

# A Note on Intensionalization\*

Makoto Kanazawa

December 27, 2009

## 1 Introduction

Ben-Avi and Winter (2007) proposed a procedure for *intensionalization*, a method for mapping an object of an “extensional” type (i.e., type based on atomic types  $e$  and  $t$ ) into an object of a corresponding “intensional” type (based on  $e$ ,  $t$ , and  $s$ ). They used this mapping to convert an extensional semantics for a fragment of natural language, where all lexical items have extensional denotations, into an intensional semantics which assigns “intensionalized” denotations to the same lexical items and which is capable of accommodating new lexical items with “truly intensional” denotations without any change in the grammar.

This note presents an alternative intensionalization procedure that is more general than Ben-Avi and Winter’s (2007) in two respects. First, Ben-Avi and Winter’s procedure is only applicable to objects of *quasi-relational* types. This limitation stems from the fact that their type change scheme replaces  $t$  by  $s \rightarrow t$ , but leaves  $e$  unchanged. In contrast, the present scheme uniformly replaces each atomic type  $a$  in an extensional type  $\alpha$  by  $s \rightarrow a$  to produce an intensional type  $\bar{\alpha}$ ; this uniformity allows the intensionalization procedure to be defined at arbitrary extensional types. Second, unlike Ben-Avi and Winter’s (2007) method, where the input is a single extensional object, the method defined here takes a set of extensional objects of type  $\alpha$  *parametrized* to objects of type  $s$ , and returns an object of type  $\bar{\alpha}$ . This allows a construction of an intensional model from a class of extensional models viewed as possible worlds; as a result of this, there is no need to stipulate a sharp distinction between lexical items that are “logical

---

\*This note is partly based on discussions I had with Philippe de Groote and Reinhard Muskens in Tokyo in early 2007, after all three of us had heard Yoad Winter present Gilad Ben-Avi and his work on intensionalization (see Ben-Avi and Winter 2007). In particular, the definition of the intensionalization and extensionalization combinators in Section 2 is originally due to Philippe. In this sense, this note should be considered a presentation of joint work. The note was first written immediately following our discussions, and was expanded with some additional material in May 2007. It was distributed among a small group of people, but was not made publicly available. The five of us (that is, Philippe, Reinhard, Yoad, Gilad, and me) are planning to coauthor a paper incorporating some of the results in this note, but in the meantime, it is probably worthwhile making the note publicly available, especially since a near-final draft (dated December 2009) of Jan van Eijck and Christina Unger’s forthcoming textbook (van Eijck and Unger, to appear) refers to the May 2007 version of this note. In addition to some reorganization of the material from the previous versions, the present version adds new material concerning preservation of validity of schemata (Section 4).

constants” and others whose denotations are unconstrained, as in Ben-Avi and Winter’s construction.

An important desideratum for intensionalization is that the resulting intensional semantics is “conservative” over the original extensional semantics. This desideratum is somewhat misstated in Ben-Avi and Winter’s (2007) paper for a reason related to their treatment of “non-logical” lexical items. They demand that validity and consequence be preserved in moving from extensional models to intensional models. Since their treatment of non-logical lexical items in effect amounts to treating them as free variables, this is an unreasonably strong requirement, one that easily fails in the “intensionalization” of as simple a language as the language of propositional logic augmented with a non-logical constant of type  $t \rightarrow t$ . In order to satisfy this desideratum, Ben-Avi and Winter (2007) had to severely restrict the admissible types of non-logical lexical items.

In the present construction of intensional models from classes of extensional models, the desideratum is, simply put, that the truth conditions of sentences in the extensional language be preserved. More precisely, this means that the truth value of a sentence  $\varphi$  in an extensional model  $M$  is to coincide with the truth value of  $\varphi$  in any intensional model constructed from extensional models in which  $M$  is the “actual world”. The preservation of validity and consequence follows as a corollary.

Even though the validity of an open formula in a class of extensional models should not be expected to be preserved in intensional models built from them, it may be of some technical interest to see to what limited extent this expectation may be satisfied. For an open formula  $\varphi$  in the extensional language to be valid in a class of intensional models, all instances of  $\varphi$ , including formulas in an expanded language containing some truly intensional items, must be valid as well. Particularly simple examples of such formulas are provided by propositional modal logic, where a propositional variable in a tautology may be replaced by any modal formula to produce a valid modal formula. This is a rather trivial case, but there are other more interesting cases. In the final section of this note, I define a class of *safe* open formulas, and show that for those formulas, validity in extensional models guarantees validity in intensional models.

It is not entirely clear to me whether a fully general intensionalization procedure such as the one given here is called for in natural language semantics. It seems to me that the usual practice in linguistics is to introduce as few instances of  $s$  as is required for proper linguistic analysis, rather than putting an  $s$  in front of every occurrence of  $e$  and  $t$ . Nevertheless, the fact that such a general method exists, with a definition so natural as to look almost inevitable,<sup>1</sup> may interest some researchers interested in the formal side of semantics.

---

<sup>1</sup>I fondly recall that when Philippe de Groote first started to write down his definition on the board during our initial discussion, he described it as “almost stupid”.

## 2 Intensionalization

We begin with some necessary definitions. The set of *types* over a set  $A$  of *atomic types* is defined as follows:<sup>2</sup>

- Atomic types:  $a \in A$ .
- Types over  $A$ :  $\alpha ::= a \mid \alpha_1 \rightarrow \alpha_2$ .

The type constructor  $\rightarrow$  is assumed to be right-associative, so that  $\alpha \rightarrow \beta \rightarrow \gamma$  stands for  $\alpha \rightarrow (\beta \rightarrow \gamma)$ . We let  $\alpha^n \rightarrow \beta$  abbreviate  $\alpha \rightarrow \dots \rightarrow \alpha \rightarrow \beta$ , with “ $\alpha \rightarrow$ ” repeated  $n$  times.

Given  $(D_a)_{a \in A}$ , the domain  $D_\alpha$  of objects of type  $\alpha$  is defined for each type  $\alpha$  over  $A$  by recursion:

$$D_{\alpha_1 \rightarrow \alpha_2} = D_{\alpha_2}^{D_{\alpha_1}}.$$

In concrete examples, we often assume  $A = \{e, t\}$  and  $D_t = \{1, 0\}$ . The elements of  $D_e$  are *individuals* and 1 and 0 are *truth values*.

Let  $s$  be a new atomic type not in  $A$ . Elements of  $D_s$  are called *indices* or *possible worlds*. The *intensionalization* of type  $\alpha$ , written  $\bar{\alpha}$ , is defined by

$$\bar{\alpha} = \alpha[a := (s \rightarrow a)]_{a \in A}.$$

We use simply typed  $\lambda$ -calculus (à la Church) as our metalanguage to denote objects in  $D_\alpha$ , for types  $\alpha$  over  $A \cup \{s\}$ . The type of a variable is indicated by a superscript at its first occurrence. If  $\psi$  is a  $\lambda$ -term of type  $\alpha \rightarrow \beta$  and  $\chi$  is a  $\lambda$ -term of type  $\alpha$ , then the application  $\psi\chi$  is a  $\lambda$ -term of type  $\beta$  and denotes the object in  $D_\beta$  that is the value of  $\psi$  on argument  $\chi$ . If  $\psi$  is a  $\lambda$ -term of  $\beta$ , then the  $\lambda$ -abstract  $\lambda x^\alpha.\psi$  is a  $\lambda$ -term of type  $\alpha \rightarrow \beta$  and denotes the function from  $D_\alpha$  to  $D_\beta$  which maps  $x$  to  $\psi$ . The application of  $\lambda$ -terms is assumed to be left-associative, so that  $\varphi\psi\chi$  stands for  $(\varphi\psi)\chi$ . Application binds stronger than  $\lambda$ -abstraction, so that  $\lambda x^\alpha.\psi\chi$  means  $\lambda x^\alpha.(\psi\chi)$ , rather than  $(\lambda x^\alpha.\psi)\chi$ . A sequence of  $\lambda$ 's is collapsed into one, so that  $\lambda x_1^{\alpha_1} \dots x_n^{\alpha_n}.\psi$  abbreviates  $\lambda x_1^{\alpha_1} \dots \lambda x_n^{\alpha_n}.\psi$ .

