A Theory of Names and True Intensionality

Reinhard Muskens

Tilburg Center for Logic and Philosophy of Science
r.a.muskens@uvt.nl
http://let.uvt.nl/general/people/rmuskens/

Abstract. Standard approaches to proper names, based on Kripke's views, hold that the semantic values of expressions are (set-theoretic) functions from possible worlds to extensions and that names are rigid designators, i.e. that their values are *constant* functions from worlds to entities. The difficulties with these approaches are well-known and in this paper we develop an alternative. Based on earlier work on a higher order logic that is *truly intensional* in the sense that it does not validate the axiom scheme of Extensionality, we develop a simple theory of names in which Kripke's intuitions concerning rigidity are accounted for, but the more unpalatable consequences of standard implementations of his theory are avoided. The logic uses Frege's distinction between sense and reference and while it accepts the rigidity of names it rejects the view that names have direct reference. Names have constant denotations across possible worlds, but the semantic value of a name is not determined by its denotation.

1 Introduction

Standard approaches to proper names, based on Kripke (1971, 1972), make the following three assumptions.

(a) The semantic values of expressions are (possibly partial) functions from possible worlds to extensions.
(b) These functions are identified with their *graphs*, as in set theory.
(c) Names are rigid designators, i.e. their extensions are world-independent.

In particular, the semantic values of names are taken to be *constant* functions from worlds to entities, possibly undefined for some worlds.

The difficulties resulting from these assumptions are well-known. On the one hand, there are general 'logical omniscience' problems with the possible worlds approach resulting from (a) + (b). Since functions, in the set-theoretic conception, are extensional entities, with their identity criteria given by input-output behaviour, the semantic values of far too many expressions will be identified. Implications and their contrapositives, for example, will be lumped together. That is incorrect since one may very well believe $p \to q$ but fail to believe $\neg q \to \neg p$, so that there is at least one property the semantic values of these expressions do not have in common.

Adding (c) as a further restriction makes things worse, since if the semantic value of a name depends only on its bearer it is predicted that names with the same bearer can be substituted for one another in *any* context. This leads to philosophers claiming and dogmatically defending the position that the Ancients *did* know that Hesperus was Phosphorus before that identity was actually discovered, an armchair intuition that does not seem to be shared by many outside the profession. It also leads to the prediction that the following are equivalent.

(1) a. We do not know *a priori* that Hesperus is Phosphorus
b. We do not know *a priori* that Phosphorus is Phosphorus

(1a) is asserted in Kripke (1972, page 308); (1b) is obviously false. Traditional theorists are therefore confronted with the challenge to come up with a logic in which the values of (1a) and (1b) can be distinguished. No precise system seems to have been developed thus far.

The substitutivity problems that follow from the adoption of (a)–(c) show that this combination cannot stand, but this does not mean, of course, that (c), the idea that names denote rigidly, has to go. In this paper I will sketch a theory that does not suffer from the many problems that are connected with identifying intensions with certain functions in extension, but in which it is still possible to consistently formalize the intuition that names denote rigidly.

2 A Truly Intensional Logic

We move to a (higher order) logic that is *truly intensional*. By this we mean that the following axiom (schema) of extensionality fails.[1]

(2) $\forall XY(\forall \vec{x}(X\vec{x} \leftrightarrow Y\vec{x}) \rightarrow \forall Z(ZX \rightarrow ZY))$

There are now several approaches to type theory that manage to avoid making (2) valid. Fitting (2002) and Benzmüller et al. (2004) are two of them, but since both of these papers interpret the central machinery of type logic in some non-standard way,[2] the logic used here will be the ITL of Muskens (2007). In this logic all operators have standard interpretations and in fact the interpretation of the logic is a rather straightforward generalisation of that of Henkin (1950), making (2) invalid but retaining all classical rules for logical operators. The following somewhat impressionistic description mainly highlights ITL's minor differences with standard simple type theory. For precise definitions consult Muskens (2007).

