Higher-orderism

Jeff Speaks
PHIL 93503

November 22, 2023

1 Basics of higher-order languages

A first order logic is one that allows quantification only into subject position, as in

> Someone is happy.
> $\exists x$ (x is happy)

A higher-order logic is one that allows quantification into non-subject position. If we consider

> a is somehow.

we could represent this in a first-order way as

> $\exists x$ (x is a property & a has x)

In a second-order language we can represent this by quantifying into predicate position:

> $\exists F\ Fa$

Similarly, we might try to represent

> Mary believes something.

by quantifying into sentence position:

$\exists p$ (Mary believes that p)

Lots of contemporary work on higher-order uses languages in which you can quantify, not just into predicate and sentence position, but into any grammatical position in a formula.

How do we interpret these new types of quantification? Via introduction and elimination rules parallel to the first order quantifier.

One might interpret this language in various ways. It could be interpreted in a 'first order way;' that is not the interpretation intended by people working on higher-order metaphysics.

Brief historical digression: Frege and the paradox of the concept 'horse.' The connection between Frege's view and the problem of the unity of the proposition.

The idea that higher-order quantification is 'ontologically innocent.'

One standard way to develop a language with these sorts of quantification involves a 'theory of types.' Types are syntactic categories. There are two basic types: type e (entity) is the type of names and individual variables, and type t is the type of formulas.

Further, for any two types, there is a type of expression which combines with expressions of the first type to yield entities of the second type. So, for example, there are expressions – monadic predicates – which combine with names (e) to form a formula (t). That type is written:

$$< e, t >$$

Or consider negation. That combines with a formula to make a formula; so it is of type $< t, t >$. Some types combine with multiple expressions to form a new expression; an example would be a sentence connective like 'and' which combines with two formulas to yield a new formula. That type is written like this:

$$< t, t, t >$$

That should be distinguished from a type like

$$<< e, t >, t >$$

which would combine with a single expression of type $< e, t >$ to make a formula.

What would be a plausible type for adverbs? How about a simple quantifier?

Somewhat confusingly, there are other ways of writing out these types. One can also write the type $< e, t >$ as

$$e \to t$$

Some also write it as

$$< e >$$

with the type of sentences written as $<>$.

Distinguish between typed languages and the use of typed languages to express higher-order theories.

Once we have these types, we can introduce quantifiers corresponding to each. Intuitively (and not quite correctly) the idea is that the quantifier for a given type will range over the entities which can be assigned as the values of expressions of that type. The reason why this is not quite correct is that these will not in general be entities, since 'they' will not be of type e. Just as there are infinitely many types, there will be infinitely many existential quantifiers. So quantifiers are often written with subscripts:

$$\exists_e x$$

$$\forall_{<e,t>} F$$

etc.

Typically these languages will also involve identity predicates which apply to things not of type e. So like quantifiers, '=' is also typically subscripted. Some take these higher order identities to correspond at least roughly to certain sentences of English. We might then write 'to be a vixen is to be a female fox' as

$$V =_{<e,t>} FF$$

or 'for it to be the case that Hesperus is visible is for it to be the case that Phosphorus is visible'

$$Vh =_t Fh$$

Higher-order languages typically also involve the expression 'λ.' This is (in the first instance) a predicate-forming operator which combines with n variables and a formula to make an n-place predicate. So for example we might form the following monadic predicate:

$$\lambda x \ (x \text{ is furry } \& \ x \text{ is hungry})$$

This is often called a 'lambda abstract.' This whole expression is of type $< e, t >$ — so it could combine with a name to make a sentence.

One can also generalize lambda abstraction so that it is a device for forming expressions of arbitrary types — not just type $< e, t >$. For example we could have

$$\lambda X \ (X(\text{Bob}) \ \& \ X(\text{Mary}))$$

What would the type of this lambda abstract be?

Lambda expressions are governed by two basic rules. The first governs application of a function to an argument:

Beta reduction

$$\lambda x_1 \ldots x_n.F(x_1 \ldots x_n)(a_1 \ldots a_n) \to F(a_1/x_1 \ldots a_n/x_n)$$

So, e.g., you can go from 'λx.philosopher(x)(Daniel)' to 'philosopher(Daniel).'

