

María Manzano and Manuel Crescencio Moreno

## IDENTITY, EQUALITY, NAMEABILITY AND COMPLETENESS. PART II\*

### Abstract

This article is a continuation of our promenade along the winding roads of identity, equality, nameability and completeness. We continue looking for a place where all these concepts converge.

We assume that identity is a binary relation between objects while equality is a symbolic relation between terms. Identity plays a central role in logic and we have looked at it from two different points of view. In one case, identity is a notion which has to be defined and, in the other case, identity is a notion used to define other logical concepts.

In our previous paper, [16], we investigated whether identity can be introduced by definition arriving to the conclusion that only in full higher-order logic with standard semantics a reliable definition of identity is possible.

In the present study we have moved to modal logic and realized that here we can distinguish in the formal language between two different equality symbols, the first one shall be interpreted as extensional genuine identity and only applies for objects, the second one applies for non rigid terms and has the characteristic of synonymy. We have also analyzed the hybrid modal logic where we can introduce rigid terms by definition and can express that two worlds are identical by using the nominals and the @ operator.

We finish our paper in the kingdom of identity where the only primitives are lambda and equality. Here we show how other logical concepts can be defined in terms of the identity relation.

---

\*This research has been possible thanks to the research projects sustained by Ministerio de Economía y Competitividad of Spain with references FFI2013-47126-P and FFI2017-82554-P

We have found at the end of our walk a possible point of convergence in the logic Equational Hybrid Propositional Type Theory (EHPTT), [14] and [15].

*Keywords:* identity, equality, completeness, nameability, first-order modal logic, hybrid logic, hybrid type theory, equational hybrid propositional type theory.

## 1. Introduction

We assume the distinction between identity and equality: the first is a binary relation between objects while the second is a symbolic relation between terms. Owing to the central role the notion of identity plays in logic, you can be interested either in how to define it using other logical concepts (identity as *definiendum*) or in the opposite scheme (identity as *definiens*).

In our previous paper, *Identity, equality, nameability and completeness*, [16], we focused on the definition of identity by means of other logical concepts in classical logic (first-order logic as well as higher-order logic). We concluded from this approach that, in case you want proper identity as interpretation of equality, you need higher-order logic with standard semantics.

The first part of this article is centered on the same issue but we have moved from classical logic to modal logic and hybrid logic. In the last part of this article, identity is not considered to be the *definiendum* but the *definiens*. Identity is there taken as a primitive logical concept and hence we are interested in the definition of the other logical concepts (connectives and quantifiers) in terms of the equality symbol (interpreted as the identity relation), using the lambda abstraction as well.

## 2. Identity and Equality in Modal Logic

In modal logic one can decide to keep the interpretation of equality as extensional identity so that the equality symbol that we have in the language designates properly the “ontological” relation of identity which is only held by an object with itself and with nothing more.<sup>1</sup> As we saw in [16], in first-order logic identity is taken as primitive because it is undefinable and

---

<sup>1</sup>That is the reason why Wittgenstein in [19] rejected the use of the equality sign: “Identity of objects I express by identity of sign, and not by using a sign of identity.

the calculus should include the rules of reflexivity of equality and equals substitution. These two rules guarantee that equality is an equivalence relation. Are they plausible axioms in modal contexts? What about formulas with the box operator?

In case we wanted to interpret equality as proper identity, we must ensure that the rules of *necessity of identity* (**NI**)

$$x = y \rightarrow \Box(x = y)$$

and *substitutivity of identicals* (**SI**)

$$x = y \rightarrow (\varphi(x) \rightarrow \varphi(y))$$

are observed.

## 2.1. First-Order Modal Logic

The rule of the substitutivity of identicals (**SI**), which is close to a first-order formulation of the principle of the indiscernibility of identicals – saying that two objects are identical when there is no property able to distinguish them – cannot capture completely the identity relation, even in classical first-order logic.

On the other hand, the rule of the necessity of identity (**NI**) seems very reasonable when we are talking about two identical things. If two things are identical, then it is necessary that they are identical. This rule gives us the possibility of substituting equals for equals in modal contexts, therefore we do not want to lose (**NI**). However, there is a problem with this rule: (**NI**) is valid in first-order modal logic when equality denotes extensional identity, but it is a source of some logical puzzles like the famous one posed by Frege regarding the “morning star” and the “evening star”. Considering that *m* stands for “morning star” and *e* for “evening star” and knowing that both names designate the very same thing, i.e., the planet Venus, we can obtain as an instance of (**NI**) the following modal sentence:<sup>2</sup>

$$m = e \rightarrow \Box(m = e)$$

The puzzle appears when we give to  $\Box$  an interpretation in terms of propositional attitudes like “the ancient Greeks knew”. In this case, from

---

Difference of objects I express by difference of signs.” (§ 5.53). In fact he indicates: “to say of two things that they are identical is nonsense, and to say of one thing that it is identical with itself is to say nothing at all” (§ 5.5303).

<sup>2</sup>A good explanation of this issue can be found in [7], pp. 140–150.

the previous sentence and the fact that  $m = e$ , which is supposed to be true because both denote Venus, then by *Modus Ponens* we have as conclusion that

$$\Box(m = e)$$

which can be read as:

The ancient Greeks knew that the morning star was the evening star

which is a false proposition, since the identification of the morning star and the evening star with the very same object was something discovered in subsequent years.

