b. ‘narrow scope’ $\forall y. \mathbf{D}(y) \rightarrow \forall x. \mathbf{F}(x) \wedge \mathbf{O}(x, y) \rightarrow \mathbf{B}(x, y)$

This kind of apparent quantificational variability motivated the claim in [Kamp 1981](#) and [Heim 1982](#) that indefinites are not actual quantifiers but variables/discourse referents. Such variables get ultimately bound either by some existential or some universal quantifier but, crucially, they do not introduce the quantifiers on their own. The idea is sketched in (4). Notice that in (4) quantification goes over sets of variables, and the variable of the indefinite may be captured or not by the universal quantifier. Moreover, the widest scope existential quantifier is assumed to be a default operator. However, also notice that it is crucial to have the restrictor $\mathbf{D}(y)$ outside the island, or else a version of the famous Donald Duck problem arises ([Reinhart 1992](#)), a reading made true by the existence of Donald Duck, mostly assumed to be a non-donkey. Hence, while the donkey readings are fairly naturally captured by the assumption that indefinites do not come with an own quantificational force, their scope island ignoring properties do still seem to pose a problem.

- (4) Every farmer who owns a donkey beats it.
- a. ‘wide scope’: $\exists\{y\}.\mathbf{D}(y) \wedge \forall\{x\}.\mathbf{F}(x) \wedge \mathbf{O}(x, y) \rightarrow \mathbf{B}(x, y)$
 - b. ‘narrow scope’ $\exists\emptyset.\forall\{x, y\}.\mathbf{D}(y) \wedge \mathbf{F}(x) \wedge \mathbf{O}(x, y) \rightarrow \mathbf{B}(x, y)$
 - c. Donald Duck reading: $\exists\{y\}.\forall\{x\}.\mathbf{D}(y) \wedge \mathbf{F}(x) \wedge \mathbf{O}(x, y) \rightarrow \mathbf{B}(x, y)$

In this paper, I will elaborate on the idea that indefinites do not have a quantificational force on their own. Instead, they act as variables which receive a default existential closure. In this theory, however, indefinites are always fully interpreted in situ. As a result of their interpretation, they impose a referential constraint on some local referential environment. Crucially, however, referential constraints can but need not be inherited to more global referential contexts in the process of composition. Apparent wide scope then arises from an interaction between quantifiers and referential contexts in their scope.

In a sense, the interpretation of indefinites that I propose is similar to the buoyancy principle postulated in [Geurts 2002](#) for DRT, according to which specific indefinites tend to float towards more global DRSSs, hence violating scope islands. As opposed to Geurts’ system, however, in the present theory, indefinites do not actually violate scope islands and the computation is driven by simple lambda-calculus (functional application and, for convenience, predicate modification). Moreover, the theory that I propose is purely static and does not make any assumptions about the role of context in the process of interpretation.² Finally, since the syntactic tree fully determines the outcome, the system is compositional in a particularly strong sense.

² This is not to suggest that there is anything wrong with dynamic semantics. However, intuitively, there is nothing in indefinites which requires a dynamic theory of meaning. Put differently, it is reasonable to attempt to understand indefiniteness in a static framework regardless as to whether, ultimately, a dynamic framework is chosen or not.

It is not the case that there is no other, competing static and fully compositional theory that is (at least in principle) able to handle the two phenomena we are interested in. For instance, [Brasoveanu & Farkas 2011](#) propose an analysis of indefinites that is similar in many ways, even though the underlying intuition is fundamentally different. While for them indefinites are essentially existential quantifiers, and existential quantifiers have specific ways of scope-taking in the general framework of independence friendly logic, in the theory proposed here, indefinites, just like other referential expressions, do not have an intrinsic existential import.

In section 2, I introduce the main assumptions of the compositional system. In section 3, I introduce quantification and indefinites. In sections 4 and 5, I show how both donkey sentences and apparent scope island violation are handled in this theory. I end with some general considerations about where the proposed theory should lead to.

2 The basics

In this section, I introduce the basic compositional system and explain how it relates to well known standard compositional systems. The aim of this section is to be able to derive the truth conditions of some simple sentences such as those in (5).

- (5) a. Peter loves Mary.
b. Peter loves himself.

It is fairly standard to consider an interpretation function for natural language which is dependent on at least an assignment function and a world of evaluation, noted as $\| \cdot \|^{w,g}$. It is also customary to assume that pronouns receive their referential value from the assignment function, as in (6a), as argued in [Heim & Kratzer 1998](#). For instance [Büring 2005](#) assumes that while proper names do not essentially depend on assignment functions, they impose a constraint that the assignment function assigns precisely their semantic value to their index, whence the representation in (6b).

- (6) a. $\|he_i\|^{w,g} = gi$
b. $\|Peter_i\|^{w,g} = \mathbf{P}$, defined iff $gi = \mathbf{P}$

As a parameter of evaluation, we can think of the assignment function as a global variable which stands for something I would like to call *referential context*. As with the other parameters of evaluation, such as the world variable or a Kaplanian context variable c , it is assumed that the rules of semantic composition do not generally manipulate the assignment parameter. A well known exception are different versions of predicate abstraction ([Heim & Kratzer 1998](#)). Similar exceptions are also known for other parameters of evaluation: the world parameter is actively manipulated in the

semantic composition by intensional expressions (von Fintel & Heim 2011) and the context parameter in special ‘monstrous’ contexts (von Stechow & Zimmermann 2005).

In this paper, the main innovation that will lead to a natural solution of the indefinite problem is that referential contexts should not only be treated as a parameter of evaluation in the semantic composition but rather as a local context that gets incrementally specified in the process of computation. This has two aspects. The first one is that we will get rid of the evaluation parameter g and instead assume that the interpretation function will always output functions from assignments. The second one is that semantic values of natural language expressions are specifically interacting with referential contexts.

It is also possible to use a lambda notation for assignments without any substantive change. The rules in (7) and (8) will create isomorphic interpretations, as it is easy to check, see also Poesio 1996. Thereby I use $\lambda\alpha$ [condition] β to write a partial function from alphas that satisfy the condition to betas.³ Since A and B are equivalent, one may generally prefer A as it leads to simpler expressions.

