Symbolic Logic 4

We are now moving beyond or rather within the sentence. The truth-functional logic we have met took sentences as its units, but the validity or otherwise of a lot of reasoning depends on how components of sentences behave.

In what is called *first-order logic* we distinguish the following elements within sentences:

- Names for singular objects of reference
- Predicates, that ascribe properties or relations to singular objects of reference
- Variables that can replace names for singular objects of reference
- Quantifiers that can link up with variables for singular objects of reference to say how many such objects have a particular predicate

In ordinary English these four categories are typically expressed by the following types of expressions:

- Names for singular objects of reference – proper names (Socrates, Bridgetown). We can afford to be fairly lax about what counts as an object of reference, thus Hodges mentions the referents of four types of phrase (proper names, non-count nouns, singular personal pronouns, and definite descriptions)[1]
- Predicates – verbal phrases (is wise, … loves …, swims)
- Variables – some uses of pronouns function like variables, but the closer analogue is in mathematics, where variables are ubiquitous: $x (y – z) = (x y) – (x z)$.
- Quantifiers – these don't have exact analogues, but they express what English says by using *all*, *every*, *some*, ...

You can see that in one sense the main addition is the notion of names, since variables and quantifiers relate to that category as well. A more complicated logic is produced if one adds variables and quantifiers related to predicates: this is called *second-order* logic. We won't be dealing with that.

Notation

We continue to use all the notation for the truth-functional mini-language. In addition we have the following:

For names we use small letters, usually from the beginning of the alphabet, though you can use ones that remind you of who is being named if you want, so *a* or *s* may be used for Socrates. These can be called *constants*, since on any particular occasion they pick out the same individuals each time.

1 His examples of non-count nouns include *butter, poverty*, *moonlight*; while definite descriptions are singular noun phrases beginning with *the*, *this*, *that*, *my*, *his*, *Birmingham's*, etc. What we exclude in this way are (a) plural noun phrases e.g., *my knees*, (b) expressions with the English equivalents of quantifiers, and (c) indefinite descriptions, *a cat.* (We will come back to definite descriptions later.)

For predicates, we tend to use two sets of capital letters: F, G, .. for one-place predicates (that require only one name to make a complete statement), and R, Q, S, .. for relational predicates (that require more than one name to make a complete sentence).

To put a name and a predicate together to make a sentence, we put the predicate first: so *Fa* may symbolise Fred smokes, or *Rab* may symbolise Alfred loves Betty.

For name-variables we use small letters from the end of the alphabet, *x*, *y,* … The variables in sentences that are attached to predicates but without any quantifiers linking to them are called *free*.

In first-order logic we usually consider only two quantifiers: the universal quantifier, $\forall$, read *for all*, and the existential quantifier, $\exists$, read, *for at least one* (or often *for some,* but understood to be true if one or more of the names that replace the variable it binds satisfies the predicate in question). Quantifiers are followed by a variable which typically links up to the same variable in the following sentence; when it does so, that variable is said to be *bound* by the quantifier. So, if *F* stands for *swims*, then $\exists$xFx, says *something swims*, and the *x* in *Fx* is bound by the quantifier expression..

Semantics

Typically when we think about individuals and generalisations about them we are thinking within a particular *universe of discourse*. If I say, 'everyone knows who the Principal is' I may be right if I am talking about current Cave Hill students; but if I was talking about everyone in Barbados or everyone in the Caribbean or everyone in the world or everyone in the history of the world, chances get less with each extension of the universe of discourse. Typically we take the quantifiers to range over a given universe of discourse. The universal quantifier says that a predicate holds of every individual in the universe of discourse. The existential quantifier says that some predicate holds of at least one individual in that universe.

Simple sentences that attach a predicate to one or more names say that the predicate is satisfied by the bearers of those names.

Translating from English

There are some standard translation schemata to get from English to first-order logic. When in English we talk about *all Fs being G*, or *every F being G*, the standard translation goes via the paraphrase *for all x, if x is F then x is G*. So we would write it $\forall$x(Fx → Gx).[2]

Some Fs are G is translated via the paraphrase *for at least one x, x is F and x is G*:$\exists$x(Fx ∧ Gx).

Note that in the universal case the quantifier binds a conditional whereas in the existential case the quantifier binds a conjunction. Why not the same sort of sentence in both?

If the existential quantifier binds a conditional we run into problems because a conditional is true when its antecedent is false. So, for instance, the false claim there is at least one even positive integer not divisible by two would turn out true if we symbolised it by writing $\exists$x(Px → Dx). Three (let *a* stand for 3) is a positive integer that is not even, so ¬Pa and so ¬Pa ∨ Da is true, but that is equivalent to (Pa → Da) which is therefore true, and the existential quantifier is true if there is at

2 Note an immediate consequence of this schema: if there are no *F*s the claim is true. The universal generalisation is only false if there is an *F* that is not *G*.

least one instance of its being true.

If we took universally quantified statements to be conjunctions we would be saying that everything in the domain of discourse is both *F* and *G*, but usually we are interested in picking out the *F*s from a wider group.