The other rule, **(SI)**, can also be exemplified with our previous notation as

$$m = e \rightarrow (\varphi(m) \rightarrow \varphi(e))$$

which can be read as

if the morning star and the evening star denote the same thing,  
then what can be said of the morning star can also be said of  
the evening star

but this is not true because if we interpret  $\varphi$  to be “was known to be the morning star by the ancient Greeks”, the truth value of  $\varphi(m)$  is true but the truth value of  $\varphi(e)$  is just false.

Therefore, regarding modal logic and equality we have some issues to solve if we do not want to reject any support to modal logic – as Quine, [17], for example, did. When we study the previous puzzles in depth, what turns out to be problematic is the introduction of terms other than variables.

We took this example because of its paradigmatical character even though it can be misleading, as here there are two different problems involved. One problem has to do with descriptions and we know that even in classical first-order logic descriptions are difficult terms to deal with. But the **(NI)** presents also some difficulties when we have function symbols and constants whose denotation vary from world to world. The same formula

$$m = e \rightarrow \Box(m = e)$$

is false when we interpret  $m$  as  $1 + 1$  and  $e$  as 10 because  $1 + 1 = 10$  is true in binary number system but false in other number systems, like the

decimal one. If we give to  $\square$  an interpretation in terms of “all number systems”, this formula

$$1 + 1 = 10 \rightarrow \square(1 + 1 = 10)$$

does not qualify for an instance of a reliable axiom. As we have already mentioned, the formula is problematic when we use terms other than rigid variables.

Since identity is not definable in first-order modal logic, it has to be introduced as primitive. If we want to interpret equality as the identity relation, then we need to consider normal models. In a normal model

$$\mathcal{M} = \langle W, R, D, F \rangle$$

the equality symbol would always be interpreted as the identity relation on the domain:

$$(F(=))(w) = Id_D$$

for each  $w \in W$ . In such a model the formula

$$x = y$$

is true at the world  $w$  in the structure  $\mathcal{M}$  under assignment  $g$  if and only if the identity relation holds between the objects denoted by the application of the assignment function  $g$  to the variables:

$$[[x = y]]^{\mathcal{M}, g, w} = T \text{ iff } g(x) = g(y)$$

Since the interpretation of variables range over the objects in the domain of the model, there is no problem in assuming **(NI)** and **(SI)**, because both rules are valid when we are dealing with objects: if two objects are identical then they are identical at every world of the model and they can also be substituted without change in the truth value of a given formula. When we move to formulas with terms other than variables, **(NI)** and **(SI)** become untenable. Intensional terms, whose interpretation varies at each world, were the cause of the puzzles we have previously indicated.

One approach to the combination of equality with terms is the proposal of Hughes and Cresswell.<sup>3</sup> The radical solution they offer is to eliminate terms other than variables, so narrowing the equality relation to objects.

---

<sup>3</sup>See [13], pp. 330–335.

This method gets rid of the problem, but constant and function symbols have to be eliminated. As a matter of fact, the equality relation is genuine identity, but intensions are not taken into account. However, the authors propose that in case we wanted to deal with intensions in our language we must renounce to the **(NI)** principle and weaken **(SI)** to formulas  $\varphi$  free of modal operators. These systems in which **(NI)** does not hold and **(SI)** is weakened are called *contingent identity systems*. In these systems, equality between terms is a contingent relation.

Another possibility is to introduce two equality symbols, one for variables,  $x = y$ , and another for intensional terms,  $\tau_1 \approx \tau_2$ . Fitting introduces a new equality symbol between intensional terms<sup>4</sup>

$$\tau_1 \approx \tau_2 \leftrightarrow_{Df} \langle \lambda x, y. y = x \rangle (\tau_1, \tau_2)$$

The interpretation in a normal model of the previous relation being

$$[[\tau_1 \approx \tau_2]]^{\mathcal{M},g,w} = T \text{ iff } [[\tau_1]]^{\mathcal{M},g,w} = [[\tau_2]]^{\mathcal{M},g,w}$$

This allows us to compare the behavior of axiom **(NI)** for both equality symbols. The axiom **(NI)** when equality is only between variables

$$(x = y) \rightarrow \Box(x = y)$$

is valid because  $x = y$  affirms that the *objects* that are the values of  $x$  and  $y$  are the same. When the new equality symbol applies for terms other than variables, axiom **(NI)**

$$\tau_1 \approx \tau_2 \rightarrow \Box(\tau_1 \approx \tau_2)$$

is not valid because what  $\tau_1 \approx \tau_2$  states is that  $\tau_1$  and  $\tau_2$  *designate* the same thing, and if they are not rigid, their designation can change from world to world.

In conclusion, whereas  $x = y$  expresses that the objects are the same and can be taken as the identity relation,  $\tau_1 \approx \tau_2$  says that the terms  $\tau_1$  and  $\tau_2$  designate the same object. In this sense, if we want to accept **(NI)** for  $\approx$  we would be expressing a notion stronger than that of simple equality, namely it would have the characteristics of *synonymy*:

$x = y$  asserts that the *objects* that are the values of  $x$  and  $y$  are the same [...]  $t \approx u$  asserts that the terms  $t$  and  $u$  *designate*

---

<sup>4</sup>See [7], pp. 204–205.

the same object, which is quite a different thing. If  $x$  and  $y$  are the same object, the object is, and always will be, one, and hence  $\Box(x = y)$  follows. But if  $t$  and  $u$  happen to designate the same object, they might fail to do so under other circumstances, so  $\Box(t \approx u)$  does not follow. In fact,  $\Box(t \approx u)$  expresses a notion considerably stronger than that of simple equality – it has the characteristics of *synonymy*. ([7], p. 205.)

