

Proof of the Completeness Theorem

We prove the Completeness Theorem.

1. The Completeness Theorem

1.1. Statement

Recall the statement that we studied carefully in the chapter “Completeness Theorem” but have not yet proved.

Theorem 1 (Completeness Theorem).

If $\Gamma \models \mathcal{Q}$, then $\Gamma \vdash \mathcal{Q}$.

In other words, everything that is true is provable.

1.2. To prove: “If $\mathcal{T}$ is consistent, then $\mathcal{T}$ has a model.”

By the closure theorem from the previous chapter, the proof of the Completeness Theorem in the general setting reduces to the setting of closed formulas. Let $\mathcal{T}$ be a theory and $\mathcal{Q}$ a closed formula.

Theorem 2 (Theorem A).

If $\mathcal{T} \models \mathcal{Q}$, then $\mathcal{T} \vdash \mathcal{Q}$.

We will therefore study the case of theories. Theorem A will be obtained from the following equivalent Theorem B:

Theorem 3 (Theorem B).

If $\mathcal{T}$ is consistent, then $\mathcal{T}$ has a model.

Thus, we will have the chain:

$$\text{Theorem B} \implies \text{Theorem A} \implies \text{Completeness Theorem}$$

Theorem B is therefore what we will prove in this chapter, after explaining how to deduce Theorem A from it.

1.3. Review of terminology

- **Theory**: a set $\mathcal{T}$ of closed formulas.
- $\mathcal{T}$ is **inconsistent** if $\mathcal{T} \vdash \perp$; otherwise, $\mathcal{T}$ is **consistent**.
- $\mathcal{T}$ has a **model** if there is a structure S such that $\models_S \mathcal{T}$; that is, a structure in which all the formulas of $\mathcal{T}$ become true.

In the previous chapter, we saw the following proposition:

Proposition 1.

$\mathcal{T} \vdash Q$ iff $\mathcal{T} \cup \{\neg Q\}$ is inconsistent.

Here is how to prove Theorem A from Theorem B (we saw the converse in the previous chapter).

Proof. We will prove Theorem A in its contrapositive form: if $\mathcal{T} \not\vdash Q$, then $\mathcal{T} \not\models Q$.

We will use negations of operators sparingly: $\mathcal{T} \not\vdash Q$ means $\text{not}(\mathcal{T} \vdash Q)$ and $\mathcal{T} \not\models Q$ means $\text{not}(\mathcal{T} \models Q)$.

Assumption: $\text{not}(\mathcal{T} \vdash Q)$. Goal: $\text{not}(\mathcal{T} \models Q)$.

$\text{not}(\mathcal{T} \vdash Q)$

therefore $\mathcal{T} \cup \{\neg Q\}$ is consistent by Proposition 1 above

therefore $\mathcal{T} \cup \{\neg Q\}$ has a model by Theorem B

therefore $\models_{\mathcal{S}} \mathcal{T} \cup \{\neg Q\}$ for some structure $\mathcal{S}$

therefore $\models_{\mathcal{S}} \mathcal{T}$ and $\models_{\mathcal{S}} \neg Q$ since $\mathcal{T}$ and $\neg Q$ are both true in $\mathcal{S}$

therefore $\text{not}(\mathcal{T} \models Q)$ since $\mathcal{T}$ is true and Q is false in $\mathcal{S}$

□

2. Complete theory

2.1. Definition

Definition.

A theory $\mathcal{T}$ is said to be *complete* if:

- $\mathcal{T}$ is consistent,
- and, for every closed formula Q , either $\mathcal{T} \vdash Q$ or $\mathcal{T} \vdash \neg Q$.

Thus, in a complete theory, for every closed formula Q , either it or its negation can be proved, but not both at the same time. The name “complete” is therefore well chosen.

2.2. Satisfaction

Here is an important consequence when working with a complete theory: formulas are true or false independently of the structure.

Proposition 2.

Let $\mathcal{T}$ be a complete theory and Q a closed formula. Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be two models of $\mathcal{T}$. Then:

$$\models_{\mathcal{S}_1} Q \quad \text{iff} \quad \models_{\mathcal{S}_2} Q.$$

Proof. Since $\mathcal{T}$ is complete, consider the two cases: either $\mathcal{T} \vdash Q$ or $\mathcal{T} \vdash \neg Q$.

- If $\mathcal{T} \vdash Q$, then $\mathcal{T} \models Q$ by the Soundness Theorem, so $\models_{\mathcal{S}_1} Q$ and $\models_{\mathcal{S}_2} Q$ because $\mathcal{S}_1$ and $\mathcal{S}_2$ are models of $\mathcal{T}$; that is, Q is true in both $\mathcal{S}_1$ and $\mathcal{S}_2$.
- If $\mathcal{T} \vdash \neg Q$, the same reasoning gives $\models_{\mathcal{S}_1} \neg Q$ and $\models_{\mathcal{S}_2} \neg Q$; that is, Q is false in both $\mathcal{S}_1$ and $\mathcal{S}_2$.

□

2.3. A lemma

In a complete theory, there are no traps. We will prove the following statements, which may seem obvious but require this notion.

Lemma 1.

Let $\mathcal{T}$ be a complete theory.

