Advances in Logical Grammar: Review of Higher Order Logic

Carl Pollard

Department of Linguistics
Ohio State University

June 11, 2012

From TLC to HOL (1/2)

- We start with a TLC and add a new type t of **truth values**. This will be a key ingredient of the logic, not just a basic type added at the user's discretion.
- Terms of type t are called **formulas**.
- So now the term 'formula' is ambiguous between types, which are formulas in IPL (the underlying type logic), and terms of type t, which are formulas in (what will turn out to be) a higher-order classical predicate logic.
- To avoid ambiguity, we'll usually call formulas of the first kind *types*, and formulas of the second kind *HOL formulas*.
- In an interpretation, the interpretation of an HOL formula is called a **truth value**. There will be axioms to ensure that there are exactly two of these.

From TLC to HOL (2/2)

- For each type A, we add a constant $=_A : A \to A \to \mathrm{t}$, written infix, e.g. $a = b$. The type subscript is usually omitted.
- In an interpretation I, $I(=_A)$ is (curry of) the identity relation on $I(A)$.
- There will be enough axioms to guarantee that all the TLC term equivalences can now be proved as equalities *in the object language itself*.
- That is, the term equivalences, formerly a set of metalanguage assertions about TLC terms, are now reformulated as a logical theory about equality.
- Amazingly, we now have a logic of truth-value terms.

Classical Connectives and Quantifiers are Definable

Here ϕ is a metavariable over HOL formulas, x is a variable of type A, and s, t are variables of type t:

1. $\mathsf{true} =_{\mathrm{def}} * = *$
2. $\forall_x.\phi =_{\mathrm{def}} \lambda_x.\phi = \lambda_x.\mathsf{true}$
3. $\mathsf{false} =_{\mathrm{def}} \forall_t.t$
4. $\phi \wedge \psi =_{\mathrm{def}} (\phi, \psi) = (\mathsf{true}, \mathsf{true})$
5. $\phi \rightarrow \psi =_{\mathrm{def}} \phi = (\phi \wedge \psi)$
6. $\phi \leftrightarrow \psi =_{\mathrm{def}} (\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)$
7. $\neg\phi =_{\mathrm{def}} \phi \rightarrow \mathsf{false}$
8. $\phi \vee \psi =_{\mathrm{def}} \neg[(\neg\phi) \wedge (\neg\psi)]$
9. $\exists_x.\phi =_{\mathrm{def}} \neg\forall_x.\neg\phi$

HOL: More than TLC + FOL (1/2)

- The connectives just defined act just like their FOL counterparts.
- The quantifers just defined act just like *their* FOL counterparts, except that they can quantify over variables of any type (including t). This is why it is called 'higher-order'.
- At the same time, an HOL is still a TLC.
- All the TLC term equivalences—most importantly, rule (β)—still hold, as equalities provable in the logic itself.

HOL: More than TLC + FOL (2/2)

- There are numerous ways to axiomatize HOL (Henkin 1950, Gallin 1975, Andrews 1986, Lambek and Scott 1986, Carpenter 1997).
- We'll remain agnostic about how to axiomatize HOL, and just mention some useful rules and axioms (or theorems, depending on the choice of axiomatization).
- We write $\vdash \phi$ to mean 'ϕ is provable in HOL'. (Note that '$\vdash$' is the same symbol used in typing judgments.)

Equality is an Equivalence Relation

1. $\vdash a = a$ (reflexivity)
2. if $\vdash a = b$, then $\vdash b = a$ (symmetry)
3. If $\vdash a = b$ and $\vdash b = c$, then $\vdash a = c$ (transitivity)

Rules for Substitution of Equals

1. if $\vdash a = c$ and $\vdash b = d$, then $\vdash (a, b) = (c, d)$
2. if $\vdash f = g$ and $\vdash a = b$, then $\vdash (f\ a) = (g\ b)$
3. if $\vdash a = b$, then $\vdash \lambda_x.a = \lambda_x.b$

Axioms for Cartesian Products

1. $\vdash a = *$ (for $\vdash a : \mathrm{T}$)
2. $\vdash \pi(a, b) = a$
3. $\vdash \pi'(a, b) = b$
4. $\vdash (\pi(c), \pi'(c)) = c$

Axioms for Lambda Conversion

1. $\vdash \lambda_{x \in A}.b = \lambda_{y \in A}.[y/x]b$ (rule α)
2. $\vdash ((\lambda_{x \in A}.b)\ a) = [a/x]b$ (rule β)
3. if $\vdash f : A \rightarrow B$ and x is not free in f, then $\vdash (\lambda_{x \in A}.f\ x) = f$ (rule η)

Axioms and Rules about Truth Values

1. $\vdash \phi = (\phi = \mathsf{true})$
2. If $\vdash \phi$ and $\vdash \phi = \psi$, then $\vdash \psi$
3. $\vdash \phi$ iff $\vdash \phi = \mathsf{true}$
4. $\vdash \forall_{s,t}.(s \leftrightarrow t) \rightarrow (s = t)$ (Truth-Value Extensionality)
5. $\vdash \forall_t.t \ \vee \neg t$ (Excluded Middle)
6. $\vdash \neg(\mathsf{true} = \mathsf{false})$ (Nondegeneracy)

Theorizing in HOL

- As with FOL, we can use HOL to write theories about the subject matter of our choice.
- Each high-order theory has its own stock of basic types, (typed) nonlogical constants, and nonlogical axioms.
- Linear grammar (LG) includes two such theories, a pheno theory (discussed immediately below) and a semantic theory (to be introduced little by little as needed).

A Simple Pheno Theory for LG (1/2)

Basic Types: In the simplest approach to pheno, the pheno theory has just one basic type s (string). (Eventually it becomes necessary to add more basic pheno types, e.g. for phonological words, clitics, pitch accents, etc.).

Nonlogical Constants:

$\vdash \mathbf{e} : \mathrm{s}$ (null string)
$\vdash \cdot : \mathrm{s} \to \mathrm{s} \to \mathrm{s}$ (concatenation)
Note: usually written infix, e.g. $s \cdot t$ for $(\cdot\ s\ t)$

constants for strings of single phonological words,
e.g. $\vdash$ pig : s for the string of /pIg/.

A Simple Pheno Theory for LG (2/2)

Nonlogical Axioms (here $s, t, u : \mathrm{s}$):

$$\vdash \forall_{stu}.(s \cdot t) \cdot u = s \cdot (t \cdot u)$$
$$\vdash \forall_{s}.(\mathbf{e} \cdot s) = s$$
$$\vdash \forall_{s}.(s \cdot \mathbf{e}) = s$$

Note: if you know a little abstract algebra, note that this theory says the set of strings forms a **monoid** with concatenation as the associative operation and the null string as the two-sided identity element.