In conclusion, in first-order modal logic the rules of **(NI)** and **(SI)** apply well to variables and rigid terms but not to non-rigid intensional terms.

## 2.2. Modal Type Theory

In *Types, Tableaux, and Gödel's God*, [8], Melvin Fitting has developed a novel approach to higher-order modal logic. Fitting's work has proved influential. Syntactically, it uses modal operators, but dispenses with the function-argument syntax usual in type theory in favor of a predicate-term syntax reminiscent of first-order logic. However, it is his semantic innovation which is likely to be enduring: the use of intensional models, a mechanism which makes it possible to avoid restrictions to rigid terms.

In his modal type theory Fitting, in [8], p. 92, introduces equality as a primitive predicate and he stipulates that it has to be interpreted as the identity relation on each domain of the family of domains in the model.

Given a modal model

$$\mathcal{M} = \langle W, R, D, F \rangle$$

if  $=_{\langle \alpha, \alpha \rangle}$  is an equality constant symbol,  $F(=_{\langle \alpha, \alpha \rangle})$  is the identity relation on  $[[\alpha, D, W]]$ . Where  $[[\alpha, D, W]]$  is the collection of objects of type  $\alpha$ , which can be extensional or intensional, with respect to  $D$  and  $W$ . This collection is defined as follows:

1. If  $\alpha$  is the type of individuals  $\iota$  – let us remember that Fitting does not have functional types nor a type of proposition – then  $[[\iota, D, W]] = D$ .
2.  $[[\langle \beta_1, \dots, \beta_n \rangle, D, W]] = \wp([[ \beta_1, D, W]] \times \dots \times [[ \beta_n, D, W]])$
3.  $[[\uparrow \alpha, D, W]] = [[\alpha, D, W]]^W$ , where  $\uparrow \alpha$  is an intensional type.

In his type theoretical calculus,<sup>5</sup> Fitting adds equality by including equality axioms in his tableau system. These equality axioms – which are formalizations of the Leibniz's principle – are sentences of the following form

<sup>5</sup>See [8], p. 115 and also p. 69.

$$\forall a \forall b [a = b \leftrightarrow \forall c (c(a) \rightarrow c(b))]$$

where  $=$  is of type  $\langle \alpha, \alpha \rangle$ , for some  $\alpha$ , then  $a$  and  $b$  are of type  $\alpha$  and  $c$  is of type  $\langle \alpha \rangle$ . In this case the equality relation is not the prototypical identity.

### 3. Equality and Identity in Hybrid Logic

#### 3.1. Nominals and the satisfaction operator in Hybrid Logic

In *Modal Logic: A Semantic Perspective* [5] Patrick Blackburn and Johan van Benthem identify the expressive weakness of basic modal logic as the lack of a mechanism to express in the language identity of worlds; neither the accessibility relation between worlds can be referred to. This lack of expressivity is an obvious weak as in the spirit of the Kripke semantics is the consideration of modalities as quantifiers over worlds. The basic hybrid language is the modal solution to this deficiency.

The two innovations of basic hybrid logic are the *nominals* and the *satisfaction operators*.

The first idea is to add special propositional symbols called *nominals* to an orthodox modal language, and to establish that in every model, nominals are true at precisely one world. A nominal *names* its denotation by being true there and there only.

The second idea is to add *satisfaction operators*. These are modalities of the form  $@_i$ , where  $i$  can be any nominal. The intended semantics is as follows: a formula of the form  $@_i\varphi$  says that  $\varphi$  is satisfied at the (unique) world named by the nominal  $i$ . Note that a formula of the form  $@_i\varphi$  is either true at all worlds, or false at all worlds: if  $\varphi$  is true at the world named by  $i$ , then  $@_i\varphi$  is true at all worlds  $w$ ; otherwise it is (everywhere) false.

In particular, formulas of form  $@_ij$ , together with  $@_i\Diamond j$  are extremely important in hybrid logic. The formula  $@_ij$  asserts that  $i$  and  $j$  name the same point. That is,  $@_ij$  is a modal way of expressing what  $i = j$  would express in classical logic. Indeed, it is easy to see that the following basic hybrid formulas, which express the reflexivity, symmetry and transitivity of equality, respectively, are all validities:

$$@_ii \quad @_ij \rightarrow @_ji \quad @_ij \wedge @_jk \rightarrow @_ik$$

On the other hand, formulas of the form  $@_i\Diamond j$  express the accessibility relation and allows to define relevant properties this relation might have;



some of them are undefinable in orthodox modal logic, like irreflexivity or trichotomy.

$$\textcircled{a}_i \neg \diamond i \quad \textcircled{a}_i j \vee \textcircled{a}_i \diamond j \vee \textcircled{a}_j \diamond i$$

Hybrid logic adds then to modal logic mechanisms for naming worlds, asserting equalities between worlds and describing accessibility relations. Since we can create also rigidified terms with the satisfaction operator of the form  $\textcircled{a}_i \tau$ , we can apply to these rigid terms the rules (NI) and (SI) without any problem.

