teorema
Vol. XXXIII/2, 2014, pp. 137-150
ISSN: 0210-1602
[BIBLID 0210-1602 (2014) 33:2; pp. 137-150]

On Skolem Functions, and Arbitrary Objects. An Analysis of a Kit Fine's Mysterious Claim

Aránzazu San Ginés

RESUMEN

En un momento de su libro de 1985 *Reasoning with Arbitrary Objects*, Kit Fine observa y enfatiza tres, en su opinión, importantes diferencias entre objetos-A y funciones de Skolem. El presente artículo está dedicado, en particular, a la discusión de una de ellas. Según Fine, existen dependencias entre objetos que no pueden ser representadas propiamente por ninguna función. En lo que sigue, analizaremos esta afirmación desde la perspectiva del lenguaje natural y discutiremos la mejora que parece introducir el uso de objetos arbitrarios frente al de funciones de Skolem en el tratamiento de la dependencia.

PALABRAS CLAVE: *lenguaje natural, dependencias, objetos arbitrarios, funciones de Skolem*

ABSTRACT

In 1985, in his book *Reasoning with Arbitrary Objects*, Kit Fine observed and stressed three, in his opinion, important differences between A-objects and Skolem functions. The present paper rests on one of them. According to Fine, there is some kind of dependence relationship between objects that cannot properly be represented by any function. We will analyze this claim from the perspective of natural language, and discuss the improvement that the use of arbitrary objects seemingly provides over Skolem functions in dealing with dependence.

KEYWORDS: Natural Language, Dependencies, Arbitrary Objects, Skolem Functions

I. HINTIKKA, DEPENDENCE, AND SKOLEM FUNCTIONS

Imagine a situation in which a teacher tells the parents of one of his students (hereafter called parents*) the following:

(1) Every student must read a book (any book!) before next Monday.

The teacher is talking about all the students (probably those in his class), about books, and about the relation connecting both of them: READ. But he is not saying anything about specific students and books related to each other. However, the parents do know now that their son must read a book, the one he chooses, before Monday. The reason is that the quantifiers in (1) not only signal the individuals the teacher is talking about, but also their interdependencies. (1) points out a dependence relationship which allows us, given a student, to associate him or her with a book.

Hintikka, who observed this more than thirty years ago, has ever since stressed the importance of dealing formally with the idea of dependent quantifier. In order to do so, he proposed and later on developed with his associates a new semantics of English quantifiers [Hintikka (1974); see also Sandu (1997)] based on an ingenious idea. To explain it, let us go back to (1). We say that the parents* understand the sentence if they are able to make out from it that given a student, he or she must choose a book and read it before Monday. If the teacher is telling the truth, then also their child will have to choose a book and read it. Furthermore, if after their reunion with the teacher they meet another mother who met him before them, and she tells them that her child does not have to read any book, they will think immediately that the teacher lied to them. It seems then, that understanding a sentence has a lot to do with knowing the way to verify and falsify it. It is like a game. Let us conceive that there are two players, *verifier* and *falsifier*, who want to win the game associated with the sentence by, respectively, verifying and falsifying it. I understand the sentence if I know how to play both roles. The sentence is true if, no matter how the *falsifier* plays, the *verifier* can always win, that is, the sentence is true if the (initial) *verifier* has a winning strategy. In particular, (1) will be true if, no matter which student is picked by the *falsifier* in the first place, the *verifier* can always find a book which the former will have to read before Monday. In the context of game-theoretical semantics (GTS), as this semantics is called, the dependence of a book upon every student becomes perfectly clear. In fact, it can be translated into a fragment of an ordinary second-order language as follows:

(1′) $\exists f \forall x \text{Read-before-Monday}(x, f(x))$

f is called a Skolem function, and it codifies the dependence relationship between quantifiers in logical terms. In our example, given a student, x, the book he will read depends only on him. Thus it must be possible to find a function f which associates every student with the book he or she will read.