1. $\mathcal{T} \vdash \neg \mathcal{P}$ iff $\text{not}(\mathcal{T} \vdash \mathcal{P})$
2. $\mathcal{T} \vdash \mathcal{P} \wedge \mathcal{Q}$ iff $(\mathcal{T} \vdash \mathcal{P} \text{ and } \mathcal{T} \vdash \mathcal{Q})$
3. $\mathcal{T} \vdash \mathcal{P} \vee \mathcal{Q}$ iff $(\mathcal{T} \vdash \mathcal{P} \text{ or } \mathcal{T} \vdash \mathcal{Q})$

Proof.

1.
 - Suppose $\mathcal{T} \vdash \neg \mathcal{P}$. By contradiction, if $\mathcal{T} \vdash \mathcal{P}$, then $\mathcal{T} \vdash (\mathcal{P} \wedge \neg \mathcal{P})$, so $\mathcal{T} \vdash \perp$. This contradicts the consistency of $\mathcal{T}$ (since $\mathcal{T}$ is complete). Therefore, $\text{not}(\mathcal{T} \vdash \mathcal{P})$.
 - Suppose $\text{not}(\mathcal{T} \vdash \mathcal{P})$. Since $\mathcal{T}$ is complete ($\mathcal{T} \vdash \mathcal{P}$ or $\mathcal{T} \vdash \neg \mathcal{P}$), we therefore have $\mathcal{T} \vdash \neg \mathcal{P}$.
2.
 - If $\mathcal{T} \vdash \mathcal{P} \wedge \mathcal{Q}$, then $\mathcal{T} \vdash \mathcal{P}$ and $\mathcal{T} \vdash \mathcal{Q}$ because $\mathcal{P} \wedge \mathcal{Q} \vdash \mathcal{P}$ and $\mathcal{P} \wedge \mathcal{Q} \vdash \mathcal{Q}$.
 - If $\mathcal{T} \vdash \mathcal{P}$ and $\mathcal{T} \vdash \mathcal{Q}$, then $\mathcal{T} \vdash \mathcal{P} \wedge \mathcal{Q}$, because $\mathcal{P}, \mathcal{Q} \vdash \mathcal{P} \wedge \mathcal{Q}$.

Note: this point does not use the assumption that $\mathcal{T}$ is complete.

3.
 - If $(\mathcal{T} \vdash \mathcal{P} \text{ or } \mathcal{T} \vdash \mathcal{Q})$, then $\mathcal{T} \vdash \mathcal{P} \vee \mathcal{Q}$ because $\mathcal{P} \vdash \mathcal{P} \vee \mathcal{Q}$ and $\mathcal{Q} \vdash \mathcal{P} \vee \mathcal{Q}$. (There is no need for $\mathcal{T}$ to be complete.)
 - Suppose $\mathcal{T} \vdash \mathcal{P} \vee \mathcal{Q}$. Suppose, for contradiction, that $\mathcal{T} \not\vdash \mathcal{P}$ and $\mathcal{T} \not\vdash \mathcal{Q}$. Since $\mathcal{T} \not\vdash \mathcal{P}$, we have $\mathcal{T} \vdash \neg \mathcal{P}$ because $\mathcal{T}$ is complete (by point 1); similarly, since $\mathcal{T} \not\vdash \mathcal{Q}$, we have $\mathcal{T} \vdash \neg \mathcal{Q}$. Thus, by point 2, $\mathcal{T} \vdash \neg \mathcal{P} \wedge \neg \mathcal{Q}$, so $\mathcal{T} \vdash \neg(\mathcal{P} \vee \mathcal{Q})$. But then we have both $\mathcal{T} \vdash \mathcal{P} \vee \mathcal{Q}$ (by assumption) and $\mathcal{T} \vdash \neg(\mathcal{P} \vee \mathcal{Q})$, so $\mathcal{T}$ is inconsistent. This is a contradiction because $\mathcal{T}$ is complete and hence consistent.

□

3. Constructing a model from a complete theory

We begin the proof of the Completeness Theorem in the form of Theorem B: “if $\mathcal{T}$ is consistent, then $\mathcal{T}$ has a model”. We begin this proof somewhat from the end. We will start from a complete theory with additional assumptions and construct a model. We will explain later how to obtain these additional assumptions. There are several proofs of the Completeness Theorem; here we adopt the one based on Henkin witnesses.

3.1. Strengthened assumptions

Let $\mathcal{L}$ be a language and $\mathcal{T}$ a theory. We assume the following properties of $\mathcal{T}$:

- (i) $\mathcal{T}$ is consistent.
- (ii) $\mathcal{T}$ is complete.
- (iii) $\mathcal{T}$ has Henkin witnesses.

Note: (ii) $\implies$ (i), but we emphasize that in the end we would like (i) to be the only assumption.

Definition.

$\mathcal{T}$ has **Henkin witnesses** if, for every formula $\mathcal{P}$ in $\mathcal{L}$ with a single free variable x , there is a constant $c_{\mathcal{P}}$ in $\mathcal{L}$ such that the following formula:

$$\exists x \mathcal{P}(x) \rightarrow \mathcal{P}(c_{\mathcal{P}})$$

belongs to $\mathcal{T}$.

Notes.