(7) Type A theories: g as a parameter of evaluation

- a. Lexical entries A

$$\|\text{Peter}_i\|^{w,g} = \mathbf{P} \text{ iff } gi = \mathbf{P} \qquad \|\text{sleeps}\|^{w,g} = \lambda x_e. \mathbf{S}(x)$$
- b. Functional application A

$$\|\widehat{\alpha \beta}\|^{w,g} = \|\alpha\|^{w,g} \|\beta\|^{w,g}, \text{ iff } \|\alpha\|^{w,g} \text{ and } \|\beta\|^{w,g} \text{ are defined}$$
- c. Predicate abstraction A

$$\|\widehat{\alpha \ i \ \beta}\|^{w,g} = \|\alpha\|^{w,g} \lambda x_e. \|\beta\|^{w,g_i/x}$$

(8) Type B theories: g as part of semantic values

- a. Lexical entries B

$$\|\text{Peter}_i\|^w = \lambda g_a [gi = \mathbf{P}] . gi \qquad \|\text{sleeps}\|^w = \lambda g_a. \lambda x_e. \mathbf{S}(x)$$
- b. Functional application B

$$\|\widehat{\alpha \beta}\|^w = \lambda g_a [g \in \mathcal{D} \ \|\alpha\|^w \text{ and } g \in \mathcal{D} \ \|\alpha\|^w] . (\|\alpha\|^w g) (\|\beta\|^w g)$$
- c. Predicate abstraction B

$$\|\widehat{\alpha \ i \ \beta}\|^w = \lambda g_a. (\|\alpha\|^w g) (\lambda x_e. \|\beta\|^w g_{i/x})$$

If we assume that semantic values of natural language expressions specifically interact with referential contexts, we will need a different system. In particular, we will use standard functional application, as in (7b), but we still use a special

³ The standard type system (e, s, t) is enriched by a , the type for assignments, which stands for functions from numbers to individuals. Types are written in subscript.

lambda notation for assignments in the lexical entries.⁴ Not all semantic values are functions from assignments. While referential expressions are partial functions from assignments, properties and transitive verbs are not. This is shown for a simple set of examples in (9).

(9) Lexical entries C:

a. Proper Names:

$$\|\text{Peter}_i\|^w = \lambda g_a [gi = \mathbf{P}] .gi$$

b. Pronouns and traces:

$$\|\text{he}_i\|^w = \lambda g_a .gi \quad \|\text{t}_i\|^w = \lambda g_a .gi \quad \|\text{himself}_i\|^w = \lambda g_a .gi$$

c. Properties and transitive verbs:

$$\|\text{sleep}\|^w = \lambda x_{\langle a,e \rangle} .\lambda g_a [g \in \mathcal{D}x] .\mathbf{S}(xg, w)$$

$$\|\text{student}\|^w = \lambda x_{\langle a,e \rangle} .\lambda g_a [g \in \mathcal{D}x] .\mathbf{S}(xg, w)$$

$$\|\text{kiss}\|^w = \lambda xy_{\langle a,e \rangle} .\lambda g_a [g \in \mathcal{D}x \wedge g \in \mathcal{D}y] .\mathbf{K}(yg, xg, w)$$

As in standard systems, referential expressions impose referential constraints on referential contexts, but properties and transitive verbs are different in the sense that they do not impose own constraints on referential contexts but provide a mechanism of inheritance. So, while in B properties are totally neutral with regard to referential contexts, in C properties have an inherited non-neutrality: they are designed as devices of storing referential constraints. Put differently, the semantic composition gathers all referential constraints encountered in a sentence as a definedness condition on the output, which is a function from assignments to truth values. This is exemplified in the example derivation below:⁵

$$(10) \quad \left\| \begin{array}{ccc} & \text{Peter}_1 & \\ & / \quad \backslash & \\ & \text{loves} & \text{Mary}_2 \end{array} \right\|^w = (\|\text{loves}\|^w \|\text{Mary}_2\|^w) \|\text{Peter}_1\|^w =$$

$$(\lambda xy_{\langle a,e \rangle} .\lambda g_a [g \in \mathcal{D}x, y] .\mathbf{L}(yg, xg, w) \lambda h_a [h2 = \mathbf{M}] .h2) \|\text{Peter}_1\|^w =$$

$$\lambda y_{\langle a,e \rangle} .\lambda g_a [g2 = \mathbf{M} \wedge g \in \mathcal{D}y] .\mathbf{L}(yg, g2, w) \lambda h_a [h1 = \mathbf{P}] .h1 =$$

$$\lambda g_a [g2 = \mathbf{M} \wedge g1 = \mathbf{P}] .\mathbf{L}(g1, g2, w)$$

For the result in (9), truth can either be a result of applying some contextually salient assignment, say g_{173} , which satisfies the constraints. Or, there can be a more general operator that turns functions from assignments into truth values. For explicitness, I assume that the syntactic operator ASSERT, generally assumed to occupy ForceP (Rizzi 1997), just like other speech act operators, has the function of matching

⁴ No predicate abstraction will be necessary.

⁵ The derivation does not proceed entirely step by step, but the omitted steps should be obvious enough to reconstruct.

referential restrictions on possible worlds and predications. The ASSERT operator is given in (11). It is easy to see that, applied to (9), the result is the standardly expected proposition given in (12):

$$\begin{aligned}
 (11) \quad & \left\| \text{ASSERT } \alpha \right\|^w = \exists g.g \in \mathcal{D} \left\| \alpha \right\|^w \wedge \left\| \alpha \right\|^w g \\
 (12) \quad & \left\| \begin{array}{c} \text{ASSERT} \\ \text{Peter}_1 \text{ loves } \text{Mary}_2 \end{array} \right\|^w = \\
 & = \exists g.g2 = \mathbf{M} \wedge g1 = \mathbf{P} \wedge \mathbf{L}(g1, g2, w) \quad \rightsquigarrow \quad \mathbf{L}(\mathbf{P}, \mathbf{M}, w)
 \end{aligned}$$

The system is naturally suitable to handle pronominal binding, determined in the syntax by co-indexing. This is exemplified in (13):

$$(13) \quad \left\| \begin{array}{c} \text{Peter}_1 \text{ loves } \text{himself}_1 \end{array} \right\|^w = \lambda g_a [g1 = \mathbf{P}] . \mathbf{L}(g1, g1, w)$$

Before concluding this section, a technical note is in order. While I will leave this exercise implicit, when defining the metalanguage care must be taken of the evaluation of logical connectives such that formulae intended to be true or false do not come out as undefined. For instance, $\forall g.g \in \mathcal{D} x \rightarrow xg$ could lead to the evaluation of xg for g 's which are not in the domain of x . But of course this is not intended: the formula should only evaluate xg for g 's which make the first part of the implication true. This can, of course, be easily implemented.