Once the dependence has been made explicit (through Skolem functions), it obviously becomes easier to avoid it. Observe, for instance, the following formula expressed in an ordinary first-order language:

(2) $\forall x \forall y \exists z \exists t \mathrm{R}(x, z, y, t)$

In order to make z depend only on x (thus being independent on y – i.e. not having access to the information provided by y), and t depend only on y (thus being independent on x – not having access to the information provided by x), we actually need to switch from ordinary first-order language to another language – preferably one in which the dependencies have been made explicit. Again then, making use of Skolem functions we get what we wanted:

(2′) $\exists f \exists g \forall x \forall y \mathrm{R}(x, f(x), y, g(y))$

Nonetheless, Hintikka [(1996), chapter 3] noticed that it is also possible to express this idea of informational independence while remaining in a first-order language. To do that, he introduces a new notation item: the slash /. This new language has been called by Hintikka and associates *Independence Friendly first-order language* (IF-language). It provides the following translation of (2′):

(2″) $\forall x \forall y (\exists z / \forall y)(\exists\ t / \forall x) \mathrm{R}(x, z, y, t)$,

where $(\exists z/\forall y)$ expresses the independence of $\exists z$ from $\forall y$, and $(\exists t/\forall x)$ the independence of $\exists t$ from $\forall x$.

According to Hintikka, the phenomenon of informational independence is ubiquitous in natural language, but it "has been hidden by the fact that informational independence is not indicated in natural languages by any uniform syntactical device" [Hintikka (1996), p. 73; see also Hintikka (1990)]. On these lines, recent work has shown the advantages that functional quantification offers to the study of natural language semantics [see, for instance, Winter (2004), and Schlenker (2006)].

However, on the following pages we will not focus on the strengths of functional quantification, but rather on its weaknesses, and in particular in connection with Skolem functions. In section II, we will explain, through various examples, an observation made by Kit Fine concerning Skolem functions, namely that there are some kind of dependence relationship between objects that "cannot properly be represented by a function at all" [p. 47]. Finally, in section III, we will use the same examples to discuss whether arbitrary objects offer, in fact, an improvement as regards the expression of dependence.

II. Making Sense of Kit Fine's Mysterious Claim

In 1985 *Reasoning with Arbitrary Objects* is published. In this book, Fine develops a theory of arbitrary objects with the idea of applying it, in a subsequent part of the book, to systems of natural deduction and ordinary reasoning. In the first chapters of the book, Fine explains the notion of arbitrary object, defends it, and gives us a syntax and a semantics to deal with it in the context of first-order languages. After defining truth in chapter 5, Fine compares his "theory on semantics for arbitrary objects" [p. 44] with some other semantics and theories. It is in this context that Skolem functions are brought into the picture. Here it is what he says:

> In certain ways, arbitrary objects behave like Skolem functions...
> There are, however, some important differences between A-objects [arbitrary objects] and Skolem functions. First, A-objects, unlike functions, are treated ontologically, on the same level as individuals, and A-names, unlike function symbols, are treated syntactically, on the same level as individual names. Second, a dependent A-object may take *several* values for given values to its dependees and not just a single value. In this respect, it is more like a multi-valued function. Finally, a dependent object of level >1, one that depends upon another dependent object, cannot properly be represented by a function at all. ...We might have, for example, that a = x, b = x^2 and c = 2b. Then *c* cannot be represented as a function with one argument, since that is to overlook the dependence on the other 'argument'. But nor can *c* be represented as a function of two arguments, since that is to overlook the dependence of one of the arguments upon the other [Fine (1985), pp. 46-47; underlined is mine].

The reason why we describe the claim underlined above as *mysterious* is easy: Fine seems pretty convinced about something which looks clearly false. To see that, it suffices to notice that c, in Fine's example, can be expressed as c = g(f(x)), where f(x) = x^2 and g(x) = 2x. May we then say that Fine is wrong? It does not seem so, although his example does not seem the most fortunate one at all either.

With the aim of overcoming this inconvenience, and taking Fine's argument further, we will introduce into the discussion three pairs of sentences (discourses):

(3) $_{1.}$ Every product has a price.
$_{2.}$ For meat it is 10 euros.

(4) $_{1.}$ Everyone has a loved one with a problematic relative.
$_{2.}$ Sooner or later, one ends up helping some of them.

(5) $_{1.}$ Everyone ends up helping a problematic relative of a loved one.
$_{2.}$ For John, it was María's.

