Higher-Order Logic as Metaphysics

Zachary Goodsell
Juhani Yli-Vakkuri

March 16, 2021

1 The Language

We work in Church's *simply typed λ-calculus*, wherein each formal expression falls into one of infinitely many disjoint syntactic categories, called *types*. The types are built recursively from the two base types e and t by the operation of putting a $\rightarrow$ between them. Intuitively, type e is the type of names, t of sentences, and $\sigma \rightarrow \tau$ the type of things that yield an expression of type τ when applied to an expression of type σ.

The formal language also includes variables of all types. For example, "(Xy)" will be a well-formed expression of type τ when "X" is of type $(\sigma \rightarrow \tau)$ and "y" of type σ. Variables, like other expressions, have their types set in stone. Due to a shortage of letters, the official variables are letters with superscript designations of their type. So we have variables like "$X^{(t\rightarrow t)}$" and "y^t" and so on. The superscript will be suppressed for brevity when the type in question either doesn't matter or can be figured out from context.

Variables are bound using *λ-abstraction*. When an expression M of type τ contains an unbound variable, "x" say of type σ, then $(\lambda x.M)$ is an expression of type $(\sigma \rightarrow \tau)$. That is, if n is also of type σ, then "$(\lambda x.M)$" can be applied to n to get the well-formed expression "$((\lambda x.M)n)$" of type τ. Intuitively, "$((\lambda x.M)n)$" means what M would have meant if every occurrence of "x" that is not bound in M were replaced with an occurrence of n. λ-abstraction is the only way of binding variables in the language. Quantifiers, which are traditionally used to bind variables in first-order logic, are instead treated as yet more expressions in the way we treated the English "every" above. For example, the first-order universal quantifier is treated as an expression of type $((e \rightarrow t) \rightarrow t)$. That is, it takes in a predicate and spits out a sentence, which intuitively means that the property expressed by the predicate is universal.

The logical primitives of the language are the following:

- For each type σ a quantifier expression $\forall_\sigma$ of type $((\sigma \to t) \to t)$;
- A connective, $\to$, of type $(t \to (t \to t))$.

$\forall_e$ is to be understood as the familiar first-order universal quantifier. For other types, the meaning of $\forall_\sigma$ is to be pinned down by its logical role, which we will stipulate later is exactly analogous to that of $\forall_e$. The connective $\to$ is to mean material implication.

This completes our description of the formal language. However, we will help ourselves to some abbreviations that help with readability. For one thing, we will omit brackets where we can. Rightwards arrows are always taken to associate rightwards: so that $e \to e \to e \to t$ is to abbreviate $(e \to (e \to (e \to t)))$, but $(e \to e) \to e \to t$ abbreviates $((e \to e) \to (e \to t))$. Application will always associate *left*wards: so that "Xyz" will abbreviate "$((Xy)z)$" rather than, say, "$(X(yz))$". That is, "Xyz" is the expression that results from first applying "X" to "y" and then the result, "(Xy)", to "z".

Next, we will omit repeated λs, so that complex λ-abstractions like "$(\lambda x.(\lambda Y.(\lambda z.Yxz)))$" will be abbreviated as "$\lambda xYz.Yxz$". When a universal quantifier of any type precedes a λ, we omit the λ, so that "$\forall x(Yx)$" abbreviates "$\forall\lambda x.Yx$". Moreover, we freely write many universal quantifiers as one, using a period to separate the quantifier from the formula it is prepended to: "$\forall xYz.(Yxz)$" abbreviates "$\forall x\forall Y\forall z(Yxz)$".

We will also write relations in infix notation where this helps readability. For example, "$p \to q$" will be written instead of the cumbersome "$\to pq$".

Finally, we will use the only primitive truth-functional connective, "$\to$", along with the quantifier "$\forall_t$", to define the other familiar truth-functional connectives of conjunction, negation, and so on. Thus, we will employ the following abbreviations:

- "$\bot$" abbreviates "$\forall_t p(p)$";
- "$\neg$" abbreviates "$\lambda p.(p \to \bot)$";
- "$\vee$" abbreviates "$\lambda pq.(\neg p \to q)$";
- "$\wedge$" abbreviates "$\lambda pq.\neg(\neg p \vee \neg q)$";
- "$\leftrightarrow$" abbreviates "$\lambda pq.((p \to q) \wedge (q \to p))$";
- "$\exists_\sigma$" abbreviates "$\lambda X.\neg\forall_\sigma y(\neg Xy)$".

