Chapter udf

Beyond First-order Logic

> This chapter, adapted from Jeremy Avigad's logic notes, gives the briefest of glimpses into which other logical systems there are. It is intended as a chapter suggesting further topics for study in a course that does not cover them. Each one of the topics mentioned here will—hopefully—eventually receive its own part-level treatment in the Open Logic Project.

byd.1 Overview

fol:byd:int:sec First-order logic is not the only system of logic of interest: there are many extensions and variations of first-order logic. A logic typically consists of the formal specification of a language, usually, but not always, a deductive system, and usually, but not always, an intended semantics. But the technical use of the term raises an obvious question: what do logics that are not first-order logic have to do with the word "logic," used in the intuitive or philosophical sense? All of the systems described below are designed to model reasoning of some form or another; can we say what makes them logical?

No easy answers are forthcoming. The word "logic" is used in different ways and in different contexts, and the notion, like that of "truth," has been analyzed from numerous philosophical stances. For example, one might take the goal of logical reasoning to be the determination of which statements are necessarily true, true a priori, true independent of the interpretation of the nonlogical terms, true by virtue of their form, or true by linguistic convention; and each of these conceptions requires a good deal of clarification. Even if one restricts one's attention to the kind of logic used in mathematics, there is little agreement as to its scope. For example, in the *Principia Mathematica*, Russell and Whitehead tried to develop mathematics on the basis of logic, in the *logicist* tradition begun by Frege. Their system of logic was a form of higher-type logic similar to the one described below. In the end they were forced to introduce axioms which, by most standards, do not seem purely logical (notably, the axiom of infinity, and the axiom of reducibility), but one might nonetheless hold that some forms of higher-order reasoning should be accepted as logical. In contrast, Quine, whose ontology does not admit "propositions" as legitimate objects of discourse, argues that second-order and higher-order logic are really manifestations of set theory in sheep's clothing; in other words, systems involving quantification over predicates are not purely logical.

For now, it is best to leave such philosophical issues for a rainy day, and simply think of the systems below as formal idealizations of various kinds of reasoning, logical or otherwise.

byd.2 Many-Sorted Logic

In first-order logic, variables and quantifiers range over a single domain. But it is often useful to have multiple (disjoint) domains: for example, you might want to have a domain of numbers, a domain of geometric objects, a domain of functions from numbers to numbers, a domain of abelian groups, and so on. fol:byd:msl:sec

Many-sorted logic provides this kind of framework. One starts with a list of "sorts"—the "sort" of an object indicates the "domain" it is supposed to inhabit. One then has variables and quantifiers for each sort, and (usually) an identity predicate for each sort. Functions and relations are also "typed" by the sorts of objects they can take as arguments. Otherwise, one keeps the usual rules of first-order logic, with versions of the quantifier-rules repeated for each sort.

For example, to study international relations we might choose a language with two sorts of objects, French citizens and German citizens. We might have a unary relation, "drinks wine," for objects of the first sort; another unary relation, "eats wurst," for objects of the second sort; and a binary relation, "forms a multinational married couple," which takes two arguments, where the first argument is of the first sort and the second argument is of the second sort. If we use variables a, b, c to range over French citizens and x, y, z to range over German citizens, then

$$\forall a\,\forall x[(\mathit{MarriedTo}(a,x) \rightarrow (\mathit{DrinksWine}(a) \lor \lnot \mathit{EatsWurst}(x))]]$$

asserts that if any French person is married to a German, either the French person drinks wine or the German doesn't eat wurst.

Many-sorted logic can be embedded in first-order logic in a natural way, by lumping all the objects of the many-sorted domains together into one first-order domain, using unary predicate symbols to keep track of the sorts, and relativizing quantifiers. For example, the first-order language corresponding to the example above would have unary predicate symbols "German" and "French," in addition to the other relations described, with the sort requirements erased. A sorted quantifier $\forall x\,\varphi$, where x is a variable of the German sort, translates to

$$\forall x\,(\mathit{German}(x) \rightarrow \varphi).$$

We need to add axioms that insure that the sorts are separate—e.g., $\forall x\,\lnot(\mathit{German}(x) \land \mathit{French}(x))$—as well as axioms that guarantee that "drinks wine" only holds of objects satisfying the predicate $\mathit{French}(x)$, etc. With these conventions and axioms, it is not difficult to show that many-sorted sentences translate to first-order sentences, and many-sorted derivations translate to first-order derivations. Also, many-sorted structures "translate" to corresponding first-order structures and vice-versa, so we also have a completeness theorem for many-sorted logic.