The second is the reverse, and tells you how to abstract a function from an expression:

Lambda abstraction

$$F(a_1/x_1 \ldots a_n/x_n) \to \lambda x_1 \ldots x_n.F(x_1 \ldots x_n)(a_1 \ldots a_n)$$

So, e.g., you can go from 'philosopher(Daniel)' to 'λx.philosopher(x)(Daniel).' Intuitively, this (roughly) takes you from an application of a predicate to a sentence which says something about the property expressed by the predicate.

One thing that can make expressions of this language hard to understand at first is that it is possible to use quantifiers which do not bind any variables. The idea is that we can separate out the variable binding and (for lack of a better term) quantifying roles of traditional quantifiers, with lambda abstracts doing the variable binding. So e.g. we might represent the relation of loving as

$$\lambda x.\lambda y(x \text{ loves } y)$$

The property of loving someone is then

$$\lambda x.\exists_e(\lambda y(x \text{ loves } y))$$

Then 'Everyone loves someone' is

$$\forall_e(\lambda x.\exists_e(\lambda y(x \text{ loves } y)))$$

2 A quick potted history

Higher-order languages like this have been around since Russell and Whitehead's *Principia Mathematica*; the lambda calculus was developed by Church in the 1930's.

Despite, this, for a long time, the kinds of quantification licensed by theories like the ones we have just sketched – quantification into something other than name position – was thought to be either unintelligible or first-order quantification in disguise. Often pinned on Quine's *Philosophy of Logic* — higher order quantification is 'set theory in sheep's clothing.'

What was his argument? Not obvious. (He says that he 'deplores' various things.) One line is: there's nothing to be the value of a higher-order variable other than a set. (There are no attributes or propositions or . . .)

Why the history is a little hard to understand.

3 What does this theory tell us about properties, relations, and propositions?

So far this is just a sketch of a certain kind of formal language. What does it have to do with the topic of this course?

Basic idea: properties (in at least one sense of the term) are entities of type $< e, t >$, and so cannot be the values of first order quantifiers. Analogously for relations and propositions.

Further: questions about properties, relations, and propositions are best pursued by developing various logics for higher-order languages, which might yield different views of these higher-order entities. One central type of question: questions of 'grain.' These are questions about what principles govern the identity and distinctness of entities of various types.

Obviously these questions are also pursued by first-orderists. But some take them to be more tractable in a higher-order setting.

Let's look at some arguments for and against this general approach.

4 Expressibility considerations

Two different claims are sometimes made about higher-order language:

- It is needed to formalize certain sentences of natural languages like English.
- It is needed to state and discuss certain issues in metaphysics. (Or: needed to state and discuss these issues in the most perspicuous way.)

One argument for higher-orderism is that it can explain and validate a certain kind of inference. Williamson gives the example of the inference from

> Al is silent and Ben isn't

to

> Al is something that Ben isn't

In the language sketched above this might be represented like this:

$Sa \,\&\, \neg Sb$
$\lambda X.(Xa \,\&\, \neg Xb)(S)$ (lambda abstraction)
$\exists_{<e,t>} \lambda X.(Xa \,\&\, \neg Xb)$ (existential generalization)

How can we write out the inference in a first-order language? We might try something like

$Sa \,\&\, \neg Sb$
$\exists x(\text{Property}(x) \,\&\, \text{Has}(x, a) \,\&\, \neg\, \text{Has}(x, b))$

But this is not logically valid, as the interpretation of the conclusion depends on the non-logical terms 'has' and 'property.' One needs something like the following additional premise:

$$\exists x \ (\text{Property}(x) \ \& \ \forall y \ (\text{Has}(y, x) \text{ iff } Sy))$$

i.e., there's some property that is had by anything iff that thing is silent. So what? The extra premise is obviously true, right? (Silence would appear to be such a property.)

But consider the more general schema of which this is an instance, in which the predicate 'is silent' is replaced by a schematic letter. One might think that we are free to use specific claims like the above only insofar as we are entitled to think that every instance of this schema is true. But the claim that every instance of this schema is true leads to contradiction. If we substitute '$\neg\text{Has}(x, x)$' for the schematic letter, we get

$$\exists x \ (\text{Property}(x) \ \& \ \forall y \ (\text{Has}(y, x) \text{ iff } \neg\text{Has}(y, y)))$$

which says that there is some property x such that, for all y, y has x iff y does not have y. But then it follows by universal instantiation that there is some property x such that x has x iff x does not have x. No good!

Williamson:

> "We need a more principled way of answering questions of the form 'Is there a property p such that for every x, x has p if and only A(x)?' ... In other words, property theory requires explicit, general *comprehension principles*. A depressing feature of most philosophical discussions of properties and relations, the problem of universals and the like is that no such comprehension principle is on the horizon."

