Incompleteness v2 Revisited

Harry Sha

Fall 2026

Section 1

Review

Incompleteness v1 (Formal)

Let Σ be a set of sentences, and $\mathcal{N}$ be the standard model. Then, the following cannot be satisfied simultaneously.

1. **Completeness**. $\mathcal{N} \models A \implies \Sigma \vdash A$
2. **Soundness**. $\Sigma \vdash A \implies \mathcal{N} \models A$
3. **Recognizable axioms**. $\Sigma \in \mathbf{TR}$
4. **Expressivity**. For each TM M, there is a formula φ_M such that $\mathcal{N} \models \varphi_M$ if and only if M halts on the emptystring.

We showed that the last property could be removed so long as we are working over the language of arithmetic since were were able to construct this formula φ_M by expressing TM configurations (and sequences of them) as numbers, and the transitions of the TM as operations of arithmetic.

Some formulas we constructed

- $\mathrm{AccHist}_M(\overline{\ulcorner w \urcorner}, h)$. h codes an accepting sequence of TM configurations for M when run on w.
- $\mathrm{RejHist}_M(\overline{\ulcorner w \urcorner}, h)$. h codes a rejecting sequence of TM configurations for M when run on w.
- $\mathrm{HaltHist}_M(\overline{\ulcorner w \urcorner}, h)$. h codes a halting sequence of TM configurations for M when run on w.
- $A_M(\overline{\ulcorner x \urcorner})$. $\exists h \mathrm{AccHist}_M(\overline{\ulcorner w \urcorner}, h)$
- $H_M(\overline{\ulcorner x \urcorner})$. $\exists h \mathrm{AccHist}_M(\overline{\ulcorner w \urcorner}, h)$

Sometimes we included t, the number of steps as an extra parameter, but we can always just pair it with h...

Incompleteness v1 (Formal, stronger)

Let Σ be a set of sentences in the language of arithmetic, and $\mathcal{N}$ be the standard model. Then, the following cannot be satisfied simultaneously.

1. **Completeness**. For all A, $\Sigma \vdash A$ or $\Sigma \vdash \neg A$
2. **Soundness**. $\mathcal{N} \models \Sigma$
3. **Recognizable axioms**. $\Sigma \in \mathbf{TR}$

Is there an effective axiomatization that exactly captures $\mathrm{Th}(\mathcal{N})$?

Incompleteness v2 (Informal)

No formal system can simultaneously have all four properties:

- **Completeness**. for all φ either φ or $\neg\varphi$ is provable.
- **Consistency**. for all φ at most one of φ and $\neg\varphi$ is provable.
- **Verifiable proofs**. there is an algorithm that checks proposed proofs.
- **Expressivity**. the system represents TM computations.

Incompleteness v2 (Formal)

Let Σ be a set of sentences in the language of arithmetic. Then, the following cannot be satisfied simultaneously.

- **Completeness.**. For all A, $\Sigma \vdash A$ or $\Sigma \vdash \neg A$
- **Consistency.** For all A, at most one of the following $\Sigma \vdash A$, $\Sigma \vdash \neg A$.
- **Recognizable axioms.** $\Sigma \in \mathbf{TR}$
- **Expressivity**. There is a formula α_x such that
  - If M_x accepts ϵ, then $\Sigma \vdash \alpha_x$.
  - If M_x rejects ϵ, then $\Sigma \vdash \neg\alpha_x$.
  - The map $x \mapsto \alpha_x$ is computable.

Note. No mention of the standard model!

Review of Proof

R:

- $\langle R \rangle = \texttt{get_source()}$
- construct $\alpha_{\langle R \rangle}$.
- For each string p in standard string order:
  - if p is a proof of $\alpha_{\langle R \rangle}$, reject;
  - if p is a proof of $\neg\alpha_{\langle R \rangle}$, accept.

Then Σ either proves $\alpha_{\langle R \rangle}$ or $\neg\alpha_{\langle R \rangle}$ by completeness, but both cases result in contradiction. (why?)

Just like last lecture, we're going to do away with the expressivity condition. The result will be an even stronger version of the incompleteness theorem.

Incompleteness v2

Let Σ be a set of sentences in the language of arithmetic. The following cannot simulatneously hold. Then, the following cannot be satisfied simultaneously.

- **Completeness.** For all A, $\Sigma \vdash A$ or $\Sigma \vdash \neg A$
- **Consistency.** For all A, at most one of the following $\Sigma \vdash A$, $\Sigma \vdash \neg A$.
- **Recognizable axioms.** $\Sigma \in \mathbf{TR}$
- **Captures basic arithmetic.** $\mathrm{Cn}(A_E) \subseteq \mathrm{Cn}(\Sigma)$

I haven't yet told you what A_E is, so let's start with that.