For each type  $\alpha$  over  $A$ , we define the *intensionalization* and *extensionalization* combinators at type  $\alpha$  by mutual recursion as follows:

$$\begin{aligned} \text{int}_\alpha &= \text{ext}_\alpha = \lambda x^{s \rightarrow \alpha}.x, \\ \text{int}_{\alpha \rightarrow \beta} &= \lambda x^{s \rightarrow \alpha \rightarrow \beta} y^{\bar{\alpha}}. \text{int}_\beta(\lambda i^s. xi(\text{ext}_\alpha y i)), \\ \text{ext}_{\alpha \rightarrow \beta} &= \lambda y^{\bar{\alpha} \rightarrow \beta} j^s x^\alpha. \text{ext}_\beta(y(\text{int}_\alpha(\lambda k^s.x)))j. \end{aligned}$$

Note that  $\text{int}_\alpha$  is of type  $(s \rightarrow \alpha) \rightarrow \bar{\alpha}$  and  $\text{ext}_\alpha$  is of type  $\bar{\alpha} \rightarrow s \rightarrow \alpha$ .

**Remark 1.** If  $\alpha = \alpha_1 \rightarrow \dots \rightarrow \alpha_n \rightarrow a$ , we have

$$\begin{aligned} \text{int}_\alpha &= \lambda x^{s \rightarrow \alpha} y_1^{\bar{\alpha}_1} \dots y_n^{\bar{\alpha}_n} i^s. xi(\text{ext}_{\alpha_1} y_1 i) \dots (\text{ext}_{\alpha_n} y_n i), \\ \text{ext}_\alpha &= \lambda y^{\bar{\alpha}} j^s x_1^{\alpha_1} \dots x_n^{\alpha_n}. y(\text{int}_{\alpha_1}(\lambda k^s.x_1)) \dots (\text{int}_{\alpha_n}(\lambda k^s.x_n))j, \end{aligned}$$

**Lemma 1.** For any type  $\alpha$  over  $A$ ,  $\lambda x^{s \rightarrow \alpha}.\text{ext}_\alpha(\text{int}_\alpha x) = \lambda x^{s \rightarrow \alpha}.x$ .

<sup>2</sup>In Montague semantics,  $\alpha \rightarrow \beta$  is usually written  $\langle \alpha, \beta \rangle$ , but I prefer the notation standard in  $\lambda$ -calculus.

*Proof.* By induction on  $\alpha$ . The atomic case is obvious. For the induction step, we have

$$\begin{aligned}
& \lambda x^{s \rightarrow \alpha \rightarrow \beta}. \text{ext}_{\alpha \rightarrow \beta}(\text{int}_{\alpha \rightarrow \beta} x) \\
&= \lambda x^{s \rightarrow \alpha \rightarrow \beta} j^s z^\alpha. \text{ext}_\beta(\text{int}_{\alpha \rightarrow \beta} x(\text{int}_\alpha(\lambda k^s.z)))j \\
&= \lambda x^{s \rightarrow \alpha \rightarrow \beta} j^s z^\alpha. \text{ext}_\beta(\text{int}_\beta(\lambda i^s.xi(\text{ext}_\alpha(\text{int}_\alpha(\lambda k^s.z))i)))j \\
&= \lambda x^{s \rightarrow \alpha \rightarrow \beta} j^s z^\alpha. (\lambda i^s.xi((\lambda k^s.z)i))j \quad \text{by induction hypothesis} \\
&= \lambda x^{s \rightarrow \alpha \rightarrow \beta} j^s z^\alpha.xjz \\
&= \lambda x^{s \rightarrow \alpha \rightarrow \beta}.x
\end{aligned}$$

□

Clearly,

$$\lambda y^{\bar{\alpha}}. \text{int}_\alpha(\text{ext}_\alpha y) = \lambda y^{\bar{\alpha}}.y$$

does not hold in general. We call an object  $y \in D_{\bar{\alpha}}$  *quasi-extensional* if  $y = \text{int}_\alpha(\text{ext}_\alpha y)$ . By Lemma 1, it is clear that  $y \in D_{\bar{\alpha}}$  is quasi-extensional if and only if  $y = \text{int}_\alpha x$  for some  $x \in D_{s \rightarrow \alpha}$ . We call  $y \in D_{\bar{\alpha}}$  *truly intensional* if it is not quasi-extensional.

Note that if  $x \in D_{t^n \rightarrow t}$  (i.e.,  $x$  is a truth function), then  $\text{int}_{t^n \rightarrow t}(\lambda k^s.x)$  coincides with its usual Boolean “generalization” to type  $(s \rightarrow t)^n \rightarrow s \rightarrow t$ . For instance, the intensionalization of conjunction  $\wedge \in D_{t \rightarrow t \rightarrow t}$  is

$$\text{int}_{t \rightarrow t \rightarrow t}(\lambda k^s.\wedge) = \lambda x^{s \rightarrow t} y^{s \rightarrow t} i^s. \wedge(xi)(yi),$$

that is, the intersection operation on subsets of  $D_s$ .

In general, the intensionalization  $\text{int}_\alpha(\lambda k^s.x)$  corresponding to a “logical constant”  $x \in D_\alpha$  does not necessarily agree with other existing ways of “lifting”  $x$  to an object in  $D_{\bar{\alpha}}$ . For instance, the first-order universal quantifier over individuals  $\forall_e \in D_{(e \rightarrow t) \rightarrow t}$  can be naturally “lifted” to

$$\lambda y^{(s \rightarrow e) \rightarrow s \rightarrow t} i^s. \forall_{s \rightarrow e}(\lambda z^{s \rightarrow e}.yzi),$$

using the universal quantifier  $\forall_{s \rightarrow e} \in D_{((s \rightarrow e) \rightarrow t) \rightarrow t}$  over *individual concepts* (i.e., functions from possible worlds to individuals), but this differs from its intensionalization,

$$\begin{aligned}
\text{int}_{(e \rightarrow t) \rightarrow t}(\lambda k^s.\forall_e) &= \lambda y^{(s \rightarrow e) \rightarrow s \rightarrow t} i^s. \forall_e(\text{ext}_{e \rightarrow t} y i) \\
&= \lambda y^{(s \rightarrow e) \rightarrow s \rightarrow t} i^s. \forall_e(\lambda z^e.y(\lambda k^s.z)i).
\end{aligned}$$

The point here is that the intensionalization of  $\lambda k^s.x$  is  $\lambda$ -definable in terms of  $x$ , but the universal quantifier over objects of a higher type is not  $\lambda$ -definable in terms of the universal quantifier over objects of a lower type.<sup>3</sup>

Another example is “generalized conjunction”, defined for each type  $\alpha$  of the form  $\alpha_1 \rightarrow \dots \rightarrow \alpha_n \rightarrow t$  by

$$\wedge_\alpha = \lambda y_1^\alpha y_2^\alpha z_1^{\alpha_1} \dots z_n^{\alpha_n}. \wedge(y_1 z_1 \dots z_n)(y_2 z_1 \dots z_n).$$

<sup>3</sup>In fact, in the particular case of the universal quantifier,  $\lambda$ -definability goes the other way around:  $\lambda k^s.\forall_\alpha$  is  $\lambda$ -definable in terms of  $\forall_{\bar{\alpha}}$  by

$$\lambda i^s x^{\alpha \rightarrow t}. \forall_{\bar{\alpha}}(\lambda y^{\bar{\alpha}}.x(\text{ext}_\alpha y i)).$$

(The same can be said of equality at different types.)