2 The Logic

2.1 Classical Higher-Order Logic

Logics are here identified with the set of (type t, possibly open) formulae *provable in* the logic. We also use *theory* interchangeably with logic.

As usual, our logic will be described in terms of *axioms* and *rules of inference*. These characterise the logic recursively: axioms are the things we start with, and rules of inference are some closure conditions. The logic is taken to be the least set of formulae that contains the axioms and is closed under the rules.

Classical higher-order logic is a minimal logic that we will assume to be true in what follows. Roughly, classical higher-order logic is the smallest (that is, weakest) logic that ensures that

- The truth-functional connectives work as expected;
- For each type σ, the quantifier "$\forall_\sigma$" obeys versions of the familiar laws of universal instantiation and universal generalisation;
- Each expression of the form "$\lambda x.Fx$" is intersubstitutable with "F" (η-equivalence);
- The application of a λ-expression, "$(\lambda x.M)n$", is intersubstitutable with "$M[n/x]$", which is the expression M but with each unbound occurrence of x replaced with an n (β-equivalence).

There are infinitely many different types, and classical higher-order logic says something about each of them, so classical higher-order logic requires infinitely many axioms and rules. Our axioms will be the following:

Propositional Calculus The standard axioms for classical propositional logic for the material conditional "$\rightarrow$" and the other, defined, truth-functional connectives.

Universal Instantiation An axiom of the following form for every type σ, and expressions M and n of types $(\sigma \rightarrow t)$ and σ respectively (including variable symbols):

$$(\forall M) \rightarrow Mn.$$

α-Equivalence An axiom of the form $\phi \leftrightarrow \psi$, whenever ϕ and ψ differ only by a change of variables.

η-Equivalence An axiom of the form $\phi \leftrightarrow \psi$, whenever

(a) ϕ and ψ are (grammatical) sentences that differ only by the substitution of M and "$\lambda x.Mx$", for M of any functional type (otherwise M cannot grammatically be applied to the variable), and

(b) There are no unbound occurrences of x in M.

β-Equivalence An axiom of the form $\phi \leftrightarrow \psi$, whenever

(a) ϕ and ψ are sentences that differ only by the substitution of "$(\lambda x.M)n$" and $M[n/x]$, again for any expressions M and n of types for which "$(\lambda x.M)n$" is grammatical, and

(b) No variable that is unbound in the occurrence of n in either expression becomes bound in the other.

The rules of inference are as follows:

Modus Ponens From ϕ and $\phi \rightarrow \psi$, infer ψ.

Universal Generalisation For each type σ: from $\phi \rightarrow \psi$, where x is a variable of type σ that has no unbound occurrences in ϕ, infer $\phi \rightarrow \forall_\sigma x\psi$.

2.2 Classicism

Classicism adds the following axiom schema:

Classical Equivalence $\phi = \psi$, and $\lambda x_1 \ldots x_n.\phi = \lambda x_1 \ldots x_n.\psi$ for any string of variables $x_1 \ldots x_n$, whenever $\phi \leftrightarrow \psi$ is a theorem of classical higher-order logic.

2.3 The Axiom of Choice

Axiom of Choice (AC) $\exists f \forall X((\exists y Xy) \rightarrow X(f(X)).$

Here is an equivalent axiom schema:

Hilbert-Ackermann Equivalence $\forall R(\forall x \exists y Rxy \leftrightarrow \exists f \forall x Rxf(x)).$

We will assume a stronger schema:

Necessary Axiom of Choice □AC $\top = \exists f \forall X((\exists y Xy) \rightarrow X(f(X)).$

Which is equivalent to the following plausible schema about propositional granularity:

Hilbert-Ackermann Identity $\forall R((\forall x \exists y Rxy) = \exists f \forall x Rxf(x)).$