- We could also write: there exists $c \in \mathcal{L}$ such that:

$$(\exists x \mathcal{P}(x)) \rightarrow \mathcal{P}(c) \in \mathcal{T},$$

if we do not wish to emphasize the dependence of c on $\mathcal{P}$.

- With Henkin witnesses, a formula “ $\exists x \mathcal{P}(x)$ ” is no longer an abstract formula, because we can obtain a concrete realization “ $\mathcal{P}(c_{\mathcal{P}})$ ”, hence the name “witness”.

Here are two more additional assumptions, this time concerning the language $\mathcal{L}$:

- (iv) $\mathcal{L}$ is a countable language (the set of symbols in $\mathcal{L}$ is countable). This implies that the set of formulas in $\mathcal{L}$ is also countable (see the chapter “Compactness Theorem”).
- (v) $\mathcal{L}$ does not contain the equality predicate “ $=$ ”. Without this assumption, the proof would become more technical because we would have to work with equivalence classes for the relation defined by $t_1 \sim t_2$ whenever $t_1 = t_2$ holds.

Despite these simplifying assumptions, the proof remains difficult.

3.2. Constructing the model

For a theory $\mathcal{T}$ satisfying assumptions (i), (ii), and (iii) in a language $\mathcal{L}$ satisfying assumptions (iv) and (v), we will construct a model $\mathcal{S}$ of $\mathcal{T}$; that is, a structure $\mathcal{S}$ such that $\models_{\mathcal{S}} \mathcal{T}$.

Constructing the structure $\mathcal{S}$ means specifying:

- a domain E ,
- an interpretation $c^{\mathcal{S}}$ of each constant c in $\mathcal{L}$,
- an interpretation $f^{\mathcal{S}}$ of each function f in $\mathcal{L}$,
- an interpretation $P^{\mathcal{S}}$ of each predicate P in $\mathcal{L}$.

We will then have to prove that every formula $\mathcal{P}$ in $\mathcal{T}$ is true when interpreted in $\mathcal{S}$.

Here we go! The construction is more or less automatic. We define the objects so that they have the desired properties. Here is the structure $\mathcal{S}$:

- The domain E is the set of closed terms in $\mathcal{L}$. Recall that terms are formed from constants and variables, to which functions can be applied recursively. Here, the closed terms are therefore only constants and iterated applications of functions to constants. Examples of closed terms: c_1 , c_2 , $f(c_1)$, $g(c_1, c_2)$, $f(f(c_1))$, $g(c_1, f(c_2))$, etc.

Note: if equality were part of the language, E would have to be the set of equivalence classes of closed terms modulo the equivalence relation $t_1 \sim t_2$ iff $\mathcal{T} \vdash (t_1 = t_2)$.

- If c is a constant symbol in $\mathcal{L}$, then c is a closed term and hence an element of E . We simply define $c^{\mathcal{S}} = c$.

- Let f be an n -ary function symbol in $\mathcal{L}$. Defining f^S means defining an “actual” function $f^S : E^n \rightarrow E$. Let $t_1, \dots, t_n$ be elements of E . They are closed terms in $\mathcal{L}$, so $f(t_1, \dots, t_n)$ is also a closed term and hence an element of E . Set $f^S(t_1, \dots, t_n) = f(t_1, \dots, t_n)$.
- Let P be an n -place predicate in $\mathcal{L}$. Defining the interpretation P^S means defining a function $P^S : E^n \rightarrow \{T, F\}$. Here is how to assign a truth value to each element $(t_1, \dots, t_n) \in E^n$: set $P^S(t_1, \dots, t_n) = T$ if $\mathcal{T} \vdash P(t_1, \dots, t_n)$, and set $P^S(t_1, \dots, t_n) = F$ otherwise.

Thus, by construction: $\models_S P(t_1, \dots, t_n)$ iff $\mathcal{T} \vdash P(t_1, \dots, t_n)$. We will later generalize this statement to arbitrary formulas.

3.3. Satisfaction

It remains to show that our structure S is indeed a model of $\mathcal{T}$; that is, $\models_S \mathcal{T}$. The proof is based on the following result:

Lemma 2.

Let $\mathcal{Q}$ be a closed formula.

$$\mathcal{T} \vdash \mathcal{Q} \quad \text{iff} \quad \models_S \mathcal{Q}$$

Assume this lemma for a moment and let us show that S is a model of $\mathcal{T}$. Indeed, let $\mathcal{P}$ be a formula in $\mathcal{T}$. Of course, $\mathcal{T} \vdash \mathcal{P}$, so the lemma gives $\models_S \mathcal{P}$. Since this is true for every formula in $\mathcal{T}$, we indeed have $\models_S \mathcal{T}$.

Proof of the lemma. For simplicity, and without loss of generality, assume that the formula $\mathcal{Q}$ contains only the connectives $\neg$ and $\wedge$, together with the quantifier $\exists$. For the connectives, see the chapter “Which Connectives Are Useful?” for the reduction to these two connectives. A formula “ $\forall x \mathcal{P}(x)$ ” is equivalent to “ $\neg(\exists x \neg \mathcal{P}(x))$ ”, which allows us to turn every $\forall$ quantifier into an $\exists$ quantifier (with additional negations).

Let us prove the result by induction on the construction of the formula $\mathcal{Q}$.

- Case where $\mathcal{Q}$ is an atomic formula.

