A Modular Approach to Intensionality*

Gilad Ben-Avi
Technion

Yoad Winter
Technion/NIAS

{bagilad,winter}@cs.technion.ac.il

Abstract

This paper introduces a procedure that takes a simple version of extensional semantics and generates from it an equivalent possible-world semantics that is suitable for treating intensional phenomena in natural language. This process of *intensionalization* allows to treat intensional phenomena as stemming exclusively from the lexical meaning of words like *believe*, *need* or *fake*. We illustrate the proposed intensionalization technique using an extensional toy fragment. This fragment is used to show that independently motivated extensional mechanisms for scope shifting and verb-object composition, once properly intensionalized, are strictly speaking responsible for certain intensional effects, including *de dicto*/*de re* ambiguities and coordinations containing intensional transitive verbs. While such extensional-intensional relations have often been assumed in the literature, the present paper offers a formal sense for this claim, facilitating the dissociation between extensional semantics and intensional semantics.

1 Introduction

The simplicity and elegance of standard versions of extensional higher-order logics make them attractive for treating many phenomena in natural language. The arguments for intensional (and hyper-intensional) semantics are of course compelling, but we would not like these considerations to complicate the analysis of properly-extensional phenomena. Unfortunately this is often the case, and especially in Montague's classical treatment in PTQ. In order to address this tension between extensional semantics and intensional semantics, this paper studies the relations between elementary extensional semantics and intensional semantics such as Montague's IL or Gallin's (1975) Ty2. We propose a general process of *intensionalization* that maps an extensional framework to such an intensional framework, and illustrate its architectural benefits using a toy fragment. More generally, we argue that also in other frameworks, there are methodological and empirical reasons for taking intensionalization procedures to be a central part of the study of intensional phenomena.

The distinction between parts of a language that exhibit intensional effects and parts that do not can often be reduced to a simple distinction between two kinds of lexical items: those that create an intensional context and those that do not. In this paper, an expression that creates an intensional context is called *intension-sensitive*. Some well-known examples are the verbs *seek* and *believe*, and the adjective *fake*. Expressions that do not create an intensional context, such as the verb *kiss* or the adjective *red*, are called *intension-insensitive*. With this distinction,

*Special thanks to Chris Barker, Nissim Francez, Philippe de Groote, Makoto Kanazawa and Reinhard Muskens for their thorough remarks. Thanks also to Johan van Benthem, Gennaro Chierchia, Edit Doron, Theo Janssen, Ed Keenan, Fred Landman, Anita Mittwoch, Larry Moss, Remko Scha and Anna Szabolcsi for discussions. The second author is grateful for the support of the Netherlands Institute of Advanced Study (NIAS), where part of the work was carried out, and to Makoto Kanazawa for his kind invitation to the third workshop on Lambda Calculus and Formal Grammar at NII (Tokyo), where parts of this paper were presented.

we assume that an extensional semantics is sufficiently adequate for expressions that consist solely of intension-insensitive lexical items, while an intensional semantics is only needed for expressions with intension-sensitive lexical items.

In this paper we propose a modular approach to the architecture of intensional systems that is based on this assumption. We start out from an extensional system that only contains intension-insensitive lexical items. We then systematically shift all extensional types and meanings to intensional types and meanings. Following Van Benthem (1988), we refer to this shifting process as *intensionalization*. This process is necessary to allow the addition of intension-sensitive lexical items to the system. For example, for a correct analysis of a sentence like (1) below, the intension-insensitive indefinite must denote an *intension* (e.g. an intensional quantifier as in Montague 1973, or a property as in Zimmermann 1993).

(1) Mary needed a doctor.

In the proposed analysis such intensions result from the intensionalization of intension-insensitive words like *a* and *doctor* and the semantic modes of composition in the grammar.

Beside types and meanings of expressions, the intensionalization process changes very little in the extensional system. For example, *de dicto*/*de re* ambiguities in sentences like (1), and coordinations of intension-sensitive and intension-insensitive transitive verbs as in (2) below, are treated as manifestations of purely extensional mechanisms in their intensionalized guise.

(2) Mary sought, found and kissed a doctor.

For coordinations as in sentence (2), it is moreover necessary to allow the type of the intension-insensitive verb *kiss* to be the same as that of the intension-sensitive verb *seek* (see for example Partee and Rooth 1983). We regard this too as resulting from a general intensionalization process, this time mapping the type and meaning of the verb *kiss* to its proper intensional type and meaning.

One central formal aspect of the intensionalization process is truth-conditional *soundness*. In order to preserve the insights of an extensional semantics, we need to guarantee that its intensionalized version is descriptively equivalent to it. In more exact terms: both the extensional semantics and its image under intensionalization should provably describe the same entailment relations between sentences. Establishing the soundness of our proposed intensionalization procedure is one of the main subjects addressed in Ben-Avi (2007), and it will only lightly be touched upon in this paper.

The paper is organized as follows. Section 2 briefly reviews some relevant works. Section 3 defines a setting for an extensional lexicon and a minimal set of derivation rules, and illustrates them using a toy lexicon. The intensionalization process is the subject of Section 4, which also demonstrates its application to the toy lexicon of Section 3. Section 5 demonstrates how intension-sensitive lexical items are added to the intensionalized lexicon, and illustrates the resulting grammatical interactions between such items and intension-insensitive expressions and extensional mechanisms.

2 From Montague's PTQ to modern type-theoretical treatments of intensionality

In the classical work of Montague (1973), henceforth PTQ, all lexical items are assigned intensional types independently of whether they are intension-sensitive or intension-insensitive. This follows from Montague's uniform category-to-type mapping together with his famous 'generalization to the worst case' strategy. The motivation for this move is exemplified by the sentences in (3) below, involving the transitive verbs (TVs) *kiss* and *seek*.

(3) a. Mary kissed a king.

b. Mary sought a king.

The object of an intension-sensitive transitive verb (ITV) like *seek* must denote some kind of intension. Specifically, in PTQ such objects denote intensional *quantifiers*. Accordingly, ITVs can be treated as relations between entities and intensional quantifiers. As exemplified in (2) above, ITVs appear in coordinations with intension-insensitive (or *extensional*) transitive verbs (ETVs).[1] To allow the derivation of such coordinations, Montague assigned ETVs the same type as that of ITVs. But this uniformity led to disparity elsewhere in the system since the truth-conditional behavior of ITVs is different than that of ETVs. For instance, (3a) asserts the existence of a king whereas (3b) does not. To capture such differences, we should make sure that while ITVs denote arbitrary relations between entities and intensional quantifiers, ETVs are restricted to only denote such relations that 'behave like' binary relations between entities. This and similar limitations on the denotations of intension-insensitive lexical items are guaranteed by Montague's (in)famous meaning postulates.

