The Notion of "Theory" in Formal Logic

Nobo Komagata, January 14, 2005

While the notion of "theory" is used in all sorts of context, we rarely agree on the term completely. As a background discussion to understand the notion better, these notes explores the idea developed in formal logic. The materials are taken from standard logic textbooks [Enderton, 2001; Ebbinghaus et al., 1984] and a textbook on program specification Loeckx et al. [1996].

1 Logic

The presentation of logic in this subsection is intentionally left rather vague to abstract away from the details. A formal presentation based on first-order logic and many-sorted signature are found in the texts.

Informally, a logic is a specification of a language and its meaning. We first introduce the components of logic.

Definition 1 (**Language**) A language is a specification of the use of symbols.

Depending on further specification, these symbols are combined into a term, a formula, or a sentence. For example, a language of first-order logic may have a term $wife(Socrates)$, a formula $man(x) \to mortal(x)$, and a sentence $\forall x\,[man(x) \to mortal(x)]$. A language of equational logic may have a formula $x + 1 = 1 + x$. Although it is not technically correct (e.g., Enderton [2001]), we call the expressions we deal in these notes (in correspondence to "statement") as "formula."[1]

Definition 2 (**Interpretation**) Interpretation is a function (mapping) that specifies the meaning of symbols *built in* the logic under consideration.

For example, a logic of arithmetic may have a special constant '0', whose meaning is fixed in the logic (e.g., the integer '0' as we understand). Many logics also have logical connectives such as '$\wedge$' (and), whose interpretation may roughly correspond to that of natural language "and."[2]

Definition 3 (**Structure**) A structure (for a logic) is a function (mapping) that defines the meaning of symbols not defined by the interpretation.

By convention, for each sort (type) of elements, a structure assigns the universe of individuals corresponding to that sort. For each operation (in first-order logic, function or relation), a structure assigns its meaning in terms of the involved individuals. For example, a first-order logic assigns to the universal quantifier '$\forall$' a set of integers (individuals). The same first-order logic may have a relation "even" that assigns "true" for any even numbers. For a certain set of algebraic symbols, we may consider a structure of an algebra, e.g., group.

Definition 4 (**Satisfaction**) A satisfaction is a relation between structures and formulas. For a structure A and a formula φ, this relation is usually written as $A \models \varphi$.

For example, a structure of group satisfies a formula "$x + (y + z) = (x + y) + z$" (associativity).

We are now in a position to define logic in terms of the above-mentioned components.

Definition 5 (**Logic**) A logic is a combined specifications of language, interpretation, and satisfaction.

The language of a logic is also called syntax. By fixing the syntax, we can tell whether a certain expression is a formula of that logic. The remaining elements define the semantics of the logic. By fixing the semantics, we can tell whether a certain formula is satisfied by a certain structure.[3]

---

[1] In first-order logic, the distinction between "formulas" and "sentences" is that in the latter, every variable is "bound" by a quantifier. In equational logic, all variables in a "formula" are assumed to be universally quantified.

[2] For the interpretation of a formula with unbound variable, the "assignment" of variable also needs to be specified. Since we focus on quantified formulas, we skip the definition of assignment.

[3] Both the language and the structure of a logic may be specified in terms of a signature Σ, the specification of the involved "sorts" and "operations." In this case, the language and the structure can be represented as $L(\Sigma)$ and $Alg(\Sigma)$, called Σ-algebra. While the traditional first-order logic deals with a single "sort," i.e., sort of individuals, modern programming specification often uses multiple "sorts," corresponding to multiple (often complex) data types.

2 Definition of "Theory"

We now proceed to the definition of "theory" in the context of formal logic. We need to introduce a few more definitions about the relation between structures and formulas.

Definition 6 (**Model**) For a logic L, a structure A is called a model of formulas Φ, if $A \vDash \varphi$ for all $\varphi \in \Phi$.

By abusing the symbol '$\vDash$', we also write $A \vDash \Phi$. For example, a structure (algebra) of group is a model of formulas $\{x + (y + z) = (x + y) + z\}$. The set of all models of Φ is written as $Mod(\Phi)$.

Definition 7 (**Logical consequence**) For a logic L, a formula φ is called a logical consequence of Φ, if for each $A \in Mod(\Phi)$, $A \vDash \varphi$.

