MA5219 LOGIC AND FOUNDATION OF MATHEMATICS I

Lecture 13: A closer look at incompleteness

Tin Lok Wong

1 October, 2018

The aim of this lecture is to look back and see how everything fits together to give the Incompleteness Theorems. We include a few related results and introduce some terminology that is commonly used in the literature.

Let us start with informal statements of the Incompleteness Theorems.

Rosser's Theorem / First Incompleteness Theorem / Gödel–Rosser Incompleteness Theorem. No sufficiently strong consistent axiomatic system is complete.

Second Incompleteness Theorem (Gödel)**.** No sufficiently strong consistent axiomatic system proves its own consistency.

After 12 lectures, we can now explain these in more precise terms.

- An 'axiomatic system' consists of a deduction system $\vdash$ based on first-order logic (see Lecture 6, pages 22f.) and a recursive set T of axioms (see Lecture 8, page 29). We insist that the axiom set must be recursive because one needs to be able to check algorithmically whether one is allowed to make a certain assumption in a proof. We do not discuss the choice of deduction system here, but first-order logic is known (from Lindström, for example) to have nice distinguishing properties. (The term 'axiomatic system' is not standard.)
- Saying the axiomatic system $(T, \vdash)$ is 'complete' means $T \vdash \sigma$ or $T \vdash \neg\sigma$ for every sentence σ in the relevant language.
- Saying the axiomatic system $(T, \vdash)$ is 'consistent' means there is no sentence σ such that $T \vdash \sigma$ and $T \vdash \neg\sigma$. For this, we introduced a symbol $\perp$ such that $T \vdash \perp$ if and only if $T \vdash \sigma$ and $T \vdash \neg\sigma$ for some sentence σ (see Lecture 7, page 24).
- The term 'consistency' in the Second Incompleteness Theorem above refers to the sentence $\neg \Box_T \perp$, where $\Box_T$ is some formula (constructed in Lecture 10) that defines in $\mathbb{N}$
$$\{\ulcorner\theta\urcorner : \theta \text{ is a formula and } T \vdash \theta\}.$$
- We require different levels of 'strength' in the First and the Second Incompleteness Theorems.
  - Our proof of the First Incompleteness Theorem (in Lecture 8, pages 29ff.) requires the axiomatic system in question to Σ_1-represent all recursive functions and be Σ_1 complete. We saw in Theorem 4.4 and Corollary 4.2 that any axiomatic system stronger than $(\mathrm{R}(\exp), \vdash)$ satisfies these conditions.
  - Our proof of the Second Incompleteness Theorem (in Lecture 9, pages 33f.) requires $\Box_T$ to satisfy Löb's derivability conditions with respect to the axiomatic system. We spent most of Lecture 11 and Lecture 12 verifying this for an axiomatic system stronger than $(\mathrm{I}\Delta_0(\exp), \vdash)$.

  The requirement for the Second Incompleteness Theorem is generally understood to be stronger than that for the First Incompleteness Theorem.

The language $\mathcal{L}_A(\exp)$ we used has $0, 1, +, \times, \exp, <$ as non-logical symbols. Another popular choice is the language of ordered rings, which is essentially $\mathcal{L}_A(\exp)$ minus exp. Using the notion of bounded quantification, we defined the formula classes $\Delta_0(\exp)$, Σ_1 and Π_1. These are part of the so-called *arithmetic hierarchy.*

Definition. Let $n \in \mathbb{N}$. Then $\Sigma_0(\exp) = \Pi_0(\exp) = \Delta_0(\exp)$,

$$\Sigma_{n+1} = \{\exists \bar{y}_0\ \forall \bar{y}_1\ \cdots\ \mathrm{Q}\bar{y}_n\ \theta(\bar{x}, \bar{y}) : \mathrm{Q} \in \{\forall, \exists\} \text{ and } \theta \in \Delta_0(\exp)\}, \text{ and}$$
$$\Pi_{n+1} = \{\forall \bar{y}_0\ \exists \bar{y}_1\ \cdots\ \mathrm{Q}'\bar{y}_n\ \theta(\bar{x}, \bar{y}) : \mathrm{Q}' \in \{\forall, \exists\} \text{ and } \theta \in \Delta_0(\exp)\}.$$

Although we already used this terminology in various places, we did not properly define what it means for a formula to define a set in a structure. We make this definition now.

Definition. Let $k \in \mathbb{N}$ and $S \subseteq \mathbb{N}^k$ An $\mathcal{L}_A(\exp)$ formula $\theta(x_1, x_2, \ldots, x_k)$ *defines* S in $\mathbb{N}$ if

$$S = \{(a_1, a_2, \ldots, a_k) \in \mathbb{N}^k : \mathbb{N} \models \theta(\bar{a})\}.$$

Let Γ be a class of $\mathcal{L}_A(\exp)$ formulas. Then S is Γ-*definable* in $\mathbb{N}$ if it can be defined by a formula in Γ in $\mathbb{N}$.