Partee and Rooth (1983, henceforth P&R) used their generalized conjunction schema to show that Montague's strategy of assigning the same semantic types to all TVs makes some wrong predictions. P&R proposed that the lexical type of every word should be the 'minimal' type that it requires. For example, according to P&R the minimal type for an ITV like *seek* is an intensional type similar to its type in PTQ. The type for an ETV like *kiss*, on the other hand, is the minimal type needed for a correct analysis of sentences like (3a) – the type of binary relations between entities. In order to enable coordinations between ITVs and ETVs, P&R define a type-shifting operator by which denotations of ETVs can be shifted to denotations of the ITV type. This operator is only one in an array of type-shifting operators that P&R define for various purposes. What is most relevant for our purposes here is P&R's assumption that (at least some) intension-insensitive lexical items should have a simple extensional type, whereas their intensional type can be derived by some non-lexical part of the grammar.

The idea of defining a general process of *intensionalization* by which extensional types and meanings are mapped to intensional types and meanings was suggested by Van Benthem (1988). Van Benthem (henceforth vB) proposes to intensionalize extensional types by using the idea that "the main thrust of intensionalization consists in *sentences* becoming *propositions*, denoting a truth value trajectory across possible worlds, rather than one single truth value" (Van Benthem 1988, p.46). Formally, this means that extensional types are intensionalized by replacing t by (st) everywhere. vB's paper does not, however, include a derivation of meanings using the type change strategy that he proposes.

As far as we know, the only full-fledged intensionalization procedure in the literature, even if not under this label, was defined by Keenan and Faltz (1985), predating vB's proposal. In the first part of their monograph, Keenan and Faltz (henceforth K&F) develop an extensional Boolean semantics of English. The aim of the second part of their book is "to create a system of model-theoretic semantic interpretation for our logical language which will preserve the advantages and insights revealed by our extensional system while allowing properly intensional facts to be represented." (Keenan and Faltz 1985, p.274). K&F define an array of operators by which every expression in their extensional system can be intensionalized. Notwithstanding its innovative aspects, K&F's intensionalization procedure is very complex, specific to their fragment, and does not generalize uniformly to modern extensional type systems.

[1]For a recent study of such multiple conjunctions and their implications for type-shifting strategies and the syntax/semantics of coordination, see Winter (2006).

A simple intensionalization procedure can be inferred from the introduction of intensional semantics in Heim and Kratzer (1998,ch.12). The typing strategy of Heim and Kratzer (henceforth H&K) is intensionally poorer than that of vB's: an extensional type σ in H&K's discussion is simply intensionalized by adding a single s in front of it, resulting in an intensional type $(s\sigma)$. H&K do not explain how ITVs like *seek*, which need to take intensions as *arguments*, can be added to their system while allowing coordinations with ETVs. To do that, an additional type-shifting strategy for ITVs similar to P&R's could probably be used on top of intensionalization, but that would come at the cost of complicating the array of operators in the grammar.

A similar architectural motivation to Keenan and Faltz' can be found in Shan (2001). Shan uses the concept of *monads* from theoretical computer science in order to extend a simple version of natural language semantics for treating various phenomena, including interrogatives, focus, quantification, variable binding and intensionality. Shan partly follows an early version of Barker (2002), who uses a comparable architectural approach for treating natural language quantifiers using the notion of *continuations*. Shan's exposition of his "intensionality monad" is terse and does not give many details about its empirical implications. As far as we were able to see, however, Shan's typing strategy is similar to Heim and Kratzer's simple procedure and is therefore likely to involve the same limitation that we mention above.

The general intensionalization of meanings that is proposed in this paper is based on vB's typing strategy, and is thus more general than the scheme exemplified by H&K. Unlike P&R's type-shifting operator for ETVs, our intensionalization applies uniformly to all lexical items. Moreover, our analysis generalizes to different treatments of intension-sensitive items, most notably to both Montague's and Zimmermann's typings of ITVs.

3 The extensional setting

To formalize an intensionalization procedure we first need to make explicit assumptions about the extensional semantics. In the setting that we define below we include only the bare semantic details that are necessary for defining the intensionalization procedure. Syntactic, pragmatic and phonological details are ignored. In this semantic setting we assign an extensional semantic *type* to every word in the lexicon.[2] The set $\mathcal{T}_{\text{ex}}$ of *extensional types* is routinely defined below.

(4) $\mathcal{T}_{\text{ex}}$ is the smallest set that satisfies $\{e,t\} \subseteq \mathcal{T}_{\text{ex}}$, and $(\sigma_1\sigma_2) \in \mathcal{T}_{\text{ex}}$ if $\sigma_1, \sigma_2 \in \mathcal{T}_{\text{ex}}$.

These types will be intensionalized according to vB's recipe. In order to also intensionalize extensional meanings, we have to specify the logical, truth-conditional, aspects of lexical meaning that are relevant for our purposes. Consider for instance the following sentences.

(5) a. Every king smiled.
b. Every bald king smiled.

To capture the observed entailment from (5a) to (5b) it is standardly assumed that the noun *king* and the verb *smile* denote arbitrary sets of entities, that the adjective *bald* denotes a *restrictive* function that maps any set of entities A to a subset of A, and that the determiner *every* denotes the subset relation between sets of entities. One way to formalize such assumptions would be to follow Montague's tradition and use lambda-terms as meaning representations. Here we prefer a direct interpretation of language expressions, mainly because with such an interpretation we do not need to be committed to a specific format for meaning representation. Formally, lexical

[2] Ambiguous words may be assigned a finite set of types, but we ignore this possibility here.

items are directly interpreted by models in the class of *intended models*, whose definition will make sure that truth-conditional restrictions on lexical meanings are respected.

A model is standardly a pair that consists of a *frame* and an *interpretation function*. A frame in an extensional system – henceforth an *extensional frame* – is the collection of domains D_σ for the extensional types σ. Standardly, we assume that D_t is the set of truth values, denoted here by $\{0,1\}$, that D_e is an arbitrary nonempty set of *entities*, and that $D_{(\tau\sigma)}$ is the set of all functions from D_τ to D_σ, denoted by $D_\sigma^{D_\tau}$. Thus, a frame is uniquely determined by the choice of the set of entities for D_e. For a fixed nonempty set E, we say that the frame in which $D_e = E$ is the *E-based extensional frame*. For reference, this is formally stated below.

(6) Given a nonempty set E, an *(E-based) extensional frame* is the set $\mathcal{F} = \{D_\sigma : \sigma \in \mathcal{T}_{\text{ex}}\}$, where $D_e = E$, $D_t = \{0,1\}$ and $D_{(\tau\sigma)} = D_\sigma^{D_\tau}$.