We have to make sure that the domain of discourse doesn't vary within an inference so it is usually best to go for a fairly broad universe..

Just as with our first mini-language there are a good number of English expressions that can be captured, with more or less faithfulness, by the machinery we have now introduced.

The word *only* has a similar effect when it is a quantifier as when it joins with *if*. *Q if P* is $P \rightarrow Q$, while *Q only if P* is $Q \rightarrow P$; similarly *all Fs are G* is $\forall x(Fx \rightarrow Gx)$ while *only Fs are G* is translated $\forall x(Gx \rightarrow Fx)$.

Scope: As Hodges notes, *all*/*any* and *every* often vary in English by reference to their scope. *I don't know anything* – ∀x(¬ I know x) – is different from *I don't know everything* – ¬ (∀x(I know x).

You also need to be careful about which variables you use when there is more than one quantifier. Hodge's rule is that *no quantifier should occur within the scope of another occurrence of a quantifier with the same variable* (p. 216). To illustrate, *every house has a deep freeze and a colour television* becomes ∀x(x is a house → (∃y(y is a deep freeze ∧ x has y) ∧ ∃y(y is a colour television ∧ x has y))). Note that all the *x*s are bound by the initial quantifier, but the two existential quantifiers only bind the two *y*s following them, so there is no problem using *y* for both of them.

Howson illustrates one whole area of possible translations that was originally urged by Davidson as a way of capturing obvious entailments with sentences using adverbs. It is clear that *Minerva is thinking deeply* entails *Minerva is thinking*. One way to capture that using our new language is to refer to an event of thinking that is had by Minerva, so we introduce three predicates, T for *is a process of thinking*, D for *is deep*, and M for *is had by Minerva*. We can then translate our first sentence as $\exists x(Tx \wedge Dx \wedge Mx)$. We will see that it is then easy to prove that $\exists x(Tx \wedge Mx)$ follows from it, and that is a translation of our second sentence, *Minerva is thinking.* You may think this is doing some damage to simple English, but it works (and logicians have been taking similar liberties for thousands of years).

Tree rules for quantifiers

Just as we used trees to simplify compound statements we can continue to use them in our new language by finding ways to get rid of the quantifiers and deal with quantifier-free sentences appropriately related to the sentences in the trunk of our tree. (Hodges speaks of the set of quantifier-free sentences corresponding to a set X of sentence s as a set of Herbrand sentneces.)

Every quantifier-free sentence in X is a Herbrand sentence. The other Herbrand sentences result from dropping the quantifiers.

The $\forall x\varphi$ rule says if you have a sentence of the form $\forall x\varphi$ in X and a designator (name) *a* that occurs in X then, if ψ is a matter of replacing every free occurrence of x in φ by *a*, then *X therefore* ψ is a valid argument.

In effect this allows you to instantiate the universal generalisation using any name you have. So if you have ∀x(Fx → Gx) and the name *a*, the sentence Fa → Ga is a Herbrand sentence. [Other books make the point that you can introduce a name here too – if something is true of everything in the universe of discourse then it is true of any item therein, named already or not.]

The ∃xφ rule says in effect that when you have an existential generalisation you can name an instance of it, making sure the name you use hasn't already been allocated to refer to something else. So if you have a sentence of the form ∃xφ in X and a designator (name) *a* that does not occur in X then, if ψ is a matter of replacing every free occurrence of x in φ by *a*, then you can add *ψ* to *X* without creating an inconsistency.

Hodges notes that when you are to use both the quantifier rules it is usually strategically better to use the ∃xφ rule first.

Previously we have had to consider not only positive claims but also negative ones. There is a convenient equivalence that means we don't need any new rules for the quantifiers. The quantifiers are interdefinable thus: ∃xφ ↔ ¬∀x¬φ ; and ∀xφ ↔ ¬∃x¬φ, so ¬∀xφ works the same as ∃x¬φ and ¬∃xφ works the same as ∀x¬φ.

An illustration

Hodges (p. 208) gives this argument:

> Bank-notes all carry a metal strip. Anything with a metal strip can be detected by X-rays. Therefore bank-notes can be detected by X-rays.

Let us use the following abbreviations:

B = is a bank-note
M = has a metal strip
D = can be detected by X-rays.

Then our premises are:

∀x(Bx → Mx)
∀x(Mx → Dx)

and the conclusion is:

∀x(Bx → Dx).

The tree here will involve negating the conclusion, ¬(∀x(Bx → Dx)), so we use our rule about negated quantifiers to get: ∃x¬(Bx → Dx).

So our tree starts with

∀x(Bx → Mx)
∀x(Mx → Dx)
∃x¬(Bx → Dx)

Following his advice we start with the existential claim, using any name not already used (there are none) and then converting the first two claims to Herbrand sentences:

¬(Bb → Db)
Bb → Mb
Mb → Db

We then use the rules we already know for propositional trees to show that this set of Herbrand sentences yields a closed tree:

```
                Bb
               ¬Db
                |
      ____________________
   ¬Bb                   Mb
    --------              |
   ---------------------------------
  ¬Mb                      Db
 --------                -------
```