[1] Our notion of *true intensionality* is just the notion of *intensionality* defined in Whitehead and Russell (1913), but using that term without modification may lead to confusion nowadays, as the word is now widely used for Carnap's imperfect approximation of the original concept. I will mostly, but not always, use *true intensionality* for *intensionality* in this paper. *Hyperintensionality*, another word for the same idea, is less than felicitous, as it suggests a property stronger than intensionality, while it is only stronger than Carnap's approximation.

[2] Fitting's (2002) interpretation of lambda abstraction is non-standard while Benzmüller et al.'s (2004) interpretation of application is.

$$\frac{\Pi \Rightarrow \Sigma}{\Pi' \Rightarrow \Sigma'}\,[W], \quad \text{if } \Pi \subseteq \Pi',\ \Sigma \subseteq \Sigma'$$

$$\frac{}{\Pi, \varphi \Rightarrow \Sigma, \varphi}\,[R] \qquad \frac{}{\Pi, \bot \Rightarrow \Sigma}\,[\bot\mathsf{L}]$$

$$\frac{\Pi, A\{x := B\}\vec{C} \Rightarrow \Sigma}{\Pi, (\lambda x.A)B\vec{C} \Rightarrow \Sigma}\,[\lambda\mathsf{L}] \qquad \frac{\Pi \Rightarrow \Sigma, A\{x := B\}\vec{C}}{\Pi \Rightarrow \Sigma, (\lambda x.A)B\vec{C}}\,[\lambda\mathsf{R}]$$

if B is free for x in A — if B is free for x in A

$$\frac{\Pi, B\vec{C} \Rightarrow \Sigma \qquad \Pi \Rightarrow \Sigma, A\vec{C}}{\Pi, A \subset B \Rightarrow \Sigma}\,[\subset\mathsf{L}] \qquad \frac{\Pi, A\vec{c} \Rightarrow \Sigma, B\vec{c}}{\Pi \Rightarrow \Sigma, A \subset B}\,[\subset\mathsf{R}]$$

if the constants $\vec{c}$ are fresh

Table 1. Gentzen rules for ITL.

Type system ITL's type system is *relational*, rather than *functional*. Given some set of basic types, further types are formed by the rule that $\langle \alpha_1 \dots \alpha_n \rangle$ is a type if $\alpha_1, \dots, \alpha_n$ are. Objects of type $\langle \alpha_1 \dots \alpha_n \rangle$ are n-ary relations in intension that take objects of type α_k in their k-th argument place. Readers familiar with functional type logics may identify $\langle \alpha_1 \dots \alpha_n \rangle$ with $\alpha_1 \to \dots \to \alpha_n \to t$, where t is the type of propositions and truth values (and association is to the right). In linguistic semantics this would be written without the arrows: $\alpha_1 \dots \alpha_n t$. So $\langle e \rangle$ is the type of unary and $\langle eee \rangle$ the type of ternary relations in intension of type e objects, while $\langle\langle e \rangle\rangle$ is the type of properties of properties of individuals (quantifiers). The type $\langle\rangle$ is a limiting case. It corresponds to t in the functional set-up. Objects of this type are propositions, and their extensions are truth values.

Language Terms of the logic are built up in the usual way from variables and non-logical constants with the help of application, λ-abstraction and a few logical constants, here $\subset$ and $\bot$ ($\subset$ is meant to denote inclusion of extensions and $\bot$ will be a proposition that is always false). Typing of terms is as expected, given the correlation between relational and functional types that was just described. For example, $(\lambda x.A)$ is of type $\langle \alpha_1 \alpha_2 \dots \alpha_n \rangle$ if A is of type $\langle \alpha_2 \dots \alpha_n \rangle$ and x is of type α_1, while (AB) is of type $\langle \alpha_2 \dots \alpha_n \rangle$ if A is of type $\langle \alpha_1 \alpha_2 \dots \alpha_n \rangle$ and B is of type α_1. Successive applications can 'eat up' all the argument places of a relation until $\langle\rangle$ is reached.