A routine induction shows that the arithmetic hierarchy exhausts all the $\mathcal{L}_A(\exp)$ formulas up to equivalence. The position in which an $\mathcal{L}_A(\exp)$ formula appears in the arithmetic hierarchy is related to the complexity of the set defined by this formula in $\mathbb{N}$. For instance, for $k \in \mathbb{N}$, by definition (Lecture 1, page 5), a subset of $\mathbb{N}^k$ is r.e. if and only if it is Σ_1-definable in $\mathbb{N}$. It is recursive if and only if it is Δ_1-*definable*, i.e., both Σ_1- and Π_1-definable, in $\mathbb{N}$. In a similar way, the class Δ_0 corresponds to the linear time hierarchy, as Wrathall (1978) showed.

In addition to recursive sets, we also defined recursive *functions.* Recall (from Lecture 4, page 14) that a function $F \colon \mathbb{N}^k \to \mathbb{N}$ is *recursive* if and only if its graph

$$\big\{\big(m_1, m_2, \ldots, m_k, F(m_1, m_2, \ldots, m_k)\big) \in \mathbb{N}^{k+1} : m_1, m_2, \ldots, m_k \in \mathbb{N}\big\}$$

is r.e. It is natural to ask why one does not define recursive functions as those functions with *recursive* graphs. The answer is that the two definitions are actually equivalent, as the next proposition shows.

Proposition 13.1. Graphs of recursive functions $F \colon \mathbb{N}^k \to \mathbb{N}$ are recursive.

Proof. Use the recursiveness of F to find a Σ_1 formula $\theta(x_1, x_2, \ldots, x_k, y)$ which defines the graph of F in $\mathbb{N}$. Since F is a function,

$$\mathbb{N} \models \forall \bar{x}\ \exists y\ \theta(\bar{x}, y) \land \forall \bar{x}, y, z\ \big(\theta(\bar{x}, y) \land \theta(\bar{x}, z) \to y = z\big).$$

So $\mathbb{N} \models \forall \bar{x}, y\ \big(\theta(\bar{x}, y) \leftrightarrow \forall z\ (\theta(\bar{x}, y) \to z = y)\big)$: the $\to$ direction follows from the second conjunct, whereas the $\leftarrow$ direction follows from the first conjunct. The right-hand side of this equivalence is equivalent to a Π_1 formula by Lemma 5.2(2). Thus the graph of F is Π_1-definable. $\square$

Recursive functions are nice to work with (partly) because they are represented by simple formulas in arithmetic. Recall from Mostowski, R. Robinson, and Tarski (1953) that an $\mathcal{L}_A(\exp)$ formula $\rho(x_1, x_2, \ldots, x_k, y)$ *represents* $F \colon \mathbb{N}^k \to \mathbb{N}$ over an $\mathcal{L}_A(\exp)$ theory T if for all $m_1, m_2, \ldots, m_k \in \mathbb{N}$,

$$T \vdash \rho(\underline{\bar{m}}, \underline{F(\bar{m})}) \land \forall y\ \big(\rho(\underline{\bar{m}}, y) \to y = \underline{F(\bar{m})}\big).$$

As we saw in Theorem 4.4, all recursive functions are represented by a Σ_1 formula over R(exp). One can directly check that our proof using witness comparison invoked only $\mathrm{R} \coloneqq \mathrm{R} \setminus (\mathrm{Rexp})$. This fact was first proved by Mostowski, R. Robinson, and Tarski (1953). They also showed that all recursive sets are represented by a Σ_1 formula over R. Let us state this only for R(exp) here.

Definition (Mostowski–R. Robinson–Tarski 1953)**.** Let $k \in \mathbb{N}$ and $S \subseteq \mathbb{N}^k$. An $\mathcal{L}_A(\exp)$ formula $\rho(x_1, x_2, \ldots, x_k)$ *represents* S over an $\mathcal{L}_A(\exp)$ theory T if for all $m_1, m_2, \ldots, m_k \in \mathbb{N}$,

- $(m_1, m_2, \ldots, m_k) \in S$ implies $T \vdash \rho(\underline{\bar{m}})$; and

- $(m_1, m_2, \ldots, m_k) \notin S$ implies $T \vdash \neg\rho(\underline{\bar{m}})$.

Theorem 13.2 (Mostowski–R. Robinson–Tarski 1953)**.** Every recursive set S is represented by a Σ_1 formula over R(exp).

