CHAPTER 10

# The Completeness Theorem



## 10.1 Introduction

The completeness theorem is one of the most fundamental results about logic. It comes in two formulations, the equivalence of which we'll prove. In its first formulation it says something fundamental about the relationship between semantic consequence and our proof system: if a sentence $A$ follows from some sentences $\Gamma$, then there is also a derivation that establishes $\Gamma \vdash A$. Thus, the proof system is as strong as it can possibly be without proving things that don't actually follow.

In its second formulation, it can be stated as a model existence result: every consistent set of sentences is satisfiable. Consistency is a proof-theoretic notion: it says that our proof system is unable to produce certain derivations. But who's to say that just because there are no derivations of a certain sort from $\Gamma$, it's guaranteed that there is a structure $M$? Before the completeness theorem was first proved—in fact before we had the proof systems we now do—the great German mathematician David Hilbert held the view that consistency of mathematical theories guarantees the existence of the objects they are about. He put it as follows in a letter to Gottlob Frege:

> If the arbitrarily given axioms do not contradict one another with all their consequences, then they are true and the things defined by the axioms exist. This is for me the criterion of truth and existence.

Frege vehemently disagreed. The second formulation of the completeness theorem shows that Hilbert was right in at least the sense that if the axioms are consistent, then *some* structure exists that makes them all true.

These aren't the only reasons the completeness theorem—or rather, its proof—is important. It has a number of important consequences, some of which we'll discuss separately. For instance, since any derivation that shows $\Gamma \vdash A$ is finite and so can only use finitely many of the sentences in $\Gamma$, it follows by the completeness theorem that if $A$ is a consequence of $\Gamma$, it is already a consequence of a finite subset of $\Gamma$. This is called *compactness.* Equivalently, if every finite subset of $\Gamma$ is consistent, then $\Gamma$ itself must be consistent.

Although the compactness theorem follows from the completeness theorem via the detour through derivations, it is also possible to use the *the proof of* the completeness theorem to establish it directly. For what the proof does is take a set of sentences with a certain property—consistency—and constructs a structure out of this set that has certain properties (in this case, that it satisfies the set). Almost the very same construction can be used to directly establish compactness, by starting from "finitely satisfiable" sets of sentences instead of consistent ones. The construction also yields other consequences, e.g., that any satisfiable set of sentences has a finite or countably infinite model. (This result is called the Löwenheim-Skolem theorem.) In general, the construction of structures from sets of sentences is used often in logic, and sometimes even in philosophy.

## 10.2 Outline of the Proof

The proof of the completeness theorem is a bit complex, and upon first reading it, it is easy to get lost. So let us outline the proof. The first step is a shift of perspective, that allows us to see a route to a proof. When completeness is thought of as "whenever $\Gamma \vDash A$ then $\Gamma \vdash A$," it may be hard to even come up with an idea: for to show that $\Gamma \vdash A$ we have to find a derivation, and it does not look like the hypothesis that $\Gamma \vDash A$ helps us for this in any way. For some proof systems it is possible to directly construct a derivation, but we will take a slightly different approach. The shift in perspective required is this: completeness can also be formulated as: "if $\Gamma$ is consistent, it is satisfiable." Perhaps we can use the information in $\Gamma$ together with the hypothesis that it is consistent to construct a structure that satisfies every sentence in $\Gamma$. After all, we know what kind of structure we are looking for: one that is as $\Gamma$ describes it!

If $\Gamma$ contains only atomic sentences, it is easy to construct a model for it. Suppose the atomic sentences are all of the form $P(a_1, \dots, a_n)$ where the $a_i$ are constant symbols. All we have to do is come up with a domain $|M|$ and an assignment for $P$ so that $M \vDash P(a_1, \dots, a_n)$. But that's not very hard: put $|M| = \mathbb{N}$, $c_i^M = i$, and for every $P(a_1, \dots, a_n) \in \Gamma$, put the tuple $\langle k_1, \dots, k_n \rangle$ into $P^M$, where $k_i$ is the index of the constant symbol $a_i$ (i.e., $a_i \equiv c_{k_i}$).