How strong is this motivation?

A related idea: higher-order languages are better than first order languages for stating certain kinds of generalizations about properties, relations, etc.

5 Higher-order views as dissolving false or confused disputes

One kind of argument you see for higher-order views is that they dissolve pseudo-problems that get generated by thinking of higher-order entities in a first order way.

5.1 Puzzles about instantiation

Lederman gives an argument of this kind in the paper we read. The first order person who believes in properties like redness is led to think that there is some special glue, instantiation, which can bind redness to apples but not bind apples to oranges. So they have to explain: why can this metaphysical glue bind together one pair but not the other?

Higher-order folks have an answer:

> ‘entities in the category of sui generis properties combine (as it were, by nature) with objects in something like the way that predicates combine with names, while entities in the category of things or objects cannot combine with one another in the same way.’

Can first order person can give a parallel speech, substituting ‘universals’ for ‘sui generis properties’? Need some argument that the first order person can’t make use of the ideology of ‘fundamentally different categories of things.’

Maybe the problem is that the first order person makes the extra claim that there is a further more general category into which individuals and universals both fall. But it seems like what is doing the explanatory work is the claim that there is a deep divide between the natures of individuals and of universals — whether there is some more general category including both of these deeply different kinds of things seems beside the point.

Maybe the idea is that if x and y can both be picked out by expressions of a single syntactic category, then they aren’t different enough to explain the facts about metaphysical glue. But not sure how much intuitive pull this really has.

5.2 *Are properties located?*

Jones (in another paper) considers these two claims:

> Immanence: All properties are spatially located.
>
> Transcendence: No properties are spatially located; despite being instantiated by located things, properties aren’t themselves located.

A first-orderist about properties has to think that both of these sentences make sense, and hence express propositions. Given this it seems that one must be true. But both lead to problems of various sorts. (Where are relations, or uninstantiated properties? How could an object fail to be spatially located?)

The idea is that the higher-orderist can dissolve the problem. They can give an interpretation of our two theses on which the ‘thing’ said to be located is not a first-order object. No reason to think that ‘is located’ is even defined for arguments of this sort, and hence no reason to think that the above sentences express propositions.

How strong of an argument is this? Does it show that the listed sentences don’t express propositions, or that some related sentences in higher-order-ese don’t? Some tension here between the (apparently intelligible) use of first-order language to communicate the subject matter of the theory and the claim that apparently parallel uses of first order language fail to convey real claims about higher-order entities.

6 Jones on the theoretical roles of propositions

A very different sort of argument comes from the paper by Jones. Jones contrasts two ‘approaches to theorizing about ontological categories’: Quineanism and type theory. (By the latter he means type theory + higher order quantification.)

Jones thinks that type theory is to be preferred on the grounds that it accommodates a key theoretical role for propositions that the Quinean approach cannot. He poses the question as: are propositions objects?

What's the theoretical role? The idea is that propositions are 'cross-type cognitive relations.' They are cognitive relations because they are 'relations of thought' between thinkers and reality. Why are they 'cross-type'? Here's the argument:

> 'One's best theory of any given aspect of reality will employ sentences. This theory must eventually be extended to encompass our cognitive relations to its various components. Because the original theory employs sentences, this extended theory will require expressions for our cognitive relations to the aspects of reality described by the original theory's sentences. In order to play this role, an expression must have the following formal profile: it has two argument positions, the first for expressions for thinkers, that is, singular terms, and the second for sentences. It is thus a cross-type relational expression of type $< e, t >$.'

(In our terminology the relation would be of type $< e, t, t >$.)

Small point: it does not seem like false beliefs relate us to aspects of reality. So it must be something like 'the sort of thing that could be an aspect of reality'?

Why this instead of

> 'In order to play this role, an expression must have the following formal profile: it has two argument positions, the first for expressions for thinkers, that is, singular terms, and the second for terms for the aspects of reality described by sentences, that is, that-clauses.'

Maybe the idea is something like this. Names pick out objects; so, if you want to report a relation between two objects, you use an expression which grammatically combines with two names to make a sentence. Sentences pick out aspects of reality; so, if you want to report a relation between an object and an aspect of reality, you use an expression which grammatically combines with a name and a sentence to make a sentence.