Intensional models Models for ITL distinguish between the *intension* of a term (given an assignment) and the *extension* associated with that intension.

$$\frac{}{\Pi \Rightarrow \Sigma, \top}\,[\top\mathsf{R}]$$

$$\frac{\Pi, \psi \Rightarrow \Sigma \qquad \Pi \Rightarrow \Sigma, \varphi}{\Pi, \varphi \to \psi \Rightarrow \Sigma}\,[\to\mathsf{L}] \qquad \frac{\Pi, \varphi \Rightarrow \Sigma, \psi}{\Pi \Rightarrow \Sigma, \varphi \to \psi}\,[\to\mathsf{R}]$$

$$\frac{\Pi, \varphi\{x := A\} \Rightarrow \Sigma}{\Pi, \forall x\varphi \Rightarrow \Sigma}\,[\forall\mathsf{L}] \qquad \frac{\Pi \Rightarrow \Sigma, \varphi\{x := c\}}{\Pi \Rightarrow \Sigma, \forall x\varphi}\,[\forall\mathsf{R}]$$

where c is fresh

$$\frac{\Pi, A \doteq B \Rightarrow \Sigma, \varphi\{x := A\}}{\Pi, A \doteq B \Rightarrow \Sigma, \varphi\{x := B\}}\,[=\mathsf{L}] \qquad \frac{}{\Pi \Rightarrow \Sigma, A = A}\,[=\mathsf{R}]$$

where $A \doteq B$ is $A = B$ or $B = A$

Table 2. Some classical rules derivable in ITL.

Fitting (2002) uses a similar way of defining models and the set-up is strongly reminiscent of that of Frege (1892).

Collections of domains will be sets $\{D_\alpha \mid \alpha \text{ is a type}\}$ of pairwise disjoint non-empty sets. There is no further restriction on collections of domains and in particular sets $D_{\langle\alpha_1\ldots\alpha_n\rangle}$ need *not* consist of relations over lower type domains, as is the case in the usual Henkin models. *Assignments* and notation for assignments are defined as usual. *Intension functions* are defined to be functions that send terms and assignments to elements of the D_α and respect the following constraints.

- $I(a, A) \in D_\alpha$, if A is of type α
- $I(a, x) = a(x)$, if x is a variable
- $I(a, A) = I(a', A)$, if a and a' agree on all variables free in A
- $I(a, A\{x := B\}) = I(a[I(a, B)/x], A)$, if B is free for x in A

These constraints are still very liberal and do not amount to the constraints imposed by the usual Tarski definition.

The next step associates extensions with intensions. For each $\alpha = \langle\alpha_1 \ldots \alpha_n\rangle$, a function $E_\alpha : D_\alpha \to \mathcal{P}(D_{\alpha_1} \times \cdots \times D_{\alpha_n})$ is called an *extension function.* A triple consisting of a collection of domains, an intension function, and a family of extension functions is called a *generalised frame.* (Note that in a generalised frame $E_{\langle\rangle} : D_{\langle\rangle} \to \{0, 1\}$, if some standard identifications are made.) Generalised frames are *intensional models* if, for all $\alpha = \langle\alpha_1 \ldots \alpha_n\rangle$, and for all terms A of type α, the extensions $E_\alpha(I(a, A))$, for which we write $V(a, A)$, satisfy the following constraints.

- $V(a, \bot) = 0$
- $V(a, AB) = \{\langle\vec{d}\rangle \mid \langle I(a, B), \vec{d}\rangle \in V(a, A)\}$
- $V(a, \lambda x_\beta.A) = \{\langle d, \vec{d}\rangle \mid d \in D_\beta \text{ and } \langle\vec{d}\rangle \in V(a[d/x], A)\}$

– $V(a, A \subset B) = 1 \Longleftrightarrow V(a, A) \subseteq V(a, B)$

These last clauses constrain extensions to behave as in the usual Tarski value definition. For the treatment of abstraction and application in a relational setting, see also Muskens (1995).

Entailment is defined in the usual way, with the help of intensional models. The rules for λ-*conversion*, (α), (β) and (η), do not automatically hold (they preserve extension, but not necessarily intension), but it is possible to consistently add them to the logic. Extensionality is not universally valid. This is because the functions $E_\alpha : D_\alpha \to \mathcal{P}(D_{\alpha_1} \times \cdots \times D_{\alpha_n})$ need not be injective. In fact, intensional models in which all extension functions are injective essentially are Henkin's general models, while a further requirement of surjectivity will give full models.