For instance,

$$\begin{aligned}
& \text{int}_{(e \rightarrow t) \rightarrow (e \rightarrow t) \rightarrow e \rightarrow t} (\lambda k^s . \wedge_{e \rightarrow t}) \\
&= \lambda y_1^{(s \rightarrow e) \rightarrow s \rightarrow t} y_2^{(s \rightarrow e) \rightarrow s \rightarrow t} z_{s \rightarrow e} i^s . \wedge_{e \rightarrow t} (\text{ext}_{e \rightarrow t} y_1 i) (\text{ext}_{e \rightarrow t} y_2 i) (\text{ext}_e z i) \\
&= \lambda y_1^{(s \rightarrow e) \rightarrow s \rightarrow t} y_2^{(s \rightarrow e) \rightarrow s \rightarrow t} z_{s \rightarrow e} i^s . \wedge_{e \rightarrow t} (\lambda x^e . y_1 (\lambda k^s . x) i) (\lambda x^e . y_2 (\lambda k^s . x) i) (z i) \\
&= \lambda y_1^{(s \rightarrow e) \rightarrow s \rightarrow t} y_2^{(s \rightarrow e) \rightarrow s \rightarrow t} z_{s \rightarrow e} i^s . \wedge (y_1 (\lambda k^s . z i) i) (y_2 (\lambda k^s . z i) i),
\end{aligned}$$

which does not equal

$$\wedge_{e \rightarrow t} = \lambda y_1^{(s \rightarrow e) \rightarrow s \rightarrow t} y_2^{(s \rightarrow e) \rightarrow s \rightarrow t} z_{s \rightarrow e} i^s . \wedge (y_1 z i) (y_2 z i).$$

### 3 Intensional Validity of Extensional Sentences

The object language of our study is that of typed  $\lambda$ -terms built up from basic expressions consisting of constants, each of some type  $\alpha$  over  $A$ , and countably many variables for each type  $\alpha$  over  $A$ . We use boldface variables  $\mathbf{v}_1^\alpha, \mathbf{v}_2^\alpha, \dots$  and boldface  $\lambda$  in the object language to avoid confusion with the metalanguage. Because of the presence of constants, whose interpretation we can pick at will, this choice of the object language is general enough to encompass most “extensional” languages of formal logic, and is also adequate as a language for representing meanings of expressions in extensional fragments of natural language in the style of Montague semantics. The intensionalization of the semantics of these languages serves as a foundation on which to build richer languages including intensional constructs within the usual framework of possible world semantics.

An *extensional model*  $M$  consists of base domains  $(D_\alpha)_{\alpha \in A}$  and an assignment of a denotation  $M(c) \in D_\alpha$  to each object language constant  $c$  of type  $\alpha$ . An *intensional model* consists of base domains  $(D_\alpha)_{\alpha \in A \cup \{s\}}$  together with an assignment of a denotation  $M(c) \in D_\alpha$  to each object language constant  $c$  of type  $\alpha$ . We are interested in those intensional models that are built from extensional models by means of intensionalization.

An object language expression  $\varphi$  has the denotation  $\llbracket \varphi \rrbracket_{M,g}$  in an (extensional or intensional) model  $M$  relative to an assignment  $g$  of values to variables. In an extensional model,  $g(\mathbf{v}_i^\alpha) \in D_\alpha$ , whereas in an intensional model,  $g(\mathbf{v}_i^\alpha) \in D_{\bar{\alpha}}$ . We let  $g[x/\mathbf{v}_i^\alpha]$  denote the assignment that is like  $g$  except that it assigns  $x$  to  $\mathbf{v}_i^\alpha$ .

$$\begin{aligned}
\llbracket c \rrbracket_{M,g} &= M(c), \\
\llbracket \mathbf{v}_i^\alpha \rrbracket_{M,g} &= g(\mathbf{v}_i^\alpha), \\
\llbracket \varphi \psi \rrbracket_{M,g} &= \llbracket \varphi \rrbracket_{M,g} \llbracket \psi \rrbracket_{M,g}, \\
\llbracket \lambda \mathbf{v}_i^\alpha . \varphi \rrbracket_{M,g} &= \begin{cases} \lambda x^\alpha . \llbracket \varphi \rrbracket_{M,g[x/\mathbf{v}_i^\alpha]} & \text{if } M \text{ is an extensional model,} \\ \lambda x^{\bar{\alpha}} . \llbracket \varphi \rrbracket_{M,g[x/\mathbf{v}_i^\alpha]} & \text{if } M \text{ is an intensional model.} \end{cases}
\end{aligned}$$

Note that if  $\varphi$  is an object language expression of type  $\alpha$ , the denotation of  $\varphi$  in an intensional model belongs to  $D_{\bar{\alpha}}$ .

If  $\varphi$  is a closed object language expression,  $\llbracket \varphi \rrbracket_{M,g}$  does not depend on  $g$ , so we let  $\llbracket \varphi \rrbracket_M = \llbracket \varphi \rrbracket_{M,g}$  for an arbitrarily chosen  $g$ .

Given an indexed collection  $\mathcal{I} = (M_i)_{i \in I}$  of extensional models with the same base domains  $(D_a)_{a \in A}$ , we create an intensional model  $M_{\mathcal{I}}$  based on  $\mathcal{I}$ , with  $D_s = I$ . For an object language constant  $c$  of type  $\alpha$ , we let

$$M_{\mathcal{I}}(c) = \text{int}_{\alpha}(\lambda i^s.M_i(c))$$

be its denotation in  $M_{\mathcal{I}}$ .

Let us consider a very simple example to illustrate the above definition. Let  $A = \{t\}$  and let the vocabulary of the object language include constants  $c$  of type  $t$  and  $q$  of type  $t \rightarrow t$ . Fix  $D_t = \{1, 0\}$ . There are eight possible extensional models for this language:  $M(c)$  is either 0 or 1, and  $M(q)$  must be one of  $\lambda x^t.0$ ,  $\lambda x^t.1$ ,  $\lambda x^t.x$ ,  $\lambda x^t.\neg x$ . Let  $\mathcal{C} = \{M_1, \dots, M_8\}$  be the set of these eight models and let  $I$  be a subset of  $\{1, \dots, 8\}$ . We can construct an intensional model  $M_{\mathcal{I}}$  out of  $\mathcal{I} = (M_i)_{i \in I}$  by the above definition. The denotation  $M_{\mathcal{I}}(c)$  of  $c$  is (the characteristic function of) a subset of  $I$ , and  $M_{\mathcal{I}}(q)$  is a function from the power set of  $I$  to the power set of  $I$ . We have

$$\begin{aligned} M_{\mathcal{I}}(q) &= \text{int}_{t \rightarrow t}(\lambda i^s.M_i(q)) \\ &= \lambda y^{s \rightarrow e}.i^s.M_i(q)(yi), \end{aligned}$$

so  $M_{\mathcal{I}}(q) yi$  does not depend on the “global” property of  $y$ , but only on its value at  $i$ . In particular,

$$\begin{aligned} \llbracket qc \rrbracket_{M_{\mathcal{I}}} &= M_{\mathcal{I}}(q)M_{\mathcal{I}}(c) \\ &= (\lambda y^{s \rightarrow e}.i^s.M_i(q)(yi))(\lambda i^s.M_i(c)) \\ &= \lambda i^s.M_i(q)M_i(c) \\ &= \lambda i^s.\llbracket qc \rrbracket_{M_i}. \end{aligned} \tag{1}$$

Having defined  $M_{\mathcal{I}}$ , one can then expand it to a model  $M'$  for a larger vocabulary including additional constants, such as the necessity operator  $\Box$  (of type  $t \rightarrow t$ , with denotation in  $D_{t \rightarrow t}$ ), whose denotation is truly intensional.

We shall show that equalities exemplified by (1) are completely general and hold for all sentences in the extensional object language.

If  $g$  is an intensional assignment (suitable for  $M_{\mathcal{I}}$ ), define an extensional assignment  $g \downarrow_i$  by

$$g \downarrow_i(\mathbf{v}_i^\alpha) = \text{ext}_{\alpha} g(\mathbf{v}_i^\alpha) i.$$

We call an intensional assignment  $g$  *quasi-extensional* if  $g(\mathbf{v}_i^\alpha)$  is quasi-extensional for all  $\mathbf{v}_i^\alpha$ .

**Lemma 2.** *Let  $\varphi$  be an object language expression of type  $\alpha$ , and let  $g$  be an intensional assignment suitable for  $M_{\mathcal{I}}$  that is quasi-extensional.*

1.  $\text{ext}_{\alpha} \llbracket \varphi \rrbracket_{M_{\mathcal{I}}, g} i = \llbracket \varphi \rrbracket_{M_i, g \downarrow_i}$ .
2. If  $\varphi$  is not a  $\lambda$ -abstract,  $\llbracket \varphi \rrbracket_{M_{\mathcal{I}}, g} = \text{int}_{\alpha}(\lambda i^s.\llbracket \varphi \rrbracket_{M_i, g \downarrow_i})$ .

*Proof.* We prove both 1 and 2 simultaneously by induction on  $\varphi$ , assuming that  $\varphi$  is in  $\beta$ -normal form. Note that if  $\varphi$  is not a  $\lambda$ -abstract, 1 follows from 2 by Lemma 1.