3 Necessity, Identity, and Distinctness

Let $\Box$ abbreviate $= \top$. Classicism ensures that $\Box$ behaves logically like a notion of necessity:

Theorem 1 (C)**.** *$\Box$ has an S4 modal logic, and obeys the necessity of identity and the converse Barcan formula.*

Proof. See Bacon 2018. $\Box$

Theorem 2 (C)**.** *$\Box$ has an S5 modal logic just in case every pair of distinct propositions is necessarily distinct.*

$$\forall p(\neg\Box p \to \Box\neg\Box p) \leftrightarrow \forall pq((p \neq q) \to \Box(p \neq q))$$

Proof. If distinct propositions are necessarily distinct, then in particular $p \neq \top$ is necessary when true. But $p \neq \top$ is just $\neg\Box p$, so $\neg\Box p$ is necessary when true. $\Box$

Theorem 3 (C + AC)**.** *Distinct things are necessarily distinct.*

Proof. For a fixed a of any type, define the relation $R := \lambda xp.(x = a \wedge p = \top) \vee (x \neq a \wedge p = \bot)$. Clearly, R relates everything to something, so by the Hilbert-Ackermann identity, there is a function f that relates every x to something it is R-related to. So if $a \neq b$, then $fa = \top$ and $fb = \bot$. As a result, if $a \neq b$, then $(a = b) = (a = b \wedge (fa = fb)) = (a = b \wedge (\top = \bot)) = \bot$. So $a \neq b$ is necessary when true. $\Box$

4 Classes and Extensions

We define an *extension* at type $\sigma_1 \to \cdots \to \sigma_n \to t$ as a function that maps everything to either $\top$ or $\bot$:

$$\mathrm{Ext}^{\sigma_1 \to \cdots \to \sigma_n \to t} := \lambda X.\forall y_1 \ldots y_n.(\Box X y_1 \ldots y_n \vee \Box\neg X y_1 \ldots y_n)$$

The theory of extensions is given by

Extension Extensionality Coextensional extensions are identical.

$$\forall XY.((\mathrm{Ext}\, X \wedge \mathrm{Ext}\, Y) \to \forall z_1 \ldots z_n.(X z_1 \ldots z_n \leftrightarrow Y z_1 \ldots z_n))$$

Extension Comprehension Every property and relation has an extension.

$$\forall X \exists Y(\mathrm{Ext}\, Y \wedge \forall z_1 \ldots z_n.(X z_1 \ldots z_n \leftrightarrow Y z_1 \ldots z_n))$$

Extension Persistence Extensions have their instances necessarily.

$$\forall X.(\mathrm{Ext}\, X \to \forall y_1 \ldots y_n.(X y_1 \ldots y_n \to \Box X y_1 \ldots y_n))$$

Extension Inextensibility If everything in an extension is necessarily some way, then necessarily everything in that extension is that way.

$$\forall X(\mathrm{Ext}\, X \to \forall Y(\forall z_1 \ldots z_n(X z_1 \ldots z_n \to Y z_1 \ldots z_n) \\ \to \Box\forall z_1 \ldots z_n(X z_1 \ldots z_n \to Y z_1 \ldots z_n)))$$

Theorem 4. *C+AC entails the theory of extensions (i.e., every theorem of the theory of extensions is a theorem of C + AC).*

5 Actuality and Possible Worlds

Let an *actuality operator* be a $t \to t$ operator that applies to all and only the truths, and applies necessarily to everything it applies to.

Theorem 5 (C + AC)**.** *There is an actuality operator.*

Proof sketch. The extension of truth ($\lambda p.p$) is an actuality operator. □

We let a *possible world* be a maximally specific possible proposition. That is,

$$\mathrm{World} := \lambda p.\forall q(\Box(p \to q) \lor \Box(p \to \neg q))$$

For example, if @ is an actuality operator, then World $\forall p(@p \to p)$.