### 3.2. Hybrid Type Theory

In *Completeness in Hybrid Type Theory*, [4], equality is introduced as primitive and its interpretation varies from world to world. Given two meaningful expressions  $a$  and  $b$  of type  $\alpha \neq t$ , we can have as a meaningful expression of type  $t$ :

$$a_\alpha = b_\alpha$$

whose interpretation is

$$[[a_\alpha = b_\alpha]]^{\mathcal{M},g,w} = T \text{ iff } [[a_\alpha]]^{\mathcal{M},g,w} = [[b_\alpha]]^{\mathcal{M},g,w}$$

In the same paper, the axioms referring to equality are restricted to rigids. Rigid expressions have the same denotation at all worlds; good examples are  $\top$ ,  $\perp$ , variables and any expressions prefixed by an  $\textcircled{a}$  operator. The set RIGIDS of **rigid meaningful expressions** is defined inductively as follows:

$$\text{RIGIDS} ::= \perp \mid \top \mid v_\alpha \mid \textcircled{a}_i \theta_\alpha \mid \lambda v_\beta a_\alpha \mid f_{\langle \beta, \alpha \rangle} b_\beta \mid a_\beta = b_\beta \mid \neg \varphi_t \mid \varphi_t \wedge \psi_t \mid \forall v_\alpha \varphi_t,$$

where  $\theta_\alpha \in \text{ME}_\alpha$  and  $a_\alpha, b_\beta, f_{\langle \beta, \alpha \rangle}, \varphi, \psi \in \text{RIGIDS}$ .

As axioms for equality we have reflexivity and rigid substitution:

1. **Reflexivity:**  $\vdash a_\alpha = a_\alpha$ .
2. **Substitution:** For  $a_\alpha, b_\alpha$  rigid,  $\vdash a_\alpha = b_\alpha \rightarrow (d_\gamma \frac{a_\alpha}{v_\alpha} = d_\gamma \frac{b_\alpha}{v_\alpha})$ .

The completeness theorem of this logic follows Henkin's pattern. In particular, we want to prove that any consistent set of sentences has a general model. The first step consists on extending the consistent set to a maximal consistent set, but not any maximal consistent set will do. To deal with the hybrid and modal components of our logic, we want the set to be named,  $\exists$ -saturated and  $\diamond$ -saturated. Using this set of sentences

as an oracle, we build the model its formulas describe. The second step is to establish equivalent relations between meaningful expressions of all types using equality. These equivalence classes are the building blocks of the (higher-order) domain of quantification. From the completeness proof of type theory, in [10], we take the idea of defining a function  $\Phi$  on equivalent classes and simultaneously defining the domains of the hierarchy as functions. From the completeness proof of first-order logic, in [9], we take the idea of building the basic domains out of equivalent classes; namely, classes of nominals for the universe of worlds, and equivalent classes of constants for the individual domain. In the present situation the terms must be rigids. Instead of following the usual procedure via canonical models we use a combined version of the completeness theorem for type theory and first-order logic.

In [6] we describe in depth the relationship between his ideas and that novel completeness proof.

#### 4. The Kingdom of Identity

In [16] we were interested in how to define identity using other logical concepts and we arrived to the conclusion that in case you want the proper identity as interpretation of equality you need higher order logic with standard semantics, and so you sacrifice the calculus.

Now we will follow the opposite scheme. Identity is taken as a primitive logical concept and we are interested in the definition of the other logical concepts (connectives and quantifiers) in terms of the equality symbol (interpreted as the identity relation), using also lambda abstraction. The idea of reducing to identity other logical concepts is an old one that goes back to Tarski, Ramsey and Quine. It was Ramsey who raised the whole subject, as both Henkin<sup>6</sup> and Andrews<sup>7</sup> exemplified with this same Ramsey's quote [18]:

The preceding and other considerations let Wittgenstein to the view that mathematics does not consist of tautologies, but of what he called 'equations', for which I should prefer to substitute 'identities'... (It) is interesting to see whether a theory

---

<sup>6</sup>In [12], p. 33.

<sup>7</sup>In [3], p. 67, he declares that in 1962 he copied that quote of Ramsey into his journal.

of mathematics could not be constructed with identities for its foundations. I have spent a lot of time developing such a theory, and found that it was faced with what seemed to me insuperable difficulties. ([18], p. 18.)

In 1963 Henkin published *A Theory of Propositional Types*, [11]. In this paper he offers a semantics and a complete calculus for a logic exclusively based on lambda and equality. The paper was followed by Andrews, *A reduction of the axioms for the theory of propositional type theory*, [1]. Peter Andrews, in a very personal and touching paper, *A Bit of History Related to Equality as a Logical Primitive*, [3], relates the early history of his and Henkin's interest on the subject in the early sixties as well as the story of his radical reduction of axioms. During the 1961–62 academic year Henkin was at the Institute for Advanced Study in Princeton on a Guggenheim Fellowship. At that time, Andrews was there working on his thesis under the advice of Alonzo Church, who had been the director of Henkin's thesis in 1947. Let us quote Andrews:

On several occasions I suggested to Henkin that he simply incorporate my proofs into his paper, but he insisted that I publish a separate paper presenting these proofs, and he wrote a very complimentary letter to Andrzej Mostowski (the editor of *Fundamenta Mathematicae*) recommending that my paper be published immediately following his own paper. He was very concerned that my paper be easy to read as well as technically correct, and made a number of suggestions about it. After we had discussed a number of ideas related to Axiom H2 ( $(A_0 \equiv T^n) \equiv A_0$ ), Henkin found a way to derive it, but he didn't want to write an addendum to my addendum to his paper, so he told me to simply include his proof of Axiom H2 in my paper. ([3], p. 69).

