Handout: Proof of the completeness theorem

Gödel's Compactness Theorem 1930. For a set $\Gamma$ of wffs and a wff $\varphi$, we have the following.

1. If $\Gamma \models \varphi$, then $\Gamma \vdash \varphi$.
2. If $\Gamma$ is consistent, then $\Gamma$ is satisfiable.

While this theorem was proved by Gödel in 1930, our proof is due to Henkin in 1949. This presentation very closely follows that in Enderton Section 2.5. The biggest difference is our use of natural deduction, but that is a relatively minor difference.

We want to focus on the main ideas of the proof and not get caught up in the details. Therefore (following Enderton) the proof is broken up into 6 steps and the details are either omitted (see Enderton Section 2.5) or written with a grey background.

1 Initial setup and outline of the proof

We are going to prove version 2 of the compactness theorem. Let $\Gamma$ be a consistent set of formulas. We need to show that $\Gamma$ is satisfiable. Our goal is to construct a model $\mathfrak{A}$ of $\Gamma$.

There are many similarities between this proof and both the proof of Compactness Theorem for Sentential Logic and the proof of the Completeness Theorem for sentential logic.

Here are some big ideas of the proof.

1. Just as in the proofs of Compactness and the Completeness Theorem for Sentential Logic we extend $\Gamma$ to a maximally consistent set $\Delta$.
2. We need to actually build a structure $\mathfrak{A}$. Should it be made of numbers? Sets? People? The brilliant idea that Henkin had was to make the universe of $|\mathfrak{A}|$ be the set of all terms. This type of model is called a term model.
3. Term models work on the principal that every object we care about has a term describing it. This is a problem when it comes to quantifiers. We fix this by adding in special constants $c$ (one for each pair $\langle \varphi, x \rangle$) which satisfy

   $$\exists x \varphi(x) \rightarrow \varphi(c)$$

   These constants act as a "name" for each type of object that $\Gamma$ says exists. (This is akin to when humans discovered that there is a solution to $x^2 + 1 = 0$. They felt it needed a name and called it $i$—even though there are actually two solutions.)

4. Last, in these term models we could have multiple terms for the same object. To fix this, we replace $|\mathfrak{A}|$ with the set of equivalence classes of terms. This trick is one of the most important in all of mathematics.
2 Step 1: Expanding the language

**Theorem** (Step 1). *Expand the language by adding in countably infinitely many constant symbols. Then \( \Gamma \) remains consistent as a set of wffs in the new language.*

We need to make sure that adding in these new constants does not make \( \Gamma \) inconsistent. The short reason of why this works is that, in our proof trees, constants behave just like variables. Since the constants are not used in \( \Gamma \), any proof of \( \Gamma \vdash \perp \) which uses these constants, can be replaced with a proof of \( \Gamma \vdash \perp \) where these constants are replaced by new variables. However, \( \Gamma \) is consistent so there is no such proof.

Details: Formally let \( c \) be a constant. We define \( \varphi^c \) to mean replace all occurrences of \( c \) in \( \varphi \) with the variable \( z \). Similarly define \( \Sigma^c \) to mean replace all occurrences of \( c \) in \( \Sigma \) with the variable \( z \). Then we prove the following by induction on proofs.

**Lemma.** If \( \Sigma \vdash \varphi \) (where \( c \) possibly occurs in \( \Sigma, \varphi \), and/or the proof tree) and \( z \) is a variable which does not occur in \( \Sigma \) or \( \varphi \) or the proof of \( \Sigma \vdash \varphi \), then \( \Sigma^c \vdash \varphi^c \) using a proof tree which does not contain \( c \).