Now, we will analyze these pairs of sentences from the point of view of natural language semantics in three steps: 1.- Dependencies; 2.- Skolemization; and 3.- Interpretation of the second sentence of every pair.

FIRST STEP – DEPENDENCIES

In the first step of our analysis, we pay attention to the dependence relationships present in the first sentence of every pair in order to represent that sentence in terms of an IF-language.

In (3_1), ‘a price’ depends on ‘every product.’ So in terms of an IF-language:

$$(3_1)' \ \forall x \exists y(\text{product}(x) \wedge \text{price}(y) \wedge \text{have}(x, y))$$

In (4_1), ‘a loved one’ depends only on ‘everyone,’ and ‘a problematic relative’ depends only on ‘a loved one.’ So in terms of an IF-language:

$$(4_1)' \ \forall x \exists y(\exists z/\forall x)(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z))$$

In (5_1), ‘a problematic relative’ depends on ‘everyone’, and also on ‘a loved one’; ‘a loved one’ depends on ‘everyone.’ So, in terms of an IF-language:

$$(5_1)' \ \forall x \exists y \exists z(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z) \wedge \text{help}(x, z))$$

SECOND STEP - SKOLEMIZATION

In this step, we want to translate $(3_1)'$, $(4_1)'$, and $(5_1)'$, that is, the representations in terms of an IF-language of (3_1), (4_1), and (5_1), into their respective Skolem forms. In line with this aim, we will follow Mann, Sandu and Sevenster [(2011), chapter 4] where skolemization is defined recursively. We proceed, therefore, as they do in their book, inside-out [cf. pp. 67-68]:

$(3_1)' \ \forall x \exists y(\text{product}(x) \wedge \text{price}(y) \wedge \text{have}(x, y))$

$\text{Sk}_{\{x, y\}}[\text{product}(x) \wedge \text{price}(y) \wedge \text{have}(x, y)]$ is
$\text{product}(x) \wedge \text{price}(y) \wedge \text{have}(x, y)$

$\text{Sk}_{\{x\}}[\exists y(\text{product}(x) \wedge \text{price}(y) \wedge \text{have}(x, y))]$ is
$\text{product}(x) \wedge \text{price}(f(x)) \wedge \text{have}(x, f(x))$

$\text{Sk}[\forall x \exists y(\text{product}(x) \wedge \text{price}(y) \wedge \text{have}(x, y))]$ is
$\forall x(\text{product}(x) \wedge \text{price}(f(x)) \wedge \text{have}(x, f(x)))$

So, finally, the Skolem form of $(3_1)'$ is:

$$(3_1)'' \ \forall x(\text{product}(x) \wedge \text{price}(f(x)) \wedge \text{have}(x, f(x)))$$

$(4_1)'$ $\forall x \exists y (\exists z/\forall x)(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z))$

$\text{Sk}_{\{x,\,y,\,z\}}[\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z)]$ is
$\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z)$
$\text{Sk}_{\{x,\,y\}}[(\exists z/\forall x)(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z))]$ is
$\text{love}(x, y) \wedge \text{relative}(y, g(y)) \wedge \text{problematic}(g(y))$
$\text{Sk}_{\{x\}}[\exists y(\exists z/\forall x)(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z))]$ is
$\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(f(x))) \wedge \text{problematic}(g(f(x)))$
$\text{Sk}[\forall x \exists y (\exists z/\forall x)(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z))]$ is
$\forall x(\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(f(x))) \wedge \text{problematic}(g(f(x))))$

So, finally, the Skolem form of $(4_1)'$ is:

$(4_1)''$ $\forall x(\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(f(x))) \wedge \text{problematic}(g(f(x))))$[1]

$(5_1)'$ $\forall x \exists y \exists z(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z) \wedge \text{help}(x, z))$