byd.3 Second-Order logic

fol:byd:sol:sec The language of second-order logic allows one to quantify not just over a domain of individuals, but over relations on that domain as well. Given a first-order language $\mathcal{L}$, for each k one adds variables R which range over k-ary relations, and allows quantification over those variables. If R is a variable for a k-ary relation, and t_1, …, t_k are ordinary (first-order) terms, $R(t_1, \dots, t_k)$ is an atomic formula. Otherwise, the set of formulas is defined just as in the case of first-order logic, with additional clauses for second-order quantification. Note that we only have the identity predicate for first-order terms: if R and S are relation variables of the same arity k, we can define $R = S$ to be an abbreviation for

$$\forall x_1 \dots \forall x_k \, (R(x_1, \dots, x_k) \leftrightarrow S(x_1, \dots, x_k)).$$

The rules for second-order logic simply extend the quantifier rules to the new second order variables. Here, however, one has to be a little bit careful to explain how these variables interact with the predicate symbols of $\mathcal{L}$, and with formulas of $\mathcal{L}$ more generally. At the bare minimum, relation variables count as terms, so one has inferences of the form

$$\varphi(R) \vdash \exists R \, \varphi(R)$$

But if $\mathcal{L}$ is the language of arithmetic with a constant relation symbol $<$, one would also expect the following inference to be valid:

$$x < y \vdash \exists R \, R(x, y)$$

or for a given formula φ,

$$\varphi(x_1, \dots, x_k) \vdash \exists R \, R(x_1, \dots, x_k)$$

More generally, we might want to allow inferences of the form

$$\varphi[\lambda \vec{x}. \, \psi(\vec{x})/R] \vdash \exists R \, \varphi$$

where $\varphi[\lambda \vec{x}. \, \psi(\vec{x})/R]$ denotes the result of replacing every atomic formula of the form $R t_1, \dots, t_k$ in φ by $\psi(t_1, \dots, t_k)$. This last rule is equivalent to having a *comprehension schema*, i.e., an axiom of the form

$$\exists R \, \forall x_1, \dots, x_k \, (\varphi(x_1, \dots, x_k) \leftrightarrow R(x_1, \dots, x_k)),$$

one for each formula φ in the second-order language, in which R is not a free variable. (Exercise: show that if R is allowed to occur in φ, this schema is inconsistent!)

When logicians refer to the "axioms of second-order logic" they usually mean the minimal extension of first-order logic by second-order quantifier rules together with the comprehension schema. But it is often interesting to study weaker subsystems of these axioms and rules. For example, note that in its full generality the axiom schema of comprehension is *impredicative*: it allows one to assert the existence of a relation $R(x_1, \dots, x_k)$ that is "defined" by a formula with second-order quantifiers; and these quantifiers range over the set of all such relations—a set which includes R itself! Around the turn of the twentieth century, a common reaction to Russell's paradox was to lay the blame on such definitions, and to avoid them in developing the foundations of mathematics. If one prohibits the use of second-order quantifiers in the formula φ, one has a *predicative* form of comprehension, which is somewhat weaker.

From the semantic point of view, one can think of a second-order structure as consisting of a first-order structure for the language, coupled with a set of relations on the domain over which the second-order quantifiers range (more precisely, for each k there is a set of relations of arity k). Of course, if comprehension is included in the derivation system, then we have the added requirement that there are enough relations in the "second-order part" to satisfy the comprehension axioms—otherwise the derivation system is not sound! One easy way to insure that there are enough relations around is to take the second-order part to consist of *all* the relations on the first-order part. Such a structure is called *full*, and, in a sense, is really the "intended structure" for the language. If we restrict our attention to full structures we have what is known as the *full* second-order semantics. In that case, specifying a structure boils down to specifying the first-order part, since the contents of the second-order part follow from that implicitly.

To summarize, there is some ambiguity when talking about second-order logic. In terms of the derivation system, one might have in mind either

1. A "minimal" second-order derivation system, together with some comprehension axioms.

2. The "standard" second-order derivation system, with full comprehension.

In terms of the semantics, one might be interested in either

1. The "weak" semantics, where a structure consists of a first-order part, together with a second-order part big enough to satisfy the comprehension axioms.