What is the advantage of the compositional system sketched above as compared to more standard ones? The main difference is that while in standard models referential expressions only impose constraints on global assignments, in this model, an assignment, as a referential context, is essentially local. The global effect arises due to inheritance. Now, if we want to consider the 'scope' of indefinites, we will need to impose a system of controls for inheritance. Put differently: an indefinite will always impose a local referential constraint, however, there will be a mechanism of control which may prevent referential constraints imposed by indefinites from becoming global.

3 Indefinites and quantification

With the basic tenets of the theory developed in the previous section, we now discuss the three components needed to see how indefinites behave in this system. First, I show how indefinite NPs can be analysed. I then introduce relative clauses as an

example of a scope island and finally, I introduce the universal quantifier.

3.1 Definites and indefinites

Following [Kamp \(1981\)](#) and [Heim \(1982\)](#), I assume that both indefinite and definite DPs are non-quantificational. Instead, they impose a referential constraint. As opposed to Heim and Kamp, however, I will opt for a uniqueness based theory of definiteness, for practical reasons. Because my theory is static, familiarity is more difficult to express. In this paper, I will not be concerned with an elaborate theory of definiteness, hence, I will use a very simple Russelian ([Russel 1905](#)) theory of uniqueness. In addition, I will assume later a rule of indexing which makes sure that definite descriptions always get maximal scope, modulo the Binder Roof Constraint ([Brasoveanu & Farkas 2011](#)).

With this background, the denotation of the indefinite article is given in (14). The first argument is the standard denotation of an NP, i.e., a property, a function from the semantic value of some referential expression (type $\langle a, e \rangle$) to a function from assignments to truth values. The most important part is the definedness condition on the embedded function, i.e., $[P(\lambda h. hi)g]$. The first argument of P here is the referential expression $\lambda h. hi$ which is simply the semantic value of a coindexed pronoun.

$$(14) \quad \|\text{some}_i\|^w = \lambda P_{\langle \langle a, e \rangle, \langle a, t \rangle \rangle} . \lambda g_a . [g \in \mathfrak{D}(P(\lambda h. hi)) \wedge P(\lambda h. hi)g] . gi$$

It is easy to see that $P(\lambda h. hi)g$ is a stronger condition than $g \in \mathfrak{D}(P(\lambda h. hi))$. In fact, we only need the latter to make sure that in the evaluation of the formula no unwanted breakdown arises due to undefinedness of some expression. For simplicity, we therefore omit that condition, hence simplifying (14) to (15), keeping in mind that the official version should remain (14):

$$(15) \quad \|\text{some}_i\|^w = \lambda P_{\langle \langle a, e \rangle, \langle a, t \rangle \rangle} . \lambda g_a . [P(\lambda h. hi)g] . gi$$

An example for a simple sentence is given in (16). It is worth noting that in (16a) the derivation of *some*₁ *man* yields a function from assignments to individuals which is only defined for assignment functions which assign to the referential index 1 a man.

$$(16) \quad \|\text{Some}_1 \text{ man loves some}_2 \text{ woman}\|^w$$

- a. $\left\| \begin{array}{c} \diagup \quad \diagdown \\ \text{some}_1 \quad \text{man} \end{array} \right\|^w = \lambda g_a . [\mathbf{M}(g1, w)] . g1$
- b. $\left\| \begin{array}{c} \diagup \quad \diagdown \\ \text{some}_2 \quad \text{woman} \end{array} \right\|^w = \lambda g_a . [\mathbf{W}(g2, w)] . g2$

$$\begin{array}{c}
\text{c.} \quad \left\| \begin{array}{c} \text{some}_1 \quad \text{man} \\ \text{loves} \\ \text{some}_2 \quad \text{woman} \end{array} \right\|^w = \\
\lambda g_a [\mathbf{W}(g2, w) \wedge \mathbf{M}(g1, w)] . \mathbf{L}(g1, g2, w)
\end{array}$$

Let us consider the denotation of the definite article. There is a slight technical difficulty in spelling out the standard Russelian uniqueness condition, $\lambda P. \lambda Q. \exists x. Px \wedge (\forall y. Py \rightarrow x = y) \wedge Qx$, in the present system. The problem is that in (15) we have applied the pronoun $\lambda h. hi$ to the property P , but as opposed to standard variables, the uniqueness of pronouns would be an absurd requirement. This leads to a constraint which is a bit longer and basically says the same. The official formula is given in (17), but we will use a simplification given in (18):

$$(17) \quad \|\text{the}_i\|^w = \lambda P. \lambda g_a \left[\begin{array}{c} g \in \mathcal{D}(P(\lambda h. hi)) \wedge P(\lambda h. hi)g \wedge \\ \forall u. g_{[i/u]} \in \mathcal{D}(P(\lambda h. hi)) \wedge P(\lambda h. hi)g_{[i/u]} \rightarrow u = gi \end{array} \right] . gi$$

$$(18) \quad \|\text{the}_i\|^w = \lambda P_{\langle \langle a, e \rangle, \langle a, t \rangle \rangle} . \lambda g_a [P(\lambda h. hi)g \wedge \forall u. P(\lambda h. hi)g_{[i/u]} \rightarrow u = gi] . gi$$

One of the reasons why the condition in (18) involves the modification of an assignment instead of simply a new universal quantification over assignments is that a definite description may contain bound pronouns, as in (19). In such a case, a new quantification over assignments would interrupt the respective dependencies, hence imposing too strong conditions. For (19) that would incorrectly predict that every person who has a mother has the same mother. Such a prediction is avoided by (18).

$$\begin{array}{c}
(19) \quad \|\text{Some}_1 \text{ man loves (the)}_2 \text{ his}_1 \text{ mother}\|^v \\
\text{a.} \quad \left\| \begin{array}{c} \text{the}_2 \\ \text{his}_1 \quad \text{mother} \end{array} \right\|^v = \\
\lambda g [\mathbf{Mo}(g2, g1, v) \wedge \forall u. \mathbf{Mo}(u, g1, v) \rightarrow u = g2] . gi \\
\text{b.} \quad \left\| \begin{array}{c} \text{some}_1 \quad \text{man} \\ \text{loves} \\ \text{the}_2 \quad \text{his}_1 \quad \text{mother} \end{array} \right\|^v = \\
\lambda g [\mathbf{M}(g1, v) \wedge \mathbf{Mo}(g2, g1, v) \wedge \forall u. \mathbf{Mo}(u, g1, v) \rightarrow u = g2] . \mathbf{L}(g1, g2, v)
\end{array}$$

It is worth noting that in the examples above we have only reproduced derivations that can be done in the standard system as well, yielding exactly the same results. It is only once we introduce relative clauses and quantification that the essential

differences due to inheritance based composition become obvious.