**Proof.** Just check that every rule holds with \( c \) replaced by \( z \). The only rules that would be suspect are the quantifier rules. We handle those here:

- **\( \forall I \):** Assume we have a proof of \( \Sigma \vdash \forall y \varphi^x_y \) ending in
  \[
  \frac{\Sigma \vdash \varphi}{\Sigma \vdash \forall y \varphi^x_y \quad \forall I}
  \]
  and following the side conditions for this rule. By the induction hypothesis, there is a proof of \( \Sigma^c \vdash \varphi^c \). Since \( z \) is neither the variable \( x \) or \( y \), the following is still a valid way to prove \( \Sigma^c \vdash (\forall y \varphi^x_y)_z^c \) (since \( (\forall y \varphi^x_y)_z^c = \forall y (\varphi^c)_y \)):
  \[
  \frac{\Sigma^c \vdash \varphi^c}{\Sigma^c \vdash \forall y (\varphi^c)_y^z \quad \forall I}
  \]

- **\( \forall E \):** Assume we have a proof of \( \Sigma \vdash \varphi^c \) ending in
  \[
  \frac{\Sigma \vdash \forall x \varphi}{\Sigma \vdash \varphi^c \quad \forall I}
  \]
  where \( t \) is substitutable for \( x \) in \( \varphi \). By the induction hypothesis, there is a proof of \( \Sigma^c \vdash \forall x \varphi^c \). Since \( z \) is not in \( \varphi \), then \( t^c_z \) is substitutable for \( x \) in \( \varphi^c \). Hence, the following is still a valid way to prove \( \Sigma^c \vdash (\varphi^c)_t^c \) (since \( (\varphi^c)_t^c = (\varphi^c)_t^c \)):
  \[
  \frac{\Sigma^c \vdash \forall x \varphi^c}{\Sigma^c \vdash (\varphi^c)_t^c \quad \forall E}
  \]

Now, letting \( \Sigma = \Gamma \) and \( \varphi = \perp \), we have the following: If \( \Gamma \) is inconsistent using a proof with the constant \( c \), then \( \Gamma \) is inconsistent using a proof without the constant \( c \). (The constant \( c \) is not in \( \Gamma \) or \( \perp \), so \( \Gamma^c_z = \Gamma \) and \( \perp^c_z = \perp \).)
3 Step 2: Add special sentences to $\Gamma$

For every wff $\varphi$ (in our new language with the constants) and for each variable $x$ we consider the formula

$$\neg \forall x \varphi \rightarrow \neg \varphi^x_c$$

were $c$ is the unique constant associated with $\varphi$ and $x$. (This formula basically says that if there exists $\exists x \neg \varphi$ then that $\neg \varphi$ holds with $c$ in place of $x$. If $\exists x \neg \varphi$ does not hold, then this is vacuously true. Such constants $c$ are called *witnessing constants* since they witness that $\exists x \neg \varphi$ holds.) Let $\Theta$ be the set of all such formulas.

This new set $\Gamma \cup \Theta$ is still consistent. (The main idea is that if were not consistent, then we could prove $\Gamma \cup \Theta \vdash \bot$. Then we can modify this to a proof of $\Gamma \vdash \Theta$.)

Details: First, double enumerate our new constants as $(c_{m,n})_{m \in \mathbb{N}, n \in \mathbb{N}}$. (So rather than list them 0, 1, 2,..., we list associate each constant with a pair of natural numbers.) For each $m$, enumerate all pairs $\langle \varphi, x \rangle$ where $\varphi$ is a wff using only the new constants $c_{i,j}$ where $i < m$ and $x$ is a variable. Then for the $n$th pair $\langle \varphi, x \rangle$, add in the sentence

$$\neg \forall x \varphi \rightarrow \neg \varphi^x_c$$

where $c$ is $c_{m,n}$. In particular, $c_{m,n}$ is not in $\varphi$.

Now, consider one of these sentences

$$\neg \forall x \varphi \rightarrow \neg \varphi^x_c$$

For any $\Sigma$ and $\psi$ we can prove the following.

**Lemma.** If $\Sigma; \neg \forall x \varphi \rightarrow \neg \varphi^x_c \vdash \psi$ (where $c$ is not in $\Sigma$, $\varphi$, or $\psi$) then $\Sigma \vdash \psi$.

