

int.1 Semantic Notions

fol:int:sem:
sec

We mentioned above that when we consider whether $\mathfrak{M}, s \models \varphi$ holds, we (for convenience) let s assign values to all **variables**, but only the values it assigns to **variables** in φ are used. In fact, it's only the values of *free* variables in φ that matter. Of course, because we're careful, we are going to prove this fact. Since **sentences** have no free variables, s doesn't matter at all when it comes to whether or not they are satisfied in a **structure**. So, when φ is a **sentence** we can define $\mathfrak{M} \models \varphi$ to mean " $\mathfrak{M}, s \models \varphi$ for all s ," which as it happens is true iff $\mathfrak{M}, s \models \varphi$ for at least one s . We need to introduce **variable** assignments to get a working definition of satisfaction for **formulas**, but for **sentences**, satisfaction is independent of the **variable** assignments.

Once we have a definition of " $\mathfrak{M} \models \varphi$," we know what "case" and "true in" mean as far as **sentences** of first-order logic are concerned. On the basis of the definition of $\mathfrak{M} \models \varphi$ for **sentences** we can then define the basic semantic notions of validity, entailment, and satisfiability. A sentence is valid, $\models \varphi$, if every **structure** satisfies it. It is entailed by a set of **sentences**, $\Gamma \models \varphi$, if every **structure** that satisfies all the **sentences** in Γ also satisfies φ . And a set of **sentences** is satisfiable if some **structure** satisfies all **sentences** in it at the same time.

Because **formulas** are inductively defined, and satisfaction is in turn defined by induction on the structure of **formulas**, we can use induction to prove properties of our semantics and to relate the semantic notions defined. We'll collect and prove some of these properties, partly because they are individually interesting, but mainly because many of them will come in handy when we go on to investigate the relation between semantics and **derivation** systems. In order to do so, we'll also have to define (precisely, i.e., by induction) some syntactic notions and operations we haven't mentioned yet.

Photo Credits

Bibliography