IFL: Logicbite 10

PL languages: variations

Peter Smith

Chapters 9 and 10 of *IFL* introduced the syntax and semantics of formal languages apt for exploring the logic of the propositional connectives 'and', 'or' and 'not'.

But a glance at the textbooks will quickly reveal that other authors do things in variously different ways. There's no single gold-standard story.

❦

Many of the differences in the details are pretty trivial. Let's quickly mention some of these, just to set them aside (I'll be relaxed about quotation marks):

(1) *IFL* uses e.g. $\mathsf{P}, \mathsf{Q}, \mathsf{R}$ for atomic wffs (a.k.a. propositional letters). The use of mid-alphabet upper case letters is widespread, but some use p, q, r or A, B, C. I then use the 'prime' character to form extra atomic wffs if needed, e.g. $\mathsf{P}', \mathsf{P}'', \mathsf{Q}'''$; some use subscript numbers, e.g. A_2, B_{43}.

(2) *IFL* uses $\wedge, \vee, \neg$ for the three connectives, the majority modern choice. The use of the wedge character for disjunction is universal (unless we go Polish, as in Exercises 10d). But old-school alternatives to $(\mathsf{P} \wedge \mathsf{Q})$ include $(\mathsf{P}\ \&\ \mathsf{Q})$, $(\mathsf{P}.\mathsf{Q})$ or just (PQ), with or without brackets. And alternatives to $\neg\mathsf{P}$ include $\neg(\mathsf{P})$, $\sim\mathsf{P}$, $-\mathsf{P}$, $\overline{p}$.

(3) The official bracketing policy in *IFL* is simple but stern. Some authors allow us to drop outermost brackets, so permitting $\mathsf{P} \wedge (\mathsf{Q} \vee \mathsf{R})$. Some relax further by dropping brackets from multiple conjunctions or multiple disjunctions, so allowing $\mathsf{P} \vee \mathsf{Q} \vee \mathsf{R}$. Some introduce the convention that $\wedge$ binds more tightly than $\vee$ so that $\mathsf{P} \wedge \mathsf{Q} \vee \mathsf{R}$ is unequivocally equivalent to the bracketed $((\mathsf{P} \wedge \mathsf{Q}) \vee \mathsf{R})$.

(4) I employ Greek letters as 'schematic variables' for use when we want to talk in logician's symbol-augmented English about wffs in our PL languages. Many stick to Roman letters. So, for example, if $\mathsf{P}, \mathsf{Q}, \mathsf{R}$ are propositional letters in PL, the role of schematic letters can be played by A, B, C, or $\mathcal{A}, \mathcal{B}, \mathcal{C}$ etc., using some such typographical difference in style to mark the difference in role.

None of these variations raises any issue of principle, and none should give you pause if you encounter it elsewhere!

❦

So far, so good. Now things get a bit more complicated, starting from the very basics. Let's ask, what are propositional letters – the likes of our $\mathsf{P}, \mathsf{Q}, \mathsf{R}$ – *for*?

If you look at even a good book like Lemmon's *Beginning Logic*, it is easy to get confused. For example, on p. 6, his P and Q are particular propositions; on p. 8, P is a *stand-in* for a particular proposition; but on p. 52, $P \vee -P$ is a *general* law which "effectively affirms that for any proposition, either it or its negation is the case."

Let's turn though to Goldfarb, who is eminently clear about what *his* 'p's and 'q's are doing:

> We have been using the letters "p", "q", "r", and so on to represent statements, and have been looking at expressions like "$p \vee q$" and "$(p.q) \supset r$", which represent compound statements. We call "p", "q", ... *sentence letters*, and the compounds constructed from them and the truth-functional connectives *truth-functional schemata*. ... Schemata are not themselves statements. Their constituents, the sentence letters,

> state nothing, but are mere stand-ins for statements. Schemata are logical diagrams of statements, diagrams obtained by abstracting from all the internal features of the statements save those relevant to the logical structures with which we are concerned. An interpretation of sentence letters is a correlation of a statement with each of the sentence letters. Given such a correlation, a schema constructed from the sentence letters is interpreted by replacing each letter with its correlated statement. Thus, under the interpretation that correlates "Figaro exulted" with "p", "Basilio fretted" with "q", and "the Count had a plan" with "r", the schema "$p.(q \vee r)$" becomes the statement
>
> (1) Figaro exulted . (Basilio fretted $\vee$ the Count had a plan)
>
> or, in ordinary language,
>
> (2) Figaro exulted and either Basilio fretted or the Count had a plan.
>
> To say that a statement has the logical form given by a certain schema, or that the statement is schematized by the schema, is just to say that there is an interpretation under which the schema becomes the statement (or, more pedantically: there is an interpretation under which the schema becomes a paraphrased form of the statement).