2. The "standard" second-order semantics, in which one considers full structures only.

When logicians do not specify the derivation system or the semantics they have in mind, they are usually referring to the second item on each list. The advantage to using this semantics is that, as we will see, it gives us categorical descriptions of many natural mathematical structures; at the same time, the derivation system is quite strong, and sound for this semantics. The drawback is that the derivation system is *not* complete for the semantics; in fact, *no* effectively given derivation system is complete for the full second-order semantics. On the other hand, we will see that the derivation system *is* complete for the weakened semantics; this implies that if a sentence is not provable, then there is *some* structure, not necessarily the full one, in which it is false.

The language of second-order logic is quite rich. One can identify unary relations with subsets of the domain, and so in particular you can quantify over these sets; for example, one can express induction for the natural numbers with a single axiom

$$\forall R\,((R(\mathsf{o}) \land \forall x\,(R(x) \to R(x'))) \to \forall x\,R(x)).$$

If one takes the language of arithmetic to have symbols $\mathsf{o}, \prime, +, \times$ and $<$, one can add the following axioms to describe their behavior:

1. $\forall x\,\neg x' = \mathsf{o}$
2. $\forall x\,\forall y\,(s(x) = s(y) \to x = y)$
3. $\forall x\,(x + \mathsf{o}) = x$
4. $\forall x\,\forall y\,(x + y') = (x + y)'$
5. $\forall x\,(x \times \mathsf{o}) = \mathsf{o}$
6. $\forall x\,\forall y\,(x \times y') = ((x \times y) + x)$
7. $\forall x\,\forall y\,(x < y \leftrightarrow \exists z\,y = (x + z'))$

It is not difficult to show that these axioms, together with the axiom of induction above, provide a categorical description of the structure $\mathfrak{N}$, the standard model of arithmetic, provided we are using the full second-order semantics. Given any structure $\mathfrak{M}$ in which these axioms are true, define a function f from $\mathbb{N}$ to the domain of $\mathfrak{M}$ using ordinary recursion on $\mathbb{N}$, so that $f(0) = \mathsf{o}^{\mathfrak{M}}$ and $f(x+1) = \prime^{\mathfrak{M}}(f(x))$. Using ordinary induction on $\mathbb{N}$ and the fact that axioms (1) and (2) hold in $\mathfrak{M}$, we see that f is injective. To see that f is surjective, let P be the set of elements of $|\mathfrak{M}|$ that are in the range of f. Since $\mathfrak{M}$ is full, P is in the second-order domain. By the construction of f, we know that $\mathsf{o}^{\mathfrak{M}}$ is in P, and that P is closed under $\prime^{\mathfrak{M}}$. The fact that the induction axiom holds in $\mathfrak{M}$ (in particular, for P) guarantees that P is equal to the entire first-order domain of $\mathfrak{M}$. This shows that f is a bijection. Showing that f is a homomorphism is no more difficult, using ordinary induction on $\mathbb{N}$ repeatedly.

In set-theoretic terms, a function is just a special kind of relation; for example, a unary function f can be identified with a binary relation R satisfying $\forall x\,\exists! y\,R(x, y)$. As a result, one can quantify over functions too. Using the full semantics, one can then define the class of infinite structures to be the class of structures $\mathfrak{M}$ for which there is an injective function from the domain of $\mathfrak{M}$ to a proper subset of itself:

$$\exists f\,(\forall x\,\forall y\,(f(x) = f(y) \rightarrow x = y) \land \exists y\,\forall x\,f(x) \neq y).$$

The negation of this sentence then defines the class of finite structures.

In addition, one can define the class of well-orderings, by adding the following to the definition of a linear ordering:

$$\forall P\,(\exists x\,P(x) \rightarrow \exists x\,(P(x) \land \forall y\,(y < x \rightarrow \lnot P(y)))).$$

This asserts that every non-empty set has a least element, modulo the identification of "set" with "one-place relation". For another example, one can express the notion of connectedness for graphs, by saying that there is no nontrivial separation of the vertices into disconnected parts:

$$\lnot\exists A\,(\exists x\,A(x) \land \exists y\,\lnot A(y) \land \forall w\,\forall z\,((A(w) \land \lnot A(z)) \rightarrow \lnot R(w, z))).$$

For yet another example, you might try as an exercise to define the class of finite structures whose domain has even size. More strikingly, one can provide a categorical description of the real numbers as a complete ordered field containing the rationals.