Now suppose $\Gamma$ contains some formula $\neg B$, with $B$ atomic. We might worry that the construction of $M$ interferes with the possibility of making $\neg B$ true. But here's where the consistency of $\Gamma$ comes in: if $\neg B \in \Gamma$, then $B \notin \Gamma$, or else $\Gamma$ would be inconsistent. And if $B \notin \Gamma$, then according to our construction of $M$, $M \nvDash B$, so $M \vDash \neg B$. So far so good.

What if $\Gamma$ contains complex, non-atomic formulas? Say it contains $A \wedge B$. To make that true, we should proceed as if both $A$ and $B$ were in $\Gamma$. And if $A \vee B \in \Gamma$, then we will have to make at least one of them true, i.e., proceed as if one of them was in $\Gamma$.

This suggests the following idea: we add additional formulas to $\Gamma$ so as to (a) keep the resulting set consistent and (b) make sure that for every possible atomic sentence $A$, either $A$ is in the resulting set, or $\neg A$ is, and (c) such that, whenever $A \wedge B$ is in the set, so are both $A$ and $B$, if $A \vee B$ is in the set, at least one of $A$ or $B$ is also, etc. We keep doing this (potentially forever). Call the set of all formulas so added $\Gamma^*$. Then our construction above would provide us with a structure $\boldsymbol{M}$ for which we could prove, by induction, that it satisfies all sentences in $\Gamma^*$, and hence also all sentence in $\Gamma$ since $\Gamma \subseteq \Gamma^*$. It turns out that guaranteeing (a) and (b) is enough. A set of sentences for which (b) holds is called *complete*. So our task will be to extend the consistent set $\Gamma$ to a consistent and complete set $\Gamma^*$.

There is one wrinkle in this plan: if $\exists x\, A(x) \in \Gamma$ we would hope to be able to pick some constant symbol $c$ and add $A(c)$ in this process. But how do we know we can always do that? Perhaps we only have a few constant symbols in our language, and for each one of them we have $\neg A(c) \in \Gamma$. We can't also add $A(c)$, since this would make the set inconsistent, and we wouldn't know whether $\boldsymbol{M}$ has to make $A(c)$ or $\neg A(c)$ true. Moreover, it might happen that $\Gamma$ contains only sentences in a language that has no constant symbols at all (e.g., the language of set theory).

The solution to this problem is to simply add infinitely many constants at the beginning, plus sentences that connect them with the quantifiers in the right way. (Of course, we have to verify that this cannot introduce an inconsistency.)

Our original construction works well if we only have constant symbols in the atomic sentences. But the language might also contain function symbols. In that case, it might be tricky to find the right functions on $\mathbb{N}$ to assign to these function symbols to make everything work. So here's another trick: instead of using $i$ to interpret $c_i$, just take the set of constant symbols itself as the domain. Then $\boldsymbol{M}$ can assign every constant symbol to itself: $c_i^{\boldsymbol{M}} = c_i$. But why not go all the way: let $|\boldsymbol{M}|$ be all *terms* of the language! If we do this, there is an obvious assignment of functions (that take terms as arguments and have terms as values) to function symbols: we assign to the function symbol $f_i^n$ the function which, given $n$ terms $t_1, \dots, t_n$ as input, produces the term $f_i^n(t_1, \dots, t_n)$ as value.

The last piece of the puzzle is what to do with =. The predicate symbol = has a fixed interpretation: $M \vDash t = t'$ iff $\mathrm{Val}^M(t) = \mathrm{Val}^M(t')$. Now if we set things up so that the value of a term $t$ is $t$ itself, then this structure will make *no* sentence of the form $t = t'$ true unless $t$ and $t'$ are one and the same term. And of course this is a problem, since basically every interesting theory in a language with function symbols will have as theorems sentences $t = t'$ where $t$ and $t'$ are not the same term (e.g., in theories of arithmetic: $(0 + 0) = 0$). To solve this problem, we change the domain of $M$: instead of using terms as the objects in $|M|$, we use sets of terms, and each set is so that it contains all those terms which the sentences in $\Gamma$ require to be equal. So, e.g., if $\Gamma$ is a theory of arithmetic, one of these sets will contain: $0$, $(0 + 0)$, $(0 \times 0)$, etc. This will be the set we assign to $0$, and it will turn out that this set is also the value of all the terms in it, e.g., also of $(0 + 0)$. Therefore, the sentence $(0 + 0) = 0$ will be true in this revised structure.