$\text{Sk}_{\{x,\,y,\,z\}}[\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z) \wedge \text{help}(x, z)]$ is
$\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z) \wedge \text{help}(x, z)$
$\text{Sk}_{\{x,\,y\}}[\exists z(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z) \wedge \text{help}(x, z))]$ is
$\text{love}(x, y) \wedge \text{relative}(y, g(x, y)) \wedge \text{problematic}(g(x, y)) \wedge$
$\text{help}(x, g(x, y))$
$\text{Sk}_{\{x\}}[\exists y \exists z(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z) \wedge \text{help}(x, z))]$ is
$\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(x, f(x))) \wedge \text{problematic}(g(x, f(x))) \wedge$
$\text{help}(x, g(x, f(x)))$
$\text{Sk}[\forall x \exists y \exists z(\text{love}(x, y) \wedge \text{relative}(y, z) \wedge \text{problematic}(z) \wedge \text{help}(x, z))]$ is
$\forall x(\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(\mathrm{x}, f(x))) \wedge$
$\text{problematic}(g(x, f(x))) \wedge \text{help}(x, g(\mathrm{x}, f(x))))$

So, finally, the Skolem form of $(5_1)'$ is:

$(5_1)''$ $\forall x(\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(x, f(x))) \wedge$
$\text{problematic}(g(x, f(x))) \wedge \text{help}(x, g(x, f(x))))$

THIRD STEP - INTERPRETATION OF THE SECOND SENTENCE

Finally, once the dependence relationships have been made explicit through Skolem functions, we use $(3_1)''$, $(4_1)''$, and $(5_1)''$ as the base for the interpretation of the second sentence of every pair.

(3_2) For meat it is 10 euros.

In the first sentence the dependence relationship between ‘every product’ and ‘a price’ has been explicitly introduced through the Skolem function f:

$(3_1)''\ \forall x(\text{product}(x) \wedge \text{price}(f(x)) \wedge \text{have}(x, f(x)))$

In (3_2), this explicit expression of dependence is recovered through the pronoun ‘it’:

$(3_2)'\ f(\text{meat}) = 10\text{euros}$

(4_2) Sooner or later, one ends up helping some of them.

In the first sentence the dependence relationships between ‘everyone’ and ‘a loved one,’ and ‘a loved one’ and ‘a problematic relative’ have been explicitly introduced through the Skolem functions f and g, respectively:

$(4_1)''\ \forall x(\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(f(x))) \wedge \text{problematic}(g(f(x))))$

Actually, from f and g we obtain a new dependence relationship ($h := g{\circ}f$) connecting an individual (anyone) with another one who is problematic.

In the second sentence it is said that:

$(4_2)'$ Given an individual x, there are some functions in the set $\{h$: $h = g{\circ}f$, where f maps x to a loved one, and g maps $f(x)$ to a problematic relative$\}$ such that help(x, $h(x)$).

Again, we are recovering the Skolem functions in $(4_1)''$, but this time through the pronoun ‘them’.

(5_2) For John, it was María’s.

In the first sentence the dependence relationships of ‘a loved one’ with ‘everyone,’ and of ‘a problematic relative’ with both ‘everyone’ and ‘a loved one’ have been explicitly introduced through the Skolem functions f and g, respectively:

$$(5_1)''\ \forall x(\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(x, f(x))) \wedge \text{problematic}(g(x, f(x))) \wedge \text{help}(x, g(x, f(x))))$$

As for the second sentence, its construction is quite similar to those in the previous examples. It seems then that one of the Skolem functions in $(5_1)''$

would have to be recovered through the pronoun 'it' in order to express (5_2). Interestingly enough, it does not work this time. Let us see why.

In order to represent what (5_2) expresses, we would need something of the following sort:

r(John) = l(María),

where r and l would express the dependence of the problematic individual we are talking about on, respectively, John and María. We would read it as: the problematic individual whom John helps, is a problematic relative of María.

We cannot get this result using the Skolem functions in $(5_1)''$. If we take a look at the different ways of using f and g to represent the meaning of (5_2), we observe that none of them is what we were looking for:

a) g(John, María) = ?	NO! g is certainly the way to recover the problematic individual. The issue now is to determine the right side of the equality.
b) g(John, María) = f(John)	NO! It does not make any sense.
c) f(John) = María	NO! However, in this case the explanation is not that simple.