If  $\varphi$  is a constant  $c$ , then 2 holds by the definition of  $M_{\mathcal{I}}(c)$ . If  $\varphi$  is a variable  $\mathbf{v}_i^\alpha$ , 2 holds since  $g$  is quasi-extensional.

Suppose  $\varphi = \psi\chi$ , where  $\psi$  is of type  $\beta \rightarrow \alpha$  and  $\chi$  is of type  $\beta$ . Since  $\varphi$  is in  $\beta$ -normal form,  $\psi$  is not a  $\lambda$ -abstract. Hence by induction hypothesis,

$$\begin{aligned} \llbracket \psi \rrbracket_{M_{\mathcal{I},g}} &= \text{int}_{\beta \rightarrow \alpha}(\lambda i^s . \llbracket \psi \rrbracket_{M_{i,g \downarrow i}}) \\ &= \lambda y_0^{\bar{\alpha}} . \text{int}_{\alpha}(\lambda i^s . \llbracket \psi \rrbracket_{M_{i,g \downarrow i}}(\text{ext}_{\beta} y_0 i)). \end{aligned}$$

The induction hypothesis about  $\chi$  gives

$$\text{ext}_{\beta} \llbracket \chi \rrbracket_{M_{\mathcal{I},g}} i = \llbracket \chi \rrbracket_{M_{i,g \downarrow i}}.$$

Thus,

$$\begin{aligned} \llbracket \psi\chi \rrbracket_{M_{\mathcal{I},g}} &= \llbracket \psi \rrbracket_{M_{\mathcal{I},g}} \llbracket \chi \rrbracket_{M_{\mathcal{I},g}} \\ &= \text{int}_{\alpha}(\lambda i^s . \llbracket \psi \rrbracket_{M_{i,g \downarrow i}}(\text{ext}_{\beta} \llbracket \chi \rrbracket_{M_{\mathcal{I},g}} i)) \\ &= \text{int}_{\alpha}(\lambda i^s . \llbracket \psi \rrbracket_{M_{i,g \downarrow i}} \llbracket \chi \rrbracket_{M_{i,g \downarrow i}}) \\ &= \text{int}_{\alpha}(\lambda i^s . \llbracket \psi\chi \rrbracket_{M_{i,g \downarrow i}}) \end{aligned}$$

and the condition in 2 holds of  $\varphi$ .

It remains to consider the case where  $\varphi$  is a  $\lambda$ -abstract  $\lambda \mathbf{u}^{\beta} . \psi$ , where  $\alpha = \beta \rightarrow \gamma$  and  $\psi$  is of type  $\gamma$ . By induction hypothesis,  $\text{ext}_{\gamma} \llbracket \psi \rrbracket_{M_{\mathcal{I},h}} i = \llbracket \psi \rrbracket_{M_{i,h \downarrow i}}$  for all assignments  $h$  that are quasi-extensional.

$$\begin{aligned} \text{ext}_{\beta \rightarrow \gamma} \llbracket \lambda \mathbf{u}^{\beta} . \psi \rrbracket_{M_{\mathcal{I},g}} i &= \lambda x^{\beta} . \text{ext}_{\gamma}(\llbracket \lambda \mathbf{u}^{\beta} . \psi \rrbracket_{M_{\mathcal{I},g}}(\text{int}_{\beta}(\lambda k^s . x))) i \\ &= \lambda x^{\beta} . \text{ext}_{\gamma}(\llbracket \psi \rrbracket_{M_{\mathcal{I},g[\text{int}_{\beta}(\lambda k^s . x)/\mathbf{u}^{\beta}]}}) i \\ &= \lambda x^{\beta} . \llbracket \psi \rrbracket_{M_{i,g[\text{int}_{\beta}(\lambda k^s . x)/\mathbf{u}^{\beta}] \downarrow i}} \quad \text{by induction hypothesis} \\ &= \lambda x^{\beta} . \llbracket \psi \rrbracket_{M_{i,g \downarrow i}[x/\mathbf{u}^{\beta}]} \\ &= \llbracket \lambda \mathbf{u}^{\beta} . \psi \rrbracket_{M_{i,g \downarrow i}}. \end{aligned}$$

(Note that  $g[\text{int}_{\beta}(\lambda k^s . x)/\mathbf{u}^{\beta}]$  is quasi-extensional.) Thus, 1 holds of  $\varphi$ .  $\square$

Note that the condition in part 2 of Lemma 2 does not hold of  $\lambda$ -abstracts. A simple counterexample is the  $\mathbf{I}$  combinator  $\lambda \mathbf{u}^{a \rightarrow b} . \mathbf{u}^{a \rightarrow b}$ :

$$\begin{aligned} &\text{int}_{(a \rightarrow b) \rightarrow a \rightarrow b}(\lambda i^s . \llbracket \lambda \mathbf{u}^{a \rightarrow b} . \mathbf{u}^{a \rightarrow b} \rrbracket_{M_i}) \\ &= \text{int}_{(a \rightarrow b) \rightarrow a \rightarrow b}(\lambda i^s . x^{a \rightarrow b} . x) \\ &= \lambda y_1^{(s \rightarrow a) \rightarrow s \rightarrow b} y_2^{s \rightarrow a} i^s . \text{ext}_{a \rightarrow b} y_1 i (\text{ext}_a y_2 i) \\ &= \lambda y_1^{(s \rightarrow a) \rightarrow s \rightarrow b} y_2^{s \rightarrow a} i^s . (\lambda z^a . y_1(\text{int}_a(\lambda k . z))i)(\text{ext}_a y_2 i) \\ &= \lambda y_1^{(s \rightarrow a) \rightarrow s \rightarrow b} y_2^{s \rightarrow a} i^s . (\lambda z^a . y_1(\lambda k . z)i)(y_2 i) \\ &= \lambda y_1^{(s \rightarrow a) \rightarrow s \rightarrow b} y_2^{s \rightarrow a} i^s . y_1(\lambda k . y_2 i)i \\ &\neq \lambda y_1^{(s \rightarrow a) \rightarrow s \rightarrow b} y_2^{s \rightarrow a} i^s . y_1 y_2 i \\ &= \lambda y_1^{(s \rightarrow a) \rightarrow s \rightarrow b} . y_1 \\ &= \llbracket \lambda \mathbf{u}^{a \rightarrow b} . \mathbf{u}^{a \rightarrow b} \rrbracket_{M_{\mathcal{I}}}, \end{aligned}$$

(The inequality assumes  $|D_a| \geq 2$  and  $|D_s| \geq 2$ .)

**Remark 2.** A special case of Lemma 2 is when  $g = h^*$  for some extensional assignment  $h$ , where  $h^*$  is defined by

$$h^*(\mathbf{v}_i^\alpha) = \text{int}_\alpha(\lambda k^s.h(\mathbf{v}_i^\alpha)).$$

In this case, we have  $g \downarrow_i = h$  for all  $i \in I$ . This special case itself can be proved directly by induction, as was done in a previous version of this note.

**Remark 3.** The content of Lemma 2 can be stated entirely within simply typed  $\lambda$ -calculus, as follows. If  $\varphi$  is a  $\lambda$ -term of type  $\alpha$ , with free variables  $z_1^{\beta_1}, \dots, z_n^{\beta_n}$ , let  $\bar{\varphi}$  be the  $\lambda$ -term of type  $\bar{\alpha}$  obtained from  $\varphi$  by replacing each occurrence of  $a \in A$  by  $s \rightarrow a$  in the type annotation of  $\varphi$ . Then we have

$$\begin{aligned} \text{ext}_\alpha \bar{\varphi}[(\text{int}_{\beta_1} x_1^{s \rightarrow \beta_1})/z_1^{\bar{\beta}_1}, \dots, (\text{int}_{\beta_n} x_n^{s \rightarrow \beta_n})/z_n^{\bar{\beta}_n}] \\ =_{\beta_\eta} \lambda i^s. \varphi[(x_1^{s \rightarrow \beta_1} i)/z_1^{\beta_1}, \dots, (x_n^{s \rightarrow \beta_n} i)/z_n^{\beta_n}], \end{aligned}$$

and if  $\varphi$  is not a  $\lambda$ -abstract,

$$\begin{aligned} \bar{\varphi}[(\text{int}_{\beta_1} x_1^{s \rightarrow \beta_1})/z_1^{\bar{\beta}_1}, \dots, (\text{int}_{\beta_n} x_n^{s \rightarrow \beta_n})/z_n^{\bar{\beta}_n}] \\ =_{\beta_\eta} \text{int}_\alpha(\lambda i^s. \varphi[(x_1^{s \rightarrow \beta_1} i)/z_1^{\beta_1}, \dots, (x_n^{s \rightarrow \beta_n} i)/z_n^{\beta_n}]). \end{aligned}$$

**Theorem 3.** For any closed object language expression  $\varphi$  of type  $a \in A$ ,  $\llbracket \varphi \rrbracket_{M_{\mathcal{I}}} = \lambda i^s. \llbracket \varphi \rrbracket_{M_i}$ .