Don't worry about the 'truth-functional' – that becomes clear in *IFL* Chapter 12. The point for now is that for Goldfarb an expression like '$p.(q \vee r)$' is not a statement or proposition: it says nothing. And therefore a derivation from '$p.(q \vee r)$' to 'p' is not really an argument either (for arguments involve contentful propositions, and this – on Goldfarb's view – involves two schemata which aren't contentful, but are mere diagrams for potential propositions).

Now this *is* one way of conceptualizing things. But it does have the result that you can then have a whole logic book full of apparent formal proofs, sequences that *look* like genuine derivations (albeit in an artificial formal language), but which in fact contains no real formal proofs, just templates or diagrams for possible proofs in (tidied up) ordinary language. This I find a very odd way of thinking about what is going on e.g. when we formalize mathematician's proofs! We are then, I thought, still in the business of directly proving contentful claims, just more rigorously.

I prefer then to take a different line in *IFL*: I think of propositional letters – and expressions belonging to more useful formal language later in the book – as belonging to interpreted formal languages, and so for me a derivation from '$\mathsf{P} \wedge (\mathsf{Q} \vee \mathsf{R})$' to '$\mathsf{P}$' is indeed a proof of one contentful proposition for another (which propositions depending of course of the language-defining glossary which is in force).

Still, the differences between Goldfarb's line and mine doesn't matter too much at the end of the day. It mostly will effect our *commentaries* on what are up to in our formal symbol-juggling in our respective later chapters, rather than the technical details of the juggling.

❦

One way the different approaches can show up is this. Goldfarb thinks in terms of there being a single symbolic apparatus, available to be interpreted in many ways (there is one schematic language for doing propositional logic); I think in terms of there being multiple languages with their different built-in interpretations.

But others who don't share Goldfarb's 'it's all schematic' approach can also take the one-language approach. Here's Nick Smith (omitting his references to connectives in addition to our basic three):

> We now summarize in a compact and more formal way what we have said about the language PL. . . . . To describe the syntax of a language is to say (1) what the basic symbols of the language are, and (2) how these symbols can be combined to form the sentences of the language. The sentences of PL are called wffs.

(1) The symbols of PL are:

- (i) basic propositions: $A, A_2, A_3, \ldots, B, B_2, B_3, \ldots, C, C_2, C_3, \ldots, Z, Z_2, Z_3, \ldots$
- (ii) connectives: $\neg, \wedge, \vee$
- (iii) punctuation symbols (parentheses): ( )

(2) Wffs of PL are defined as follows:

- (i) Any basic proposition is a wff.
- (ii) If α and β are wffs, then so are: $\neg\alpha$, $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$.
- (iii) Nothing else is a wff.

Which is all very familiar, except for one significant difference. For note the talk here is of a *single* PL language, which has an *unlimited* number of basic atoms (indeed, an infinite vocabulary). By contrast, as I said, I talk in *IFL* about PL languages, *plural*, with each one having its particular *limited* vocabulary (with a finite supply of atoms).

So which is it to be: a single PL language (Nick Smith) or multiple languages (Peter Smith)? Among authors we've met before, Goldfarb, Lemmon, and Teller also write as if there is a single formal language for propositional logic, though they have somewhat different stories about how this one language is used. Many others, including the authors of *The Logic Book*,[1] take the same one-language line.

I seem to be an outlier here. Though I don't feel too abashed about this! Two points:

(1) When we come to *apply* our formal propositional language(s) in regimenting particular arguments, we'll only need to use a limited number of atoms. And we'll only need to consider assignments of truth values to these atoms. So if we have infinitely more redundant atoms hanging around, we need to have some official policy allowing us to forget about *them* (allowing us to not worry about about assignments of values to them). Yet it is surely neater not to bring them into the story in the first place. Rather, use a limited PL language which is built for the needs of the case.

(2) More seriously, when we come to investigating quantificational logic, we introduce richer formal QL languages. And it is entirely natural (and common) to talk in the plural *here*. Mathematicians talk of 'the language of set theory', 'the language of first-order arithmetic', 'the language of ring theory' and so on. Formalized, these will be different QL languages, each with their own (very) limited specialist vocabulary. I can see no very good and principled reason for talking of the many topic-specific QL languages like this, while saying that there is just a single PL all-purpose language.

But to repeat, at the end of the day, nothing too important depends on which way we jump on this sort of point. I just want you to be alert to the fact that different logic books do present things in slightly different ways, depending on how exactly they conceive of the role of formal languages in propositional logic.