One argument to not accept c) could be that there is no direct reference to the problematic individual in f(John) = María. However, it would be a bad argument, since the problematic individual can be easily identified after a simple computation:

f(John) = María → g(John, f(John)) = g(John, María) is a relative of María → g(John, María) is problematic → John helps g(John, María)[2]

We need to go a bit further to show that c) is not what we were looking for. Imagine the following dialogue:

(A) Everyone ends up helping a problematic relative of a loved one. For John it was María's.

(B)Who? Roy?

(A) No! Greg.

When (A) and (B) talk about Roy or Greg, they are talking about problematic relatives of María. This information is somehow part of the discourse itself, information which we will nevertheless lose, if we represent it in the following way:

$$\forall x(\text{love}(x, f(x)) \wedge \text{relative}(f(x), g(x, f(x))) \wedge \text{problematic}(g(x, f(x))) \wedge \text{help}(x, g(x, f(x))) \wedge f(\text{John}) = \text{María} \wedge g(\text{John, María}) \neq \text{Roy} \wedge g(\text{John, María}) = \text{Greg})$$

We do not get from this representation the information that Roy is a relative of María.

The problem arises because in (5_1) four dependence relationships have been put in play, but only two of them have acquired explicit form as Skolem functions. Bringing this observation to (5_2), we find that we know the dependence relationship of María on John (f), and also that of the problematic individual on both John and María (g), but we also observe that we do not know how the problematic individual depends on John alone, nor how the problematic individual depends on María alone. The functions r and l above seem to make reference to just these unknown relationships. That is why we fail to represent (5_2). Eventually, this is also the interpretation, under which Fine's claim makes sense to me.

In the next section, we will recall Fine's definition of A-model, and wonder whether an analysis in terms of arbitrary objects really offers, as Fine seems to suggest, a deeper and more accurate representation of the phenomenon of dependence. For that, we will bring the examples (3), (4), and (5) up again, and discuss their interpretation in terms of A-models.

III. Dependence and A-models

Let us begin by recalling what an A-model is [see Fine (1985), chapter 2]. An $\mathcal{A}$-model (generic or arbitrary-object-related terms model) $\mathcal{M}^+$ is of the form ($\mathcal{I}$, ..., $\mathcal{A}$, <, $\mathcal{V}$), where:

(i) ($\mathcal{I}$, ...) is a classical model $\mathcal{M}$ for $\mathcal{L}$ ($\mathcal{L}$ a first order language);

(ii) $\mathcal{A}$ is a set of objects disjoint from $\mathcal{I}$ (the arbitrary objects);

(iii) < is a relation on $\mathcal{A}$. It is the dependence relation between arbitrary objects. $b < a$ indicates that the value of b depends upon the value of a. Diagrams like the one below can be used to represent the relation:

(iv) $\mathcal{V}$ (family of value assignments) is a non-empty set of partial functions from $\mathcal{A}$ into $\mathcal{I}$, i.e. functions ν for which $\mathrm{Dm}(\nu) \subseteq \mathcal{A}$ and $\mathrm{Rg}(\nu) \subseteq \mathcal{I}$ (that is, $\mathcal{V}$ gives us the possible values that an arbitrary object can have);

(v) (a) (Transitivity) $a < b$ & $b < c$ implies $a < c$;

(b) (Foundation) The converse of the relation < is well-founded, i.e. there is no infinite sequence of $\mathcal{A}$-objects $a_1, a_2, \ldots$ for which $a_1 < a_2 < \ldots$;

(vi) (Restriction) $\mathcal{V}$ is closed under restriction, i.e. $\nu \in \mathcal{V}$ and $B \subseteq \mathcal{A}$ implies that $\nu|B \in \mathcal{V}$ (where $\nu|B$ is the restriction of ν to B);

(vii) (Partial Extendibility) If $\nu \in \mathcal{V}$, then there is a $\nu^+ \in \mathcal{V}$ for which $\nu \subseteq \nu^+$ and $[\mathrm{Dm}(\nu)] \subseteq \mathrm{Dm}(\nu^+)$ (where $[\mathrm{Dm}(\nu)]$ is the smallest closed set to contain $\mathrm{Dm}(\nu)$, and a subset B of $\mathcal{A}$ is said to be closed if whenever $a \in B$ and $a < b$ then $b \in B$);

(viii) (Piecing) Let $\{\nu_\rho : \rho \in \Omega\}$, for Ω a not empty set, be an indexed subset of $\mathcal{V}$ subject to the requirements that (a) each $\mathrm{Dm}(\nu_\rho)$ is closed and (b) the union $\cup\nu_\rho$ is a function. Then $\nu = \cup\nu_\rho$ is also a member of $\mathcal{V}$.