In short, second-order logic is much more expressive than first-order logic. That's the good news; now for the bad. We have already mentioned that there is no effective derivation system that is complete for the full second-order semantics. For better or for worse, many of the properties of first-order logic are absent, including compactness and the Löwenheim–Skolem theorems.

On the other hand, if one is willing to give up the full second-order semantics in terms of the weaker one, then the minimal second-order derivation system is complete for this semantics. In other words, if we read $\vdash$ as "proves in the minimal system" and $\vDash$ as "logically implies in the weaker semantics", we can show that whenever $\Gamma \vDash \varphi$ then $\Gamma \vdash \varphi$. If one wants to include specific comprehension axioms in the derivation system, one has to restrict the semantics to second-order structures that satisfy these axioms: for example, if Δ consists of a set of comprehension axioms (possibly all of them), we have that if $\Gamma \cup \Delta \vDash \varphi$, then $\Gamma \cup \Delta \vdash \varphi$. In particular, if φ is not provable using the comprehension axioms we are considering, then there is a model of $\lnot\varphi$ in which these comprehension axioms nonetheless hold.

The easiest way to see that the completeness theorem holds for the weaker semantics is to think of second-order logic as a many-sorted logic, as follows. One sort is interpreted as the ordinary "first-order" domain, and then for each k we have a domain of "relations of arity k." We take the language to have built-in relation symbols "$\mathit{true}_k(R, x_1, \dots, x_k)$" which is meant to assert that R holds of x_1, …, x_k, where R is a variable of the sort "k-ary relation" and x_1, …, x_k are objects of the first-order sort.

With this identification, the weak second-order semantics is essentially the usual semantics for many-sorted logic; and we have already observed that many-sorted logic can be embedded in first-order logic. Modulo the translations back and forth, then, the weaker conception of second-order logic is really a form of first-order logic in disguise, where the domain contains both "objects" and "relations" governed by the appropriate axioms.

byd.4 Higher-Order logic

fol:byd:hol:sec Passing from first-order logic to second-order logic enabled us to talk about sets of objects in the first-order domain, within the formal language. Why stop there? For example, third-order logic should enable us to deal with sets of sets of objects, or perhaps even sets which contain both objects and sets of objects. And fourth-order logic will let us talk about sets of objects of that kind. As you may have guessed, one can iterate this idea arbitrarily.

In practice, higher-order logic is often formulated in terms of functions instead of relations. (Modulo the natural identifications, this difference is inessential.) Given some basic "sorts" A, B, C, … (which we will now call "types"), we can create new ones by stipulating

> If σ and τ are finite types then so is $\sigma \to \tau$.

Think of types as syntactic "labels," which classify the objects we want in our domain; $\sigma \to \tau$ describes those objects that are functions which take objects of type σ to objects of type τ. For example, we might want to have a type Ω of truth values, "true" and "false," and a type $\mathbb{N}$ of natural numbers. In that case, you can think of objects of type $\mathbb{N} \to \Omega$ as unary relations, or subsets of $\mathbb{N}$; objects of type $\mathbb{N} \to \mathbb{N}$ are functions from natural numbers to natural numbers; and objects of type $(\mathbb{N} \to \mathbb{N}) \to \mathbb{N}$ are "functionals," that is, higher-type functions that take functions to numbers.

As in the case of second-order logic, one can think of higher-order logic as a kind of many-sorted logic, where there is a sort for each type of object we want to consider. But it is usually clearer just to define the syntax of higher-type logic from the ground up. For example, we can define a set of finite types inductively, as follows:

1. $\mathbb{N}$ is a finite type.
2. If σ and τ are finite types, then so is $\sigma \to \tau$.
3. If σ and τ are finite types, so is $\sigma \times \tau$.