*Before we prove the lemma, consider this exercise.*

**Exercise.** $\vdash \exists y (\neg \forall x \varphi \rightarrow \neg \varphi^x_y)$ where $y$ is any variable not free in $\varphi$. (This is the drinker’s paradox we mentioned in class. There is someone at the bar such that if that not everyone is drinking, then that person is not drinking.)

**Proof of Lemma.** Recall we derived a rule for $\exists E$. One way to write this rule is as follows.

$$\frac{\Sigma \vdash \exists y \varphi \quad \Sigma, \varphi^y \vdash \theta}{\Sigma \vdash \theta}$$

where $x$ is not free in $\Sigma$ or $\theta$, $x$ is substitutable for $y$ in $\varphi$, and either $y = x$ or $x$ is not free in $\varphi$.

By assumption we know that

$$\Sigma; \neg \forall x \varphi \rightarrow \neg \varphi^x_c \vdash \psi$$

By a lemma in the last step, we can replace $c$ with any variable $z$ to get

$$\Sigma; \neg \forall x \varphi \rightarrow \neg \varphi^x_z \vdash \psi$$
since $c$ is not in $\Sigma$ or $\psi$ or $\varphi$. By the exercise, we have

$$\Sigma \vdash \exists y (\neg \forall x \varphi \rightarrow \neg \varphi^*_y)$$

Putting all this together with the derived rule for $\exists E$, we get the following proof of $\Sigma \vdash \varphi$:

$$\Sigma \vdash \exists y (\neg \forall x \varphi \rightarrow \neg \varphi^*_y) \quad \Sigma, \neg \forall x \varphi \rightarrow \neg \varphi^*_y \vdash \psi \quad \exists E$$

Now if $\Gamma \cup \Theta \vdash \bot$, then by finite character we use only finitely many formulas in $\Theta$ to prove $\bot$. By applying the previous lemma finitely many times, we can eliminate those formulas in $\Theta$ from our assumptions to get a proof of $\Gamma \vdash \bot$. But this contradicts that $\Gamma$ is consistent.

4  Step 3: Extend $\Gamma \cup \Theta$ to a maximally consistent set $\Delta$

Extend $\Gamma \cup \Theta$ to a consistent set $\Delta \supseteq \Gamma \cup \Theta$ such that for every wff $\varphi$ (in the larger language with constants) either $\varphi \in \Delta$ or $\neg \varphi \in \Delta$.

Details: We have done this many times before. For more details, see our proof of the completeness theorem for sentential logic. Also see the book (Section 2.5) as well as the proof of the compactness theorem for sentential logic.

**Lemma.** $\Delta \vdash \varphi$ implies $\varphi \in \Delta$

**Proof.**

$$\Delta \vdash \varphi \Rightarrow \Delta \not\vdash \varphi \quad \text{consistency}$$

$$\Rightarrow (\neg \varphi) \notin \Delta$$

$$\Rightarrow \varphi \in \Delta \quad \Delta \text{ maximal}$$

5  Step 4: Use $\Delta$ to construct a term model $\mathfrak{A}$ (but with weak notion of $=$)

Define $\mathfrak{A}$ and $s$ as follows. First, remove $=$ from our language an replace it with a binary predicate $E$. Let $\mathfrak{A}$ and $s$ as follows:

- $|\mathfrak{A}|$ is the set of all terms in the new language (with the new constants).
• \( \langle u, t \rangle \in E^A \iff (u = t) \in \Delta \)

• For each other \( n \)-place predicate symbol \( P \)

\[
\langle t_1, \ldots, t_n \rangle \in P^A \iff (Pt_1 \ldots t_n) \in \Delta.
\]

• For each \( n \)-place function symbol \( f \)

\[
f^A(t_1, \ldots, t_n) = ft_1 \ldots t_n
\]

• (So for constant symbols \( c, c^A = c \).)

• Define \( s : V \to |A| \) as \( s(x) = x \).