The identity relation on the set of truth values  $\{T, F\}$  is the biconditional connective. Using identity on truth values we cannot define connectives whose truth tables have a value  $T$  on an odd number of lines, not even with identity and negation. In Lesniewski logic, with identity and quantification over propositions, negation can be defined by  $\forall X_0(\neg X_0 \equiv (X_0 \equiv \forall Y_0 Y_0))$ . When quantifiers over higher propositional types are allowed, expressivity improves and Tarski was able to define conjunction by

$$\forall X_0 Y_0 (X_0 \wedge Y_0 \equiv \forall G_{(0)} [(G_{(0)} X_0 \equiv G_{(0)} T_0) \equiv Y_0])$$

Instead of using quantifiers, one can use Church's lambda abstraction and define not only connectives but also quantifiers in terms of identity and lambda. This was first observed by Quine.

#### 4.1. Lambda and Equality in Propositional Type Theory

We briefly present the hierarchy of propositional types as well as the language and its semantics using Henkin's definition in [11].

The hierarchy of propositional types,  $\mathfrak{PT}$ , is the least class of sets containing  $\mathcal{D}_0$ , which is closed under passage from  $\mathcal{D}_\alpha$  and  $\mathcal{D}_\beta$  to  $\mathcal{D}_{\beta\alpha}$ .  $\mathcal{D}_0$  is the truth values set,  $\mathcal{D}_0 = \{T, F\}$ , while  $\mathcal{D}_{\alpha\beta}$  is the set of all functions which maps  $\mathcal{D}_\beta$  to  $\mathcal{D}_\alpha$ .<sup>8</sup> The formal language contains variables for each propositional type, the lambda abstractor,  $\lambda$ , and a collection of equality constants,  $\mathcal{Q}_{(0\alpha)\alpha}$ , one for each type  $\alpha$ .

In this language, all connectives and quantifiers are definable; that is, using only the biconditional  $\mathcal{Q}_{(00)0}$  and  $\lambda$ , the remaining connectives as well as the quantifiers  $\forall X_\alpha$  – for each propositional variable of any propositional type  $\alpha$  – are presented as defined operators.

DEFINITION 1 (Logical constants). *In [11], pp. 326–327, Henkin defines the following expressions:*<sup>9</sup>

1.  $T^N ::= ((\lambda X_0 X_0) \equiv (\lambda X_0 X_0))$
2.  $F^N ::= ((\lambda X_0 X_0) \equiv (\lambda X_0 T^N))$
3.  $\neg ::= (\lambda X_0 (F^N \equiv X_0))$
4.  $\wedge ::= \lambda X_0 (\lambda Y_0 (\lambda f_{00} (f_{00} X_0 \equiv Y_0)) \equiv (\lambda f_{00} (f_{00} T^N)))$
5.  $\forall X_\alpha A_0 ::= ((\lambda X_\alpha A_0) \equiv (\lambda X_\alpha T^N))$

The intuition behind these definitions is rather clear, with the only possible exception of  $\wedge$ , the conjunction operator. This is probably why, in his expository paper of 1975 [12], Henkin gave another and easier definition of conjunction, this time using set-relational abstractor instead of lambda notation.

The use of Church's  $\lambda$ -notation for functional abstraction is not as widely known as the notions of set – and relational –

<sup>8</sup>Nowadays we don't use this reverse notation for types and we say that the domain of functions from  $\mathcal{D}_\alpha$  to  $\mathcal{D}_\beta$  is  $\mathcal{D}_{\alpha\beta}$ .

<sup>9</sup>We would use the symbol  $\equiv$  instead of  $\mathcal{Q}_{(0\alpha)\alpha}$ , for any  $\alpha$ .

abstraction made familiar by Russell and Whitehead through the **Principia**. It has therefore seemed to me worthwhile to sketch below a ‘translation’ of the ideas of my earlier paper (as improved by Andrews) ([12], pp. 33–34).

The new definition is in page 36 of the same paper [12]:

$$\wedge \text{ abbreviates } \{X_0 Y_0 \mid \forall G_{(0)}[(G_{(0)} X_0 \equiv G_{(0)} T_0) \equiv Y_0]\}$$

The definition of  $\wedge$  offered by Andrews in page 349 of [2] is

$$\wedge_{000} \text{ stands for } [\lambda X_0 \lambda Y_0 (\lambda g_{000}(g_{000} T^N T^N) \equiv [\lambda g_{000}(g_{000} X_0 Y_0)])]$$

and his explanation reads: “The expression  $[\lambda g_{000}(g_{000} X_0 Y_0)]$  can be used to represent the ordered pair  $\langle X_0, Y_0 \rangle$  and the conjunction  $X_0 \wedge Y_0$  is true iff  $X_0$  and  $Y_0$  are both true, i.e., iff  $\langle T^N, T^N \rangle \equiv \langle X_0, Y_0 \rangle$ . Hence  $X_0 \wedge Y_0$  can be expressed by the formula  $[\lambda g_{000}(g_{000} X_0 Y_0)] \equiv [\lambda g_{000}(g_{000} T_0 T_0)]$ .” ([2], p. 160).