Then $\mathcal{Q} = P(t_1, \dots, t_n)$, where P is a predicate and $t_1, \dots, t_n$ are closed terms. By construction of S ,

$$\mathcal{T} \vdash P(t_1, \dots, t_n) \quad \text{iff} \quad \models_S P(t_1, \dots, t_n),$$

so the lemma is true in this case.

- Case where $\mathcal{Q} = \neg \mathcal{P}$.

By induction, assume the lemma is true for $\mathcal{P}$, and prove it for $\mathcal{Q} = \neg \mathcal{P}$.

$$\begin{aligned} & \mathcal{T} \vdash \mathcal{Q} \\ \text{iff} & \quad \mathcal{T} \vdash \neg \mathcal{P} \\ \text{iff} & \quad \text{not}(\mathcal{T} \vdash \mathcal{P}) \quad \text{by Lemma 1, since } \mathcal{T} \text{ is complete} \\ \text{iff} & \quad \text{not}(\models_S \mathcal{P}) \quad \text{by the induction hypothesis} \\ \text{iff} & \quad \models_S \neg \mathcal{P} \\ \text{iff} & \quad \models_S \mathcal{Q} \end{aligned}$$

The key step is the transition from the second line to the third, which uses assumption (ii), that $\mathcal{T}$ is complete, through Lemma 1.

- Case where $Q = \mathcal{P}_1 \wedge \mathcal{P}_2$.

By induction, assume the lemma is true for $\mathcal{P}_1$ and $\mathcal{P}_2$.

$$\begin{aligned}
 & \mathcal{T} \vdash Q \\
 \text{iff} \quad & \mathcal{T} \vdash \mathcal{P}_1 \wedge \mathcal{P}_2 \\
 \text{iff} \quad & (\mathcal{T} \vdash \mathcal{P}_1 \text{ and } \mathcal{T} \vdash \mathcal{P}_2) \quad \text{by Lemma 1} \\
 \text{iff} \quad & (\models_{\mathcal{S}} \mathcal{P}_1 \text{ and } \models_{\mathcal{S}} \mathcal{P}_2) \quad \text{by the induction hypothesis} \\
 \text{iff} \quad & \models_{\mathcal{S}} \mathcal{P}_1 \wedge \mathcal{P}_2 \\
 \text{iff} \quad & \models_{\mathcal{S}} Q
 \end{aligned}$$

- Case where $Q = \exists x \mathcal{P}(x)$.

Assume the lemma is true for $\mathcal{P}$.

- If $\mathcal{T} \vdash Q$, that is, if $\mathcal{T} \vdash \exists x \mathcal{P}(x)$, then assumption (iii) on Henkin witnesses tells us that there is a $c_{\mathcal{P}} \in \mathcal{L}$ such that “ $(\exists x \mathcal{P}(x)) \rightarrow \mathcal{P}(c_{\mathcal{P}})$ ” is a formula in $\mathcal{T}$. Since $\mathcal{T} \vdash \exists x \mathcal{P}(x)$ and $\mathcal{T} \vdash [(\exists x \mathcal{P}(x)) \rightarrow \mathcal{P}(c_{\mathcal{P}})]$, we have $\mathcal{T} \vdash \mathcal{P}(c_{\mathcal{P}})$. Therefore, by the induction hypothesis, $\models_{\mathcal{S}} \mathcal{P}(c_{\mathcal{P}})$, and by the definition of satisfaction for $\exists$, we deduce $\models_{\mathcal{S}} \exists x \mathcal{P}(x)$; that is, $\models_{\mathcal{S}} Q$.
- If $\models_{\mathcal{S}} Q$, then $\models_{\mathcal{S}} \exists x \mathcal{P}(x)$, so there is a closed term t , whose interpretation is $t^{\mathcal{S}} \in E$, such that $\models_{\mathcal{S}} \mathcal{P}(t)$. By the induction hypothesis, $\mathcal{T} \vdash \mathcal{P}(t)$. Thus, by the $\exists$ introduction rule, $\mathcal{T} \vdash \exists x \mathcal{P}(x)$, and hence $\mathcal{T} \vdash Q$.

□

4. Constructing a complete theory

Let us explain how to make a theory complete by adding formulas.

4.1. Construction

Let $\mathcal{T}$ be a consistent theory in a language $\mathcal{L}$. We are going to construct a theory $\mathcal{T}'$, in the same language $\mathcal{L}$, such that $\mathcal{T} \subset \mathcal{T}'$ and $\mathcal{T}'$ is complete. We assume that the language $\mathcal{L}$ is countable, so the set of formulas is countable as well. Consider an enumeration $\mathcal{P}_0, \mathcal{P}_1, \dots$ of all the closed formulas of $\mathcal{L}$. We construct $\mathcal{T}'$ as the limit of $(\mathcal{T}_n)_{n \geq 0}$, defined recursively as follows:

- $\mathcal{T}_0 = \mathcal{T}$
- - $\mathcal{T}_{n+1} = \mathcal{T}_n \cup \{\mathcal{P}_n\}$ if this union is consistent,
 - $\mathcal{T}_{n+1} = \mathcal{T}_n \cup \{\neg \mathcal{P}_n\}$ otherwise.

Finally, let $\mathcal{T}' = \bigcup_{n \geq 0} \mathcal{T}_n$.