Proofs The Gentzen calculus in Table 1 is generalised complete for the semantic notion of entailment just defined (see Muskens (2007) for a proof). Table 2 gives derived rules for some operators defined from the two primitives $\subset$ and $\bot$. The identity here is Leibniz identity, having the same properties, i.e. $A = B$ is short for $\forall Z(ZA \to ZB)$.

3 Names in a Truly Intensional Setting

Given a truly intensional logic such as the one just defined, a theory of names can take the following form.

- Ordinary proper names are *predicates*.
- They are *singular* in the sense that their extensions are either empty or singletons.
- Meanings are represented by lambda terms and combine with the help of application and *type shifters*.
- Among the type shifters is Partee's type shifter A, i.e. $\lambda P'P.\exists x(P'x \wedge Px)$ (Partee, 1986).[3]

In defending a theory of names as predicates I side with Aristotle, I think, and the idea seems linguistically natural, as names accept modification, combine with determiners, etc., just like common nouns. Singularity can be enforced by adopting the following constraint, for all names N.

(3) $\forall xy((\mathsf{N}x \wedge \mathsf{N}y) \to x = y)$

The type shifter A provides the glue that is needed to get predication going. (4) provides a simple example. Let's say *Zeus* translates as the predicate Z (4a); then combining with Partee's type shifter $\lambda P'P.\exists x(P'x \wedge Px)$ leads to the translation

[3] We generally use Q as a variable over type $\langle\langle e\rangle\rangle$ (quantifiers), P as a variable of type $\langle e\rangle$ (properties of individuals), R as a variable of type $\langle ee\rangle$ (binary relations in intension of individuals), and x, y and z as variables of type e (individuals).

in (4b) and a further combination with the translation of *smiles*, S say, to that in (4c) (here it is assumed that the rules of λ-conversion have indeed been added to the logic).

(4) a. Zeus $\rightsquigarrow \mathsf{Z}$
b. Zeus $\rightsquigarrow \lambda P.\exists x(\mathsf{Z}x \wedge Px)$
c. Zeus smiles $\rightsquigarrow \exists x(\mathsf{Z}x \wedge \mathsf{S}x)$

(4c) also illustrates how non-referring names are dealt with. Atheists denying the existence of Zeus can consistently claim the statement $\exists x(\mathsf{Z}x \wedge \mathsf{S}x)$ to be false, a possibility that was also provided for in Russell (1905), but is not available in theories that translate names as individual constants.

Let us look at identity statements, such as the infamous *Hesperus is Phosphorus* case. In (5a) we translate *Phosphorus* as Φ, a translation that, I take it, can be inherited by *is Phosphorus*.[4] The translation in (5c) is then obtained in a way analogous to the one in (4).

(5) a. Phosphorus $\rightsquigarrow \Phi$
b. is Phosphorus $\rightsquigarrow \Phi$
c. Hesperus is Phosphorus $\rightsquigarrow \exists x(\mathsf{H}x \wedge \Phi x)$
d. Hesperus is Hesperus $\rightsquigarrow \exists x(\mathsf{H}x \wedge \mathsf{H}x)$

Note that it is consistent to assume that the semantic value of *Hesperus is Phosphorus*, $\exists x(\mathsf{H}x \wedge \Phi x)$, and that of *Hesperus is Hesperus*, given in (5d) as $\exists x(\mathsf{H}x \wedge \mathsf{H}x)$, are completely distinct propositions, even if the first is true. Also, while $\forall x(\mathsf{H}x \leftrightarrow \Phi x)$ follows from $\exists x(\mathsf{H}x \wedge \Phi x)$ and the singularity requirement for name denotations (from which we get $\forall xy((\mathsf{H}x \wedge \mathsf{H}y) \rightarrow x = y)$ and $\forall xy((\Phi x \wedge \Phi y) \rightarrow x = y)$), it does *not* follow that $\mathsf{H} = \Phi$ and H may well have properties that Φ lacks or vice versa. Co-extensionality crucially does not entail identity, having the same properties, in our theory and the theory allows for the possibility that Phosphorus has, but Hesperus fails to have, the property (λX.we do not know *a priori* that Hesperus is X), as in (1).