*Proof.* Immediate from Lemma 2. □

Now assume  $t \in A$  and fix  $D_t = \{1, 0\}$ . We call an object language expression of type  $t$  a *formula*, and a closed formula a *sentence*. A *pointed possible world model* is a pair of the form  $(M_{\mathcal{I}}, i)$  where  $i \in I$ . The extensional model  $M_i$  is the *actual world* of a pointed possible world model  $(M_{\mathcal{I}}, i)$ . We say that a sentence  $\varphi$  is true in a pointed possible world model  $(M_{\mathcal{I}}, i)$  if  $\llbracket \varphi \rrbracket_{M_{\mathcal{I}}} i = 1$ .

**Corollary 4.** For every sentence  $\varphi$  in the object language and every extensional model  $M$ , the following are equivalent:

1.  $\varphi$  is true in  $M$ .
2.  $\varphi$  is true in any pointed possible world model whose actual world is  $M$ .

Let  $\mathcal{C}$  be a class of extensional models. Call a sentence  $\varphi$  *extensionally valid in  $\mathcal{C}$*  if  $\llbracket \varphi \rrbracket_M = 1$  for all  $M \in \mathcal{C}$ , and *intensionally valid in  $\mathcal{C}$*  if  $\llbracket \varphi \rrbracket_{M_{\mathcal{I}}} = \lambda i^s. 1$  for all indexed collections  $\mathcal{I} = (M_i)_{i \in I}$  consisting of models in  $\mathcal{C}$  that share the same base domains. Similarly,  $\psi$  is an *extensional consequence of  $\varphi_1, \dots, \varphi_n$  in  $\mathcal{C}$*  if  $\llbracket \varphi_1 \rrbracket_M = \dots = \llbracket \varphi_n \rrbracket_M = 1$  implies  $\llbracket \psi \rrbracket_M = 1$  for all  $M \in \mathcal{C}$ , and  $\psi$  is an *intensional consequence of  $\varphi_1, \dots, \varphi_n$  in  $\mathcal{C}$*  if  $\llbracket \varphi_1 \rrbracket_{M_{\mathcal{I}}} \cap \dots \cap \llbracket \varphi_n \rrbracket_{M_{\mathcal{I}}} \subseteq \llbracket \psi \rrbracket_{M_{\mathcal{I}}}$  for all indexed collections  $\mathcal{I}$  consisting of models in  $\mathcal{C}$  with the same base domains.

**Corollary 5.** Let  $\mathcal{C}$  be a class of extensional models and let  $\varphi, \varphi_1, \dots, \varphi_n, \psi$  be sentences.

1. If  $\varphi$  is extensionally valid in  $\mathcal{C}$ , then  $\varphi$  is intensionally valid in  $\mathcal{C}$ .

2. If  $\psi$  is an extensional consequence of  $\varphi_1, \dots, \varphi_n$  in  $\mathcal{C}$ , then  $\psi$  is an intensional consequence of  $\varphi_1, \dots, \varphi_n$  in  $\mathcal{C}$ .

In the presence of conjunction ( $\wedge$ ) and implication ( $\rightarrow$ ) in the object language, the consequence relation between  $\varphi_1, \dots, \varphi_n$  and  $\psi$  can be defined as the validity of  $\varphi_1 \wedge \dots \wedge \varphi_n \rightarrow \psi$ , both in the extensional and in the intensional sense. (Recall that truth-functional connectives behave as desired in intensional models.) This allows us to concentrate on validity.

## 4 Intensionally Valid Schemata

Corollary 5 does not quite give what Ben-Avi and Winter (2007) were aiming for, because in their method of intensionalization, the denotation of a non-logical constant of type  $\alpha$  is not restricted to quasi-extensional objects in  $D_{\bar{\alpha}}$ . They start from a class  $\mathcal{C}$  of extensional models that is closed under arbitrary change in the denotations of non-logical constants and obtain by intensionalization a class  $\mathcal{C}'$  of intensional models that is again closed under arbitrary change in the denotations of non-logical constants. (In their method, intensionalization is only used to determine the denotations of logical constants in intensional models.) Replacing non-logical constants with free variables, we can say in our setting that what they were aiming for was preservation of validity of (and the consequence relation among) open formulas or *schemata*. This is clearly an unreasonably high demand and is impossible to achieve in any general terms; be that as it may, it will be instructive to see the limited extent to which the present method of intensionalization preserves validity of open formulas.

The generalization of the notion of validity to open formulas is the standard one. Let  $\mathcal{C}$  be a class of extensional models. For an object language expression  $\varphi$  of type  $t$ , we say that  $\varphi$  is *extensionally valid in  $\mathcal{C}$*  if  $\llbracket \varphi \rrbracket_{M,g} = 1$  for all  $M \in \mathcal{C}$  and all extensional assignments  $g$  suitable for  $M$ ; we say that  $\varphi$  is *intensionally valid in  $\mathcal{C}$*  if  $\llbracket \varphi \rrbracket_{M_{\mathcal{I}},g} = \lambda i^s.1$  for all indexed collections  $\mathcal{I}$  consisting of models in  $\mathcal{C}$  built on the same base domains and all intensional assignments  $g$  suitable for  $M_{\mathcal{I}}$ .

Lemma 2 does not imply that the validity of an open formula is preserved when one moves from extensional models to intensional models created out of them, because not all intensional assignments are quasi-extensional. For example, let

$$\varphi = \rightarrow(\wedge(\mathbf{u}^{t \rightarrow t} \top)(\mathbf{u}^{t \rightarrow t} \perp))(\mathbf{u}^{t \rightarrow t} \mathbf{v}^t),$$

or, in a more readable style,

$$(\mathbf{u}^{t \rightarrow t} \top \wedge \mathbf{u}^{t \rightarrow t} \perp) \rightarrow \mathbf{u}^{t \rightarrow t} \mathbf{v}^t. \quad (2)$$

Let  $\rightarrow, \wedge, \top, \perp$  have the usual interpretation in  $M_i$  for all  $i \in I$ . Then  $\varphi$  is extensionally valid in  $\mathcal{I} = \{M_i \mid i \in I\}$ , but it is easy to see that there are intensional assignments  $g$  such that

$$\llbracket \varphi \rrbracket_{M_{\mathcal{I}},g} \neq \lambda i^s.1.$$

The reason that an extensionally valid formula  $\varphi$  with  $\text{FV}(\varphi) \neq \emptyset$  need not be intensionally valid is related to the failure of the equality  $\llbracket \psi \rrbracket_{M_{\mathcal{I}}} = \text{int}_{\alpha}(\lambda i^s. \llbracket \psi \rrbracket_{M_i})$  when  $\psi$  is a (closed)  $\lambda$ -abstract. Let  $\text{FV}(\varphi) =$

$\{\mathbf{u}_1^{\alpha_1}, \dots, \mathbf{u}_n^{\alpha_n}\}$ . We have  $\llbracket \varphi \rrbracket_{M_i, g} = 1$  for all extensional assignments  $g$  if and only if  $\llbracket \lambda \mathbf{u}_1^{\alpha_1} \dots \mathbf{u}_n^{\alpha_n} . \varphi \rrbracket_{M_i} = \lambda x_1^{\alpha_1} \dots x_n^{\alpha_n} . 1$ . Also,  $\llbracket \varphi \rrbracket_{M_{\mathcal{I}}, g} = \lambda i^s . 1$  for all intensional assignments  $g$  if and only if  $\llbracket \lambda \mathbf{u}_1^{\alpha_1} \dots \mathbf{u}_n^{\alpha_n} . \varphi \rrbracket_{M_{\mathcal{I}}} = \lambda y_1^{\overline{\alpha_1}} \dots y_n^{\overline{\alpha_n}} i^s . 1$ . Now suppose  $\llbracket \varphi \rrbracket_{M_i, g} = 1$  for all  $i \in I$  and all extensional assignments  $g$ . Then  $\llbracket \lambda \mathbf{u}_1^{\alpha_1} \dots \mathbf{u}_n^{\alpha_n} . \varphi \rrbracket_{M_i} = \lambda x_1^{\alpha_1} \dots x_n^{\alpha_n} . 1$ , and this clearly implies

$$\text{int}_{\alpha_1 \rightarrow \dots \rightarrow \alpha_n \rightarrow t} (\lambda i^s . \llbracket \lambda \mathbf{u}_1^{\alpha_1} \dots \mathbf{u}_n^{\alpha_n} . \varphi \rrbracket_{M_i}) = \lambda y_1^{\overline{\alpha_1}} \dots y_n^{\overline{\alpha_n}} i^s . 1.$$