Proposition 3.

Each $\mathcal{T}_n$ is consistent, and $\mathcal{T}'$ is complete.

4.2. Proof

Proof.

- Let us prove by induction that the $\mathcal{T}_n$ are consistent. This is true for $\mathcal{T}_0 = \mathcal{T}$ by assumption. Suppose it is true for $\mathcal{T}_n$. In the first case, $\mathcal{T}_{n+1} = \mathcal{T}_n \cup \{\mathcal{P}_n\}$ is consistent by construction. In the second case, $\mathcal{T}_n \cup \{\mathcal{P}_n\}$ is not consistent, so $\mathcal{T}_n \vdash \neg \mathcal{P}_n$ (Proposition 1). If $\mathcal{T}_{n+1} = \mathcal{T}_n \cup \{\neg \mathcal{P}_n\}$ were not consistent either, then $\mathcal{T}_n \vdash \neg(\neg \mathcal{P}_n)$, and hence $\mathcal{T}_n \vdash \mathcal{P}_n$. We would then have $\mathcal{T}_n \vdash \neg \mathcal{P}_n$ and $\mathcal{T}_n \vdash \mathcal{P}_n$, contradicting the induction hypothesis that $\mathcal{T}_n$ is consistent. Thus, in every case, $\mathcal{T}_{n+1}$ is consistent.
- $\mathcal{T}'$ is consistent. Indeed, otherwise we would have $\mathcal{T}' \vdash \perp$. Since a proof consists of finitely many steps, and therefore involves finitely many formulas, these formulas are contained in some $\mathcal{T}_n$ for sufficiently large n . We would then have $\mathcal{T}_n \vdash \perp$, contradicting the consistency of $\mathcal{T}_n$.
- $\mathcal{T}'$ is complete. Indeed, let $\mathcal{Q}$ be a closed formula of $\mathcal{L}$. It is one of the $\mathcal{P}_n$. Therefore, $\mathcal{P}_n \in \mathcal{T}_{n+1}$ or $\neg \mathcal{P}_n \in \mathcal{T}_{n+1}$. Thus, either $\mathcal{Q}$ or $\neg \mathcal{Q}$ is an element of $\mathcal{T}'$. Consequently, $\mathcal{T}' \vdash \mathcal{Q}$ or $\mathcal{T}' \vdash \neg \mathcal{Q}$.

□

A useful observation for what follows: if $\mathcal{T}$ has Henkin witnesses, then so does $\mathcal{T}'$. Indeed, “ $(\exists x \mathcal{P}(x)) \rightarrow \mathcal{P}(c_{\mathcal{P}})$ ” belongs to $\mathcal{T}$ and therefore to $\mathcal{T}'$.

5. Constructing witnesses

Start with a theory $\mathcal{T}$ in a language $\mathcal{L}$. We will extend the language in order to obtain a new theory $\mathcal{T}'$ in a language $\mathcal{L}'$ that has Henkin witnesses; that is:

$$“\exists x \mathcal{P}(x) \rightarrow \mathcal{P}(c_{\mathcal{P}})” \text{ is a formula in } \mathcal{T}'.$$

In addition, $\mathcal{T}'$ must be consistent, just like $\mathcal{T}$.

5.1. Extending the language and the theory

$\mathcal{L}'$ will be constructed from $\mathcal{L}$ by adding new constant symbols $c_0, c_1, \dots$. Before giving the full proof, here is the fundamental step.

Let $\mathcal{T}$ be a consistent theory in a language $\mathcal{L}$. Let $\mathcal{P}$ be a formula of $\mathcal{L}$ whose only free variable is x . Let $c_{\mathcal{P}}$ be a new constant symbol (one that does not belong to $\mathcal{L}$). Let $\mathcal{L}' = \mathcal{L} \cup \{c_{\mathcal{P}}\}$.

Claim. $\mathcal{T}' = \mathcal{T} \cup \{\exists x \mathcal{P}(x) \rightarrow \mathcal{P}(c_{\mathcal{P}})\}$ is still consistent.

By construction, we have a Henkin witness (just for the formula $\mathcal{P}$), and consistency has been preserved.

Proof. Suppose, for a contradiction, that $\mathcal{T}'$ is not consistent:

$$\begin{aligned} & \mathcal{T} \cup \{\exists x \mathcal{P}(x) \rightarrow \mathcal{P}(c_{\mathcal{P}})\} \vdash \perp \\ \text{therefore} \quad & \mathcal{T} \vdash \neg(\exists x \mathcal{P}(x) \rightarrow \mathcal{P}(c_{\mathcal{P}})) \\ \text{therefore} \quad & \mathcal{T} \vdash \exists x \mathcal{P}(x) \wedge \neg \mathcal{P}(c_{\mathcal{P}}) \quad \text{since } \neg(\mathcal{A} \rightarrow \mathcal{B}) \equiv \mathcal{A} \wedge \neg \mathcal{B} \\ \text{therefore} \quad & \mathcal{T} \vdash \exists x \mathcal{P}(x) \text{ and } \mathcal{T} \vdash \neg \mathcal{P}(c_{\mathcal{P}}) \\ \text{therefore} \quad & \mathcal{T} \vdash \exists x \mathcal{P}(x) \text{ and } \mathcal{T} \vdash \forall x \neg \mathcal{P}(x) \quad \text{by the } \forall \text{ introduction rule} \\ \text{therefore} \quad & \mathcal{T} \vdash \exists x \mathcal{P}(x) \text{ and } \mathcal{T} \vdash \neg(\exists x \mathcal{P}(x)) \quad \text{by the negation of } \forall \\ \text{therefore} \quad & \mathcal{T} \vdash \mathcal{Q} \text{ and } \mathcal{T} \vdash \neg \mathcal{Q} \quad \text{for } \mathcal{Q} = \exists x \mathcal{P}(x) \end{aligned}$$