Atomicity Possibility is coextensive with being true at some world.

$$\forall p(\Diamond p \leftrightarrow \exists w(\mathrm{World}\, w \land \Box(w \to p))$$

Theorem 6 (C + □AC)**.** *Atomicity is true.*

Proof. By necessitating □AC we get that there is necessarily an actuality operator. Thus it is necessary that there is a true world, or

$$\Box\exists w(w \land \mathrm{World}\, w)$$

Therefore, if $\Diamond p$, then

$$\Diamond\exists w(p \land w \land \mathrm{World}\, w)$$

and so $\Diamond\exists w(\text{World}\, w \wedge \Box(p \to w))$. Since $\Box$ has an S5 modal logic in $\mathsf{C} + \Box$AC, this yields Atomicity. $\Box$

This proof is the same as that of Gallin 1975, so in fact all we needed was S5 for $\Box$ and the necessitated theory of extensions. Gallin also shows the opposite directon: Classicism plus S5 plus Atomicity gives you the necessitated theory of extensions.

6 Foundations of Mathematics

6.1 The Necessity of Logic

Theorem 7 ($\mathsf{C}+\Box$AC)**.** *$\Box\phi \vee \Box\neg\phi$, where "$\phi$" is a closed sentence with only logical vocabulary.*

Proof sketch. Showing $\Box\phi \vee \Box\neg\varphi$ is the same as showing that $\phi = \pi\phi$ for any permutation of propositions given by a permutation of possible worlds.

Where π_t is such a permutation of propositions and π_e a permutation of individuals, we define permutations of the higher types recursively as follows:

$$\pi_{\sigma\to\tau} := \lambda X^{\sigma\to\tau} y^{\sigma}.\pi_\tau(X\pi_\sigma^{-1}y)$$

This gives us the identity $\pi(Xy) = \pi(X)\pi(y)$.

We then show that $\pi\forall = \forall$, $\pi\!\to\, =\!\to$, and $\pi M = M$ for any combinator M. If ϕ has only logical vocabulary, it can be written in the form $((Ma_1)\dots)a_n)$, where M is a combinator and a_1 through a_n are logical constants. Therefore, $\pi(\phi) = (\pi(M)\pi(a_1))\dots)\pi(a_n))$, which by the previous fact is just ϕ, as required. $\Box$

6.2 Consistency Implies Truth

Corollary 8. *$(\Diamond\exists x_1 \dots x_n.\phi) \to \exists x_1 \dots x_n.\phi$, where "$\phi$" is purely logical.*

Definition 1 (First-order formulae)**.** The *first-order* formulae relative to some finite set of variables Σ of types $e \to t$, $e \to e \to t$ and so on, are the least set of formulae that contain all the atomic sentences of the form $x^e = y^e$, and $Xy_1 \dots y_n$ for $X \in \Sigma$ that is also closed under Boolean combinations of formulae, and first-order quantification.

Definition 2 (Interpretation/Truth)**.** An *interpretation* of a finite set of variables Σ of type $e \to t$, $e \to e \to t$, and so on, consists of an $e \to t$ extension specifying the domain of quantification, and a function specifying for each variable an extension of the appropriate type. Truth on an interpretation is then defined in the Tarskian way.

Theorem 9 (C+□AC+Infinity)**.** *If some class of first-order sentences is syntactically consistent, then it is true on some interpretation.*

6.3 Equivalences Between $\mathsf{C} + \Box AC$ *and Set Theory*

Work in set theory. Given a set of worlds W and a set of individuals D^e, we can define a set of propositions D^t by taking the powerset of worlds, a set of properties $D^{e\to t}$ by taking the set of functions from D^e to D^t, and so on. Universal quantification over individuals can be defined as the function from properties to propositions that maps an $f \in D^{e\to t}$ to $\{w \in W : \forall x \in D^e(w \in f(x))\}$, and so on for other types of quantification. In this way, we can translate every formula of the higher-order language to a sentence in the language of set theory with W and D^e free, call this $\varphi^\dagger$.