To finish, let us introduce two more items of nomenclature [Fine (1985), chapter 2]:

<u>Value-assignments of $\mathcal{V}$ defined on $B \subseteq \mathcal{A}$</u>: $\mathcal{V}_B := \{\nu \in \mathcal{V} : \mathrm{Dm}(\nu) = B\}$
<u>Value dependence of $a \in \mathcal{A}$ upon $B \subseteq \mathcal{A}$</u>: $\mathrm{VD}(a, B)$. It is the function defined, for $\nu \in \mathcal{V}_B$, by:

$$\mathrm{VD}(a, B)(\nu) := \{\mathrm{i} \in \mathcal{I} : \{(a, \mathrm{i})\} \cup \nu \in \mathcal{V}\}$$

(that is, all the values that a can take, given the values taken by the elements in B).

The time has come to return to our examples, and observe whether or not the use of arbitrary objects and models introduces an improvement on their interpretation:

(3) 1. Every product has a price.
 2. For meat it is 10 euros.

Let $\mathcal{M}^+ = (\mathcal{I}, ..., \mathcal{A}, <, \mathcal{V})$ be a generic model.
Let a, b be in $\mathcal{A}$, where a is an arbitrary product, and b is an arbitrary price.
(3_1) can then be interpreted:

(3_{1*}) $\text{product}(a) \wedge \text{price}(b) \wedge \text{have}(a, b)$

In (3_2) we are saying that: if the product is meat, then the price it has must be 10 euros. Using the tools provided by $\mathcal{M}^+$ we can interpret (3_2) as:

(3_{2*}) $\text{VD}(b, \{a\})(\nu) = \{10 \text{ euros}\}$, where $\nu : \{a\} \xrightarrow[\nu(a)=meat]{} I$

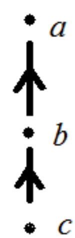
Let us see now what happens with (4):

(4)1. Everyone has a loved one with a problematic relative.
 2. Sooner or later, one ends up helping some of them.

Let $\mathcal{M}^+ = (\mathcal{I}, ..., \mathcal{A}, <, \mathcal{V})$ be a generic model.
Let a, b, c be in $\mathcal{A}$, arbitrary persons.
(4) can then be interpreted:

(4_{1*}) $\text{love}(a, b) \wedge \text{relative}(b, c) \wedge \text{problematic}(c)$

(4_{2*}) $\forall \nu \in \mathcal{V}_{\{a\}}\ \exists \text{i} \in \text{VD}(c, \{a\})(\nu) \text{ s.t. help}(\nu(a), \text{i})$



Observe that in (4_{1*}), as previously in (3_{1*}), the only elements are arbitrary objects, that is, elements of $\mathcal{A}$, thus using arbitrary objects to introduce the kind of objects that we are talking about and the dependencies between them.

Next, (3_{2*}) and (4_{2*}) make the discourse more precise by bringing up the elements in $\mathcal{I}$. In (3_2) the reason of bringing in elements of $\mathcal{I}$ is 'meat' and '10 euros', both of them possible values in $\mathcal{I}$ of a and b, respectively. In (4_2) the reason is 'some of them'. Through this expression, we introduce into the discourse the values of c.

Finally, the example in which we are more interested:

(5) 1. Everyone ends up helping a problematic relative of a loved one.
2. For John, it was María's.