This contradicts the consistency of $\mathcal{T}$. Conclusion: $\mathcal{T}'$ is consistent. Note the use of $\forall$ introduction to pass from “ $\neg \mathcal{P}(c_{\mathcal{P}})$ ” to “ $\forall x \neg \mathcal{P}(x)$ ”. This is legitimate because the constant $c_{\mathcal{P}}$ does not occur in $\mathcal{T}$. Indeed, the formulas in $\mathcal{T}$ are formulas of $\mathcal{L}$, which does not contain the constant $c_{\mathcal{P}}$. □

One might think that we could do this for every formula $\mathcal{P}_0, \mathcal{P}_1, \dots$ of $\mathcal{L}$, but matters are a little more subtle. Indeed, at each step we enlarge the language, so there are more and more formulas, which complicates the recursion.

5.2. Construction

Let $\mathcal{T}$ be a consistent theory in a language $\mathcal{L}$. We will recursively construct a sequence of theories $(\mathcal{T}_n)_{n \geq 0}$ and a sequence of languages $(\mathcal{L}_n)_{n \geq 0}$ as follows:

- $\mathcal{L}_0 = \mathcal{L}$ and $\mathcal{T}_0 = \mathcal{T}$,
- and then, for $n \geq 0$:
 - $\mathcal{L}_{n+1} = \mathcal{L}_n \cup \{c_{\mathcal{P}}\}_{\mathcal{P}}$, where the union ranges over all formulas $\mathcal{P}$ of $\mathcal{L}_n$ having exactly one free variable,
 - $\mathcal{T}_{n+1} = \mathcal{T}_n \cup \{\exists x \mathcal{P}(x) \rightarrow \mathcal{P}(c_{\mathcal{P}})\}_{\mathcal{P}}$.

Finally, let $\mathcal{L}' = \bigcup_{n \geq 0} \mathcal{L}_n$ and $\mathcal{T}' = \bigcup_{n \geq 0} \mathcal{T}_n$.

Claim. $\mathcal{T}'$ has Henkin witnesses.

Proof. Let $\mathcal{P}$ be a formula of $\mathcal{L}'$ whose only free variable is x . It contains finitely many symbols of $\mathcal{L}'$, so for sufficiently large n , $\mathcal{P}$ is a formula of $\mathcal{L}_n$. Therefore, $\exists x \mathcal{P}(x) \rightarrow \mathcal{P}(c_{\mathcal{P}})$ is a formula in $\mathcal{T}_{n+1}$, and hence in $\mathcal{T}'$. $\square$

Claim. $\mathcal{T}'$ is consistent.

Proof. We will show that each $\mathcal{T}_n$ is consistent. This will prove that $\mathcal{T}'$ is consistent. Indeed, if we had $\mathcal{T}' \vdash \perp$, then a proof of this would only involve a finite number of formulas from $\mathcal{T}'$, which are therefore all included in some $\mathcal{T}_n$ for a sufficiently large n . For this n , we have $\mathcal{T}_n \vdash \perp$, which contradicts the consistency of $\mathcal{T}_n$.

Let us show by induction that $\mathcal{T}_n$ (for $n \geq 0$) is consistent. For $n = 0$, $\mathcal{T}_0 = \mathcal{T}$, so $\mathcal{T}_0$ is consistent by hypothesis.

Suppose $\mathcal{T}_n$ is consistent and, for the sake of contradiction, that $\mathcal{T}_{n+1}$ is not. A proof of $\mathcal{T}_{n+1} \vdash \perp$ involves only a finite number of formulas. There thus exist a finite number of formulas $\mathcal{P}_1, \dots, \mathcal{P}_k$ of $\mathcal{L}_n$ such that, letting $\mathcal{H}_i$ denote the Henkin axiom “ $\exists x \mathcal{P}_i(x) \rightarrow \mathcal{P}_i(c_{\mathcal{P}_i})$ ”, we have:

$$\mathcal{T}_n \cup \{\mathcal{H}_1, \dots, \mathcal{H}_k\} \vdash \perp$$

Let $\mathcal{S}_m = \mathcal{T}_n \cup \{\mathcal{H}_1, \dots, \mathcal{H}_m\}$ for $0 \leq m \leq k$. Let us show by finite induction on m that $\mathcal{S}_m$ is consistent.