### 4.2. Nameability

This very succinct language that only uses  $\lambda$  and  $\equiv$  is so powerful that you can give a proper name to every object in  $\mathfrak{P}\mathfrak{T}$ . This is possible because in this hierarchy each type is finite and the already mentioned problem of having more objects than formulas does not appear. Names and denotations do match: “In particular, we shall associate, with each element  $\mathbf{x}$  of an arbitrary type  $D_\alpha$ , a closed formula  $\mathbf{x}^n$  of type  $\alpha$  such that  $(\mathbf{x}^n)^d = \mathbf{x}$ ” ([11], p. 326).

Names for the basic objects  $T$  and  $F$  of type 0 are given in definition 1 above. Now we wonder what is the name of the function  $\mathbf{f}$  which maps every element of the finite type  $\mathcal{D}_\alpha$  (say  $\mathcal{D}_\alpha = \{\mathbf{y}_1, \dots, \mathbf{y}_q\}$ ) into  $\mathcal{D}_\beta$ . The name for the function  $\mathbf{f}$  which maps every  $\mathbf{y}_i$  to the corresponding  $\mathbf{f}(\mathbf{y}_i)$  in  $\mathcal{D}_\beta$  is given by the formula

$$\begin{aligned} \mathbf{f}^n := & \lambda X_\alpha. \lambda Z_\beta. [(X_\alpha \equiv (\mathbf{y}_1)^n) \wedge (Z_\beta \equiv \mathbf{f}(\mathbf{y}_1)^n)] \vee \dots \\ & \dots \vee [(X_\alpha \equiv (\mathbf{y}_q)^n) \wedge (Z_\beta \equiv \mathbf{f}(\mathbf{y}_q)^n)] \end{aligned}$$

To introduce  $\mathbf{f}^n$  we need to formalize the following: when variable  $X_\alpha$  is just the name of object  $\mathbf{y}_i$  – that is,  $X_\alpha \equiv (\mathbf{y}_i)^n$  – function  $\mathbf{f}$  matches it to the unique  $Z_\beta$  naming  $\mathbf{f}(\mathbf{y}_i)$  – that is,  $Z_\beta \equiv \mathbf{f}(\mathbf{y}_i)^n$ . The names of the objects of types  $\alpha$  and  $\beta$  as well as the descriptor operator are used in

the inductive definition of names for type  $(\beta\alpha)$ . To introduce the last one Henkin defines an election function  $\mathbf{t}^{(\alpha)}$  for each type:

For any arbitrary type  $\alpha$  let  $\mathbf{t}^{(\alpha)}$  be the function of  $\mathcal{D}_{\alpha(0\alpha)}$  such that, for any  $\mathbf{f} \in \mathcal{D}_{0\alpha}$ ,  $(\mathbf{t}^{(\alpha)}\mathbf{f})$  is the unique element  $\mathbf{x} \in \mathcal{D}_{\alpha}$  for which  $(\mathbf{f}\mathbf{x}) = T$ , in case there is such a unique element  $\mathbf{x}$ , or else  $(\mathbf{t}^{(\alpha)}\mathbf{f}) = \mathbf{a}_{\alpha}$  if there is no  $\mathbf{x}$ , or if there are more than one  $\mathbf{x}$ , such that  $(\mathbf{f}\mathbf{x}) = T$ . We shall show inductively that for each  $\alpha$  there is a closed formula  $\iota_{\alpha(0\alpha)}$  such that  $(\iota_{\alpha(0\alpha)})^d = \mathbf{t}^{(\alpha)}$ . Then for any formula  $A_0$  and variable  $X_{\alpha}$  we shall set  $(jX_{\alpha}A_0) = (\iota_{\alpha(0\alpha)}(\lambda X_{\alpha}A_0))$  ([11], p. 328).

In the above definition  $\mathbf{a}_{\alpha}$  serves as the denotation of improper descriptions. The setting of these  $\mathbf{a}_{\alpha}$  is done by induction on types: for type 0 we just take  $\mathbf{a}_0 = F$ , for type  $(\alpha\beta)$  we take the constant function  $\mathbf{f}_{\alpha\beta}$  with value  $\mathbf{a}_{\alpha}$  for every element of  $\mathcal{D}_{\beta}$ , where  $\mathbf{a}_{\alpha}$  is the already chosen element in  $\mathcal{D}_{\alpha}$ . Thus,  $\mathbf{f}_{\alpha\beta}\mathbf{x} = \mathbf{a}_{\alpha}$  for each  $\mathbf{x} \in \mathcal{D}_{\beta}$ .

### 4.3. A calculus for PTT based on identity

For the theory of propositional types, Henkin not only defines all logical constants and quantifiers with lambda and equality, he also gave, for the first time, a complete axiomatic treatment of propositional type theory based on  $\lambda$  and equality rules. The axioms were chosen to express basic properties of equality. Let us quote Henkin's seven axioms and *Replacement Rule*.