Intuitively, $\mathbb{N}$ denotes the type of the natural numbers, $\sigma \to \tau$ denotes the type of functions from σ to τ, and $\sigma \times \tau$ denotes the type of pairs of objects, one from σ and one from τ. We can then define a set of terms inductively, as follows:

1. For each type σ, there is a stock of variables x, y, z, … of type σ
2. $\mathsf{0}$ is a term of type $\mathbb{N}$

3. S (successor) is a term of type $\mathbb{N} \to \mathbb{N}$

4. If s is a term of type σ, and t is a term of type $\mathbb{N} \to (\sigma \to \sigma)$, then R_{st} is a term of type $\mathbb{N} \to \sigma$

5. If s is a term of type $\tau \to \sigma$ and t is a term of type τ, then $s(t)$ is a term of type σ

6. If s is a term of type σ and x is a variable of type τ, then $\lambda x.\, s$ is a term of type $\tau \to \sigma$.

7. If s is a term of type σ and t is a term of type τ, then $\langle s, t \rangle$ is a term of type $\sigma \times \tau$.

8. If s is a term of type $\sigma \times \tau$ then $p_1(s)$ is a term of type σ and $p_2(s)$ is a term of type τ.

Intuitively, R_{st} denotes the function defined recursively by

$$R_{st}(0) = s$$
$$R_{st}(x+1) = t(x, R_{st}(x)),$$

$\langle s, t \rangle$ denotes the pair whose first component is s and whose second component is t, and $p_1(s)$ and $p_2(s)$ denote the first and second elements ("projections") of s. Finally, $\lambda x.\, s$ denotes the function f defined by

$$f(x) = s$$

for any x of type σ; so item (6) gives us a form of comprehension, enabling us to define functions using terms. Formulas are built up from identity predicate statements $s = t$ between terms of the same type, the usual propositional connectives, and higher-type quantification. One can then take the axioms of the system to be the basic equations governing the terms defined above, together with the usual rules of logic with quantifiers and identity predicate.

If one augments the finite type system with a type Ω of truth values, one has to include axioms which govern its use as well. In fact, if one is clever, one can get rid of complex formulas entirely, replacing them with terms of type Ω! The proof system can then be modified accordingly. The result is essentially the *simple theory of types* set forth by Alonzo Church in the 1930s.

As in the case of second-order logic, there are different versions of higher-type semantics that one might want to use. In the full version, variables of type $\sigma \to \tau$ range over the set of *all* functions from the objects of type σ to objects of type τ. As you might expect, this semantics is too strong to admit a complete, effective derivation system. But one can consider a weaker semantics, in which a structure consists of sets of elements T_τ for each type τ, together with appropriate operations for application, projection, etc. If the details are carried out correctly, one can obtain completeness theorems for the kinds of derivation systems described above.

Higher-type logic is attractive because it provides a framework in which we can embed a good deal of mathematics in a natural way: starting with $\mathbb{N}$, one can define real numbers, continuous functions, and so on. It is also particularly attractive in the context of intuitionistic logic, since the types have clear "constructive" interpretations. In fact, one can develop constructive versions of higher-type semantics (based on intuitionistic, rather than classical logic) that clarify these constructive interpretations quite nicely, and are, in many ways, more interesting than the classical counterparts.

byd.5 Intuitionistic Logic

fol:byd:il:sec In contrast to second-order and higher-order logic, intuitionistic first-order logic represents a restriction of the classical version, intended to model a more "constructive" kind of reasoning. The following examples may serve to illustrate some of the underlying motivations.

Suppose someone came up to you one day and announced that they had determined a natural number x, with the property that if x is prime, the Riemann hypothesis is true, and if x is composite, the Riemann hypothesis is false. Great news! Whether the Riemann hypothesis is true or not is one of the big open questions of mathematics, and here they seem to have reduced the problem to one of calculation, that is, to the determination of whether a specific number is prime or not.

What is the magic value of x? They describe it as follows: x is the natural number that is equal to 7 if the Riemann hypothesis is true, and 9 otherwise.

Angrily, you demand your money back. From a classical point of view, the description above does in fact determine a unique value of x; but what you really want is a value of x that is given *explicitly*.