An *interpretation function* is a mapping from the lexicon into a frame. For any given frame $\mathcal{F}$ it is possible to define many such interpretations. However, as mentioned above, not all such interpretations are admissible. The interpretation should agree with the truth-conditional restrictions imposed on the lexical items. We formalize these restrictions by assigning a functional $\{\{\alpha\}\}$ to every lexical item α, which maps any frame $\mathcal{F}$ to the subset $\{\{\alpha\}\}^{\mathcal{F}}$ of $\bigcup \mathcal{F}$ that consists of all admissible interpretations for α.

As an example, consider the simple lexicon in Table 1. The words in this lexicon can be classi-

word α	**Type**	$\{\{\alpha\}\}^{\mathcal{F}}$	λ-term
Mary, *John*	e	D_e	
red, *bald*	et	D_{et}	
king, *queen*	et	D_{et}	
smile, *jump*	et	D_{et}	
kiss, *eat*	$e(et)$	$D_{e(et)}$	
every	$(et)((et)t)$	$\{\textbf{every}\}$	$\lambda A_{et}\lambda B_{et}.\forall x_e[A(x) \rightarrow B(x)]$
a	$(et)((et)t)$	$\{\textbf{some}\}$	$\lambda A_{et}\lambda B_{et}.\exists x_e[A(x) \wedge B(x)]$

Table 1: An extensional lexicon

fied according to how they restrict their possible interpretations. Proper names, common nouns and (in)transitive verbs, as well as other expressions of open lexical categories, allow any interpretation in the domain of their type. Such lexical items are standardly interpreted as *nonlogical constants*: lexical expressions to which the interpretation function can freely assign any object in the respective domain. For example, the 3rd column in Table 1 indicates that, given a frame $\mathcal{F}$, the set $\{\{king\}\}^{\mathcal{F}}$ is $D_{et} \in \mathcal{F}$. This means that the noun *king* can be interpreted as any set of entities. The same holds for intransitive verbs. Similarly, a transitive verb like *kiss* can be interpreted as any binary relation between entities.

A similar nonlogicality assumption can be adopted for adjectives like *bald* or *red*, but it involves slightly more complex considerations. For the sake of the example (only), we assume that all intension-insensitive adjectives are *intersective* (cf. Kamp and Partee 1995): they all describe functions f_X of type $(et)(et)$, such that X is a set of entities and for all $A \in D_{et}$: $f_X(A) = A \cap X$. Thus, we assume that x is a bald king iff x is bald and x is a king, y is a red car iff y is red and y is a car, etc. Using this assumption about intersectivity, the lexical meaning of an intension-insensitive adjective can be represented using an assumption about an arbitrary set of entities X, and a systematic mapping from X to the intersective meaning of the adjective. To achieve that, we assume that intension-insensitive adjectives, like intension-insensitive intransitive verbs and common nouns, are nonlogical constants of type et. The mapping of adjectives to type $(et)(et)$ will be implemented below as a phonologically-silent item (ε_{adj}) in the lexicon.

In distinction to the nonlogical constants above, there are other words, including most determiners, prepositions, sentential connectives and other lexical expressions of closed categories, that have a single fixed interpretation in a frame. Such words are often related to notions of logicality and treated using functionals that are referred to as *logical constants*. Here we use this terminology in a narrow technical sense, as will be clarified below. For studies of logicality see for instance Sher (1991) and Keenan (2000), as well as the references therein. For instance, it is standard to assume that the determiner *every* is interpreted in every model as the subset relation on the domain of entities: the function that maps a pair of sets of entities to *true* just in case the first set is a subset of the second set. This means that unlike the nonlogical constants discussed above, the denotation of *every* in a model is only dependent on the (E-based) extensional frame, and is independent of the interpretation function in the model. Thus, the meaning of *every* is specified so that $\{\{every\}\}^{\mathcal{F}}$ is a singleton. It is customary (and beneficial) to define this function using the lambda-calculus with equality or, alternatively, by adding standard quantifiers to the pure lambda-calculus. In Table 1 we use the latter option, but without commitment to this formalism as a general means for meaning representation: for our general purposes in this paper it is not necessary to develop a theory of logical constants, let alone adopt a uniform notation of their meanings. In order to introduce the proposed intensionalization procedure it is sufficient to assume that $\{\{every\}\}^{\mathcal{F}}$ is a singleton for any frame $\mathcal{F}$, and that its member is some function **every** in $D_{(et)((et)t)} \in \mathcal{F}$. The actual definition of **every** and other logical constants do not have to be specified for the intensionalization procedure that we propose. Definitions of logical constants are given in this paper only for the sake of demonstrating the intensionalizations of familiar examples.

We now generalize, adopting the following terminology for the rest of this paper.

(7) Let α be a lexical item of type σ.

a. We say that α is a *nonlogical constant* if for every frame $\mathcal{F}$: $\{\{\alpha\}\}^{\mathcal{F}} = D_\sigma \in \mathcal{F}$.

b. We say that α is a *logical constant* if for every frame $\mathcal{F}$ there is $\varphi \in D_\sigma \in \mathcal{F}$, such that $\{\{\alpha\}\}^{\mathcal{F}} = \{\varphi\}$.

The collection of functionals $\{\{\{\alpha\}\} : \alpha \in \Sigma\}$ over a lexicon Σ defines the class of *intended models* for Σ. A model is an intended model for a lexicon Σ just in case it interprets every lexical item α in Σ in accordance with the constraints imposed on α. Formally:

(8) A model $\mathcal{M} = \langle \mathcal{F}, I \rangle$ is an *intended model* for a lexicon Σ iff $I(\alpha) \in \{\{\alpha\}\}^{\mathcal{F}}$ for every $\alpha \in \Sigma$.

To derive interpreted sentences from interpreted lexical items, let us add a simple notion of a grammar, keeping in mind that, as before, we only introduce the bare semantic notions that are necessary in order to define an intensionalization process in a most general way. We define two derivation rules: one for functional application and another for conjunction. For the sake of presentation we only deal with the conjunctive word *and*. Each of the two rules describes how a compound derivation Δ and its denotation $[\![\Delta]\!]^{\mathcal{M}}$ in an intended model $\mathcal{M}$ are obtained from simpler (sub-)derivations and their denotations. The basic derivations are the words in the lexicon. Trivially, the expression obtained by such a derivation is the word itself, its type is as appears in the lexicon, and the denotation in an intended model is whatever object the interpretation function assigns to the lexical item. These basic derivations are officially specified in (9).