It is easy to see that \( \bar{s}(t) = t \) for all terms \( t \).

**Lemma.** Let \( \varphi^* \) be the result of replacing \( = \) with \( E \) in \( \varphi \). Then

\[\models_A \varphi^*[s] \iff \varphi \in \Delta.\]

**Proof.** We omit many of the details, all of which can be found in the book (Section 2.5). The proof is by induction. We will only do the case where \( \varphi \) is \( \forall x \psi \). Then \( \varphi^* = \forall x \psi^* \).

Consider the constant \( c \) such that

\[\neg \forall x \psi \rightarrow \neg \psi^*_c \in \Theta \subseteq \Delta.\]

(\( \Rightarrow \)):

\[
\begin{align*}
\models_A \forall x \psi^*[s] & \Rightarrow \models_A \psi^*[s|c] \\
& \Rightarrow \models_A (\psi^*_c)^s \subseteq \Delta \quad \text{substition lemma} \\
& \Rightarrow \models_A (\psi^*_c)^s \subseteq \Delta \quad \text{IH} \\
& \Rightarrow (\neg \psi^*_c) \notin \Delta \quad \text{consistency} \\
& \Rightarrow (\neg \forall x \psi) \notin \Delta \quad \neg \forall x \psi \rightarrow \neg \psi^*_c \in \Delta \\
& \Rightarrow \forall x \psi \in \Delta
\end{align*}
\]

(\( \Leftarrow \)) See the book (Section 2.5).

If \( \Gamma \) does not contain \( = \), then we are done since \( A \) satisfies each formula in \( \Gamma \) with \( s \). (Since \( \psi^* = \psi \) for all \( \psi \in \Gamma \).) However, we are not done if \( \Gamma \) contains \( = \). For example, what if the sentence \( x + 0 = x \) was in \( \Gamma \). In term model \( A \), the terms \( x + 0 \) and \( x \) are not equal.
6 Step 5: Replace $\mathcal{A}$ with the structure $\mathcal{A}/E$.

The relation $E$ satisfies the following properties because of all the inference rules for $\equiv$.

1. $E^\mathcal{A}$ is reflexive, symmetric, transitive on $|\mathcal{A}|$. (That is, $(|\mathcal{A}|; E^\mathcal{A})$ is an equivalence relation.)

2. If $\langle t_1, \ldots, t_n \rangle \in P^\mathcal{A}$ and $t_i E^\mathcal{A} t'_i (1 \leq i \leq n)$, then $\langle t'_1, \ldots, t'_n \rangle \in P^\mathcal{A}$.

3. If $t_i E^\mathcal{A} t'_i (1 \leq i \leq n)$, then $f^\mathcal{A}(t_1, \ldots, t_n) = f^\mathcal{A}(t'_1, \ldots, t'_n)$.

Since $(|\mathcal{A}|; E^\mathcal{A})$ is an equivalence class, $|\mathcal{A}|$ can be broken up into parts called equivalence classes

$$[t] = \{t' \in |\mathcal{A}| : t E^\mathcal{A} t'\}$$

Using these above properties we can define the structure $\mathcal{A}/E$ as follows:

1. $|\mathcal{A}/E|$ is the set of equivalence classes $[t]$ for terms $t \in |\mathcal{A}|$.

2. For each $n$-place predicate symbol $P$,

$$\langle [t_1], \ldots, [t_n] \rangle \in P^{\mathcal{A}/E} \iff \langle t_1, \ldots, t_n \rangle \in P^\mathcal{A}$$

3. For each $n$-place function symbol $f$,

$$f^{\mathcal{A}/E}([t_1], \ldots, [t_n]) = [f^\mathcal{A}(t_1, \ldots, t_n)]$$

4. (For constant symbols $c$, $c^{\mathcal{A}/E} = [c^\mathcal{A}]$.)