- For $m = 0$, $\mathcal{S}_0 = \mathcal{T}_n$ is consistent by the induction hypothesis.
- Suppose $\mathcal{S}_{m-1}$ is consistent. The axiom $\mathcal{H}_m$ is $\exists x \mathcal{P}_m(x) \rightarrow \mathcal{P}_m(c_{\mathcal{P}_m})$. By definition, $c_{\mathcal{P}_m}$ is a new constant symbol that does not belong to $\mathcal{L}_n$. Furthermore, since the formulas $\mathcal{P}_i$ are pairwise distinct, the constant $c_{\mathcal{P}_m}$ is different from the constants $c_{\mathcal{P}_1}, \dots, c_{\mathcal{P}_{m-1}}$. Thus, the constant $c_{\mathcal{P}_m}$ does not appear anywhere in the theory $\mathcal{S}_{m-1}$. According to the fundamental claim proved at the beginning of the section, adding $\mathcal{H}_m$ to the consistent theory $\mathcal{S}_{m-1}$ yields a consistent theory. Therefore, $\mathcal{S}_m = \mathcal{S}_{m-1} \cup \{\mathcal{H}_m\}$ is consistent.

Consequently, $\mathcal{S}_k = \mathcal{T}_n \cup \{\mathcal{H}_1, \dots, \mathcal{H}_k\}$ is consistent. This contradicts the fact that $\mathcal{S}_k \vdash \perp$. Conclusion: $\mathcal{T}_{n+1}$ is indeed consistent. $\square$

5.3. Complete proof of the theorem

We have given the details of the proof of the Completeness Theorem, but somewhat out of order. Here is the proof plan in the proper order.

- **Assumptions.** $\mathcal{T}$ is a consistent theory, and $\mathcal{L}$ is a countable language.
- **Step 1: Add Henkin witnesses.** We refer to the beginning of Section 5. We obtain a theory $\mathcal{T}'$ in a new language $\mathcal{L}'$, with $\mathcal{T} \subset \mathcal{T}'$ and $\mathcal{L} \subset \mathcal{L}'$. $\mathcal{T}'$ has Henkin witnesses and remains consistent. $\mathcal{L}'$ is obtained from $\mathcal{L}$ by adding countably many constants $c_{\mathcal{P}}$, so $\mathcal{L}'$ remains countable.
- **Step 2: Make the theory complete.** See Section 4. Starting with $\mathcal{T}'$ and $\mathcal{L}'$, we obtain a complete theory $\mathcal{T}''$ with $\mathcal{T}' \subset \mathcal{T}''$. Henkin witnesses are also preserved (see the observation at the end of Section 4). The language is unchanged; let $\mathcal{L}'' = \mathcal{L}'$.
- **Step 3: Construct the model.** This is Section 3, applied to $\mathcal{T}''$ and $\mathcal{L}''$. We obtain a model $\mathcal{S}''$ of $\mathcal{T}''$ in $\mathcal{L}''$. This model naturally restricts to a model $\mathcal{S}$ of $\mathcal{T}$ in $\mathcal{L}$. Indeed, $\mathcal{L}''$ is obtained from $\mathcal{L}$ by adding constants $c_{\mathcal{P}}$. Consider the structure $\mathcal{S}$ that is identical to $\mathcal{S}''$ except that these constants (which do not exist in $\mathcal{L}$) are not interpreted. Let $\mathcal{P}$ be a formula of $\mathcal{T}$ in $\mathcal{L}$. It is therefore also a formula of $\mathcal{T}''$ in $\mathcal{L}''$. Since $\mathcal{S}''$ is a model of $\mathcal{T}''$, we have $\models_{\mathcal{S}''} \mathcal{P}$. But since $\mathcal{P}$ is a formula of $\mathcal{L}$, it contains none of the additional constant symbols, so we also have $\models_{\mathcal{S}} \mathcal{P}$. Thus, $\mathcal{S}$ is a model of $\mathcal{T}$.

This completes the proof of Gödel's Completeness Theorem. This theorem was the subject of his dissertation, completed in 1929 at the age of 23 and published the following year. This also concludes this part on first-order logic. In the next part, we continue with what made Gödel famous throughout the world, far beyond mathematics: the Incompleteness Theorem.

Exercises

The Completeness Theorem

Exercise 1. Recall Theorems A and B:

- Theorem A. If $\mathcal{T} \models \mathcal{Q}$, then $\mathcal{T} \vdash \mathcal{Q}$.
- Theorem B. If $\mathcal{T}$ is consistent, then $\mathcal{T}$ has a model.

Assume Theorem A. Deduce Theorem B.

Exercise 2. A theory $\mathcal{T}$ (which may contain infinitely many formulas) is said to be *finitely consistent* if, for every finite subset $\Gamma \subset \mathcal{T}$, Γ is consistent.

Show that a theory $\mathcal{T}$ is consistent if and only if it is finitely consistent.

Complete theory

Exercise 3. Let $\mathcal{L}$ be a first-order language and let $\mathcal{S}$ be a structure for this language. The *theory* of $\mathcal{S}$, denoted $\mathcal{T}(\mathcal{S})$, is defined as the set of all closed formulas $\mathcal{P}$ in $\mathcal{L}$ that are true in $\mathcal{S}$:

$$\mathcal{T}(\mathcal{S}) = \{\mathcal{P} \text{ closed formula} \mid \models_{\mathcal{S}} \mathcal{P}\}$$

Show that $\mathcal{T}(\mathcal{S})$ is a complete theory.