So here's what we'll do. First we investigate the properties of complete consistent sets, in particular we prove that a complete consistent set contains $A \wedge B$ iff it contains both $A$ and $B$, $A \vee B$ iff it contains at least one of them, etc. (Proposition 10.2). Then we define and investigate "saturated" sets of sentences. A saturated set is one which contains conditionals that link each quantified sentence to instances of it (Definition 10.5). We show that any consistent set $\Gamma$ can always be extended to a saturated set $\Gamma'$ (Lemma 10.6). If a set is consistent, saturated, and complete it also has the property that it contains $\exists x\, A(x)$ iff it contains $A(t)$ for some closed term $t$ and $\forall x\, A(x)$ iff it contains $A(t)$ for all closed terms $t$ (Proposition 10.7). We'll then take the saturated consistent set $\Gamma'$ and show that it can be extended to a saturated, consistent, and complete set $\Gamma^*$ (Lemma 10.8). This set $\Gamma^*$ is what we'll use to define our term model $M(\Gamma^*)$. The term model has the set of closed terms as its domain, and the interpretation of its predicate symbols is given by the atomic sentences in $\Gamma^*$ (Definition 10.9). We'll use the properties of saturated, complete consistent sets to show that indeed $M(\Gamma^*) \vDash A$ iff $A \in \Gamma^*$ (Lemma 10.11), and thus in particular, $M(\Gamma^*) \vDash \Gamma$. Finally, we'll consider how to define a term model if $\Gamma$ contains $=$ as well (Definition 10.15) and show that it satisfies $\Gamma^*$ (Lemma 10.17).

## 10.3 Complete Consistent Sets of Sentences

**Definition 10.1 (Complete set).** A set $\Gamma$ of sentences is *complete* iff for any sentence $A$, either $A \in \Gamma$ or $\neg A \in \Gamma$.

Complete sets of sentences leave no questions unanswered. For any sentence $A$, $\Gamma$ "says" if $A$ is true or false. The importance of complete sets extends beyond the proof of the completeness theorem. A theory which is complete and axiomatizable, for instance, is always decidable.

Complete consistent sets are important in the completeness proof since we can guarantee that every consistent set of sentences $\Gamma$ is contained in a complete consistent set $\Gamma^*$. A complete consistent set contains, for each sentence $A$, either $A$ or its negation $\neg A$, but not both. This is true in particular for atomic sentences, so from a complete consistent set in a language suitably expanded by constant symbols, we can construct a structure where the interpretation of predicate symbols is defined according to which atomic sentences are in $\Gamma^*$. This structure can then be shown to make all sentences in $\Gamma^*$ (and hence also all those in $\Gamma$) true. The proof of this latter fact requires that $\neg A \in \Gamma^*$ iff $A \notin \Gamma^*$, $(A \lor B) \in \Gamma^*$ iff $A \in \Gamma^*$ or $B \in \Gamma^*$, etc.

In what follows, we will often tacitly use the properties of reflexivity, monotonicity, and transitivity of $\vdash$ (see sections 8.8 and 9.7).

**Proposition 10.2.** *Suppose $\Gamma$ is complete and consistent. Then:*

1. *If $\Gamma \vdash A$, then $A \in \Gamma$.*

2. *$A \wedge B \in \Gamma$ iff both $A \in \Gamma$ and $B \in \Gamma$.*

3. *$A \vee B \in \Gamma$ iff either $A \in \Gamma$ or $B \in \Gamma$.*

4. *$A \rightarrow B \in \Gamma$ iff either $A \notin \Gamma$ or $B \in \Gamma$.*

*Proof.* Let us suppose for all of the following that $\Gamma$ is complete and consistent.

1. If $\Gamma \vdash A$, then $A \in \Gamma$.

   Suppose that $\Gamma \vdash A$. Suppose to the contrary that $A \notin \Gamma$. Since $\Gamma$ is complete, $\neg A \in \Gamma$. By Propositions 8.20 and 9.20, $\Gamma$ is inconsistent. This contradicts the assumption that $\Gamma$ is consistent. Hence, it cannot be the case that $A \notin \Gamma$, so $A \in \Gamma$.