By still abusing the symbol '$\vDash$', we also write $\Phi \vDash \varphi$. For example, "$x + (y + z) = (x + y) + z$" is a logical consequence of the axioms of group.

Let us now proceed to define the bare-bone definition of "theory." Informally, a theory is a set of formulas that is closed under logical consequence, or complete with respect to validity.

Definition 8 (**Theory**) For a logic L, a theory is a set of formulas Φ such that for each formula φ, $\Phi \vDash \varphi$ implies $\varphi \in \Phi$.

In other words, if we can derive from a set of formulas a formula that does not belong to the set, the set is not a theory. At this point, the only requirement is that the set of formulas is validly closed. It does not even need to be consistent. Thus, an inconsistent set of formulas is still a theory as long as it satisfies the above definition. This definition of "theory" is good only if there is a mechanism to represent such a closed set. In some cases, we may list all the formulas extensionally. But it is not usually the way an interesting theory is described.

A more concise representation of a theory is based on the idea of axiomatization. Let us proceed with a few more definitions necessary for this move.

Definition 9 (**Theory of structures**) For a logic L, the theory of a class of structures $\mathcal{C}$ is the set: $Th(\mathcal{C}) = \{\varphi \mid \mathcal{C} \vDash \varphi\}$.

Definition 10 (**Consequences of formulas**) $Cn(\Phi) = Th(Mod(\Phi))$.

Definition 11 (**Axiomatizable theory**) A theory Φ is axiomatizable iff there is a decidable set Ψ such that $\Phi = Cn(\Psi)$.

A theory may or may not be axiomatizable. In addition, the set of axioms may be finite or infinite. Naturally, many interesting cases of axiomatization is finite. For example, the theory of group is finitely axiomatizable (in a certain logic). This point is particularly important because finite axiomatization allows us to *predict* beyond what has already been described.

3 Axiomatization via Calculus

Although the above presentation provides a view about theory sufficient as a background, it is often inconvenient or impossible to deal with models in a direct way. Fortunately, the study of logic can provide useful results on the relation between semantic analysis via models and syntactic (mechanical) analysis via formulas.

Let us first introduce the notion of "calculus." Informally, a calculus is a purely syntactic manipulation of symbols to derive a new formula from a set of formulas.

Definition 12 (**Axioms**) We consider a finite set of axiom schemata, each of which consists of a decidable subset of the formulas of L.

Here we talk about axiom schemata rather than axiom instances. This is convenient because we can often consolidate infinitely many axioms as a single axiom scheme. For simplicity, we may call axiom schemata as axiom. We have seen an axiom of group theory.

Definition 13 (**Rules of inference**) A rule of inference is a finite, decidable relation between a set of formulas and a formula.

A typical rule of inference is ‘modus ponens’, i.e., from “if p, then q” and “p”, to derive “q”.

Definition 14 (**Calculus**) A calculus is a combined specification of axioms and rules of inference.

Definition 15 (**Derivation**) Derivation of a formula φ from a set of formulas Φ in a calculus is specified by axioms and rules of inference, and is written as $\Phi \vdash \varphi$.

Definition 16 (**Soundness**) A calculus for a logic is sound if $\Phi \vdash \varphi$ implies $\Phi \models \varphi$.

Definition 17 (**Completeness**) A calculus for a logic is complete if $\Phi \models \varphi$ implies $\Phi \vdash \varphi$.

A sound and complete calculus of a logic guarantees that mechanical operations in the calculus is semantically valid. In practice, it is often considered sufficient to use a sound calculus (i.e., mechanical operations are always correct, but there may not be a proof for some true formula).

Now, let us return to the topic of axiomatization. An axiomatizable theory can be completely specified by a sound and complete calculus. A sound calculus can describe an axiomatizable theory, but the specification may not be complete. Thus, the ability of a mechanical specification (axiomatization) of a theory via calculus depends on the properties associated with the calculus.

Bibliography

Heinz-Dieter Ebbinghaus, Jèorg Flum, and Wolfgang Thomas. 1984. *Mathematical logic*. Undergraduate texts in mathematics. Springer-Verlag, New York.

Herbert B. Enderton. 2001. *A mathematical introduction to logic*. Harcourt, San Diego, CA, 2nd edition.

Jacques Loeckx, Hans-Dieter Ehrich, and Markus Wolf. 1996. *Specification of abstract data types*. Wiley, Chichester, UK.