Exercise. Show that the "captures basic arithmetic condition" is implied by the Completeness and Soundness conditions from Incompleteness v1.

Section 2

A_E

The set of axioms A_E

$$\forall x \quad \mathbf{S}x \neq \mathbf{0} \tag{S1}$$

$$\forall x \forall y \quad (\mathbf{S}x = \mathbf{S}y \rightarrow x = y) \tag{S2}$$

$$\forall x \forall y \quad (x < \mathbf{S}y \leftrightarrow x \leq y) \tag{L1}$$

$$\forall x \quad x \not< \mathbf{0} \tag{L2}$$

$$\forall x \forall y \quad (x < y \vee x = y \vee y < x) \tag{L3}$$

$$\forall x \quad x + \mathbf{0} = x \tag{A1}$$

$$\forall x \forall y \quad x + \mathbf{S}y = \mathbf{S}(x + y) \tag{A2}$$

$$\forall x \quad x \cdot \mathbf{0} = \mathbf{0} \tag{M1}$$

$$\forall x \forall y \quad x \cdot \mathbf{S}y = x \cdot y + x \tag{M2}$$

$$\forall x \quad x\mathbf{E}\mathbf{0} = \mathbf{S}\mathbf{0} \tag{E1}$$

$$\forall x \forall y \quad x \,\mathbf{E}\, \mathbf{S}y = x \,\mathbf{E}\, y \cdot x \tag{E2}$$

Obervations

- A_E is finite.
- $\mathcal{N} \models A_E$

Question. What is the relationship between $\mathrm{Th}(\mathcal{N})$ and $\mathrm{Cn}(A_E)$.

Review of the Goal

It turns out, A_E is already strong enough so that we can construct α_x with the properties that we wanted:

- If M_x accepts ϵ, then $A_E \vdash \alpha_x$.
- If M_x rejects ϵ, then $A_E \vdash \neg\alpha_x$.

So what kinds of things can A_E prove?

Recall $\overline{n}$ is short for $S...S0$ where there are n Ss.

In the pre-lecture questions, I asked you to prove $\overline{1} + \overline{1} = \overline{2}$.

For any natural number k, $A_E \vdash \forall x(x < \overline{k+1} \leftrightarrow x = \overline{0} \vee ... \vee x = \overline{k})$

For all terms t without any variables, $A_E \vdash t = \overline{k}$ for some $k \in \mathbb{N}$.

For any quantifier-free formula A such that $\mathcal{N} \models A$, $A_E \vdash A$.

Define an existential formula as a formula of the form $\exists x_1 ... \exists x_k B$ where B is quantifier-free

For any existential formula A such that $\mathcal{N} \models A$, $A_E \vdash A$.

Define a bounded formula as a formula in which all quantifiers appear in the following way.

$$\forall x(x < t \to \ldots), \exists x(x < t \wedge \ldots)$$

For any bounded formula A such that $\mathcal{N} \models A$, $A_E \vdash A$.

The formulas from last time

- $\mathrm{AccHist}_M(\overline{\ulcorner w \urcorner}, h)$. h codes an accepting sequence of TM configurations for M when run on w.
- $\mathrm{RejHist}_M(\overline{\ulcorner w \urcorner}, h)$. h codes a rejecting sequence of TM configurations for M when run on w.

Note these are both bounded!

Attempt

Take $\alpha_x = \exists h.\mathrm{AccHist}_{M_x}(\overline{\ulcorner \epsilon \urcorner}, h)$?

Aside: Non Standard Models

To illustrate the previous problem. Consider the following construction of a model of $\mathrm{Th}(\mathcal{N})$. Add a constant c, and consider the set.

Take $T = \mathrm{Th}(\mathcal{N}) \cup \{\overline{n} < c : n \in \mathbb{N}\}$

The initial proof

Gödel's initial proof of the incompleteness theorem sidestepped this problem using a stronger assumption. Namely, he assumed something called ω-consistency which says that if the system can prove $P(\overline{0}), P(\overline{1}), ...$, then you can't prove $\exists x \neg P(x)$.

Combined with the completeness assumption, it says if you can prove $P(\overline{0}), P(\overline{1}),...$, then you can also prove $\forall x P(x)$.

Later, Rosser showed that this assumption is not needed.