But since  $\llbracket \lambda \mathbf{u}_1^{\alpha_1} \dots \mathbf{u}_n^{\alpha_n} . \varphi \rrbracket_{M_{\mathcal{I}}} = \text{int}_{\alpha_1 \rightarrow \dots \rightarrow \alpha_n \rightarrow t} (\lambda i^s . \llbracket \lambda \mathbf{u}_1^{\alpha_1} \dots \mathbf{u}_n^{\alpha_n} . \varphi \rrbracket_{M_i})$  need not hold, we cannot infer  $\llbracket \lambda \mathbf{u}_1^{\alpha_1} \dots \mathbf{u}_n^{\alpha_n} . \varphi \rrbracket_{M_{\mathcal{I}}} = \lambda y_1^{\overline{\alpha_1}} \dots y_n^{\overline{\alpha_n}} i^s . 1$ .

The open formula (2) should be clearly distinguished from its closure

$$\forall_{t \rightarrow t} (\lambda \mathbf{u}^{t \rightarrow t} . \forall_t (\lambda \mathbf{v}^t . (\mathbf{u}^{t \rightarrow t} \top \wedge \mathbf{u}^{t \rightarrow t} \perp) \rightarrow \mathbf{u}^{t \rightarrow t} \mathbf{v}^t)) \quad (3)$$

or the open formulas in one free variable

$$\forall_{t \rightarrow t} (\lambda \mathbf{u}^{t \rightarrow t} . (\mathbf{u}^{t \rightarrow t} \top \wedge \mathbf{u}^{t \rightarrow t} \perp) \rightarrow \mathbf{u}^{t \rightarrow t} \mathbf{v}^t), \quad (4)$$

$$\forall_t (\lambda \mathbf{v}^t . (\mathbf{u}^{t \rightarrow t} \top \wedge \mathbf{u}^{t \rightarrow t} \perp) \rightarrow \mathbf{u}^{t \rightarrow t} \mathbf{v}^t). \quad (5)$$

The three formulas (3), (4), (5), unlike (2), remain valid in intensional models. As mentioned above, the intensionalization of the universal quantifier  $\forall_{\alpha}$  still only quantifies over objects in  $D_{\alpha}$ , so the intensional validity of (3), (4), (5) does not imply the intensional validity of (2).

The intensional validity of (4) and (5) illustrates the fact that extensionally valid open formulas may remain intensionally valid in certain restricted cases. In what follows, we give one sufficient condition for an extensionally valid  $\varphi$  to be intensionally valid.

Fix an indexed collection  $\mathcal{I} = (M_i)_{i \in I}$  of extensional models. We call an object language constant  $c$  *rigid* (in  $\mathcal{I}$ ) if  $M_i(c) = M_j(c)$  for all  $i, j \in I$ .

Let  $V$  be a set of object language variables. Let  $\varphi$  be an object language expression. We define two predicates  $V$ -safe and  $V$ -protected by simultaneous induction as follows:

- $\varphi$  is  $V$ -safe if and only if one of the following conditions holds:
  1.  $\varphi$  is a constant or a variable.
  2.  $\varphi = \psi \chi$  and either
    - $\psi$  is  $V$ -protected and  $\chi$  is  $V$ -safe, or
    - $\psi$  is  $V$ -safe, all constants that occur in  $\chi$  are rigid,  $\text{FV}(\chi) \cap V = \emptyset$ , and  $\chi$  is  $\emptyset$ -protected.
  3.  $\varphi = \lambda \mathbf{v} . \psi$  and  $\psi$  is  $V$ -safe.
- $\varphi$  is  $V$ -protected if and only if one of the following conditions holds:
  1.  $\varphi$  is a constant or a variable not in  $V$ .
  2.  $\varphi = \psi \chi$  and  $\psi$  is  $V$ -protected and  $\chi$  is  $V$ -safe.
  3.  $\varphi = \lambda \mathbf{v} . \psi$  and  $\psi$  is  $V \cup \{\mathbf{v}\}$ -protected.

More informally, if  $\lambda \mathbf{u}_1^{\beta_1} \dots \mathbf{u}_n^{\beta_n} . \chi$  occurs in a  $V$ -safe formula as an argument of a variable in  $V$ , then  $\chi$  cannot contain any non-rigid constants or variables in  $V$ , and  $\chi$  must be a  $\{\mathbf{u}_1^{\beta_1}, \dots, \mathbf{u}_n^{\beta_n}\}$ -safe formula that does not start with one of  $\mathbf{u}_1^{\beta_1}, \dots, \mathbf{u}_n^{\beta_n}$ .

**Lemma 6.** *Let  $V$  be a set of object language variables, and let  $\varphi$  be an object language expression of type  $\alpha$ . Suppose that  $g$  is an intensional assignment such that for all variables  $\mathbf{u}^\delta \in \text{FV}(\varphi) - V$ , we have  $g(\mathbf{u}^\delta) = \text{int}_\delta(\lambda k^s.x)$  for some  $x \in D_\delta$ . The following hold:*

1. *If  $\varphi$  is  $V$ -protected,  $\llbracket \varphi \rrbracket_{M_{\mathcal{I},g}} = \text{int}_\alpha(\lambda i^s. \llbracket \varphi \rrbracket_{M_{i,g\downarrow_i}})$ .*
2. *If  $\varphi$  is  $V$ -safe,  $\text{ext}_\alpha \llbracket \varphi \rrbracket_{M_{\mathcal{I},g}} i = \llbracket \varphi \rrbracket_{M_{i,g\downarrow_i}}$ .*

*Proof.* We prove 1 and 2 by simultaneous induction on  $\varphi$ . Note that the condition in 1 implies the condition in 2, so when  $\varphi$  is  $V$ -protected, we only need to prove the condition in 1.

*Induction basis.* If  $\varphi$  is a constant  $c$ , then  $\llbracket c \rrbracket_{M_{\mathcal{I},g}} = M_{\mathcal{I}}(c) = \text{int}_\alpha(\lambda i^s.M_i(c)) = \text{int}_\alpha(\lambda i^s.\llbracket c \rrbracket_{M_{i,g\downarrow_i}})$  by the definition of  $M_{\mathcal{I}}(c)$ . If  $\varphi$  is a variable  $\mathbf{v}^\alpha$ , then  $\text{ext}_\alpha \llbracket \mathbf{v}^\alpha \rrbracket_{M_{\mathcal{I},g}} i = \text{ext}_\alpha g(\mathbf{v}^\alpha) i = g\downarrow_i(\mathbf{v}^\alpha) = \llbracket \mathbf{v}^\alpha \rrbracket_{M_{i,g\downarrow_i}}$  by the definition of  $g\downarrow_i$ . If  $\mathbf{v}^\alpha \notin V$ , then by the assumption on  $g$ , we have  $\llbracket \mathbf{v}^\alpha \rrbracket_{M_{\mathcal{I},g}} = g(\mathbf{v}^\alpha) = \text{int}(\lambda i^s.g\downarrow_i(\mathbf{v}^\alpha)) = \text{int}(\lambda i^s.\llbracket \mathbf{v}^\alpha \rrbracket_{M_{i,g\downarrow_i}})$ . Thus, the desired condition holds in all cases.

*Induction step.*

Case 1.  $\varphi = \psi\chi$ , where  $\psi$  is of type  $\beta \rightarrow \alpha$  and  $\chi$  is of type  $\beta$ .