3.1 Worlds, Necessity, and Rigidity

Possible worlds are not needed to obtain true intensionality, and in fact cannot provide it, but they are immensely useful for modeling all kinds of *modal* phenomena. Here we construct them as certain properties of propositions (see

[4] A traditional way to obtain the translation of *is Phosphorus* from that of *Phosphorus* is to start with (5a), to then observe that Partee's A shifter allows for an interpretation of *Phosphorus* as $\lambda P.\exists x(\Phi x \wedge Px)$, as in the *Zeus* case. To the latter we could apply the linear combinator $\lambda Q\lambda R\lambda x.Q(\lambda y.Ryx)$, which is generally useful for combining transitive verb meanings with the meanings of their direct objects. This would result in a translation $\lambda R\lambda x.\exists z(\Phi z \wedge Rzx)$, which, combined with the translation of *is*, $\lambda xy.x = y$, would lead to $\lambda x.\exists z(\Phi z \wedge z = x)$. The latter is extensionally, but not intensionally, equivalent to Φ.

Muskens (2007) for more details). Propositions have type $\langle\rangle$, so properties of propositions have type $\langle\langle\rangle\rangle$, and the property of being a world, a property of properties of propositions has type $\langle\langle\langle\rangle\rangle\rangle$. We will write Ω for this special property and stipulate the following.

W1 $\forall w(\Omega w \rightarrow \neg w\bot)$
W2 $\forall w(\Omega w \rightarrow (w(A \subset B) \leftrightarrow \forall \vec{x}(w(A\vec{x}) \rightarrow w(B\vec{x}))))$

W1 requires world extensions to be consistent while addition of W2 makes worlds 'distribute over logical operators'. Statements such as the following become derivable.

a. $\forall w(\Omega w \rightarrow (w(\neg\varphi) \leftrightarrow \neg(w\varphi)))$
b. $\forall w(\Omega w \rightarrow (w(\varphi \wedge \psi) \leftrightarrow ((w\varphi) \wedge (w\psi))))$
c. $\forall w(\Omega w \rightarrow (w(\forall x\varphi) \leftrightarrow \forall x(w\varphi)))$
d. $\forall w(\Omega w \rightarrow (w(\exists x\varphi) \leftrightarrow \exists x(w\varphi)))$

The first of these statements says that worlds are complete, while the last two are 'Henkin properties' that enforce, for example, that if an existential proposition is an element of the extension of a given world some proposition witnessing the existential must also be an element. In general, given W1 and W2, worlds single out sets of propositions that could be simultaneously true. The term $\lambda p.p$ (with p of type $\langle\rangle$) will be a world if we assume $\Omega(\lambda p.p)$ and it will then have the function of the *actual* world, as, in any model, $\lambda p.p$ will hold of φ iff φ is indeed true. Let us make $\Omega(\lambda p.p)$ into an official postulate and let's consider two more.

W3 $\Omega(\lambda p.p)$
W4 $\forall ww'((\Omega w \wedge \Omega w') \rightarrow (w(w'\varphi) \leftrightarrow (w'\varphi)))$
W5 $\forall w(\Omega w \rightarrow \forall w'(\Omega w' \leftrightarrow w(\Omega w')))$

W4 says that whether a proposition holds in a world is a global property, and W5 says something similar about the question whether a property of propositions is a world.

Once worlds are introduced, it becomes useful to associate *domains* with them. Some objects may exist in some worlds but not in others. We introduce a constant E of type $\langle e\rangle$ that will function as an existence predicate. Quantification over *existing* objects can then be obtained by relativizing to E. For example, the type shifter A may now be redefined as $\lambda P'P.\exists x(\mathsf{E}x \wedge P'x \wedge Px)$. This will lead to slightly revised translations, e.g. *Hesperus is Phosphorus* will now go to $\exists x(\mathsf{E}x \wedge \mathsf{H}x \wedge \mathsf{\Phi}x)$.