To take another, perhaps less contrived example, consider the following question. We know that it is possible to raise an irrational number to a rational power, and get a rational result. For example, $\sqrt{2}^2 = 2$. What is less clear is whether or not it is possible to raise an irrational number to an *irrational* power, and get a rational result. The following theorem answers this in the affirmative:

Theorem byd.1. *There are irrational numbers a and b such that a^b is rational.*

Proof. Consider $\sqrt{2}^{\sqrt{2}}$. If this is rational, we are done: we can let $a = b = \sqrt{2}$. Otherwise, it is irrational. Then we have

$$(\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = \sqrt{2}^{\sqrt{2}\cdot\sqrt{2}} = \sqrt{2}^2 = 2,$$

which is certainly rational. So, in this case, let a be $\sqrt{2}^{\sqrt{2}}$, and let b be $\sqrt{2}$. □

Does this constitute a valid proof? Most mathematicians feel that it does. But again, there is something a little bit unsatisfying here: we have proved the existence of a pair of real numbers with a certain property, without being able to say *which* pair of numbers it is. It is possible to prove the same result, but in such a way that the pair a, b *is* given in the proof: take $a = \sqrt{3}$ and $b = \log_3 4$. Then

$$a^b = \sqrt{3}^{\log_3 4} = 3^{1/2 \cdot \log_3 4} = (3^{\log_3 4})^{1/2} = 4^{1/2} = 2,$$

since $3^{\log_3 x} = x$.

Intuitionistic logic is designed to model a kind of reasoning where moves like the one in the first proof are disallowed. Proving the existence of an x satisfying $\varphi(x)$ means that you have to give a specific x, and a proof that it satisfies φ, like in the second proof. Proving that φ or ψ holds requires that you can prove one or the other.

Formally speaking, intuitionistic first-order logic is what you get if you restrict a derivation system for first-order logic in a certain way. Similarly, there are intuitionistic versions of second-order or higher-order logic. From the mathematical point of view, these are just formal deductive systems, but, as already noted, they are intended to model a kind of mathematical reasoning. One can take this to be the kind of reasoning that is justified on a certain philosophical view of mathematics (such as Brouwer's intuitionism); one can take it to be a kind of mathematical reasoning which is more "concrete" and satisfying (along the lines of Bishop's constructivism); and one can argue about whether or not the formal description captures the informal motivation. But whatever philosophical positions we may hold, we can study intuitionistic logic as a formally presented logic; and for whatever reasons, many mathematical logicians find it interesting to do so.

There is an informal constructive interpretation of the intuitionist connectives, usually known as the BHK interpretation (named after Brouwer, Heyting, and Kolmogorov). It runs as follows: a proof of $\varphi \wedge \psi$ consists of a proof of φ paired with a proof of ψ; a proof of $\varphi \vee \psi$ consists of either a proof of φ, or a proof of ψ, where we have explicit information as to which is the case; a proof of $\varphi \to \psi$ consists of a procedure, which transforms a proof of φ to a proof of ψ; a proof of $\forall x\, \varphi(x)$ consists of a procedure which returns a proof of $\varphi(x)$ for any value of x; and a proof of $\exists x\, \varphi(x)$ consists of a value of x, together with a proof that this value satisfies φ. One can describe the interpretation in computational terms known as the "Curry–Howard isomorphism" or the "formulas-as-types paradigm": think of a formula as specifying a certain kind of data type, and proofs as computational objects of these data types that enable us to see that the corresponding formula is true.

Intuitionistic logic is often thought of as being classical logic "minus" the law of the excluded middle. This following theorem makes this more precise.

Theorem byd.2. *Intuitionistically, the following axiom schemata are equivalent:*

1. $(\neg\varphi \to \bot) \to \varphi$.

2. $\varphi \lor \lnot\varphi$

3. $\lnot\lnot\varphi \to \varphi$

Obtaining instances of one schema from either of the others is a good exercise in intuitionistic logic.

The first deductive systems for intuitionistic propositional logic, put forth as formalizations of Brouwer's intuitionism, are due, independently, to Kolmogorov, Glivenko, and Heyting. The first formalization of intuitionistic first-order logic (and parts of intuitionist mathematics) is due to Heyting. Though a number of classically valid schemata are not intuitionistically valid, many are.