Case 1a.  $\psi$  is  $V$ -protected and  $\chi$  is  $V$ -safe. In this case,  $\varphi$  is  $V$ -protected (as well as  $V$ -safe). By induction hypothesis,  $\llbracket \psi \rrbracket_{M_{\mathcal{I},g}} = \text{int}_{\beta \rightarrow \alpha}(\lambda i^s.\llbracket \psi \rrbracket_{M_{i,g\downarrow_i}})$  and  $\text{ext}_\beta \llbracket \chi \rrbracket_{M_{\mathcal{I},g}} i = \llbracket \chi \rrbracket_{M_{i,g\downarrow_i}}$ . Hence

$$\begin{aligned} \llbracket \psi\chi \rrbracket_{M_{\mathcal{I},g}} &= \llbracket \psi \rrbracket_{M_{\mathcal{I},g}} \llbracket \chi \rrbracket_{M_{\mathcal{I},g}} \\ &= \text{int}_{\beta \rightarrow \alpha}(\lambda i^s.\llbracket \psi \rrbracket_{M_{i,g\downarrow_i}} \llbracket \chi \rrbracket_{M_{\mathcal{I},g}}) \\ &= \text{int}_\alpha(\lambda i^s.\llbracket \psi \rrbracket_{M_{i,g\downarrow_i}} (\text{ext}_\beta \llbracket \chi \rrbracket_{M_{\mathcal{I},g}} i)) \\ &= \text{int}_\alpha(\lambda i^s.\llbracket \psi \rrbracket_{M_{i,g\downarrow_i}} \llbracket \chi \rrbracket_{M_{i,g\downarrow_i}}) \\ &= \text{int}_\alpha(\lambda i^s.\llbracket \psi\chi \rrbracket_{M_{i,g\downarrow_i}}), \end{aligned}$$

and the condition in 1 is satisfied.

Case 1b.  $\psi$  is  $V$ -safe, all constants that occur in  $\chi$  are rigid,  $\text{FV}(\chi) \cap V = \emptyset$ , and  $\chi$  is  $\emptyset$ -protected. In this case,  $\varphi$  is  $V$ -safe. Note that  $\text{FV}(\chi) \cap V = \emptyset$  implies that  $g(\mathbf{u}^\delta) = \lambda k^s.x$  for some  $x \in D_\delta$  for all  $\mathbf{u}^\delta \in \text{FV}(\chi)$ . Hence, the induction hypothesis applies to both  $\psi$  and  $\chi$  and we get  $\text{ext}_{\beta \rightarrow \alpha} \llbracket \psi \rrbracket_{M_{\mathcal{I},g}} = \llbracket \psi \rrbracket_{M_{i,g\downarrow_i}}$  and  $\llbracket \chi \rrbracket_{M_{\mathcal{I},g}} = \text{int}_\beta(\lambda i^s.\llbracket \chi \rrbracket_{M_{i,g\downarrow_i}})$ . The fact that  $g(\mathbf{u}^\delta) = \lambda k^s.x$  for some  $x \in D_\delta$  for all  $\mathbf{u}^\delta \in \text{FV}(\chi)$  also implies that  $g\downarrow_i$  and  $g\downarrow_j$  agree on  $\text{FV}(\chi)$  for all  $i, j \in I$ . Since  $M_i(c) = M_j(c)$  for all  $i, j \in I$  for all constants  $c$  in  $\chi$ , we see that  $\llbracket \chi \rrbracket_{M_{i,g\downarrow_i}} = \llbracket \chi \rrbracket_{M_{j,g\downarrow_j}}$  for all  $i, j \in I$ . Thus,

$$\begin{aligned} \text{ext}_\alpha \llbracket \psi\chi \rrbracket_{M_{\mathcal{I},g}} i &= \text{ext}_\alpha (\llbracket \psi \rrbracket_{M_{\mathcal{I},g}} \llbracket \chi \rrbracket_{M_{\mathcal{I},g}}) i \\ &= \text{ext}_\alpha (\llbracket \psi \rrbracket_{M_{\mathcal{I},g}} \text{int}_\beta(\lambda k^s.\llbracket \chi \rrbracket_{M_{k,g\downarrow_k}})) i \\ &= \text{ext}_\alpha (\llbracket \psi \rrbracket_{M_{\mathcal{I},g}} \text{int}_\beta(\lambda k^s.\llbracket \chi \rrbracket_{M_{i,g\downarrow_i}})) i \\ &\quad \text{since } \llbracket \chi \rrbracket_{M_{k,g\downarrow_k}} = \llbracket \chi \rrbracket_{M_{i,g\downarrow_i}} \text{ for all } k \in I \\ &= \text{ext}_{\beta \rightarrow \alpha} \llbracket \psi \rrbracket_{M_{\mathcal{I},g}} i \llbracket \chi \rrbracket_{M_{i,g\downarrow_i}} \\ &= \llbracket \psi \rrbracket_{M_{i,g\downarrow_i}} \llbracket \chi \rrbracket_{M_{i,g\downarrow_i}} \\ &= \llbracket \psi\chi \rrbracket_{M_{i,g\downarrow_i}}, \end{aligned}$$

and the condition in 2 holds.

Case 2.  $\varphi = \lambda \mathbf{v}^\beta.\psi$ , where  $\psi$  is of type  $\gamma$  and  $\alpha = \beta \rightarrow \gamma$ .

Case 2a.  $\psi$  is  $V \cup \{\mathbf{v}^\beta\}$ -protected. In this case,  $\varphi$  is  $V$ -protected. By induction hypothesis,  $\llbracket \psi \rrbracket_{M_{\mathcal{I},h}} = \text{int}(\lambda i^s. \llbracket \psi \rrbracket_{M_i, h \downarrow_i})$  holds of all  $h$  satisfying the following condition:

$$\text{for all } \mathbf{u}^\delta \in \text{FV}(\psi) - (V \cup \{\mathbf{v}^\beta\}), \text{ there is an } x \in D_\delta \text{ such that } h(\mathbf{u}^\delta) = \text{int}_\delta(\lambda k^s. x). \quad (6)$$

We see

$$\begin{aligned} \llbracket \lambda \mathbf{v}^\beta. \psi \rrbracket_{M_{\mathcal{I},g}} &= \lambda y^{\bar{\beta}}. \llbracket \psi \rrbracket_{M_{\mathcal{I},g[y/\mathbf{v}^\beta]}} \\ &= \lambda y^{\bar{\beta}}. \text{int}_\gamma(\lambda i^s. \llbracket \psi \rrbracket_{M_i, g[y/\mathbf{v}^\beta] \downarrow_i}) \\ &= \lambda y^{\bar{\beta}}. \text{int}_\gamma(\lambda i^s. \llbracket \psi \rrbracket_{M_i, g \downarrow_i[\text{ext}_\beta y i / \mathbf{v}^\beta]}) \\ &= \lambda y^{\bar{\beta}}. \text{int}_\gamma(\lambda i^s. \llbracket \lambda \mathbf{v}^\beta. \psi \rrbracket_{M_i, g \downarrow_i}(\text{ext}_\beta y i)) \\ &= \text{int}_{\beta \rightarrow \gamma}(\lambda i^s. \llbracket \lambda \mathbf{v}^\beta. \psi \rrbracket_{M_i, g \downarrow_i}). \end{aligned}$$

(Note that  $h = g[y/\mathbf{v}^\beta]$  satisfies (6).) Hence the condition in 1 is satisfied.

Case 2b.  $\psi$  is  $V$ -safe. In this case,  $\varphi$  is  $V$ -safe. By induction hypothesis,  $\text{ext}_\gamma \llbracket \psi \rrbracket_{M_{\mathcal{I},h}} i = \llbracket \psi \rrbracket_{M_i, h \downarrow_i}$  holds of all  $h$  satisfying the following condition:

$$\text{for all } \mathbf{u}^\delta \in \text{FV}(\psi) - V, \text{ there is an } x \in D_\delta \text{ such that } h(\mathbf{u}^\delta) = \text{int}_\delta(\lambda k^s. x). \quad (7)$$

We have

$$\begin{aligned} \text{ext}_{\beta \rightarrow \gamma} \llbracket \lambda \mathbf{v}^\beta. \psi \rrbracket_{M_{\mathcal{I},g}} i &= \lambda x^\beta. \text{ext}_\gamma(\llbracket \lambda \mathbf{v}^\beta. \psi \rrbracket_{M_{\mathcal{I},g}(\text{int}_\beta(\lambda k^s. x))}) i \\ &= \lambda x^\beta. \text{ext}_\gamma \llbracket \psi \rrbracket_{M_{\mathcal{I},g[\text{int}_\beta(\lambda k^s. x) / \mathbf{v}^\beta]}} i \\ &= \lambda x^\beta. \llbracket \psi \rrbracket_{M_i, g[\text{int}_\beta(\lambda k^s. x) / \mathbf{v}^\beta] \downarrow_i} \\ &= \lambda x^\beta. \llbracket \psi \rrbracket_{M_i, g \downarrow_i[x / \mathbf{v}^\beta]} \\ &= \llbracket \lambda \mathbf{v}^\beta. \psi \rrbracket_{M_i, g \downarrow_i}. \end{aligned}$$

(Note that  $\text{FV}(\psi) \subseteq \text{FV}(\varphi) \cup \{\mathbf{v}^\beta\}$  and  $h = g[\text{int}_\beta(\lambda k^s. x) / \mathbf{v}^\beta]$  satisfies (7).) Thus, the condition in 2 is satisfied.

