

byd.1 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, \dots, 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 Rt_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 proof system, then we have the added requirement that there are enough relations in the “second-order part” to satisfy the comprehension axioms—otherwise the proof 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 proof system, one might have in mind either

1. A “minimal” second-order proof system, together with some comprehension axioms.
2. The “standard” second-order proof 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 proof 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 proof system is quite strong, and sound for this semantics. The drawback is that the proof system is *not* complete for the semantics; in fact, *no* effectively given proof system is complete for the full second-order semantics. On the other hand, we will see that the proof 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

$$\forall R ((R(0) \wedge \forall x (R(x) \rightarrow R(x'))) \rightarrow \forall x R(x)).$$

If one takes the language of arithmetic to have symbols $0, ', +, \times$ and $<$, one can add the following axioms to describe their behavior:

1. $\forall x \neg x' = 0$
2. $\forall x \forall y (s(x) = s(y) \rightarrow x = y)$
3. $\forall x (x + 0) = x$
4. $\forall x \forall y (x + y') = (x + y)'$
5. $\forall x (x \times 0) = 0$
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{A} in which these axioms are true, define a function f from \mathbb{N} to the *domain* of \mathfrak{A} using ordinary recursion on \mathbb{N} , so that $f(0) = 0^{\mathfrak{A}}$ and $f(x + 1) = \iota^{\mathfrak{A}}(f(x))$. Using ordinary induction on \mathbb{N} and the fact that axioms (1) and (2) hold in \mathfrak{A} , we see that f is *injective*. To see that f is *surjective*, let P be the set of elements of $|\mathfrak{A}|$ that are in the range of f . Since \mathfrak{A} is full, P is in the second-order *domain*. By the construction of f , we know that $0^{\mathfrak{A}}$ is in P , and that P is closed under $\iota^{\mathfrak{A}}$. The fact that the induction axiom holds in \mathfrak{A} (in particular, for P) guarantees that P is equal to the entire first-order *domain* of \mathfrak{A} . 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{A} for which there is an injective function from the *domain* of \mathfrak{A} to a proper subset of itself:

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

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:

$$\forall P (\exists x P(x) \rightarrow \exists x (P(x) \wedge \forall y (y < x \rightarrow \neg P(y)))).$$

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:

$$\neg \exists A (\exists x A(x) \wedge \exists y \neg A(y) \wedge \forall w \forall z ((A(w) \wedge \neg A(z)) \rightarrow \neg R(w, z))).$$

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 proof 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 proof system is complete for this semantics. In other words, if we read \vdash as “proves in the minimal system” and \models as “logically implies in the weaker semantics”, we can show that whenever $\Gamma \models \varphi$ then $\Gamma \vdash \varphi$. If one wants to include specific comprehension axioms in the proof 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 $\Gamma \cup \Delta \models \varphi$, 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 $\neg\varphi$ 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 “ $true_k(R, x_1, \dots, x_k)$ ” which is meant to assert that R holds of x_1, \dots, x_k , where R is a variable of the sort “ k -ary relation” and x_1, \dots, 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.

Photo Credits

Bibliography