- 5.1.1. AXIOM SCHEMA 1.  $A_{\alpha} \equiv A_{\alpha}$
- 5.1.2. AXIOM SCHEMA 2.  $(A_0 \equiv T^N) \equiv A_0$
- 5.1.3. AXIOM SCHEMA 3.  $(T^N \wedge F^n) \equiv F^n$
- 5.1.4. AXIOM SCHEMA 4.  $(g_{00}T^N \wedge g_{00}F^N) \equiv (\forall X_0(g_{00}X_0))$
- 5.1.5. AXIOM SCHEMA 5.  $(X_{\beta} \equiv Y_{\beta}) \rightarrow ((f_{\alpha\beta} \equiv g_{\alpha\beta}) \rightarrow ((f_{\alpha\beta}X_{\beta} \equiv g_{\alpha\beta}Y_{\beta})))$
- 5.1.6. AXIOM SCHEMA 6.  $(\forall X_{\beta}(f_{\alpha\beta}X_{\beta} \equiv g_{\alpha\beta}X_{\beta}) \rightarrow (f_{\alpha\beta} \equiv g_{\alpha\beta}))$
- 5.1.7. AXIOM SCHEMA 7.  $((\lambda X_{\beta}B_{\alpha})A_{\beta}) \equiv C_{\alpha}$ , where  $C_{\alpha}$  is obtained from  $B_{\alpha}$  by replacing each occurrence of  $X_{\beta}$  in  $B_{\alpha}$  by an occurrence of  $A_{\beta}$ , providing no such occurrence of  $X_{\beta}$  is within a part of  $B_{\alpha}$  which is a formula beginning ' $\lambda Y_{\gamma}$ ' where  $Y_{\gamma}$  is a variable free in  $A_{\beta}$ .

5.2. By the Rule of Replacement we refer to the ternary relation on formulas of type 0 which holds for  $\langle A'_0, C_0, D_0 \rangle$  if and only if  $A'_0 = (A_\alpha \equiv B_\alpha)$  for some formulas  $A_\alpha$  and  $B_\alpha$  and  $D_0$  is obtained from  $C_0$  by replacing one occurrence of  $A_\alpha$  by an occurrence of  $B_\alpha$ . When this situation holds for  $\langle A_\alpha \equiv B_\alpha, C_0, D_0 \rangle$  we shall say that  $D_0$  is obtained by Rule R from  $A_\alpha \equiv B_\alpha$  and  $C_0$ . ([11], p. 330).

The already mentioned Andrews improvement, [1], consists in a reduction of the axioms. In particular, Andrews eliminates Axioms 1, 2 and 3, simplifies Axiom 5 in this form

$$(f_{\alpha 0} \equiv g_{\alpha 0}) \rightarrow (h_{0(\alpha 0)} f_{\alpha 0} \equiv h_{0(\alpha 0)} g_{\alpha 0})$$

and replaces Axiom 6 by

$$\forall X_\beta ((f_{\alpha\beta} X_\beta \equiv g_{\alpha\beta} X_\beta) \equiv (f_{\alpha\beta} \equiv g_{\alpha\beta}))$$

What are the basic properties of equality chosen for this calculus? Axiom 4 expresses the fact that there are only two truth values,  $T$  and  $F$  in the domain  $\mathcal{D}_0$  and could be understood as a law of excluded middle. Axiom 5 expresses a form of Leibniz' law of equality. Axiom 6 is just extensionality, and the rule of Replacement allows to replace equals by equals.

What about reflexivity? As you see it was Axiom 1 in Henkin calculus but Andrews proved in [1] that it was derivable from Axiom 7 – in this form  $((\lambda X_0 A_\alpha) X_0) \equiv A_\alpha$  – and the Rule of replacement of equals by equals.

#### 4.4. Completeness in Propositional Type Theory

Now we would like to explain the method Henkin developed in this beautiful proof.

The important result from where the completeness theorem easily follows has the form of a substitution lemma, but instead of being a semantic metatheorem, it is a theorem of the calculus:

LEMMA. Let  $A_\alpha$  be any formula and  $\varphi$  an assignment. Let  $A_\alpha^{(\varphi)}$  be the formula obtained from  $A_\alpha$  by substituting, for each free occurrence of any variable  $X_\beta$  in  $A_\alpha$ , the formula  $(\varphi X_\beta)^n$ . Then  $\vdash A_\alpha^{(\varphi)} \equiv (V(A_\alpha, \varphi))^n$ . ([11], p. 341).

The obvious question we ask is, how to prove that  $\models A_0$  implies  $\vdash A_0$  for any formula of type 0?

PROPOSITION 2. LEMMA *implies completeness.*

PROOF: If  $A_0$  is closed then  $\models A_0$  implies  $V(A_\alpha, \varphi) = T$  for any assignment  $\varphi$ . Thus the lemma gives  $\vdash A_0 \equiv (V(A_\alpha, \varphi))^n$  which turns to be  $\vdash A_0 \equiv T^n$ , where  $T^n$  is the name of the truth value true.

But using the calculus, in particular, axiom 2 – of this form,  $(A_0 \equiv T^n) \equiv A_0$  – and rule of replacement R we obtain the desired result,  $\vdash A_0$ .