2. Exercise.

3. First we show that if $A \vee B \in \Gamma$, then either $A \in \Gamma$ or $B \in \Gamma$. Suppose $A \vee B \in \Gamma$ but $A \notin \Gamma$ and $B \notin \Gamma$. Since $\Gamma$ is complete, $\neg A \in \Gamma$ and $\neg B \in \Gamma$. By Propositions 8.23 and 9.23, item (1), $\Gamma$ is inconsistent, a contradiction. Hence, either $A \in \Gamma$ or $B \in \Gamma$.

   For the reverse direction, suppose that $A \in \Gamma$ or $B \in \Gamma$. By Propositions 8.23 and 9.23, item (2), $\Gamma \vdash A \vee B$. By (1), $A \vee B \in \Gamma$, as required.

4. Exercise. □

## 10.4 Henkin Expansion

Part of the challenge in proving the completeness theorem is that the model we construct from a complete consistent set $\Gamma$ must make all the quantified formulas in $\Gamma$ true. In order to guarantee this, we use a trick due to Leon Henkin. In essence, the trick consists in expanding the language by infinitely many constant symbols and adding, for each formula with one free variable $A(x)$ a formula of the form $\exists x\, A(x) \rightarrow A(c)$, where $c$ is one of the new constant symbols. When we construct the structure satisfying $\Gamma$, this will guarantee that each true existential sentence has a witness among the new constants.

**Proposition 10.3.** *If $\Gamma$ is consistent in $\mathcal{L}$ and $\mathcal{L}'$ is obtained from $\mathcal{L}$ by adding a countably infinite set of new constant symbols $d_0$, $d_1$, …, then $\Gamma$ is consistent in $\mathcal{L}'$.*

**Definition 10.4 (Saturated set).** A set $\Gamma$ of formulas of a language $\mathcal{L}$ is *saturated* iff for each formula $A(x) \in \mathrm{Frm}(\mathcal{L})$ with one free variable $x$ there is a constant symbol $c \in \mathcal{L}$ such that $\exists x\, A(x) \rightarrow A(c) \in \Gamma$.

The following definition will be used in the proof of the next theorem.

**Definition 10.5.** Let $\mathcal{L}'$ be as in Proposition 10.3. Fix an enumeration $A_0(x_0)$, $A_1(x_1)$, … of all formulas $A_i(x_i)$ of $\mathcal{L}'$ in which one variable $(x_i)$ occurs free. We define the sentences $D_n$ by induction on $n$.

Let $c_0$ be the first constant symbol among the $d_i$ we added to $\mathcal{L}$ which does not occur in $A_0(x_0)$. Assuming that $D_0, \ldots, D_{n-1}$ have already been defined, let $c_n$ be the first among the new constant symbols $d_i$ that occurs neither in $D_0, \ldots, D_{n-1}$ nor in $A_n(x_n)$.

Now let $D_n$ be the formula $\exists x_n\, A_n(x_n) \rightarrow A_n(c_n)$.

**Lemma 10.6.** *Every consistent set $\Gamma$ can be extended to a saturated consistent set $\Gamma'$.*

*Proof.* Given a consistent set of sentences $\Gamma$ in a language $\mathcal{L}$, expand the language by adding a countably infinite set of new constant symbols to form $\mathcal{L}'$. By Proposition 10.3, $\Gamma$ is still consistent in the richer language. Further, let $D_i$ be as in Definition 10.5. Let

$$\Gamma_0 = \Gamma$$
$$\Gamma_{n+1} = \Gamma_n \cup \{D_n\}$$

i.e., $\Gamma_{n+1} = \Gamma \cup \{D_0, \dots, D_n\}$, and let $\Gamma' = \bigcup_n \Gamma_n$. $\Gamma'$ is clearly saturated.

If $\Gamma'$ were inconsistent, then for some $n$, $\Gamma_n$ would be inconsistent (Exercise: explain why). So to show that $\Gamma'$ is consistent it suffices to show, by induction on $n$, that each set $\Gamma_n$ is consistent.

The induction basis is simply the claim that $\Gamma_0 = \Gamma$ is consistent, which is the hypothesis of the theorem. For the induction step, suppose that $\Gamma_n$ is consistent but $\Gamma_{n+1} = \Gamma_n \cup \{D_n\}$ is inconsistent. Recall that $D_n$ is $\exists x_n\, A_n(x_n) \rightarrow A_n(c_n)$, where $A_n(x_n)$ is a formula of $\mathcal{L}'$ with only the variable $x_n$ free. By the way we've chosen the $c_n$ (see Definition 10.5), $c_n$ does not occur in $A_n(x_n)$ nor in $\Gamma_n$.