3.2 Relative clauses

The syntax of relative clauses that I assume is simplistic but widely standard, as shown in (20): the relative pronoun is co-indexed with a trace or pronoun.



In a standard system of type A, a special rule is needed to create a property out of the denotation of the clause: $\mathbf{L}(g1, \mathbf{J}, w) \rightsquigarrow \lambda x. \mathbf{L}(x, \mathbf{J}, w)$. This is because, in order to turn $g1$ into a variable, *who* would need to be able to directly manipulate the global parameter of evaluation, which is not granted by functional application in the A system. In a system of type B, as well, a special rule is necessary, even though *who* would be able to manipulate an assignment directly. The problem is that the rule of functional application in B lowers the type of the sister of *who*, hence in fact sabotaging the (in principle) higher expressive power.

As opposed to this, in the system of type C proposed here, we have the higher types of system B but also the harmless rule for functional application of system A. This means that we need no special rule for relative pronouns, as we can create a property out of a clause with a pronoun by a simple lexical entry, as shown in (21). While this is not of particular relevance for the purposes of this paper, it is worth noting that from the point of view of parsimony, this is an advantage of system C.

$$(21) \quad \|\text{who}_i\|^w = \|\text{such that}_i\|^w = \lambda H_{\langle a, t \rangle} . \lambda x . \lambda g [g \in \mathcal{D}H \wedge g \in \mathcal{D}x \wedge gi = xg] . Hg$$

In (21), the relative pronoun does not actually interfere with H itself. By opening a slot for a referential expression, it only has to make sure that the newly to be introduced referential constraint will be added to the ones priorly available. Moreover, it will require that the new referential expression is evaluated to the same individual as the index of the relative pronoun. This additional constraint could be omitted if we would postulate that relative pronouns are co-indexed with the determiner of the NP they are a part of. But it seems more straightforward not to request such an indexing rule. The relative clause in (20) is computed in (22).

$$(22) \quad \left\| \begin{array}{c} \text{who}_1 \\ \text{t}_1 \\ \text{loves} \\ \text{Jane}_2 \end{array} \right\|^v = \lambda xg [g2 = \mathbf{J} \wedge g \in \mathcal{D}x \wedge g1 = xg] . \mathbf{L}(g1, g2, v)$$

In order to be able to combine the arising property with a noun like *man*, we may assume the predicate modification rule in (23),⁶ which yields the derivation in (24).

(23) Predicate modification C:
 If the type of $\|\alpha\|^w = \|\beta\|^w = \langle \langle a, e \rangle, \langle a, t \rangle \rangle$ then
 $\|\widehat{\alpha} \widehat{\beta}\|^w = \lambda x_{\langle a, e \rangle} . \lambda g_a [g \in \mathcal{D} \|\alpha\|^w x \wedge g \in \mathcal{D} \|\beta\|^w x] . \|\alpha\|^w xg \wedge \|\beta\|^w xg$

(24) $\left\| \begin{array}{c} \text{man} \\ \text{who}_1 \\ \text{t}_1 \\ \text{loves} \text{ Jane}_2 \end{array} \right\|^w = \lambda x_{\langle a, e \rangle} . \lambda g_a [g \in \mathcal{D} x \wedge g2 = \mathbf{J} \wedge g1 = xg] . \mathbf{M}(xg, w) \wedge \mathbf{L}(g1, g2, w)$

Notice that even though predicate modification is, as such, a symmetric rule, combining a relative clause with a noun yields an asymmetric representation, i.e., the referential constraints of the relative clause appear as inherited referential constraints and the predications have different arguments: the predication introduced by *man* has xg as referential argument, meanwhile *loves* has $g1$ as an argument. Since, however, this is only defined for assignments such that $xg = g1$, ultimately, the result will end up being symmetric. So, for instance:

(25) $\left\| \text{Some}_3 \text{ man who}_1 \text{ t}_1 \text{ loves Jane}_2 \text{ sleeps} \right\|^w = \lambda g_a [g2 = \mathbf{J} \wedge g1 = g3 \wedge \mathbf{M}(g3, w) \wedge \mathbf{L}(g1, g2, w)] . \mathbf{S}(g3, w)$

3.3 Quantification

Up to now, we have discussed only two types of expressions: referential expressions, which impose referential constraints, and properties (in the most general sense), which inherit referential constraints to more global referential contexts. Quantifiers are a different type of expression: they manipulate referential contexts directly, hence, establishing three different referential contexts: their scope, their restrictor and the output referential context, very similar to [Kamp 1981](#), [Heim 1982](#) and also [Farkas 1997, 2002](#). In order to be able to establish the relations between these contexts, we will use some special indexing. First, I will introduce some notations which will be used to model the permeability of referential contexts, then I introduce the specific indexing used in quantification and finally, I give some lexical entries and example derivations.

⁶ Of course, this rule is just for convenience. One can easily raise the type of relative clauses to combine with nouns more directly.

3.3.1 Permeability

We can define two notions to mediate between referential contexts: *exceptive equality* and *selective compatibility*. The notion of exceptive equality is a more general notion of equality. We say that h equals g except A , noted as $h=_{A}g$, as in (26), iff h and g are identical except for the values assigned by indices in the set A . Thereby A is a set of natural numbers. It is easy to show that $=$ is the special case $=_{\emptyset}$.

$$(26) \quad \text{Exceptive equality:} \\ h=_{A}g \text{ iff } \forall i. i \notin A \rightarrow hi = gi$$

The notion of selective compatibility is a more general version of compatibility. h is in the domain of H , $h \in \mathcal{D}H$ whenever there are no referential constraints on H which rule out h . Crucially, we may be interested in such assignment functions which would be in the domain of H if we were to disregard some of the referential constraints. $h \in_{A} \mathcal{D}H$ says that h is in the domain of H if we disregard those referential constraints which concern the indices in A .⁷ It is easy to see that $h \in \mathcal{D}H$ is the same as $h \in_{\emptyset} \mathcal{D}H$.