Rosser's Trick

For every TM M, and input w, define

$$\Phi_M(\overline{\ulcorner w \urcorner}) = \exists a.(\mathrm{AccHist}_M(\overline{\ulcorner w \urcorner}, a) \wedge \forall r < a.\neg \mathrm{RejRun}_M(\overline{\ulcorner w \urcorner}, r))$$

Lemma (Smaller opposing witness)

For every TM M, and input string w.

- M *accepts* w *implies* $A_E \vdash \Phi_M(\overline{\ulcorner w \urcorner})$.
- M *rejects* w *implies* $A_E \vdash \neg\Phi_M(\overline{\ulcorner w \urcorner})$.
- Φ_M *is computable given a descprition of* M.

In otherwords, $\Phi_{M_x}(\overline{\ulcorner \epsilon \urcorner})$ is the formula, we want.

Proof

$$\Phi_M(\overline{\ulcorner w \urcorner}) = \exists a.(\mathrm{AccHist}_M(\overline{\ulcorner w \urcorner}, a) \wedge \forall r < a.\neg \mathrm{RejRun}_M(\overline{\ulcorner w \urcorner}, r))$$

Section 3

You don't need E

We can even remove E and the related axioms from A_E, and all the proofs from before still work. In particular, we can simulate E with just multiplication and addition.

Sketch

Let $A_M \subseteq A_E$ be the subset without the exponentiation axioms. We'll show that there is a formula $\mathrm{Exp}(x, y, z)$ not containing E such that

- $A_M \vdash \mathrm{Exp}(\overline{m}, \overline{n}, \overline{m^n})$.
- $A_M \vdash \forall z \mathrm{Exp}(\overline{m}, \overline{n}, z) \to z = \overline{m^n}$.

An initial idea is just to set $\mathrm{Exp}(x, y, z) = z = x...x$, but this doesn't work...

Sketch

See Enderton for the details!

Section 4

Second Incompleteness

The Second Incompleteness Theorem (Informal)

Suppose Σ satisfies the following

- **Consistent**. For all sentences A, it's not the case that $\Sigma \vdash A$ and $\Sigma \vdash \neg A$.
- $\Sigma \in \mathbf{TR}$.
- **Captures basic arithmetic.** $\mathrm{Cn}(A_E) \subseteq \mathrm{Cn}(\Sigma)$,
- Σ also satisfies some slightly stronger conditions.

Then $\Sigma \nvdash \mathrm{Con}(\Sigma)$

$\mathrm{Con}(\Sigma)$

Σ being consistent in the statement of the theorem is a meta-level definition. However, if we want to say something about Σ not proving its own consistency, we must formulate being consistent as a FOL sentence.

The slightly stronger conditions

1. If $C(a)$ is bounded, and $\sigma = \exists a(C(a))$, then $\Sigma \vdash \sigma \rightarrow \mathrm{Prov}_\Sigma(\overline{\ulcorner \sigma \urcorner})$
2. If a TM M enters an accepting state after its proof checker verifies a proof of a formula A, then $\Sigma \vdash \mathrm{Accept}_M(\overline{\ulcorner w \urcorner}) \rightarrow \mathrm{Prov}_\Sigma(\overline{\ulcorner A \urcorner})$
3. $\Sigma \vdash \mathrm{Prov}_\Sigma(\overline{\ulcorner A \urcorner}) \wedge \mathrm{Prov}_\Sigma(\overline{\ulcorner \neg A \urcorner}) \rightarrow \mathrm{Prov}_\Sigma(\overline{\ulcorner A \wedge \neg A \urcorner})$

Proof

Use the same program as before.

R:

- $\langle R \rangle =$ `get_source()`
- construct $\alpha_{\langle R \rangle}$.
- For each string p in standard string order:
  - if p is a proof of $\alpha_{\langle R \rangle}$, reject;
  - if p is a proof of $\neg\alpha_{\langle R \rangle}$, accept.

Recall, we ended up choosing
$\alpha_{\langle R \rangle} = \Phi_{\langle R \rangle}(\overline{\ulcorner w \urcorner}) = \exists a.(\mathrm{AccHist}_R(\overline{\ulcorner w \urcorner}, a) \wedge \forall r < a.\neg \mathrm{RejRun}_R(\overline{\ulcorner w \urcorner}, r))$

For simplicity, let's abbreviate that entire formula as A

Lemma: $\Sigma \vdash A \to \neg\mathsf{Con}(A)$

$A = \exists a.(\mathrm{AccHist}_R(\overline{\ulcorner w \urcorner}, a) \wedge \forall r < a.\neg\mathrm{RejRun}_R(\overline{\ulcorner w \urcorner}, r))$

Conclusion