If $\Gamma_n \cup \{D_n\}$ is inconsistent, then $\Gamma_n \vdash \neg D_n$, and hence both of the following hold:

$$\Gamma_n \vdash \exists x_n\, A_n(x_n) \qquad \Gamma_n \vdash \neg A_n(c_n)$$

Since $c_n$ does not occur in $\Gamma_n$ or in $A_n(x_n)$, Theorems 8.25 and 9.25 applies. From $\Gamma_n \vdash \neg A_n(c_n)$, we obtain $\Gamma_n \vdash \forall x_n\, \neg A_n(x_n)$. Thus we have that both $\Gamma_n \vdash \exists x_n\, A_n(x_n)$ and $\Gamma_n \vdash \forall x_n\, \neg A_n(x_n)$, so $\Gamma_n$ itself is inconsistent. (Note that $\forall x_n\, \neg A_n(x_n) \vdash \neg \exists x_n\, A_n(x_n)$.) Contradiction: $\Gamma_n$ was supposed to be consistent. Hence $\Gamma_n \cup \{D_n\}$ is consistent. □

We'll now show that *complete*, consistent sets which are saturated have the property that it contains a universally quantified sentence iff it contains all its instances and it contains an existentially quantified sentence iff it contains at least one instance. We'll use this to show that the structure we'll generate from a complete, consistent, saturated set makes all its quantified sentences true.

**Proposition 10.7.** *Suppose $\Gamma$ is complete, consistent, and saturated.*

1. $\exists x\, A(x) \in \Gamma$ *iff* $A(t) \in \Gamma$ *for at least one closed term* $t$.
2. $\forall x\, A(x) \in \Gamma$ *iff* $A(t) \in \Gamma$ *for all closed terms* $t$.

*Proof.*

1. First suppose that $\exists x\, A(x) \in \Gamma$. Because $\Gamma$ is saturated, $(\exists x\, A(x) \to A(c)) \in \Gamma$ for some constant symbol $c$. By Propositions 8.24 and 9.24, item (1), and Proposition 10.2(1), $A(c) \in \Gamma$.

   For the other direction, saturation is not necessary: Suppose $A(t) \in \Gamma$. Then $\Gamma \vdash \exists x\, A(x)$ by Propositions 8.26 and 9.26, item (1). By Proposition 10.2(1), $\exists x\, A(x) \in \Gamma$.

2. Exercise. □



## 10.5 Lindenbaum's Lemma

We now prove a lemma that shows that any consistent set of sentences is contained in some set of sentences which is not just consistent, but also complete. The proof works by adding one sentence at a time, guaranteeing at each step that the set remains consistent. We do this so that for every $A$, either $A$ or $\neg A$ gets added at some stage. The union of all stages in that construction then contains either $A$ or its negation $\neg A$ and is thus complete. It is also consistent, since we made sure at each stage not to introduce an inconsistency.

**Lemma 10.8 (Lindenbaum's Lemma).** *Every consistent set $\Gamma$ in a language $\mathcal{L}$ can be extended to a complete and consistent set $\Gamma^*$.*

*Proof.* Let $\Gamma$ be consistent. Let $A_0$, $A_1$, … be an enumeration of all the sentences of $\mathcal{L}$. Define $\Gamma_0 = \Gamma$, and

$$\Gamma_{n+1} = \begin{cases} \Gamma_n \cup \{A_n\} & \text{if } \Gamma_n \cup \{A_n\} \text{ is consistent;} \\ \Gamma_n \cup \{\neg A_n\} & \text{otherwise.} \end{cases}$$

Let $\Gamma^* = \bigcup_{n \geq 0} \Gamma_n$.

Each $\Gamma_n$ is consistent: $\Gamma_0$ is consistent by definition. If $\Gamma_{n+1} = \Gamma_n \cup \{A_n\}$, this is because the latter is consistent. If it isn't, $\Gamma_{n+1} = \Gamma_n \cup \{\neg A_n\}$. We have to verify that $\Gamma_n \cup \{\neg A_n\}$ is consistent. Suppose it's not. Then *both* $\Gamma_n \cup \{A_n\}$ and $\Gamma_n \cup \{\neg A_n\}$ are inconsistent. This means that $\Gamma_n$ would be inconsistent by Propositions 8.20 and 9.20, contrary to the induction hypothesis.