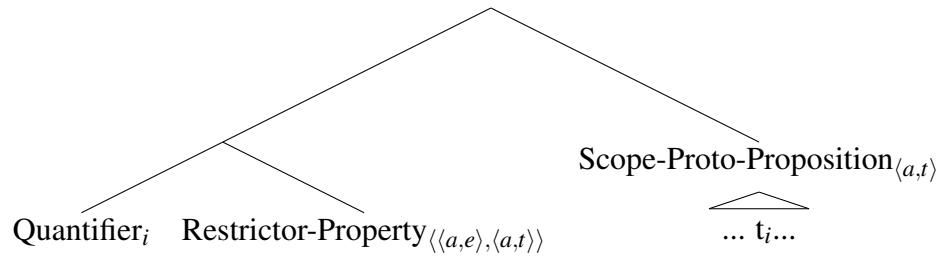
$$(27) \quad \text{Selective compatibility:} \\ h \in_{A} \mathcal{D}H \text{ iff } \exists g. g=_{A}h \wedge g \in \mathcal{D}H$$

3.3.2 Indexing and syntax

Given that the current theory has higher typed-semantic values and that referential contexts and inheritance are built into the semantics, the compositional system is powerful, hence, there are various ways in which one can go about quantification. It is possible to use in situ theories of quantification and it is possible to use quantifier-movement-based theories. I will propose a version of quantification which assumes that quantifiers always raise, that is to say, even in simple cases of subject quantifiers, movement will be assumed. We will generally assume that the scope argument of each bona fide quantifier is a proto-proposition, i.e., it has type $\langle a, t \rangle$, and that the quantifier will always have a trace within the scope. The restrictor of each quantifier remains a property, i.e., it has type $\langle \langle a, e \rangle, \langle a, t \rangle \rangle$, as shown in (28).

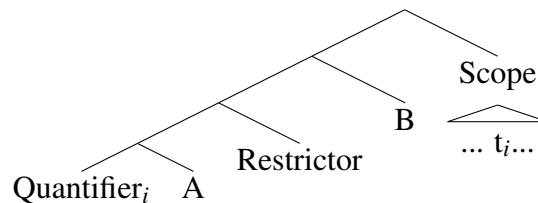
⁷ This particular formalisation of this notion tends to lead to non-emptiness entailments, i.e., if we disregard one referential constraint we implicitly assume that it could have been satisfied, but it is not yet clear whether this is actually desirable. If this turns out to be a real problem, introducing the individuals \top and \perp , such that \top makes every formula true and \perp makes every formula false, helps in avoiding unwanted entailments.

(28)



Given this skeleton, we may assume that quantifiers, have two additional index-set arguments. The role of index set A is to specify those indices which impose referential restrictions which should not be inherited from the restrictor to the global context. The role of index set B is to specify those indices which impose referential restrictions which should not be inherited from the scope to the global context. This leads to the configuration in (29), but notice that any other configuration in which A and B are visible to the quantifier determiner is acceptable.⁸

(29)



Then at least the following indexing rules apply:

(30) Indexing Rules:

- a. A and B never contain indices of proper names or quantifiers (other than the active one).⁹
- b. A and B may only contain indices of definite NPs if the quantifier binds into their restrictor.
- c. A and B never contain indices which do not appear in the restrictor and scope respectively.
- d. A may not contain *i* if *i* is the index of a pronoun in the restrictor which is bound from outside the restrictor.
- e. B may not contain *i* if *i* is the index of a pronoun in the scope which is bound from outside the scope.

The use of these index sets is conceptually very similar to what happens in DRT or FCS. If we consider standard DRT, A is the set of discourse referents declared in the restrictor box and of B is the set of discourse referents declared in the scope box.

⁸ In *Onea 2012* I propose a version with a more transparent syntactic structure concerning the index sets, however, in that version, no donkey anaphora come for free.

⁹ We do not consider generic uses of proper names.

3.3.3 Lexical entries and examples

The lexical entry for *every*, the only quantifier discussed in this paper, is very similar to the treatment of quantifiers in various versions of dynamic semantics. In particular, quantification goes over assignments. The main difference to the DRT treatment of quantifiers is that in DRT, quantification extends embedding functions, however, assignments in this theory are always total functions, hence they cannot be extended. The notions of exceptive equality and selective compatibility are used instead.¹⁰

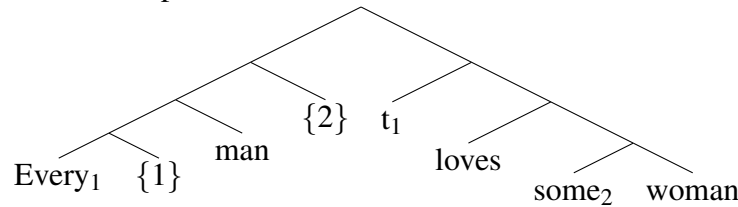
$$(31) \quad \begin{aligned} \|\text{every}_i\|^w = & \lambda A. \lambda P_{\langle \langle a,e \rangle, \langle a,t \rangle \rangle}. \lambda B. \lambda H_{\langle a,t \rangle}. \lambda h_a. [h \in_{B \setminus \{i\}} \mathcal{D}H \wedge h \in_{A \cup \{i\}} \mathcal{D}(P(\lambda g.gi))] . \\ & \forall h'. h' =_{A \cup \{i\}} h \wedge h' \in \mathcal{D}(P(\lambda g.g1)) \wedge P(\lambda g.g1) h' \rightarrow \\ & \exists h''. h' =_{B \setminus \{i\}} h'' \wedge h'' \in \mathcal{D}H \wedge Hh'' \end{aligned}$$

Notice the special treatment of the *i* index. The formula can be significantly simplified if we assume that the quantifier index is always a part of the A index set and the quantifier index is never part of the B index. The formulation in (31) tacitly repairs wrong indexing, but if these indexing rules hold, we can simplify to (32). In the following, we will make sure that the indexing allows the use of (32).