Having worlds at our disposal, we can now express that φ is globally necessary by writing $\forall w(\Omega w \rightarrow w\varphi)$ and we may abbreviate this as $\Box\varphi$.[5] The following scheme says that names have singleton extensions in all worlds.

(6) $\Box\exists x\forall y(\mathsf{N}y \leftrightarrow y = x)$

[5] Muskens (2007) discusses modalities based on accessibility relations, but here we can make do without these.

This entails (3) but no longer leaves open the possibility of empty denotation that was useful for non-referring names. Since we now have an existence predicate at our disposal, that possibility is no longer needed.

We now come to rigidity. There are various ways to model variants of the notion. Here is a strong and straightforward one.

(7) $\exists x \Box \forall y(\mathsf{N}y \leftrightarrow y = x)$

The idea is that for all names there is a possible object o such that the name's extension is $\{o\}$ across all possible worlds. Clearly, in the presence of this requirement $\exists x(\mathsf{E}x \wedge \mathsf{H}x \wedge \Phi x)$ will entail $\Box\exists x(\mathsf{H}x \wedge \Phi x)$, so if Hesperus is Phosphorus, it is necessary that Hesperus is Phoshorus wherever it exists and the usual Kripkean intuitions are formalised.

On the other hand codesignating names cannot be replaced for one another in arbitrary contexts. While Hesperus and Phosphorus have the same extension in all possible worlds, they may still have distinct intensions, as intension is not determined by extension, not even by extension in all possible worlds. And since $\exists x(\mathsf{E}x \wedge \mathsf{H}x \wedge \Phi x)$, and $\exists x(\mathsf{E}x \wedge \mathsf{H}x \wedge \mathsf{H}x)$ are completely distinct propositions it is possible, for example, to bear the relation of belief to the second but not to the first.

4 Conclusion

In this paper I have shown that Kripke's intuitions with respect to the rigid designation of proper names can be formalised in a way that does not result in a theory predicting the intersubstitutivity of codesignating names in arbitrary contexts. This means that this intersubstitutivity does not follow from the intuitions. The theory I have developed accepts rigidity of names, but rejects the Millian idea of direct reference, the idea that the meaning of a name is its bearer or at least is determined by its bearer. In the present theory a person can have many names, all with different intensions.

Bibliography

C. Benzmüller, C. E. Brown, and M. Kohlhase. Higher Order Semantics and Extensionality. *Journal of Symbolic Logic*, 69, 2004.

M. Fitting. *Types, Tableaus, and Gödels God.* Kluwer Academic Publishers, Dordrecht, 2002.

G. Frege. Über Sinn und Bedeutung. In G. Patzig, editor, *Funktion, Begriff, Bedeutung. Fünf Logische Studien.* Vanden Hoeck, Göttingen, 1892.

L. Henkin. Completeness in the Theory of Types. *Journal of Symbolic Logic*, 15:81–91, 1950.

S. Kripke. Identity and Necessity. In M. Munitz, editor, *Identity and Individuation*, pages 135–164. New York University Press, 1971.

S. Kripke. Naming and Necessity. In D. Davidson and G. Harman, editors, *Semantics of Natural Language*, pages 253–355. Reidel, Dordrecht, 1972.

R.A. Muskens. *Meaning and Partiality.* CSLI, Stanford, 1995.

Reinhard Muskens. Intensional Models for the Theory of Types. *Journal of Symbolic Logic*, 72(1):98–118, 2007.

B. Partee. Noun Phrase Interpretation and Type Shifting Principles. In J. Groenendijk, D. de Jongh, and M. Stokhof, editors, *Studies in Discourse Representation and the Theory of Generalized Quantifiers*, pages 115–143. Foris, Dordrecht, 1986.

B. Russell. On Denoting. *Mind*, 14(56):479–493, 1905.

A. N. Whitehead and B. Russell. *Principia Mathematica.* Cambridge University Press, 1913.