(9) a. Every lexical item α is a derivation of the expression α. The type of the derivation is the lexical type of α.
b. For every intended model $\mathcal{M}$, $[\![\alpha]\!]^{\mathcal{M}} = I_{\mathcal{M}}(\alpha)$.

For example, *king* is a derivation of the expression *king* over the lexicon in Table 1. The type of this derivation is (et) and its denotation in a given intended model is the unary predicate assigned by the interpretation function.

The rule of *functional application* allows two derivations to be combined whenever one of them is of a functional type $(\sigma\tau)$ and the other one is of type σ. The type of the resulting derivation is τ, and its denotation is obtained by applying the functional denotation to the other denotation. The derived expression is a concatenation of the expressions derived by the two sub-derivations. This rule for functional application is formally stated below.

(10) a. If Δ_1 is a derivation of an expression ε_1 of type $(\sigma\tau)$ and Δ_2 is a derivation of an expression ε_2 of type σ, then $[\Delta_1\ \Delta_2]$ (respectively, $[\Delta_2\ \Delta_1]$) is a derivation of the expression $\varepsilon_1\ \varepsilon_2$ (respectively, $\varepsilon_2\ \varepsilon_1$) of type τ.
b. For every intended model $\mathcal{M}$, $[\![[\Delta_1\ \Delta_2]]\!]^{\mathcal{M}} = [\![[\Delta_2\ \Delta_1]]\!]^{\mathcal{M}} = [\![\Delta_1]\!]^{\mathcal{M}}([\![\Delta_2]\!]^{\mathcal{M}})$.

For example, the noun phrase *every king* has the derivation [*every king*] of type $(et)t$ over the lexicon in Table 1. The denotation of this derivation in an intended model $\mathcal{M} = \langle \mathcal{F}, I \rangle$ is the generalized quantifier expressible by the lambda-term $\lambda B_{et}.\forall x_e[I_{\mathcal{M}}(king)(x) \rightarrow B(x)]$.

The toy grammar introduced so far does not allow to derive all grammatical strings over the lexicon in Table 1. For instance, a transitive sentence like (11) below is not derivable by the rules introduced so far.

(11) A queen kissed every king.

Similarly, intersective modification with adjectives (e.g. *bald king* in (5b)), is not treated by the assumptions introduced so far. In order to deal with such examples, without complicating too much the introduction of our proposed grammatical architecture, we introduce some phonologically-silent lexical items. These are introduced in Table 2 as an *ad hoc* extension of the lexicon from Table 1. Note that all these operators are treated as logical constants.

word α	**Type**	$\{\{\alpha\}\}^{\mathcal{F}}$	λ-term
ε_{ONS}	$(e(et))(((et)t)(et))$	{**ons**}	$\lambda R_{e(et)}\lambda F_{(et)t}\lambda x_e.F(\lambda y_e.R(y)(x))$ mapping a binary predicate between entities to a binary predicate between entities and quantifiers (the quantifier taking narrow scope)
ε_{OWS}	$(((et)t)(et))$ $(((et)t)(((et)t)t))$	{**ows**}	$\lambda R_{(((et)t)(et))}\lambda F_{(et)t}\lambda Q_{(et)t}.$ $F(\lambda y_e.Q(\lambda x_e.R(\lambda A_{et}.A(y))(x)))$ mapping a binary predicate between entities and quantifiers to a binary predicate between quantifiers (the object quantifier taking wide scope)
$\varepsilon_{\text{lift}}$	$e((et)t)$	{**lift**}	$\lambda x_e\lambda A_{et}.A(x)$ lifting an entity to a quantifier
ε_{adj}	$(et)((et)(et))$	{**adj**}	$\lambda A_{et}\lambda B_{et}\lambda x_e.A(x) \wedge B(x)$ mapping a set to an intersective modifier

Table 2: Extending the lexicon from Table 1 with empty words as type shifting operators.

The operation of the empty words ε_{ONS} and ε_{OWS} can be demonstrated with sentence (11) above. There are two different ways to analyze this sentence. The *object narrow scope* (ONS) analysis of (11) means that there is a specific queen that kissed every king. The *object wide scope* (OWS) analysis of (11), on the other hand, states that for every king there was a queen that kissed him. Over our extended lexicon there are two different derivations for sentence (11). One of them, (12a) below, uses the ε_{ONS} operator and produces the ONS interpretation. The other, in (12b), uses both ε_{ONS} and ε_{OWS} to generate the OWS interpretation. This use of (extensional) operators on predicates in order to derive ONS and OWS analyses essentially follows the (intensional) operators proposed in Hendriks (1993).

(12) a. [[*a queen*] [[ε_{ONS} *kissed*] [*every king*]]]
b. [[*a queen*] [[ε_{OWS} [ε_{ONS} *kissed*]] [*every king*]]]

The empty word ε_{adj} can be used to shift the set denoted by an intension-insensitive adjective to an intersective function of type $(et)(et)$. The latter can modify the set denoted by a common noun in the usual way. For example, derivation (13b) of sentence (5b) is true whenever derivation (13a) of sentence (5a) is true, in accordance with the observed entailment from (5a) to (5b).

(13) a. [[*every king*] *smiled*]
b. [[*every* [[ε_{adj} *bald*] *king*]] *smiled*]

Last, the empty word $\varepsilon_{\text{lift}}$ denotes the operator that shifts an entity to a generalized quantifier. This is necessary for treating a sentence like *Mary and every queen smiled*, because two expressions can be coordinated only if they are of the same semantic type.

A second derivation rule in our example introduces the conjunctive word *and*. Recall that one of our main concerns is to enable a coordination of intension-insensitive words with intension-sensitive words, like in *Mary sought, found and ate a fish*. The syncategorematic introduction of conjunction that we use here is convenient for the sake of exposition of our intensionalization procedure, as it does not require adding a polymorphic entry to the lexicon or using several entries for an arbitrary number of types. Similar rules of derivation can be formulated for other Boolean words such as *or* and *not*. For the purposes of this paper, however, it is enough to restrict attention to the conjunctive *and*. In order for two expressions to be conjoinable, they must be of the same type. Furthermore, this type must be *Boolean* (or *t-ending*). The definition of *Boolean types* in (14) below assumes a general set of basic types, which include the type t of truth values. In an extensional system, the set of basic types is $\{e,t\}$, but as soon as we start talking about intensional frames we shall add a basic type s for possible worlds.

(14) Let $\mathcal{B}$ be a finite set of basic types such that $t \in \mathcal{B}$. A type σ over $\mathcal{B}$ is *Boolean* iff either $\sigma = t$ or $\sigma = (\sigma_1\sigma_2)$ for a Boolean type σ_2.

Note that if σ is a Boolean type, then there are $n \geq 0$ and types $\sigma_1,\ldots\sigma_n$ s.t. $\sigma = (\sigma_1 \ldots (\sigma_n t)\ldots)$.