$$(32) \quad \begin{aligned} \|\text{every}_i\|^w = & \lambda A. \lambda P_{\langle \langle a,e \rangle, \langle a,t \rangle \rangle}. \lambda B. \lambda H_{\langle a,t \rangle}. \lambda h_a. [h \in_B \mathcal{D}H \wedge h \in_A \mathcal{D}(P(\lambda g.gi))] . \\ & \forall h'. h' =_A h \wedge h' \in \mathcal{D}(P(\lambda g.gi)) \wedge P(\lambda g.gi) h' \rightarrow \exists h''. h' =_B h'' \wedge h'' \in \mathcal{D}H \wedge Hh'' \end{aligned}$$

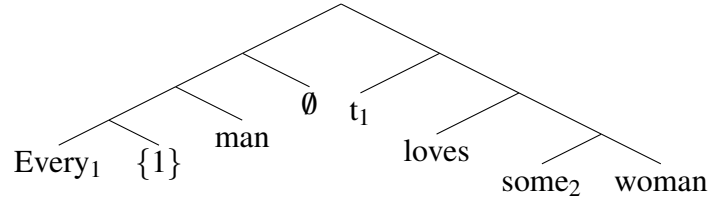
We now can consider an example. For (33), the difference between narrow and wide scope for the indefinite simply boils down to whether the referential index of the indefinite appears in the B set or not.

- (33) Every man loves some woman.
 a. Narrow scope for *some woman*:



- b. Wide scope for *some woman*:

¹⁰ I do not give the types of A and B. We can either think of them as $\langle a,t \rangle$ or we can define an additional type for sets.



For both readings, the scope argument is unchanged; its semantic value is given in (34). Similarly, the restrictor is also the same, hence its semantic value is given in (35):

$$\begin{aligned}
 (34) \quad & \left\| \begin{array}{c} \text{t}_1 \\ \text{loves} \\ \text{some}_2 \text{ woman} \end{array} \right\|^w = \lambda h [\mathbf{W}(h2, w)] . \mathbf{L}(h1, h2, w) \\
 (35) \quad & \left\| \begin{array}{c} \text{Every}_1 \{1\} \text{ man} \end{array} \right\|^w = \\
 & \lambda B . \lambda H . \lambda h . [h \in_B \mathcal{D}H] . \forall h' . h' =_{\{1\}} h \wedge \mathbf{M}(h'1, w) \rightarrow \exists g . h' =_{Bg} \wedge g \in \mathcal{D}H \wedge Hg
 \end{aligned}$$

This leads to the readings in (36). In the calculation, I omitted the existential quantification in the wide scope reading because it happens to be vacuous, as the reader may check.

- (36) a. Narrow scope reading:
 $\lambda h . [h \in_{\{2\}} \{g | \mathbf{W}(g2, w)\}] .$
 $\forall h' . h' =_{\{1\}} h \wedge \mathbf{M}(h'1, w) \rightarrow \exists h'' . h' =_{\{2\}} h'' \wedge \mathbf{W}(h''2, w) \wedge \mathbf{L}(h''1, h''2, w)$
- b. Wide scope reading:
 $\lambda h . [\mathbf{W}(h2, w)] . \forall h' . h' =_{\{1\}} h \wedge \mathbf{M}(h'1, w) \rightarrow \mathbf{W}(h'2, w) \wedge \mathbf{L}(h'1, h'2, w)$

Notice that the second occurrence of the predicate \mathbf{W} in the wide scope reading is also vacuous, since h' will already have inherited from h that $h'2$ is a woman. Therefore we can delete the second copy.

Once we add **ASSERT**, the result for the wide reading is transparent, however the narrow scope reading comes with a potentially surprising non-emptiness requirement for the set of women. In (36a) the intuition is that the $h \in_{\{2\}} \{g | \mathbf{W}(g2, w)\}$ should simply be disregarded, since it says that h is in the set of assignments which assign a woman to 2 disregarding the requirement that $h2$ should be a woman. However, due to the way $=_A$ has been formally implemented, ignoring this requirement would be a computational error. As mentioned in footnote 7, this consequence can be avoided using a more involved formal implementation, however, for now, it does not change

the truth conditions and I therefore ignore it.

- (37) a. Narrow scope reading with ASSERT:
 $\exists h. \exists g. \mathbf{W}(g2, w) \wedge$
 $\forall h'. h' = \{1\} h \wedge \mathbf{M}(h'1, w) \rightarrow \exists h''. h'' = \{2\} h'' \wedge \mathbf{W}(h''2, w) \wedge \mathbf{L}(h''1, h''2, w)$
- b. Wide scope reading with ASSERT:
 $\exists h. \mathbf{W}(h2, w) \wedge \forall h'. h' = \{1\} h \wedge \mathbf{M}(h'1, w) \rightarrow \mathbf{L}(h'1, h'2, w)$

This is the point at which the main advantage of the theory proposed here becomes obvious. Treating referential contexts as local variables which are incrementally enriched in the process of the computation allows us to fully compositionally derive both narrow and wide scope for indefinites. Thereby, there is a very clear difference between the way indefinites take ‘scope’ and the way quantifiers take scope. While the scope of quantifiers is controlled by movement, the ‘scope’ of indefinites is controlled by indexing alone.

One of the essential consequences of this decision is that in order for indefinites to show ‘specific’ wide scope readings, they do not need to somehow be extracted from a scope island. All the information needed at the higher level (i.e., in establishing the index set, which is outside the respective scope island) is the index of the respective indefinite NP. Once the index is known, a decision can be made as to whether the indefinite will have narrow or wide scope with respect to any quantifier. It is important to see that this is conceptual progress, since knowing the index is comparable to knowing the name of a variable, hence, the resources used to derive an exceptional scope reading for indefinites are comparable to the resources used for the reading (4c), however, this time, the Donald Duck problem no longer arises. Hence, while the indexing used here is very similar to the sets of discourse referents used in DRT, the basic semantics is fundamentally different, since in DRT, we need to have both the name of the discourse referent and the condition reflecting the descriptive material in the indefinite NP to take wide scope, whereas in this system the index will suffice. This is shown on some examples in the next section.