For every $n$ and every $i < n$, $\Gamma_i \subseteq \Gamma_n$. This follows by a simple induction on $n$. For $n = 0$, there are no $i < 0$, so the claim holds automatically. For the inductive step, suppose it is true for $n$. We have $\Gamma_{n+1} = \Gamma_n \cup \{A_n\}$ or $= \Gamma_n \cup \{\neg A_n\}$ by construction. So $\Gamma_n \subseteq \Gamma_{n+1}$. If $i < n$, then $\Gamma_i \subseteq \Gamma_n$ by inductive hypothesis, and so $\subseteq \Gamma_{n+1}$ by transitivity of $\subseteq$.

From this it follows that every finite subset of $\Gamma^*$ is a subset of $\Gamma_n$ for some $n$, since each $B \in \Gamma^*$ not already in $\Gamma_0$ is added at some stage $i$. If $n$ is the last one of these, then all $B$ in the finite subset are in $\Gamma_n$. So, every finite subset of $\Gamma^*$ is consistent. By Propositions 8.17 and 9.17, $\Gamma^*$ is consistent.

Every sentence of $\mathrm{Frm}(\mathcal{L})$ appears on the list used to define $\Gamma^*$. If $A_n \notin \Gamma^*$, then that is because $\Gamma_n \cup \{A_n\}$ was inconsistent. But then $\neg A_n \in \Gamma^*$, so $\Gamma^*$ is complete. □

## 10.6 Construction of a Model

Right now we are not concerned about $=$, i.e., we only want to show that a consistent set $\Gamma$ of sentences not containing $=$ is satisfiable. We first extend $\Gamma$ to a consistent, complete, and saturated set $\Gamma^*$. In this case, the definition of a model $M(\Gamma^*)$ is simple: We take the set of closed terms of $\mathcal{L}'$ as the domain. We assign every constant symbol to itself, and make sure that more generally, for every closed term $t$, $\mathrm{Val}^{M(\Gamma^*)}(t) = t$. The predicate symbols are assigned extensions in such a way that an atomic sentence is true in $M(\Gamma^*)$ iff it is in $\Gamma^*$. This will obviously make all the atomic sentences in $\Gamma^*$ true in $M(\Gamma^*)$. The rest are true provided the $\Gamma^*$ we start with is consistent, complete, and saturated.

**Definition 10.9 (Term model).** Let $\Gamma^*$ be a complete and consistent, saturated set of sentences in a language $\mathcal{L}$. The *term model* $M(\Gamma^*)$ of $\Gamma^*$ is the structure defined as follows:

1. The domain $|M(\Gamma^*)|$ is the set of all closed terms of $\mathcal{L}$.
2. The interpretation of a constant symbol $c$ is $c$ itself: $c^{M(\Gamma^*)} = c$.
3. The function symbol $f$ is assigned the function which, given as arguments the closed terms $t_1, \dots, t_n$, has as value the closed term $f(t_1, \dots, t_n)$:
$$f^{M(\Gamma^*)}(t_1, \dots, t_n) = f(t_1, \dots, t_n)$$
4. If $R$ is an $n$-place predicate symbol, then
$$\langle t_1, \dots, t_n \rangle \in R^{M(\Gamma^*)} \text{ iff } R(t_1, \dots, t_n) \in \Gamma^*.$$

A structure $M$ may make an existentially quantified sentence $\exists x\, A(x)$ true without there being an instance $A(t)$ that it makes true. A structure $M$ may make all instances $A(t)$ of a universally quantified sentence $\forall x\, A(x)$ true, without making $\forall x\, A(x)$ true. This is because in general not every element of $|M|$ is the value of a closed term ($M$ may not be covered). This is the reason the satisfaction relation is defined via variable assignments. However, for our term model $M(\Gamma^*)$ this wouldn't be necessary—because it is covered. This is the content of the next result.

**Proposition 10.10.** *Let $M(\Gamma^*)$ be the term model of Definition 10.9.*

1. $M(\Gamma^*) \vDash \exists x\, A(x)$ *iff* $M \vDash A(t)$ *for at least one term* $t$.
2. $M(\Gamma^*) \vDash \forall x\, A(x)$ *iff* $M \vDash A(t)$ *for all terms* $t$.