In the derivation rule for conjunction that we introduce in (16) below, we use the well-known *Generalized Conjunction* operator from Partee and Rooth (1983). This operator, denoted here '$\sqcap$', is recursively defined in (15) for Boolean types σ, where '$\wedge$' is the standard propositional conjunction.

(15) $$\sqcap_{\sigma(\sigma\sigma)} = \begin{cases} \wedge & \sigma = t \\ \lambda X_\sigma \lambda Y_\sigma \lambda Z_{\sigma_1}.\sqcap_{\sigma_2(\sigma_2\sigma_2)}(X(Z))(Y(Z)) & \sigma = (\sigma_1\sigma_2) \end{cases}$$

(16) a. If Δ_1 is a derivation of an expression ε_1 and Δ_2 is a derivation of an expression ε_2, both of a Boolean type σ, then $[\Delta_1$ and $\Delta_2]$ is a derivation of type σ of the expression ε_1 *and* ε_2.

b. For every intended model $\mathcal{M}$, $[\![[\Delta_1 \text{ and } \Delta_2]]\!]^{\mathcal{M}} = \sqcap_{\sigma(\sigma\sigma)}([\![\Delta_1]\!]^{\mathcal{M}})([\![\Delta_2]\!]^{\mathcal{M}})$.

Having defined a setting for an extensional semantics, we can now introduce our proposed intensionalization procedure for this setting and discuss its implications.

4 Intensionalization

In extensional systems like the one introduced in the previous section, all lexical items are treated as intension-insensitive. Our main aim is to extend such a system with intension-sensitive lexical items such as the TVs *seek* and *need* or the adjectives *alleged* and *fake*. Suppose we try to do that without any further modification in the extensional system. Clearly, this cannot work. Let us remind ourselves why, considering a well-known example.

(17) a. Mary seeks a doctor.

b. Mary seeks a lawyer.

When we say that a TV like *seek* in (17), or the more colloquial verb *look for*, is intension-sensitive, we mean that a sentence like (17a) can be true while sentence (17b) is false, even in a situation in which the doctors and the lawyers are the same. It can be that Mary is sick and is looking for a doctor without ever recognizing that finding a doctor would also lead her to a lawyer. There is no easy way to represent this difference between (17a) and (17b) with an extensional semantics like the one that we described in the last section. Any intended model in which the nouns *doctor* and *lawyer* denote the same set of entities would assign the same truth value to (17a) and (17b).

To enable the introduction of intension-sensitive lexical items like *seek* into the system, we should let their arguments denote intensional objects. In this section we take the first step towards the introduction of such intension-sensitive lexical items into the system by introducing our proposed semantics of *intensionalization*. With this procedure we will be able to modify the types and meanings of lexical items in an extensional system like the one introduced in Section 3, so that the resulting system is equivalent to the original extensional system, and at the same time can be extended to a properly intensional semantics by only adding intension-sensitive items to its lexicon.

The intensionalization that we propose follows Van Benthem's (1988) typing recipe. We denote the set of all functional types over e, s and t by $\mathcal{T}_{\text{in}}$ – the same definition as in (4), but with $\{e,s,t\}$ replacing $\{e,t\}$. We denote the intensionalization of an extensional type $\sigma \in \mathcal{T}_{\text{ex}}$ by $\ulcorner\sigma\urcorner \in \mathcal{T}_{\text{in}}$. The type $\ulcorner\sigma\urcorner$, following Van Benthem, is obtained from σ by substituting every occurrence of t by (st). For example, $\ulcorner t\urcorner = st$, which means that the type of truth-values is intensionalized to the type of propositions. Another example is $\ulcorner(et)\urcorner = e(st)$. Thus, the type of (characteristic functions of) sets of entities is intensionalized to the type of *properties*.[3] Further, rather similarly to PTQ, the intensionalized version of the type $(et)t$ of extensional quantifiers is $(e(st))(st)$ – the type of functions that map properties to propositions.

Using this global type-change recipe, we should now intensionalize the *meanings* of lexical items so that we end up with a system equivalent to the original one. Recall that the meaning of a lexical item α was defined using a functional $\{\!\{\alpha\}\!\}$ that maps an extensional frame $\mathcal{F}$ to a subset $\{\!\{\alpha\}\!\}^{\mathcal{F}}$ of $\bigcup\mathcal{F}$. $\{\!\{\alpha\}\!\}^{\mathcal{F}}$ consists of all and only those elements of $\bigcup\mathcal{F}$ that are considered as legitimate interpretations of α. The intensionalization of $\{\!\{\alpha\}\!\}$ will be a similar functional

[3]Note that there is no substantial difference between the type $e(st)$ that we assume here for properties and the more standard type for one-place properties – $s(et)$: the domains for these types are isomorphic.

$\{\{\alpha\}\}_{\text{in}}$, which maps an *intensional* frame to the set of legitimate interpretations of α in it. The formal definition of an intensional frame is similar to the definition of an extensional frame in (6), with the additional domain for possible worlds.

(18) Given a nonempty sets E of entities and W of possible worlds, an *(E,W-based) intensional frame* is the set $\mathcal{F} = \{D_\sigma : \sigma \in \mathcal{T}_{\text{in}}\}$, where $D_e = E$, $D_s = W$, $D_t = \{0,1\}$ and $D_{(\tau\sigma)} = D_\sigma^{D_\tau}$

To facilitate the intensionalization of meanings, we make a simplifying assumption: that each lexical item is either a logical or a nonlogical constant. The motivation is to preserve (non)logicality under intensionalization: extensional logical constants will be mapped to intensional logical constants; extensional nonlogical constants will be mapped to intensional nonlogical constants. The question of how to treat other kinds of lexical items is more complicated, and is left for future research.[4]

For an extensional nonlogical constant to remain nonlogical in the intensional frame, all we should do is restrict its interpretation to the whole domain of its intensionalized type. More precisely, intensionalization of nonlogical constants goes as follows:

(19) If $\{\{\alpha\}\}^{\mathcal{F}} = D_\sigma$ for every extensional frame $\mathcal{F}$, then $\{\{\alpha\}\}^{\mathcal{F}}_{\text{in}} = D_{\ulcorner\sigma\urcorner}$ for every intensional frame $\mathcal{F}$.