❦

Amplifying just a bit, here in crude outline is the story about the semantics of PL languages as sketched in *IFL*, Chapter 10.

(1) We give the atoms of a PL language an *interpretation* by using a glossary or translation key. The meaning of the atoms together with the fixed meaning of the connectives then determines the meaning of complex sentences of our language.

(2) The interpretations of the atoms of our language together with the state of the world will fix a *valuation* of those atoms, where a valuation assigns each atom the value *true* or *false*. And this truth-value assignment for the atoms will in turn determine corresponding truth values for all the wffs of the PL language.

[1] This text by Merrie Bergmann, James Moor and Jack Nelson is very widely used and is now in its sixth edition. It strikes me as doing a good job of making elementary logic unnecessarily hard going.

(3) And the different possible ways the world might be determine different possible valuations for the atoms.

Now looking ahead: we are going to be interested in working out which arguments are valid in virtue of the way the connectives feature in them. For arguments rendered into a PL language, we then ask the question: does every possible valuation of the relevant atoms which makes the premisses true make the conclusion true too?

So, in headline terms, we assess validity-in-virtue-of-the-connectives by looking at valuations. Interpretations of atoms will drop out of the picture, precisely because we are not interested for the moment in arguments whose validity depends on the internal meaning of consistent atomic propositions in premisses and conclusions, but just on how they are joined together by the connectives.

Sorry: that's no doubt a bit telegraphic. All, I hope, will become clear by the time you've read *IFL* Chapter 15! For the moment what you need to take away is the thought that *valuations* (as opposed to interpretations-via-a-glossary) become the central concept we care about when assessing validity-in-virtue-of-the-connectives for PL arguments. And that's why some, when they set up the official semantics for their PL languages, jump straight to valuations, omitting to give any meanings to their propositional symbols. And just to be annoying, a few authors who jump straight to valuations call *them* interpretations instead!

Nick Smith talks of glossaries, and Paul Teller of transcriptions – so like *IFL*, they give their propositional letters meanings (varying from context to context). *The Logic Book*, by contrast, is something of a mess. The opening chapter talks of propositional letters being used to abbreviate English sentences, and so being given interpretations (in context): but by Chapter 3, when giving the official semantics of *SL* (the book's unique official language for sentential logic), meaning-giving interpretations have disappeared and we just get truth-value assignments. Something similar happens e.g. in Richmond Thomason's more mathematically-flavoured classic *Symbolic Logic: An Introduction.*[2] Early on, his '*p*'s and '*q*'s feature in *translations* of everyday arguments: but in his later official semantics, there are just valuations. Geoffrey Hunter in his *Metalogic* is one of those who call valuations 'interpretations'.[3] And Quine, in his *Methods of Logic*[4] says outright "By interpretation of the letter '*p*' (or '*q*' etc.) may be meant may be meant specification of an actual statement which is to be imagined in place of the letter. By interpretation of '*p*' may also be meant simply specification of a truth value for '*p*'." Which you might think is in danger of occluding an important distinction!

Once more, at the end of the day, nothing too important depends on exactly what we say here. Just be clear that, in some terminology other, that there is a distinction between on the one hand assigning senses to atoms (if we do that) and on the other hand assigning truth-values. And you need to come to understand why it's the second that matters in defining validity-in-virtue-of-the-connectives for PL arguments – while also noting that they won't really *be* arguments, properly understood, if their propositional symbols aren't assigned meanings!

[2]This 1970 book is intended to be introductory, but is quite tough going and makes unnecessary heavy weather of some topics. But back in the day, I learnt a *lot* from it. The author has since written widely on logical topics, and for a decade was the chief editor of the *Journal of Philosophical Logic*.

[3]Geoffrey Hunter's 1971 book, a second-level logic text, is excellent, and still eventually well worth reading. Geoffrey long ago told me a salutary story. At the beginning of his intro logic course, having explained the idea of a valid argument, he gave out a sheet of examples to see which arguments beginners naively judged to be valid and which not. Then, at the end of the course, he gave out the same example sheet, asked which arguments were valid – and people on average did *worse*!

Well, I guess you can see why! Students learn some shiny new tools and are then tempted to apply the tools mindlessly, so e.g. faced with inferences involving conditionals, despite all the warnings you will soon meet, they turn the handle, mechanically do a 'truth-table', and out comes a silly answer.

[4]Willard Van Orman Quine is one of the most important and influential philosophers of the second half of the twentieth century. His star is no longer in the ascendant: but his *Word and Object* remains a must-read classic for philosophers, if only in order to understand what various later writers are reacting against. Quine's influential *Methods of Logic* was first published in 1952.