The *double-negation translation* describes an important relationship between classical and intuitionist logic. It is defined inductively follows (think of φ^N as the "intuitionist" translation of the classical formula φ):

$$\begin{aligned}
\varphi^N &\equiv \lnot\lnot\varphi \quad \text{for atomic formulas } \varphi \\
(\varphi \land \psi)^N &\equiv (\varphi^N \land \psi^N) \\
(\varphi \lor \psi)^N &\equiv \lnot\lnot(\varphi^N \lor \psi^N) \\
(\varphi \to \psi)^N &\equiv (\varphi^N \to \psi^N) \\
(\forall x\, \varphi)^N &\equiv \forall x\, \varphi^N \\
(\exists x\, \varphi)^N &\equiv \lnot\lnot\exists x\, \varphi^N
\end{aligned}$$

Kolmogorov and Glivenko had versions of this translation for propositional logic; for predicate logic, it is due to Gödel and Gentzen, independently. We have

Theorem byd.3. *1. $\varphi \leftrightarrow \varphi^N$ is provable classically*

2. If φ is provable classically, then φ^N is provable intuitionistically.

We can now envision the following dialogue. Classical mathematician: "I've proved φ!" Intuitionist mathematician: "Your proof isn't valid. What you've really proved is φ^N." Classical mathematician: "Fine by me!" As far as the classical mathematician is concerned, the intuitionist is just splitting hairs, since the two are equivalent. But the intuitionist insists there is a difference.

Note that the above translation concerns pure logic only; it does not address the question as to what the appropriate *nonlogical* axioms are for classical and intuitionistic mathematics, or what the relationship is between them. But the following slight extension of the theorem above provides some useful information:

Theorem byd.4. *If Γ proves φ classically, Γ^N proves φ^N intuitionistically.*

In other words, if φ is provable from some hypotheses classically, then φ^N is provable from their double-negation translations.

To show that a sentence or propositional formula is intuitionistically valid, all you have to do is provide a proof. But how can you show that it is not valid? For that purpose, we need a semantics that is sound, and preferably complete. A semantics due to Kripke nicely fits the bill.

We can play the same game we did for classical logic: define the semantics, and prove soundness and completeness. It is worthwhile, however, to note the following distinction. In the case of classical logic, the semantics was the "obvious" one, in a sense implicit in the meaning of the connectives. Though one can provide some intuitive motivation for Kripke semantics, the latter does not offer the same feeling of inevitability. In addition, the notion of a classical structure is a natural mathematical one, so we can either take the notion of a structure to be a tool for studying classical first-order logic, or take classical first-order logic to be a tool for studying mathematical structures. In contrast, Kripke structures can only be viewed as a logical construct; they don't seem to have independent mathematical interest.

A Kripke structure $\mathfrak{M} = \langle W, R, V \rangle$ for a propositional language consists of a set W, partial order R on W with a least element, and an "monotone" assignment of propositional variables to the elements of W. The intuition is that the elements of W represent "worlds," or "states of knowledge"; an element $v \geq u$ represents a "possible future state" of u; and the propositional variables assigned to u are the propositions that are known to be true in state u. The forcing relation $\mathfrak{M}, w \Vdash \varphi$ then extends this relationship to arbitrary formulas in the language; read $\mathfrak{M}, w \Vdash \varphi$ as "φ is true in state w." The relationship is defined inductively, as follows:

1. $\mathfrak{M}, w \Vdash p_i$ iff p_i is one of the propositional variables assigned to w.
2. $\mathfrak{M}, w \nVdash \bot$.
3. $\mathfrak{M}, w \Vdash (\varphi \land \psi)$ iff $\mathfrak{M}, w \Vdash \varphi$ and $\mathfrak{M}, w \Vdash \psi$.
4. $\mathfrak{M}, w \Vdash (\varphi \lor \psi)$ iff $\mathfrak{M}, w \Vdash \varphi$ or $\mathfrak{M}, w \Vdash \psi$.
5. $\mathfrak{M}, w \Vdash (\varphi \to \psi)$ iff, whenever $w' \geq w$ and $\mathfrak{M}, w' \Vdash \varphi$, then $\mathfrak{M}, w' \Vdash \psi$.

It is a good exercise to try to show that $\neg(p \land q) \to (\neg p \lor \neg q)$ is not intuitionistically valid, by cooking up a Kripke structure that provides a counterexample.

byd.6 Modal Logics

Consider the following example of a conditional sentence: fol:byd:mod:sec

> If Jeremy is alone in that room, then he is drunk and naked and dancing on the chairs.