This translation is of metamathematical interest to higher-order logicians. Informal sentences about functions and such can be formalised in set theory or in higher-order logic. If an informal sentence is formalisable in higher-order logic as φ, then that sentence ought to be formalised as $\varphi^\dagger$, with W and D^e taken to denote to the "sets" of worlds and of "individuals" respectively. Therefore, if set theorists accept a sentence of the form $\forall W \forall D^e \varphi^\dagger$, we can read off of this a corresponding commitment in the higher-order language. This notion of equivalence can therefore be used to see the differences in commitments between $\mathsf{C} + \Box$AC and various set theories. On this point we have a quick result and a conjecture which we are fairly confident in.

Theorem 10. $\mathsf{C} + \Box AC$ *is strictly weaker than the theory* $\{\varphi : \mathsf{ZFC} \vdash \forall W \forall D^e \varphi^\dagger\}$.

Proof. ZFC proves that $\mathsf{C} + \Box$AC is consistent, and therefore proves that for any W and D^e, there is no code for the inconsistency of $\mathsf{C} + \Box$AC in the higher-order natural numbers. But $\mathsf{C} + \Box$AC doesn't prove this by the incompleteness theorem. □

On the other hand, let ZDQB be ZFC minus the axiom of replacement, and with the axiom schema of separation replaced with the weaker schema:

Bounded Separation

$$\forall x \exists y (\forall z (z \in y \leftrightarrow (z \in x \land \varphi)))$$

> where "φ" is any formula with y not free, and with all quantifiers restricted to members of some set or other.

Conjecture 1. $\mathsf{C} + \Box AC = \{\varphi : \mathsf{ZDQB} \vdash \forall W \forall D^e \varphi^\dagger\}$.

Proof idea. If $\mathsf{ZDQB} \nvdash \forall W \forall D^e \varphi^\dagger$, then there is a model where $\neg\varphi^\dagger$ holds of some W and D^e. We define a class model of $\mathsf{C} + \Box\text{AC}$ in this model by interpreting quantification over type t and e as quantification restricted to W and to D^e, and over type $\sigma \to \tau$ as quantification restricted to (set-theoretic) functions from D^σ to D^τ. The closure conditions on models of set theory ensure that this forms a model of $\mathsf{C} + \Box\text{AC}$ where $\neg\varphi$ holds.

If $\mathsf{C} + \Box\text{AC} \nvdash \varphi$, then there is a Henkin model where $\neg\varphi$. We therefore have a set W and a set D^e such that quantification over type $\sigma \to \tau$ is in general treated as restricted quantification over functions $f : D^\sigma \to D^\tau$. The closure conditions on this model guarantee that these sets form part of a model of ZDQB where $\neg\varphi^\dagger$ holds. □

7 Choice and Plenitude

Bacon and Dorr have considered a weakening of choice:

Plenitude Every functional n-ary relation specifies a function.

$$\forall X((\forall y_1 \dots y_{n-1} \exists! y_n X y_1 \dots y_n) \to \exists f \forall y_1 \dots y_{n-1} X y_1 \dots y_{n-1} f(y_1 \dots y_{n-1}))$$

By a result in Gallin 1975 (Thm. 11.5), Plenitude is equivalent to the theory of extensions in Classicism. Choice strengthens Plenitude by the following schema:

Relational Choice Every functional n-ary relation has a functional subrelation:

$$\forall X \exists Y \forall z_1 \dots z_n ((Y z_1 \dots z_n \to X z_1 \dots z_n) \\ \wedge (\exists z'.X z_1 \dots z_{n-1} z' \leftrightarrow \exists! z'.Y z_1 \dots z_{n-1} z'))$$

Which is equivalent to the global wellordering schema:

Global Wellordering There is a wellordering of the things of type σ.

Theorem 11. *Relational Choice and Global Wellordering are equivalent in Classical HOL,* $\mathsf{C} + AC = \mathsf{C} +$ *Plenitude + Rel. Choice, and* $\mathsf{C} + \Box AC = \mathsf{C} + \Box$*Plenitude + Rel. Choice.*

References

Bacon, A. (2018). "The Broadest Necessity". In: *Journal of Philosophical Logic* 47.5, pp. 733–783.

Gallin, D. (1975). *Intensional and Higher-Order Modal Logic: With Applications to Montague Semantics*. American Elsevier Pub. Co.