Exercise 4. Let $\mathcal{T}$ be a consistent theory. Show the equivalence between:

- (i) $\mathcal{T}$ is complete,
- (ii) for every closed formula $\mathcal{Q}$, if $\mathcal{T} \cup \{\mathcal{Q}\}$ is consistent, then $\mathcal{T} \vdash \mathcal{Q}$.

Constructing a model from a complete theory

Exercise 5. Consider a language $\mathcal{L}$ without an equality symbol, containing a constant 0, a unary function symbol s , and a unary predicate P . Let $\mathcal{T}$ be a complete theory in $\mathcal{L}$ that contains, in particular, the following two axioms (formulas):

- $\mathcal{A}_1 : P(0)$
- $\mathcal{A}_2 : \forall x (P(x) \leftrightarrow \neg P(s(x)))$

1. Construct the canonical model $\mathcal{S}$ associated with $\mathcal{T}$ (as in Section 3.2).
2. Construct a model $\mathcal{S}'$ with domain $\mathbb{N}$, where $s(x)$ corresponds to $x + 1$ (the successor), and explain what $P(x)$ represents.

Exercise 6. Complete the proof of Lemma 2 for the missing connectives, as well as for the quantifier $\exists$; that is, for the formulas: $\mathcal{Q} = \mathcal{P}_1 \vee \mathcal{P}_2$, $\mathcal{Q} = \mathcal{P}_1 \rightarrow \mathcal{P}_2$, $\mathcal{Q} = \mathcal{P}_1 \leftrightarrow \mathcal{P}_2$, $\mathcal{Q} = \exists x \mathcal{P}(x)$.

Constructing a complete theory

Exercise 7. Work in the simpler setting of propositional logic. Let $\mathcal{L}$ be a language containing only the two propositional variables A and B .

1. Explain why every formula is a closed formula. Give examples of formulas.
2. Explain why constructing a model of a theory $\mathcal{T}$ amounts to assigning a T/F value to A and a T/F value to B that make all the formulas of $\mathcal{T}$ true.
3. Explain why, if $\mathcal{T}$ has a model, then $\mathcal{T}$ is consistent.

We now want to construct (the beginning of) a complete theory starting from $\mathcal{T}_0 = \{A\}$. Consider the following choice for the enumeration $(\mathcal{P}_n)_{n \geq 0}$ of the formulas of $\mathcal{L}$:

$$B, \quad A \vee B, \quad \neg A, \quad A \rightarrow B, \quad \dots$$

For $n \geq 0$, define:

- $\mathcal{T}_{n+1} = \mathcal{T}_n \cup \{\mathcal{P}_n\}$ if this union is consistent,
- $\mathcal{T}_{n+1} = \mathcal{T}_n \cup \{\neg \mathcal{P}_n\}$ otherwise.

4. Explain why $\mathcal{T}_0$ is consistent.
5. Construct $\mathcal{T}_1, \dots, \mathcal{T}_4$.
6. Will the complete theory $\mathcal{T} = \bigcup_{n \geq 0} \mathcal{T}_n$ prove $A \wedge B$?
7. Start again with $\mathcal{T}'_0 = \{A\}$, but this time consider the enumeration:

$$\neg B, \quad A \vee B, \quad \neg A, \quad A \rightarrow B, \quad \dots$$

Construct $\mathcal{T}'_1, \dots, \mathcal{T}'_4$.

8. Will the complete theory $\mathcal{T}' = \bigcup_{n \geq 0} \mathcal{T}'_n$ be identical to $\mathcal{T}$?

Constructing witnesses

Exercise 8. Consider an initial language $\mathcal{L}_0$ containing a single unary predicate $P(x)$, with no constants or functions. Let the theory $\mathcal{T}_0$ consist of the following two axioms:

$$\mathcal{A}_1 : \exists x P(x) \quad \text{and} \quad \mathcal{A}_2 : \exists x \neg P(x)$$

1. Explain why $\mathcal{T}_0$ is consistent (for example, by constructing a model of it).
2. Does the language $\mathcal{L}_0$ have any closed terms? Does the theory $\mathcal{T}_0$ have Henkin witnesses in $\mathcal{L}_0$?
3. Apply Henkin's method to the two formulas $P(x)$ and $\neg P(x)$. Let $\mathcal{L}_1 = \mathcal{L}_0 \cup \{c_1, c_2\}$ be the language extended by two constants c_1 and c_2 . Which Henkin axioms have been added? What is the resulting theory $\mathcal{T}_1$?
4. Show that $\mathcal{T}_1$ proves $P(c_1) \wedge (\neg P(c_2))$. Deduce that, in every model of $\mathcal{T}_1$, the interpretation of c_1 and the interpretation of c_2 must be two distinct elements of the domain.

Exercise 9. Let $\mathcal{L}'$ be an extension of a language $\mathcal{L}$. Let $\mathcal{T}$ be a theory in the language $\mathcal{L}'$ that is complete and has Henkin witnesses. Let $\mathcal{P}(x)$ be a formula of the initial language $\mathcal{L}$ (whose only free variable is x).

Show that if $\mathcal{T}$ does not prove $\forall x \mathcal{P}(x)$, then there is a Henkin constant c of the language $\mathcal{L}'$ such that:

$$\mathcal{T} \vdash \neg \mathcal{P}(c).$$

What is c called in ordinary language?