This is an example of a conditional assertion that may be materially true but nonetheless misleading, since it seems to suggest that there is a stronger link between the antecedent and conclusion other than simply that either the antecedent is false or the consequent true. That is, the wording suggests that the claim is not only true in this particular world (where it may be trivially true, because Jeremy is not alone in the room), but that, moreover, the conclusion *would have* been true *had* the antecedent been true. In other words, one can take the assertion to mean that the claim is true not just in this world, but in any "possible" world; or that it is *necessarily* true, as opposed to just true in this particular world.

Modal logic was designed to make sense of this kind of necessity. One obtains modal propositional logic from ordinary propositional logic by adding a box operator; which is to say, if φ is a formula, so is $\Box\varphi$. Intuitively, $\Box\varphi$ asserts that φ is *necessarily* true, or true in any possible world. $\Diamond\varphi$ is usually taken to be an abbreviation for $\neg\Box\neg\varphi$, and can be read as asserting that φ is *possibly* true. Of course, modality can be added to predicate logic as well.

Kripke structures can be used to provide a semantics for modal logic; in fact, Kripke first designed this semantics with modal logic in mind. Rather than restricting to partial orders, more generally one has a set of "possible worlds," P, and a binary "accessibility" relation $R(x, y)$ between worlds. Intuitively, $R(p, q)$ asserts that the world q is compatible with p; i.e., if we are "in" world p, we have to entertain the possibility that the world could have been like q.

Modal logic is sometimes called an "intensional" logic, as opposed to an "extensional" one. The intended semantics for an extensional logic, like classical logic, will only refer to a single world, the "actual" one; while the semantics for an "intensional" logic relies on a more elaborate ontology. In addition to structureing necessity, one can use modality to structure other linguistic constructions, reinterpreting $\Box$ and $\Diamond$ according to the application. For example:

1. In provability logic, $\Box\varphi$ is read "φ is provable" and $\Diamond\varphi$ is read "φ is consistent."
2. In epistemic logic, one might read $\Box\varphi$ as "I know φ" or "I believe φ."
3. In temporal logic, one can read $\Box\varphi$ as "φ is always true" and $\Diamond\varphi$ as "φ is sometimes true."

One would like to augment logic with rules and axioms dealing with modality. For example, the system **S4** consists of the ordinary axioms and rules of propositional logic, together with the following axioms:

$$\Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi)$$
$$\Box\varphi \to \varphi$$
$$\Box\varphi \to \Box\Box\varphi$$

as well as a rule, "from φ conclude $\Box\varphi$." **S5** adds the following axiom:

$$\Diamond\varphi \to \Box\Diamond\varphi$$

Variations of these axioms may be suitable for different applications; for example, S5 is usually taken to characterize the notion of logical necessity. And the nice thing is that one can usually find a semantics for which the derivation system is sound and complete by restricting the accessibility relation in the Kripke structures in natural ways. For example, **S4** corresponds to the class of Kripke structures in which the accessibility relation is reflexive and transitive. **S5** corresponds to the class of Kripke structures in which the accessibility relation is *universal*, which is to say that every world is accessible from every other; so $\Box\varphi$ holds if and only if φ holds in every world.

byd.7 Other Logics

As you may have gathered by now, it is not hard to design a new logic. You too can create your own a syntax, make up a deductive system, and fashion a semantics to go with it. You might have to be a bit clever if you want the derivation system to be complete for the semantics, and it might take some effort to convince the world at large that your logic is truly interesting. But, in return, you can enjoy hours of good, clean fun, exploring your logic's mathematical and computational properties. fol:byd:oth:sec

Recent decades have witnessed a veritable explosion of formal logics. Fuzzy logic is designed to model reasoning about vague properties. Probabilistic logic is designed to model reasoning about uncertainty. Default logics and nonmonotonic logics are designed to model defeasible forms of reasoning, which is to say, "reasonable" inferences that can later be overturned in the face of new information. There are epistemic logics, designed to model reasoning about knowledge; causal logics, designed to model reasoning about causal relationships; and even "deontic" logics, which are designed to model reasoning about moral and ethical obligations. Depending on whether the primary motivation for introducing these systems is philosophical, mathematical, or computational, you may find such creatures studies under the rubric of mathematical logic, philosophical logic, artificial intelligence, cognitive science, or elsewhere.

The list goes on and on, and the possibilities seem endless. We may never attain Leibniz' dream of reducing all of human reason to calculation—but that can't stop us from trying.

Photo Credits

Bibliography