*Proof.* 1. By Proposition 5.42, $M(\Gamma^*) \vDash \exists x\, A(x)$ iff for at least one variable assignment $s$, $M(\Gamma^*), s \vDash A(x)$. As $|M(\Gamma^*)|$ consists of the closed terms of $\mathcal{L}$, this is the case iff there is at least one closed term $t$ such that $s(x) = t$ and $M(\Gamma^*), s \vDash A(x)$. By Proposition 5.46, $M(\Gamma^*), s \vDash A(x)$ iff $M(\Gamma^*), s \vDash A(t)$, where $s(x) = t$. By Proposition 5.41, $M(\Gamma^*), s \vDash A(t)$ iff $M(\Gamma^*) \vDash A(t)$, since $A(t)$ is a sentence.

2. Exercise. □



**Lemma 10.11 (Truth Lemma).** *Suppose $A$ does not contain $=$. Then $M(\Gamma^*) \vDash A$ iff $A \in \Gamma^*$.*

*Proof.* We prove both directions simultaneously, and by induction on $A$.

1. $A \equiv \bot$: $M(\Gamma^*) \nvDash \bot$ by definition of satisfaction. On the other hand, $\bot \notin \Gamma^*$ since $\Gamma^*$ is consistent.

2. $A \equiv R(t_1, \dots, t_n)$: $M(\Gamma^*) \vDash R(t_1, \dots, t_n)$ iff $\langle t_1, \dots, t_n \rangle \in R^{M(\Gamma^*)}$ (by the definition of satisfaction) iff $R(t_1, \dots, t_n) \in \Gamma^*$ (by the construction of $M(\Gamma^*)$).

3. $A \equiv \neg B$: $M(\Gamma^*) \vDash A$ iff $M(\Gamma^*) \nvDash B$ (by definition of satisfaction). By induction hypothesis, $M(\Gamma^*) \nvDash B$ iff $B \notin \Gamma^*$. Since $\Gamma^*$ is consistent and complete, $B \notin \Gamma^*$ iff $\neg B \in \Gamma^*$.

4. $A \equiv B \wedge C$: exercise.

5. $A \equiv B \vee C$: $M(\Gamma^*) \vDash A$ iff $M(\Gamma^*) \vDash B$ or $M(\Gamma^*) \vDash C$ (by definition of satisfaction) iff $B \in \Gamma^*$ or $C \in \Gamma^*$ (by induction hypothesis). This is the case iff $(B \vee C) \in \Gamma^*$ (by Proposition 10.2(3)).

6. $A \equiv B \rightarrow C$: exercise.

7. $A \equiv \forall x\, B(x)$: exercise.

8. $A \equiv \exists x\, B(x)$: $\mathfrak{M}(\Gamma^*) \vDash A$ iff $\mathfrak{M}(\Gamma^*) \vDash B(t)$ for at least one term $t$ (Proposition 10.10). By induction hypothesis, this is the case iff $B(t) \in \Gamma^*$ for at least one term $t$. By Proposition 10.7, this in turn is the case iff $\exists x\, B(x) \in \Gamma^*$. □

## 10.7 Identity

The construction of the term model given in the preceding section is enough to establish completeness for first-order logic for sets $\Gamma$ that do not contain $=$. The term model satisfies every $A \in \Gamma^*$ which does not contain $=$ (and hence all $A \in \Gamma$). It does not work, however, if $=$ is present. The reason is that $\Gamma^*$ then may contain a sentence $t = t'$, but in the term model the value of any term is that term itself. Hence, if $t$ and $t'$ are different terms, their values in the term model—i.e., $t$ and $t'$, respectively—are different, and so $t = t'$ is false. We can fix this, however, using a construction known as "factoring."

**Definition 10.12.** Let $\Gamma^*$ be a consistent and complete set of sentences in $\mathcal{L}$. We define the relation $\approx$ on the set of closed terms of $\mathcal{L}$ by

$$t \approx t' \quad \text{iff} \quad t = t' \in \Gamma^*$$

**Proposition 10.13.** *The relation $\approx$ has the following properties:*

1. *$\approx$ is reflexive.*

2. *$\approx$ is symmetric.*

3. *$\approx$ is transitive.*