4 Exceptional scope of indefinites

We now consider some examples of exceptional scope for indefinites. We will thereby use a relative clause in the restrictor of *every* as a scope island. Since in the present system the scope of indefinites is not established by movement, it is not surprising that the two trees are directly interpreted as the wide and narrow scope reading respectively.

- (38) Every man who loves some woman sleeps.
 a. Wide scope reading

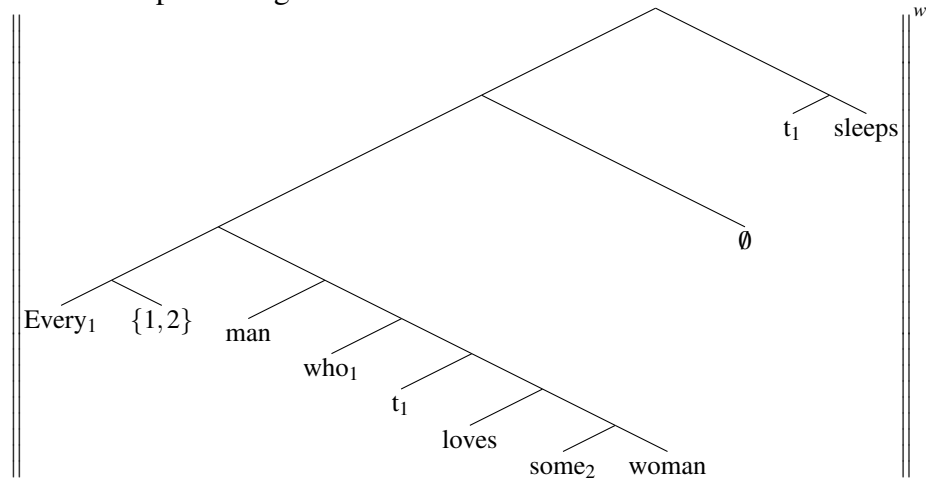
Indefinite donkeys on islands



$\lambda h. [h \in_{\{1\}} \{g | \mathbf{W}(g2, w)\}] . \lambda v.$

$\forall h'. h' =_{\{1\}} h \wedge \mathbf{W}(h'2, w) \wedge \mathbf{M}(h'1, w) \wedge \mathbf{L}(h'1, h'2, w) \rightarrow \mathbf{S}(h'1, w)$

b. Narrow scope reading



$\lambda h. [h \in_{\{1,2\}} \{g | \mathbf{W}(g2, w)\}] .$

$\forall h'. h' =_{\{1,2\}} h \wedge \mathbf{W}(h'2, w) \wedge \mathbf{M}(h'1, w) \wedge \mathbf{L}(h'1, h'2, w) \rightarrow \mathbf{S}(h'1, w)$

If the indefinite is in the scope of a quantifier and not in the restrictor, precisely the same will happen. This is only shown by the LFs. The computation should be straightforward:

(39) Every man loves the picture some woman painted.

- a. Wide scope reading: [[[[[every₁ {1}] man] \emptyset] [t₁ [loves [the₂ [picture [that₂ [[some₃ woman] [painted t₂]]]]]]]]]]]
- b. Narrow scope reading: [[[[[every₁ {1}] man] {3}] [t₁ [loves [the₂ [picture [that₂ [[some₃ woman] [painted t₂]]]]]]]]]]]

And, finally, so called intermediate readings are also possible, as suggested by the structures in (40).

- (40) Every boy believes that every man loves the picture some woman painted.
- a. Wide scope reading for *some woman*:

$$\text{[[[[[every}_4 \{4\}] \text{ boy}] \emptyset] [t_4 \text{ [believes [that [[[[[every}_1 \{1\}] \text{ man}] \emptyset] [t_1 \text{ [loves [the}_2 \text{ [picture [that}_2 \text{ [[some}_3 \text{ woman] [painted } t_2]]]]]]]]]]]]]]]]]]]$$
 - b. Narrow scope reading for *some woman*:

$$\text{[[[[[every}_4 \{4\}] \text{ boy}] \emptyset] [t_4 \text{ [believes [that [[[[[every}_1 \{1\}] \text{ man}] \{3\}] [t_1 \text{ [loves [the}_2 \text{ [picture [that}_2 \text{ [[some}_3 \text{ woman] [painted } t_2]]]]]]]]]]]]]]]]]]]$$
 - c. Intermediate scope reading for *some woman*:

$$\text{[[[[[every}_4 \{4\}] \text{ boy}] \{3\}] [t_4 \text{ [believes [that [[[[[every}_1 \{1\}] \text{ man}] \emptyset] [t_1 \text{ [loves [the}_2 \text{ [picture [that}_2 \text{ [[some}_3 \text{ woman] [painted } t_2]]]]]]]]]]]]]]]]]]]$$

5 Donkey sentences

We now turn to donkey sentences, such as (41). This sentence has a natural wide scope reading for *some woman* which allows the usual pronominal binding of *her*. Crucially, however, even if *some woman* gets narrow scope, the pronominal binding of *her* is possible. The reason for this is that the scope of a quantifier is evaluated in referential contexts which may differ in any index which is relevant for the restrictor from those assignment functions which verify the restrictor, as in DRT.

- (41) Every man who loves some woman misses her.
- a. Narrow scope

$$\text{[[[[[Every}_1 \{1,2\}] \text{ [man [who}_1 [t_1 \text{ [loves [some}_2 \text{ woman]]]]]]] \emptyset] [t_1 \text{ [misses her}_2]]].$$

$$\lambda h. [h \in_{\{1,2\}} \{g | \mathbf{W}(g2)_w\}].$$

$$\forall h'. h' =_{\{1,2\}} h \wedge \mathbf{W}(h'2)_w \wedge \mathbf{M}(h'1)_w \wedge \mathbf{L}(h'1)(h'2)_w \rightarrow \mathbf{miss}(h'1)(h'2)_w$$
 - b. Wide scope

$$\text{[[[[[Every}_1 \{1\}] \text{ [man [who}_1 [t_1 \text{ [loves [some}_2 \text{ woman]]]]]]] \emptyset] [t_1 \text{ [misses her}_2]]] \lambda h. [h \in_{\{1\}} \{g | \mathbf{W}(g2)_w\}].$$

$$\forall h'. h' =_{\{1\}} h \wedge \mathbf{M}(h'1)_w \wedge \mathbf{L}(h'1)(h'2)_w \rightarrow \mathbf{miss}(h'1)(h'2)_w$$

6 Outlook

In this paper, I have proposed a theory of semantic composition for LFs which naturally derives two properties of indefinites which have been puzzling in the literature: that they license donkey anaphora and that they may scope out of scope islands. To achieve this, I have used one of the main intuitions of dynamic semantics, that

indefinites are not bona fide quantifiers. Instead, they introduce a discourse referent, which is modelled as a referential constraint. Crucially, referential constraints introduced by indefinites (and any other referential expressions) are local (and not global).