For the intensionalization of logical constants we need to define a mapping from extensional domains to intensional domains. To see why, consider any logical constant α of an arbitrary extensional type σ. Because we want the intensionalization of α to be a logical constant as well, it follows that in every E,W-based intensional frame $\mathcal{F}$ there should be some object $g \in D_{\ulcorner\sigma\urcorner}$ such that $\{\{\alpha\}\}^{\mathcal{F}}_{\text{in}} = \{g\}$. It is expected that this object g is systematically derived from the unique interpretation of α in the corresponding E-based extensional frame $\mathcal{F}'$ using some mapping from $D_\sigma \in \mathcal{F}'$ to $D_{\ulcorner\sigma\urcorner} \in \mathcal{F}$.

To motivate our proposed definition of this mapping, consider for example the determiner *every* as appearing in the extensional lexicon from Table 1. For this determiner, we need to map the object **every** $\stackrel{def}{=} \lambda A_{et}\lambda B_{et}.\forall x_e[A(x) \rightarrow B(x)]$ in $D_{(et)((et)t)}$ to a unique member of the intensional domain $D_{\ulcorner(et)((et)t)\urcorner}$. The intensional denotation that we are after is similar to the denotation of *every* in PTQ.[5] This is the function that when applying to two properties P and Q, returns the proposition that is *true* in a world w just in case the predicate *extensions* in w of P and Q satisfy the containment requirement of **every**. In symbols, we would like to end up with $L(\mathbf{every}) = \lambda w_s\lambda P_{e(st)}\lambda Q_{e(st)}.\mathbf{every}((P)^w)((Q)^w)$, where $(P)^w$ is $\lambda x_e.P(w)(x)$ – the extension of the property P in a given index w, and similarly for $(Q)^w$.

Our next aim is to generalize this relatively simple example to any logical constant of any type. From the example with *every*, which uses w-extensions properties, we conclude that for the definition of the intensionalization mapping $L(\cdot)$ we first need to define a dual *extensionalization mapping* from intensional domains to extensional domains. To facilitate the definition of this mapping, we follow a tentative proposal in Van Benthem (1988), and restrict our attention to the *quasi-relational* types of Muskens (1989). The set $\mathcal{T}_{\text{qr}}$ of quasi-relational types is defined as below, using the set $\mathcal{T}_e$ of *e-based types* – the subset of $\mathcal{T}_{\text{ex}}$ consisting only of e-occurrences.

[4] Makoto Kanazawa (p.c.), based on work in progress with Philippe de Groote and Reinhard Muskens, suggests a way of intensionalizing meanings that unlike the present proposal does not need to stipulate different treatments for logical constants and non-logical constants. Furthermore, Kanazawa et al's intensionalization may be preferable to ours in some other important respects. We are currently studying the implications of their proposal.

[5] Determiners are introduced syncategorematically in Montague (1973), whereas here they are part of the lexicon. But this hardly matters for the semantic analysis.

(20) $\mathcal{T}_{\text{qr}} \subset \mathcal{T}_{\text{ex}}$ is the set of extensional types that satisfies $t \in \mathcal{T}_{\text{qr}}$, and $(\sigma_1\sigma_2) \in \mathcal{T}_{\text{qr}}$ if $\sigma_1 \in \mathcal{T}_e \cup \mathcal{T}_{\text{qr}}$ and $\sigma_2 \in \mathcal{T}_{\text{qr}}$.

In the sequel we shall make use of the observation that every quasi-relational type σ can be written as $(\sigma_1 \cdots (\sigma_n t) \cdots)$, for some $n \geq 0$, where each σ_i is either e-based or quasi-relational. In this case the domain D_σ is isomorphic to the cartesian product $D_{\sigma_1} \times \cdots \times D_{\sigma_n}$. In some cases it may be illuminating to consider a function in D_σ as the corresponding relation, and we shall do so without further notice.

The assumption that all items in an extensional lexicon are of quasi-relational types facilitates the following definition of a w-extension. In this definition, w is an arbitrary possible world in W and σ is either a quasi-relational type in $\mathcal{T}_{\text{qr}}$ or an e-based type in $\mathcal{T}_e$.

(21) **The w-extension of** $g \in D_{\ulcorner\sigma\urcorner}$:

a. If $\sigma \in \mathcal{T}_e$ then $(g)^w = g$;

b. if $\sigma = t$ then $(g)^w = g(w)$;

c. if $\sigma = (\sigma_1 \cdots (\sigma_n t) \cdots)$, $n \geq 1$, then

$$(g)^w = \lambda x^1_{\sigma_1} \cdots \lambda x^n_{\sigma_n}.\exists z_1 \cdots \exists z_n. \bigwedge_{i=1}^{n} ((z_i)^w = x_i) \wedge g(z_1) \cdots (z_n)(w)$$

Thus, a tuple $\langle x_1, \ldots, x_n \rangle$ is in the w-extension of an intensional relation g, iff there is a tuple $\langle z_1, \ldots, z_n, w \rangle$ in g such that the w-extensions of the z_is are the x_is, respectively.

The intensionalization mapping $L(\cdot)$ is now defined as follows, where σ is either a quasi-relational type or an e-based type.

(22) **The intensionalization of** $f \in D_\sigma$:

a. if $\sigma \in \mathcal{T}_e$ then $L(f) = f$;

b. if $\sigma = (\sigma_1 \cdots (\sigma_n t) \cdots)$, $n \geq 0$, then

$$L(f) = \lambda x^1_{\ulcorner\sigma_1\urcorner} \cdots \lambda x^n_{\ulcorner\sigma_n\urcorner} \lambda w_s. f((x^1)^w) \cdots ((x^n)^w)$$

Thus, a tuple $\langle x_1, \ldots, x_n, w \rangle$ is in the intensionalization of a relation f, iff the w-extensions of $x_1, \ldots, x_n$ are in f.

We have now completed the introduction of the proposed intensionalization process. Let us summarize it.

1. We start with an extensional system (defined in Section 3), in which every lexical item is:

   (a) of a quasi-relational type or an e-based type;

   (b) either a nonlogical constant or a logical constant.

2. We modify the type of each lexical item by changing t to (st) everywhere.

3. We modify the meaning of each lexical item in the following way:

   (a) A nonlogical constant remains a nonlogical constant.

(b) A logical constant with a denotation $f \in D_\sigma$ is intensionalized to a logical constant with a denotation $L(f) \in D_{\ulcorner\sigma\urcorner}$.[6]

The definitions of intended models and derivations in intensionalized systems are just like the corresponding definitions in extensional systems (cf. (8), (9), (10) and (16)). An *entailment* relation is defined between derivations of sentences over an extensional lexicon in the familiar way: Δ_1 entails Δ_2 (both of type t) iff $[\![\Delta_1]\!]^{\mathcal{M}} \leq [\![\Delta_2]\!]^{\mathcal{M}}$ for every intended model $\mathcal{M}$. Over an intensionalized lexicon, we say that Δ_1 entails Δ_2 (both of type (st)) iff $[\![\Delta_1]\!]^{\mathcal{M}}(w) \leq [\![\Delta_2]\!]^{\mathcal{M}}(w)$ for every intended model $\mathcal{M}$ and every $w \in W$.