Let $h: |\mathcal{A}| \rightarrow |\mathcal{A}|/E$ be $h(t) = [t]$. Then $h$ is a homomorphism of $|\mathcal{A}|$ onto $|\mathcal{A}|/E$ and $E^{\mathcal{A}/E}$ is the same as $\equiv$ on $\mathcal{A}/E$. Therefore,

$$\varphi \in \Delta \iff \models_\mathcal{A} \varphi^*[s]$$

$$\iff \models_{\mathcal{A}/E} \varphi^*[h \circ s] \quad \text{homomorphism theorem (}h\text{ is onto, }\varphi\text{ has no }\equiv\text{)}$$

$$\iff \models_{\mathcal{A}/E} \varphi[h \circ s] \quad E^{\mathcal{A}/E} \text{ is same as } \equiv$$

(See Enderton (Section 2.5) for details.)

7 Step 6: Restrict $\mathcal{A}/E$ to the original language. It satisfies $\Gamma$ with $h \circ s$.

(See Enderton (Section 2.5) for details.)

This completes the proof of the Completeness Theorem.
8 An application of the Completeness and Soundness Theorems

The Completeness and Soundness theorems provide a seamless bridge between model theory (semantics) and proof theory (syntax). Indeed every result in one has a corresponding result in the other. The compactness theorem is one example. It is a model theoretic result which corresponds to the proof theoretic result of Finite Character.

Here is another example:

**Theorem.** If $\Gamma \vdash \varphi$ in the language using a predicate $P$ (so $P$ may be used in the proof of $\Gamma \vdash \varphi$) and $P$ is in neither $\Gamma$ nor $\varphi$, then there is a proof of $\Gamma \vdash \varphi$ which does not contain $P$.

There are two proofs of this theorem.

- The “hard” approach is to systemically modify the proof to avoid $P$. (This is a by induction on proofs.)
- The “soft” approach is to use the soundness and completeness theorem to turn this into a model theory problem.

**Proof.** (Using soft approach.) Assume $\Gamma \vdash \varphi$ in the language with $P$. By the Soundness Theorem, $\Gamma \models \varphi$ in the language with $P$. By definition, for all structures $\mathfrak{A}$ in the language with $P$ and all $s: V \to |\mathfrak{A}|$, if $\mathfrak{A}$ satisfies $\Gamma$ with $s$, then $\mathfrak{A}$ satisfies $\varphi$ with $s$.

We want to show $\Gamma \vdash \varphi$ in the language without $P$. By the Completeness Theorem, it is enough to prove $\Gamma \models \varphi$ in the language without $P$. By definition, we want to show for all structures $\mathfrak{A}$ in the language without $P$ and all $s: V \to |\mathfrak{A}|$, if $\mathfrak{A}$ satisfies $\Gamma$ with $s$, then $\mathfrak{A}$ satisfies $\varphi$ with $s$.

Fix an arbitrary structure $\mathfrak{A}$ in the language without $P$ and an arbitrary function $s: V \to |\mathfrak{A}|$. Assume $\mathfrak{A}$ satisfies $\Gamma$ with $s$. Let $\mathfrak{A}^*$ be the same structure as $\mathfrak{A}$ except that we add in the relation $P_{\mathfrak{A}}^* = \emptyset$. (Here we set $P_{\mathfrak{A}}^*$ to be the always false relation. It doesn’t matter what relation we use here.) Now $\mathfrak{A}^*$ is a structure in the language with $P$. Since $\Gamma$ does not contain $P$ and $\mathfrak{A}$ satisfies $\Gamma$ with $s$, so does $\mathfrak{A}^*$ satisfy $\Gamma$ with $s$. Then since $\Gamma \vdash \varphi$ in the language without $P$ we have that $\mathfrak{A}^*$ satisfies $\varphi$ with $s$. Finally, since $P$ is not in $\varphi$, we have that $\mathfrak{A}$ also satisfies $\varphi$ with $s$.

This shows that $\Gamma \vdash \varphi$ in the language without $P$. By the completeness theorem, $\Gamma \vdash \varphi$ in the language without $P$. \qed