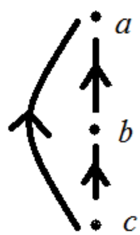
Let $\mathcal{M}^+ = (\mathcal{I}, ..., \mathcal{A}, <, \mathcal{V})$ be a generic model.
Let a, b, c be in $\mathcal{A}$, arbitrary persons.
(5) can then be interpreted:

(5_{1*}) love(a, b) $\wedge$ relative(b, c) $\wedge$ problematic(c) $\wedge$ help(a, c)

(5_{2*}) VD(c, $\{a\}$)(ν) $\subseteq$ VD(c, $\{b\}$)(ν'), where

$\nu : \{a\} \xrightarrow[\nu(a) = John]{} \mathcal{I}$

$\nu' : \{b\} \xrightarrow[\nu'(b) = María]{} \mathcal{I}$



Unlike with the previous interpretation of (5), in which we used Skolem functions, now we have been able to provide an interpretation of (5_2) consistent with both our intuitions and the interpretations of (3) and (4). The reason is simple: when we work with arbitrary objects and models, we do have access to each one of the dependence relationships put into play. As we have already seen, this is not always the case when we work with Skolem functions.

Finally, coming back to the dialogue we mentioned in the previous section:

(A) Everyone ends up helping a problematic relative of a loved one. For John it was María's.

(B)Who? Roy?

(A) No! Greg.

This time, it is possible to naturally retain the information which we lose when we work with Skolem functions. So Roy belongs to VD(c, $\{b\}$)(ν') but not to VD(c, $\{a\}$)(ν). Greg belongs to both.

Conclusion

At the beginning of this article, we wondered whether arbitrary objects offered any improvement over functional quantification, and in particular over Skolem functions, in relation to the expression of dependencies between quantifiers. In order to study this issue, we focused on one interesting, though not completely transparent claim, which appears in Kit Fine's *Reasoning with Arbitrary Objects*. In it, Fine points out an important difference between arbitrary objects and Skolem functions. Our aim in section II was to make sense of the claim, and to do that we introduced three very similar examples into discussion. Finally, in section III we achieved an answer to our question. An answer which becomes the conclusion of the present paper: the use of arbitrary objects seems indeed to provide an improvement, at least in certain cases, over Skolem functions.

Departamento de Filosofía I & Department of Human Development
Universidad de Granada & Teachers College Columbia University
Campus de Cartuja, E-18011 Granada
E-mail: aransangines@gmail.com

Acknowledgements

I would like to thank especially Gabriel Sandu. He was the first one to draw my attention to the Fine's *mysterious claim* which has been discussed in this paper. I am also grateful to Martin Andor, for his comments, Allison Faye, who has checked the English of my examples, and to Esther Romero for valuable comments and more. My thanks must also go to all the participants in the Workshop, and especially to Jaakko Hintikka, who was very generous with his time and comments throughout the three days of Workshop. This work has been partly supported by the research group HUM 432 of the Plan Andaluz de Investigación, Desarrollo e Innovación (PAIDI), and by the project P08.HUM.04099 of the Junta de Andalucía Consejería de Innovación, Ciencia y Empresa, each of which is gratefully acknowledged. This paper would not have been possible either without the Fulbright/Ministerio de Educación postdoctoral grant.

Notes

[1] Cf. signaling example in [Mann, Sandu and Sevenster (2011), pp. 73-74].
[2] I would like to thank Gabriel Sandu for pointing this out to me.

References

Fine, K. (1985), *Reasoning with Arbitrary Objects*, Oxford, B. Blackwell.

Hintikka, J. (1974), "Quantifiers vs. Quantification Theory," *Linguistic Inquiry*, vol. 5, pp. 153-177.

— (1990), "Paradigms for Language Theory," *Acta Philosophica Fennica*, vol. 49, pp. 181-209.

— (1996), *The Principles of Mathematics Revisited*, Cambridge, Cambridge University Press.

Mann, A. L., Sandu, G. and Sevenster, M. (2011), *Independence-Friendly Logic. A Game-Theoretic Approach*, Cambridge, Cambridge University Press.

Sandu, G. (1997), "On the Theory of Anaphora: Dynamic Predicate Logic vs. Game-theoretical Semantics," *Linguistic and Philosophy*, vol. 20, pp. 147-174.

Schlenker, P. (2006), "Scopal Independence: A Note on Branching and Wide Scope Readings of Indefinites and Disjunctions," *Journal of Semantics*, vol. 23, pp. 281-314.

Winter, Y. (2004), "Functional Quantification," *Research on Language and Computation*, vol. 2, pp. 331-363.