Recall that our goal in developing an intensionalization procedure is to end up with a *sound* process that does not change the truth-conditional behavior of the extensional system. In Ben-Avi (2007) it is proved that the above intensionalization process is sound in the sense that it preserves entailments between derivations of sentences. For the proof, we add one more restriction on the types of nonlogical constants. We assume that if this type is quasi-relational, then all its arguments are of an e-based type. For example, $(e(et))$ or $((ee)t)$ are legal types for a nonlogical constant, while $(et)((et)t)$ or $t(tt)$ are not. This restriction reflects our assumption that intension-insensitive relational nonlogical constants are basically relations between entities, or functions defined in terms of which.

To see the benefits of the intensionalization procedure, let us get back to the toy extensional lexicon of the previous section. The lexicon in Table 3 is the result of intensionalizing the extensional lexicon in Tables 1 and 2. For a logical constant α in this table, we write its constant interpretation as $L(f)$, where f is its constant extensional interpretation. A routine but somewhat tedious calculation shows that the relevant functions are as follows:

$$\begin{aligned}
L(\textbf{every}) &= \lambda\mathcal{A}_{e(st)}\lambda\mathcal{B}_{e(st)}\lambda w_s.\forall x_e[\mathcal{A}(x)(w) \rightarrow \mathcal{B}(x)(w)] \\
L(\textbf{some}) &= \lambda\mathcal{A}_{e(st)}\lambda\mathcal{B}_{e(st)}\lambda w_s.\exists x_e[\mathcal{A}(x)(w) \wedge \mathcal{B}(x)(w)] \\
L(\textbf{ons}) &= \lambda\mathcal{R}_{e(e(st))}\lambda\mathcal{F}_{(e(st))(st)}\lambda x_e\lambda w_s.(\mathcal{F})^w(\lambda y_e.\mathcal{R}(y)(x)(w)) \\
L(\textbf{ows}) &= \begin{array}{l}\lambda\mathcal{R}_{(((e(st))(st))(e(st)))}\lambda\mathcal{F}_{(e(st))(st)}\lambda Q_{(e(st))(st)}\lambda w_s. \\ \quad(\mathcal{F})^w(\lambda y_e.(Q)^w(\lambda x_e.(\mathcal{R})^w(\lambda A_{et}.A(y))(x)))\end{array} \\
L(\textbf{lift}) &= \lambda x_e\lambda\mathcal{A}_{e(st)}.\mathcal{A}(x) \\
L(\textbf{adj}) &= \lambda\mathcal{A}_{e(st)}\lambda\mathcal{B}_{e(st)}\lambda x_e\lambda w_s.\mathcal{A}(x)(w) \wedge \mathcal{B}(x)(w)
\end{aligned}$$

One simple example that demonstrates the soundness of the intensionalization process is the entailment between the derivations (13a) and (13b), which is respected both by the extensional system and by its intensionalized version. Another example involves the derivations (12a) and (12b) of the sentence *A queen kissed every king*. We leave it for the reader to verify (or to consult Ben-Avi 2007) that the ONS derivation (12a) of this sentence entails the OWS derivation (12b) also in the intensionalized system.

5 Extending the intensionalized system

Our main reason to develop a sound intensionalization procedure is to allow a simple introduction of intension-sensitive entries into the lexicon, without any further modification in the intensionalized system. Recall that our main argument against the typing strategy of Heim and Kratzer (1998) was that it prevents ITVs to be of the same type as that of ETVs. On the other

[6] More formally, let α be a logical constant. Let $\mathcal{F}$ be an E,W-based intensional frame, and $\mathcal{F}'$ the corresponding E-based extensional frame. We define $\{\!\{\alpha\}\!\}^{\mathcal{F}}_{\text{in}} = \{L(f)\}$, where f is the single element in $\{\!\{\alpha\}\!\}^{\mathcal{F}'}$.

word α	Type	$\{\{\alpha\}\}^{\mathcal{F}}$
Mary, John,...	e	D_e
red, sick,...	$e(st)$	$D_{e(st)}$
king, queen,...	$e(st)$	$D_{e(st)}$
smile,...	$e(st)$	$D_{e(st)}$
kiss,...	$e(e(st))$	$D_{e(e(st))}$
every	$(e(st))((e(st))(st))$	$\{L(\mathbf{every})\}$
a	$(e(st))((e(st))(st))$	$\{L(\mathbf{some})\}$
ε_{ONS}	$(e(e(st)))(((e(st))(st))(e(st)))$	$\{L(\mathbf{ons})\}$
ε_{OWS}	$(((e(st))(st))(e(st)))(((e(st))(st))(((e(st))(st))(st)))$	$\{L(\mathbf{ows})\}$
$\varepsilon_{\text{lift}}$	$e((e(st))(st))$	$\{L(\mathbf{lift})\}$
ε_{adj}	$(e(st))((e(st))(e(st)))$	$\{L(\mathbf{adj})\}$

Table 3: An intensionalization of the extensional lexicon from Tables 1 and 2.

hand, the typing strategy of Van Benthem (1988) that we have followed enables a simple and natural introduction of intension-sensitive words like *seek* and *need* without any further modification in the system, while allowing these TVs to be of the same type as (intensionalized) ETVs like *kiss*. In this section we demonstrate this by integrating ITVs into the lexicon of Table 3. As we shall see, *de dicto*/*de re* ambiguities and coordinations of ITVs with ETVs are treated without any further modifications of the system.

One simple way to add ITVs to the lexicon from Table 3 is to let them denote nonlogical constants of type $((e(st))(st))(e(st))$. By this we implement a treatment of ITVs like in PTQ, where the object of such verbs is assumed to denote an *intensional quantifier*. It should be emphasized, however, that this is not an assumption of our intensionalization procedure but a simple way to accommodate ITVs into the toy lexicon that we are using for exemplification. The treatment of *de dicto*/*de re* ambiguities under this technique is demonstrated in (23) below, where sentence (23a) has the two derivations (23b) and (23c).

(23) a. Mary sought a king.
b. [*Mary* [*sought* [*a king*]]]
c. [[$\varepsilon_{\text{lift}}$ *Mary*] [[ε_{OWS} *sought*] [*a king*]]]

In an intended model $\mathcal{M}$, (23b) is interpreted as (24b), and (23c) – as (24c). Interpretation (24b) represents a *de dicto* reading of sentence (23a), whereas (24c) represents a *de re* reading of the sentence.