The proposition theorist erects a 'veil of propositions' between subjects and the aspects of reality described by sentences. So the proposition theorist owes a general explanation of how relations to propositions can also relate thinkers to these aspects of reality. But then we will need some claim like

> $\forall p$ (to believe that p is to Bel the proposition that p)

But this quantifies into sentence position – victory for the type theorist!

Suppose we think that propositions are entities of type t. Then: no question about what sort of object they are, since they are not objects. Is there still a question about what they are, whether they represent, etc.? I'm not sure I see why not. Could say: these questions are all posed in a first order way, and so can't be used to ask intelligible questions about the type theorist's propositions. Fair enough – but type theorists typically explicate their view informally using first order language. Either that is not a coherent way of gesturing at the theory, or it is a way of coherently gesturing that anyone can use.

Why a standard view of the syntax of attitude ascriptions is in some tension with the 'cross-type' view. Possible replies: standard syntax has it wrong; ordinary attitude ascriptions don't really report the important cross-type relations, but instead report some less fundamental thing. The latter seems to be Jones' view (175). The idea is that to handle attitude ascriptions we recognize both Quinean e-type proposition and the type theorists t-type propositions. Attitude ascriptions report relations to the former. But these relations between subjects and e-propositions represent the cross-type relations, given a mapping from e-propositions to t-propositions.

There's at least some tension here with the idea that the Quinean is the mystery-monger. If these cross-type cognitive relations are so important why does our language make it so hard to express them? Why are we always talking about these e-propositions when we could be talking about the aspects of reality corresponding to sentences?

Can we still ask questions about the nature of the e-propositions? Jones: no, since any choice of a domain D and one-to-one mapping from D to the t-propositions will do. Not quite sure how to think about the semantics of attitude ascriptions on this view.

Strictly speaking, expressions like 'cross-type cognitive relations to aspects of reality' don't make much sense, if 'aspects of reality' is a first order quantifier. Ditto for 'propositions are entities of type t.' How worked up should one get about this?

(I also note in passing that the views attributed to Speaks and Richard on p. 172 are not held by these important thinkers.)

7 The challenge of propositional attitudes

Lederman explores a foundational problem for higher order metaphysics which results from thinking about propositional attitudes. The problem starts with two plausible principles:

> Atomic congruence
>
> $a = b \rightarrow Fa = Fb$

Here the expressions can be of arbitrary type, with the identity symbols subscripted accordingly. The second principle is close to trivial:

> Material Equivalence
>
> $p =_t q \rightarrow (p \leftrightarrow q)$

Now consider the following take on attitude ascriptions:

> Naive relationism
>
> Attitude verbs are of type $< t, < e, t >>$.

So, they combine with sentences (t) to form monadic predicates $< e, t >$. Supposing this to be true, we can presumably formulate sentence operators corresponding to attitude ascriptions. These might correspond to phrases like 'Plato believes that.' Abbreviating that as 'B,' and 'is visible in the evening' as 'V,' the following looks plausible:

[A] $h = p \,\&\, B(Vh) \,\&\, \neg B(Vh)$

But this is inconsistent with our first two principles. By Atomic Congruence, $Vh =_t Vp$. By a second application of Atomic Congruence, $B(Vh) =_t B(Vh)$. By Material Equivalence, $B(Vh) \leftrightarrow B(Vh)$; but that contradicts [A].

Two main ways out: reject the higher-order translations of Fregean claims about attitude ascriptions like [A], or reject Atomic Congruence.

Background thought: the higher-order theorist presents a metaphysical theory using a new language, without providing a translation from that language into an antecedently understood language. This leads to two challenges. The metasemantic challenge: can we be sure that we have succeeded in giving meaning to the expressions of our new language? The epistemological challenge: there are many mutually inconsistent theories statable in higher-orderese. How will we tell which one is true?

Suppose we reject Atomic Congruence. This seems to make both challenges tougher. It makes the metasemantic challenge tougher because now the identity facts for basic categories places fewer constraints on the identity facts for derived categories, thus opening up more possible interpretations consistent with the former. If the metasemantic challenge is in some way solved, so that one of the possible interpretations is 'correct,' the problem remains of how to know which one it is.

Suppose instead we reject [A]. Three ways to do this: (i) deny that propositional attitudes are genuinely relations to entities of type t, so that attitude ascriptions are no evidence for sameness and distinctness here, (ii) say that they are, and that the English sentence corresponding to [A] is also false, or (iii) say that the English sentence is true, that attitudes are relations to entities of type t, and but deny that [A] captures the relations reported by the corresponding English sentence correctly.