In case  $A_0$  were a valid formula but not a sentence, we pass from  $A_0$  to the sentence  $\forall X_{\gamma_1} \dots X_{\gamma_r} A_0$  – where  $freeVar(A_0) = \{X_{\gamma_1} \dots X_{\gamma_r}\}$  –. We know that  $\models \forall X_{\gamma_1} \dots X_{\gamma_r} A_0$ , and using the previous argument,  $\vdash \forall X_{\gamma_1} \dots X_{\gamma_r} A_0$ . Applying the rules of the calculus, we obtain,  $\vdash A_0$ .

## 5. Conclusion

We consider the idea of reducing other logical concepts to identity as a very fertile notion. Following this line of thought, Manzano, Martins and Huertas, in [14], have developed a logic based on Propositional Type Theory (PTT) and on Equational Logic (EL); in both logics, identity plays a central role. The resulting logic: Equational Hybrid Propositional Type Theory (EHPTT), incorporates also hybrid operators in order to deal with the difficulties arising in intensional contexts. The language EHPTT includes as logical symbols  $\equiv$  and  $\lambda$  from PTT and  $\diamond$  and  $@$  from HL. The structures used to interpret the language contain a hierarchy of propositional types, an algebra (a non-empty set with functions) and a Kripke frame. The completeness proof for this logic is proved in [15], incorporating results and techniques introduced by Henkin in [9], [10] and [11]. EHPTT is a good example of how to combine the different elements we have been finding out through our promenade along the roads of identity, equality, nameability and completeness.

## References

- [1] P. B. Andrews, *A reduction of the axioms for the theory of propositional type theory*, **Fundamenta Mathematicae** 52 (1963), pp. 354–350.
- [2] P. B. Andrews, **An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof**, Academic Press, Orlando, San Diego, New York, Austin, Boston, London, Sydney, Tokyo, Toronto (1986).



- [3] P. B. Andrews, *A Bit of History Related to Equality as a Logical Primitive*, [in:] **The Life and Work of Leon Henkin. Essays on His Contributions**, Springer Basil, (2014), pp. 67–71.
- [4] C. Areces, P. Blackburn, A. Huertas and M. Manzano, *Completeness in Hybrid Type Theory*, **Journal of Philosophical Logic** 43/2–3 (2014), pp. 209–238. DOI: 10.1007/s10992-012-9260-4.
- [5] P. Blackburn and J. van Benthem, *Modal Logic: A Semantic Perspective*, [in:] **Handbook of Modal Logic**, Elsevier, (2007), pp. 1–84.
- [6] P. Blackburn, A. Huertas, M. Manzano and F. Jorgensen, *Henkin and Hybrid Logic*, [in:] **The Life and Work of Leon Henkin. Essays on His Contributions**, Springer Basil, (2014), pp. 279–306.
- [7] M. Fitting and R. L. Mendelsohn, **First-Order Modal Logic**, Kluwer Academic Publishers, Dordrecht, Boston, London (1998).
- [8] M. Fitting, **Types, Tableaus, and Gödel’s God**, Kluwer Academic Publishers, Dordrecht, Boston, London (2002).
- [9] L. Henkin, *The completeness of the first order functional calculus*, **The Journal of Symbolic Logic** 14 (1949), pp. 159–166.
- [10] L. Henkin, *Completeness in the theory of types*, **The Journal of Symbolic Logic** 15 (1950), pp. 81–91.
- [11] L. Henkin, *A Theory of Propositional Types*, **Fundamenta Mathematicae** 52 (1963), pp. 323–344. *Errata*, **Fundamenta Mathematicae** 53 (1964), p. 119.
- [12] L. Henkin, *Identity as a logical primitive*, **Philosophia. Philosophical Quarterly of Israel** 5/1–2 (1975), pp. 31–45.
- [13] G. E. Hughes and M. J. Cresswell, **A New Introduction to Modal Logic**, Routledge, London, New York (1996).
- [14] M. Manzano, M. A. Martins and A. Huertas, *A Semantics for Equational Hybrid Propositional Type Theory*, **Bulletin of the Section of Logic** 43:3/4 (2014), pp. 121–138.
- [15] M. Manzano, M. A. Martins and A. Huertas, *Completeness in Equational Hybrid Propositional Type Theory*. To appear in **Studia Logica**.
- [16] M. Manzano and M. C. Moreno, *Identity, Equality, Nameability and Completeness*, **Bulletin of the Section of Logic** 46:3/4 (2017), pp. 169–195.
- [17] W. V. O. Quine, *Reference and Modality*, [in:] **From a Logical Point of View**, Harvard University Press, Cambridge, MA (1953), pp. 139–159.
- [18] F. P. Ramsey, *The Foundation of Mathematics*, **Proceedings of the London Mathematical Society, Ser. 2**, Vol. 25, Part 5, (1926), pp. 338–384. DOI: 10.1112/plms/s2-25.1.338

- [19] L. Wittgenstein, **Tractatus Logico-Philosophicus**, Routledge & Kegan Paul, London (1922). Originally published as *Logisch-Philosophische Abhandlung*, **Annalen der Naturphilosophische** XIV: 3/4 (1921).

Department of Philosophy  
University of Salamanca  
Salamanca, Spain  
e-mail: [mara@usal.es](mailto:mara@usal.es)  
e-mail: [manuelcrescencio@gmail.com](mailto:manuelcrescencio@gmail.com)