Of course, such a compositional system raises a number of questions. Most prominently, there is a question as to whether the technical possibilities opened here come with natural ways to capture constraints on the interpretation of indefinites. We know from intensive research on indefinite determiners that various indefinite determiners come with various types of constraints. For instance, Russian *nibud* indefinites can only take scope under some quantifiers, especially under intensional operators (Onea & Geist 2011). As opposed to this *gewiss*-indefinites in German, for instance, generally take widest possible scope (Ebert, Ebert & Hinterwimmer 2012).

References

- Bende-Farkas, Agnes & Hans Kamp. 2001. Indefinites and binding: From specificity to incorporation. Lecture notes (revised version) for 13th ESSLLI, Helsinki. <http://www.ims.uni-stuttgart.de/~hans/publications.html>.
- Brasoveanu, Adrian & Donka Farkas. 2011. How indefinites choose their scope. *Linguistics and Philosophy* 34. 1–55.
- Büring, Daniel. 2005. *Binding Theory*. Cambridge, New York: Cambridge University Press.
- Dekker, Paul. 2004. The pragmatic dimension of indefinites. *Research on Language and Computation* 2. 365–399.
- Ebert, Christian, Cornelia Ebert & Stefan Hinterwimmer. 2012. The semantics and pragmatics of “bestimmt” and “gewiss”. In *Proceedings of NELS 40*, MIT.
- Egli, Urs. 1979. The Stoic concept of anaphora. In Armin von Stechow (ed.), *Semantics from Different Points of View*, 266–283. Berlin: Springer.
- Elbourne, Paul. 2001. E-type anaphora as np-deletion. *Natural Language Semantics* 9. 241–288.
- Endriss, Cornelia. 2009. *Quantificational topics - A Scopal Treatment of Exceptional Wide Scope Phenomena* Studies in Linguistics and Philosophy. Springer.
- Farkas, Donka. 1997. Dependent indefinites. In F. Corblin, D. Godard & J.-M. Marandin (eds.), *Empirical Issues in Formal Syntax and Semantics*, 243–267. Peter Lang Publishers.
- Farkas, Donka. 2002. Specificity distinctions. *Journal of Semantics* 19. 213–243.
- von Fintel, Kai & Irene Heim. 2011. Intensional semantics. Lecture notes, MS. MIT
- Geach, Peter. 1962. *Reference and Generality*. Ithaca: Cornell University Press.
- Geurts, Bart. 2002. Indefinites, presupposition, and scope. In Rainer Bäuerle

- & Thomas Ede Zimmermann (eds.), *Presuppositions and Discourse*, Oxford: Elsevier.
- Heim, Irene. 1982. *The Semantics of Definite and Indefinite Noun Phrases*: University of Massachusetts PhD. dissertation.
- Heim, Irene. 1990. E-type pronouns and donkey anaphora. *Linguistics and Philosophy* 13. 137–177.
- Heim, Irene. 2011. Definiteness and indefiniteness. In Klaus von Heusinger, Claudia Maienborn & Paul Portner (eds.), *Semantics*, vol. 33 HSK, 996–1025. Berlin: de Gruyter.
- Heim, Irene & Angelika Kratzer. 1998. *Semantics in Generative Grammar*. Oxford: Blackwell.
- von Heusinger, Klaus. 2002. Specificity and definiteness in sentence and discourse Structure. *Journal of Semantics* 19. 1–30.
- von Heusinger, Klaus. 2011. Specificity. In Klaus von Heusinger, Claudia Maienborn & Paul Portner (eds.), *Semantics: An International Handbook of Natural Language Meaning*, vol. 2, 1024–1057. Berlin: de Gruyter.
- Kamp, Hans. 1981. A Theory of Truth and Semantic Representation. In J. Groenendijk & M. Stokhoff (eds.), *Formal Methods in the Study of Language*, 277–322. Amsterdam: Mathematical Centre, Tract 135.
- Kratzer, Angelika. 1998. Scope or pseudo-scope? Are there wide-scope indefinites? In S. Rothstein (ed.), *Events in Grammar*, 163–196. Dordrecht: Kluwer.
- Kratzer, Angelika & Junko Shimoyama. 2002. Indeterminate phrases. The view from Japanese. In Yokio Otsu (ed.), *Proceedings of the Third Tokyo Conference on Psycholinguistics*, 1–25. Tokyo: Hituzi Siobo.
- Onea, Edgar. 2012. *Potential questions in discourse and grammar*. MS.
- Onea, Edgar & Ljudmila Geist. 2011. Indefinite determiners and the pragmatics of referential anchoring. *International Review of Pragmatics* 3. 194–227.
- Poesio, Massimo 1996. Semantic ambiguity and perceived ambiguity. In *Semantic Ambiguity and Underspecification*, .
- Reinhart, Tanya. 1992. Wh-in-situ: An apparent paradox. In Paul Dekker & Martin Stokhof (eds.), *Proceedings of the Eighth Amsterdam Colloquium*, 483–492. University of Amsterdam ILLC-Publications.
- Rizzi, Luigi 1997. The fine structure of the left periphery. In L. Haegeman (ed.), *Elements of Grammar*, 281–337. Dordrecht: Kluwer.
- Russel, Bertrand. 1905. On denoting. *Mind* 14. 479–493.
- von Stechow, Armin & Ede Zimmermann. 2005. A problem for a compositional treatment of de re attitudes. In G. Carlson & F. Pelletier (ed.), *Reference and Quantification: The Partee Effect*, 207–2228. Stanford: CSLI Publications.

Indefinite donkeys on islands

Edgar Onea
Nikolausberger Weg 23
D-37073 Göttingen
Tel: +49551 / 39-20023
edgar.oney@zentr.uni-goettingen.de