(24) b. $(I_{\mathcal{M}}(seek))(\lambda\mathcal{B}_{e(st)}\lambda w_s.\exists y_e[(I_{\mathcal{M}}(king))(y)(w)\wedge\mathcal{B}(y)(w)])(I_{\mathcal{M}}(Mary))$
c. $\lambda w_s\exists y_e[(I_{\mathcal{M}}(king))(y)(w)\wedge(I_{\mathcal{M}}(seek))^w(\lambda A_{et}.A(y))(I_{\mathcal{M}}(Mary))]$

Note that the *de re* interpretation (24c) is created by the same mechanism that creates object wide scope interpretations in the extensional system (cf. (12b)). This property of the system that we describe is in accordance with Montague's strategy in PTQ, where the *quantifying in* mechanism is responsible both for the creation of scope ambiguities and for the creation of *de dicto*/*de re* ambiguities. However, in distinction with the proposals by Montague, Hendriks (1993) and others, intensionalization spares us the need to define an intricate intensional version of the scope shifting mechanism.

The typing strategy that we follow also facilitates a straightforward treatment of coordinations between ITVs and ETVs. For example, the sentence *Mary sought and kissed a king* has the two derivations in (25). Derivation (25a) represents the reading in which Mary sought a king *de dicto*, while derivation (25b) represents the reading in which Mary sought a king *de re*.

(25) a. [*Mary* [[*sought* and [ε_{ONS} *kissed*]] [*a king*]]]

a. [*Mary* [[[ε_{OWS} *sought*] and [ε_{OWS} [ε_{ONS} *kissed*]]] [*a king*]]]

As mentioned above, our intensionalization process is not restricted to the Montagovian treatment of ITVs. This is demonstrated in Ben-Avi (2007), where the same intensionalization process is applied to the treatment of Zimmermann (1993), who assumes that an ITV takes a property as its object argument. Ben-Avi (2007) demonstrates that supporting Zimmermann's treatment in an extensional system resonates well with the widely-assumed process of *semantic incorporation* (Van Geenhoven 1998, Van Geenhoven and McNally 2005). In this process, an ETV can compose with predicative indefinites by way of existential quantification. Formally, the extensional incorporation operator on ETVs is defined as follows.

(26) $INC = \lambda R_{e(et)} \lambda P_{et}.\lambda y_e.\exists x_e[R(x)(y) \wedge P(x)]$

This definition, once intensionalized, also allows a simple meaning derivation for coordinations like *sought and kissed a king*: using Zimmermann's assumption, the verb *seek* takes the property denotation of *a king*, whereas the intensionalized incorporation operator allows the verb *kiss* to conjoin with *seek* and take the same property as argument. Arguably, this account is as natural as the derivations in (25), based on the Montagovian treatment of ITVs. We take it that this simplicity further supports our claim that intensionalization should be an inseparable part of any comprehensive theory of intensionality.

6 Conclusions

So far, the study of intensionalization has not been a central part of the massive semantic literature on intensionality. In this paper we argued that such a process is necessary if we want to understand better the separation between extensional semantics and intensional semantics. We propose that intensionality phenomena are lexically driven, and that it is mostly this fact that allowed Montague to use essentially extensional mechanisms for treating long-standing puzzles like *de dicto*/*de re* ambiguities. The study of intensionalization provides a missing link in this story: it explains what is "extensional" in those mechanisms. By doing that, it articulates the lexically-driven nature of intensionality phenomena in natural language.

References

Barker, C.: 2002, Continuations and the nature of quantification, *Natural Language Semantics* **10**, 211–242.

Ben-Avi, G.: 2007, Types and Meanings in Quantifier Scope, Selection and Intensionality. PhD Thesis in preparation, Technion.

Gallin, D.: 1975, *Intensional and Higher-Order Modal Logic*, Vol. 19 of *North-Holland Mathematics Studies*, North-Holland, Amsterdam.

Heim, I. and Kratzer, A.: 1998, *Semantics in Generative Grammar*, Blackwell Publishers, Oxford.

Hendriks, H.: 1993, *Studied Flexibility: categories and types in syntax and semantics*, PhD thesis, University of Amsterdam.

Kamp, H. and Partee, B.: 1995, Prototype theory and compositionality, *Cognition* **57**, 129–191.

Keenan, E.: 2000, Logical objects, *in* C. A. Anderson and M. Zeleny (eds), *Logic, Meaning and Computation: Essays in Memory of Alonzo Church*, Kluwer, Amsterdam, pp. 151–183.

Keenan, E. and Faltz, L.: 1985, *Boolean Semantics for Natural Language*, D. Reidel, Dordrecht.

Montague, R.: 1973, The proper treatment of quantification in ordinary English, *in* J. Hintikka, J. Moravcsik and P. Suppes (eds), *Approaches to Natural Languages: proceedings of the 1970 Stanford workshop on grammar and semantics*, D. Reidel, Dordrecht. Reprinted in R. Thomason, editor (1974), *Formal Philosophy: selected papers of Richard Montague*, Yale, New Haven.

Muskens, R.: 1989, A Relational Formulation of the Theory of Types, *Linguistics and Philosophy* **12**, 325–346.

Partee, B. and Rooth, M.: 1983, Generalized conjunction and type ambiguity, *in* R. Bauerle, C. Schwarze and A. von Stechow (eds), *Meaning, Use and Interpretation of Language*, De Gruyter, Berlin.

Shan, C. C.: 2001, Monads for natural language semantics, *in* K. Striegnitz (ed.), *Proceedings of the ESSLLI-2001 student session*, pp. 285–298. Downloadable from `http://arxiv.org/pdf/cs.CL/0205026`.

Sher, G.: 1991, *The Bounds of Logic: a generalized viewpoint*, MIT Press, Cambridge, Massachusetts.

Van Benthem, J.: 1988, Strategies of intensionalization, *in* I. Bodnár, A. Máté and L. Pólos (eds), *Intensional Logic, History of Philosophy, and Methodology: To Imre Ruzsa on the occasion of his 65th birthday*, Department of Symbolic Logic, Eötvös University, Budapest, pp. 41–59.

Van Geenhoven, V.: 1998, *Semantic Incorporation and Indefinite Descriptions: semantic and syntactic aspects of noun incorporation in West Greenlandic*, CSLI Publications, Stanford.

Van Geenhoven, V. and McNally, L.: 2005, On the property analysis of opaque complements, *Lingua* **115**, 885–914.

Winter, Y.: 2006, Multiple coordination: meaning composition vs. the syntax-semantics interface. Unpublished ms., downloadable from `http://www.cs.technion.ac.il/~winter`.

Zimmermann, T. E.: 1993, On the proper treatment of opacity in certain verbs, *Natural Language Semantics* **11**, 149–179.