This completes the induction step.  $\square$

**Remark 4.** Remark 3 applies, mutatis mutandis, to Lemma 6 as well.

**Theorem 7.** *Let  $\varphi$  be a formula of the object language that is  $\text{FV}(\varphi)$ -safe. If  $\varphi$  is extensionally valid in a class  $\mathcal{C}$  of extensional models, then  $\varphi$  is intensionally valid in  $\mathcal{C}$ .*

*Proof.* Let  $\mathcal{I} = (M_i)_{i \in I}$  be an indexed collection of extensional models in  $\mathcal{C}$  that share the same base domains. Let  $g$  be an arbitrary intensional assignment suitable for  $M_{\mathcal{I}}$ . Then  $\llbracket \varphi \rrbracket_{M_{\mathcal{I},g}} = \text{ext}_t \llbracket \varphi \rrbracket_{M_{\mathcal{I},g}} = \lambda i^s. \llbracket \varphi \rrbracket_{M_i, g \downarrow_i}$  by Lemma 6. Thus, if  $\llbracket \varphi \rrbracket_{M_i, h} = 1$  for all  $i \in I$  and all extensional assignments  $h$  suitable for  $M_i$ , then  $\llbracket \varphi \rrbracket_{M_{\mathcal{I},g}} = \lambda i^s. 1$ .  $\square$

Here are some examples illustrating the scope of applicability of Theorem 7. First, all tautologies of propositional logic are intensionally valid. It is easy to see that all formulas built from propositional variables in  $V$  are  $V$ -safe, because propositional variables are  $V$ -safe and truth-functional connectives are  $V$ -protected. In fact, we need not invoke Theorem 7 in this

case, because all objects in  $D_{s \rightarrow t}$  are quasi-extensional. Of course, the fact that propositional tautologies are intensionally valid just means that the power set of  $D_s$  is a Boolean algebra.

A less trivial example is Aristotelian syllogisms, which are of the form

$$Q_1 \mathbf{u}_1^{e \rightarrow t} \mathbf{u}_2^{e \rightarrow t} \wedge Q_2 \mathbf{u}_3^{e \rightarrow t} \mathbf{u}_4^{e \rightarrow t} \rightarrow Q_3 \mathbf{u}_5^{e \rightarrow t} \mathbf{u}_6^{e \rightarrow t},$$

where  $\mathbf{u}_1^{e \rightarrow t}, \dots, \mathbf{u}_6^{e \rightarrow t}$  are not necessarily distinct variables and  $Q_1, Q_2, Q_3$  are not necessarily distinct constants of type  $(e \rightarrow t) \rightarrow (e \rightarrow t) \rightarrow t$ . Formulas of this form are  $\{\mathbf{u}_1, \dots, \mathbf{u}_6\}$ -safe, so if they are extensionally valid, one can instantiate  $\mathbf{u}_1, \dots, \mathbf{u}_6$  by expressions denoting truly intensional *properties* (functions from individual concepts to sets of possible worlds).

What about first-order logic? Of the usual Hilbert-style axioms,

$$\forall x (\varphi(x) \rightarrow \psi(x)) \rightarrow (\forall x \varphi(x) \rightarrow \forall x \psi(x))$$

is rendered as

$$\forall_e (\lambda \mathbf{v}^e . \mathbf{u}_1^{e \rightarrow t} \mathbf{v}^e \rightarrow \mathbf{u}_2^{e \rightarrow t} \mathbf{v}^e) \rightarrow (\forall_e (\lambda \mathbf{v}^e . \mathbf{u}_1^{e \rightarrow t} \mathbf{v}^e) \rightarrow \forall_e (\lambda \mathbf{v}^e . \mathbf{u}_2^{e \rightarrow t} \mathbf{v}^e)),$$

where  $\rightarrow$  (of type  $t \rightarrow t \rightarrow t$ ) is written as an infix operator. This object language formula is  $\{\mathbf{u}_1^{e \rightarrow t}, \mathbf{u}_2^{e \rightarrow t}\}$ -safe and is hence intensionally valid, assuming the usual interpretation of  $\forall_e$  and  $\rightarrow$ .

In contrast,

$$\forall x \varphi(x) \rightarrow \varphi(t)$$

is rendered as

$$\forall_e (\lambda \mathbf{v}^e . \mathbf{u}^{e \rightarrow t} \mathbf{v}^e) \rightarrow \mathbf{u}^{e \rightarrow t} \mathbf{t}^e,$$

which is not  $\{\mathbf{u}^{e \rightarrow t}, \mathbf{t}^e\}$ -safe. Indeed, it is not intensionally valid, because not all individual concepts are constant functions.

Another axiom that is not intensionally valid, this time from first-order logic with equality, is

$$s = t \rightarrow (\varphi(s) \rightarrow \varphi(t)),$$

which is rendered as

$$\mathbf{s}^e = \mathbf{t}^e \rightarrow (\mathbf{v}^{e \rightarrow t} \mathbf{s}^e \rightarrow \mathbf{v}^{e \rightarrow t} \mathbf{t}^e),$$

where  $=$  stands for the equality between individuals in  $D_e$ . This formula is not  $\{\mathbf{s}^e, \mathbf{t}^e, \mathbf{v}^{e \rightarrow t}\}$ -safe and is not intensionally valid. This is just the well-known failure of substitutivity in intensional contexts.

Here are a couple of more artificial examples. Let  $\mathbf{I}$  be an object language constant that denotes the identity function on  $D_{e \rightarrow t}$  in all models in  $\mathcal{C}$ . Then

$$\mathbf{u}^{((e \rightarrow t) \rightarrow e \rightarrow t) \rightarrow t} \mathbf{I} \rightarrow \mathbf{u}^{((e \rightarrow t) \rightarrow e \rightarrow t) \rightarrow t} (\lambda \mathbf{v}^{e \rightarrow t} . \mathbf{v}^{e \rightarrow t})$$

is extensionally valid in  $\mathcal{C}$ , but not intensionally so. This is because the intensionalization of the identity function on  $D_{e \rightarrow t}$  is not the identity function on  $D_{(s \rightarrow e) \rightarrow s \rightarrow t}$ . Note that this formula is not  $\{\mathbf{u}^{((e \rightarrow t) \rightarrow e \rightarrow t) \rightarrow t}\}$ -safe because  $\lambda \mathbf{v}^{e \rightarrow t} . \mathbf{v}^{e \rightarrow t}$  is not  $\emptyset$ -protected.

Another example is

$$(\mathbf{u}^{t \rightarrow t} \top \wedge \mathbf{u}^{t \rightarrow t} \perp) \rightarrow \mathbf{u}^{t \rightarrow t} (\neg \mathbf{v}^t),$$

where  $\top, \perp, \wedge, \rightarrow, \neg$  have the usual interpretation. This formula is extensionally valid, but not intensionally so. Observe that it is not  $\{\mathbf{u}^{t \rightarrow t}, \mathbf{v}^t\}$ -safe because  $\text{FV}(\neg \mathbf{v}^t) \cap \{\mathbf{u}^{t \rightarrow t}, \mathbf{v}^t\} \neq \emptyset$ .

## References

- Ben-Avi, Gilad and Yoad Winter. 2007. The semantics of intensionalization.  
In Reinhard Muskens, editor, *Workshop on New Directions in Type-Theoretic Grammars*, pages 98–112.
- van Eijck, Jan and Christina Unger. To appear. *Computational Semantics with Functional Programming*. Cambridge: Cambridge University Press.