Proof. Let $\chi_S\colon \mathbb{N}^k \to \mathbb{N}$ be the characteristic function of S, i.e., for all $m_1, m_2, \ldots, m_k \in \mathbb{N}$,

$$\chi_S(m_1, m_2, \ldots, m_k) = \begin{cases} 0, & \text{if } (m_1, m_2, \ldots, m_k) \notin S; \\ 1, & \text{if } (m_1, m_2, \ldots, m_k) \in S. \end{cases}$$

Since S is recursive, one can algorithmically check whether a tuple in $\mathbb{N}^k$ is in S or not. So the following describes an algorithm which checks whether a given input $(m_1, m_2, \ldots, m_k, n) \in \mathbb{N}^k$ is in the graph of χ_S or not:

if $(m_1, m_2, \ldots, m_k) \in S$
  then if $n = 1$ **then return** `true` **end if**
  else if $n = 0$ **then return** `true` **end if**
end if
return `false`

It follows that χ_S is recursive by the Church–Turing Thesis. Apply Theorem 4.4 to find $\rho(\bar{x}, y) \in \Sigma_1$ which represents χ_S over R(exp). We verify that $\rho(\bar{x}, \underline{1})$ represents S over R(exp).

If $(m_1, m_2, \ldots, m_k) \in S$, then $\chi_S(\bar{m}) = 1$ and hence R(exp) $\vdash \rho(\underline{\bar{m}}, \underline{1})$ since ρ represents χ_S over R(exp). So suppose $(m_1, m_2, \ldots, m_k) \in \mathbb{N}^k \setminus S$. Then $\chi_S(\bar{m}) = 0$ and hence

$$\text{R(exp)} \vdash \forall y \ \big(\rho(\underline{\bar{m}}, y) \to y = \underline{0}\big)$$

because ρ represents χ_S over R(exp). Recall R(exp) $\vdash \underline{0} \neq \underline{1}$ by (R$\neq$). Combining the two using ($\forall$R) and (MT), we deduce that R(exp) $\vdash \neg\rho(\underline{\bar{m}}, \underline{1})$, as required. ☐

Assignment 13.3. Fix $k \in \mathbb{N}$ and a recursive subset $S \subseteq \mathbb{N}^k$. Let $\alpha(x_1, x_2, \ldots, x_k, z)$ and $\alpha'(x_1, x_2, \ldots, x_k, z)$ be Δ_0(exp) formulas such that

$$S = \{(m_1, m_2, \ldots, m_k) \in \mathbb{N}^k : \mathbb{N} \models \exists z \ \alpha(\bar{m}, z)\} = \{(m_1, m_2, \ldots, m_k) \in \mathbb{N}^k : \mathbb{N} \models \neg\exists z' \ \alpha'(\bar{m}, z')\}.$$

Prove that the Σ_1 formula

$$\rho(\bar{x}) \quad := \quad \exists z \ \big(\alpha(\bar{x}, z) \land \forall z'{<}z \ \neg\alpha'(\bar{x}, z')\big)$$

represents S over R(exp). (Hint: the proof of Theorem 4.4 and the proof of the First Incompleteness Theorem in Lecture 8 have a similar flavour.) [5 points]

Arithmetic can talk about syntactic objects via their Gödel numbers. In our system of Gödel numbering (from Lecture 5, page 18), we view each string of symbols in $\mathcal{L}_{\mathrm{A}}$(exp) as a number written in hexadecimal representation via Table 8.1. Gödel's original system (1931) is based on the unique factorization of a number into a product of primes: code the string $m_1 m_2 \ldots m_\ell$ where $m_1, m_2, \ldots, m_\ell \in \mathbb{N}$ by the natural number

$$2^{m_1} 3^{m_2} 5^{m_3} \cdots {p_\ell}^{m_\ell}$$

where p_ℓ denotes the ℓth prime number. Note that his system allows an infinite conversion table, but our system does not.

With Gödel numbering, every $\mathcal{L}_{\mathrm{A}}$(exp) formula $\varphi(y)$ can be viewed as a function on the set of all $\mathcal{L}_{\mathrm{A}}$(exp) sentences mapping $\sigma \mapsto \varphi(\underline{\sigma})$. The Diagonal Lemma (from Lecture 5, page 19) says that the graph of such a function must intersect the diagonal

$$\{(\sigma, \sigma) : \sigma \text{ is an } \mathcal{L}_{\mathrm{A}}\text{(exp) sentence}\}$$

nontrivially, up to provable equivalence. This intersection represents the fixed points of the function $\sigma \mapsto \varphi(\underline{\sigma})$, hence the alternative name *Fixed-Point Theorem.* The Diagonal Lemma guarantees the existence of self-referential sentences. Roughly speaking, it tells us that for every $\mathcal{L}_\mathrm{A}(\exp)$ property φ, there is an $\mathcal{L}_\mathrm{A}(\exp)$ sentence which expresses

this sentence has property φ.

The proof involves a clever use of the diagonization function, which maps each $\mathcal{L}_\mathrm{A}(\exp)$ formula $\theta(x)$ to $\theta(\underline{\theta})$.

A straightforward consequence of the Diagonal Lemma is Tarski's theorem on the undefinability of truth (from Lecture 5, page 19), which says that the set of (Gödel numbers of) true sentences is not definable by a formula in any model of R(exp). The proof is an application of the Liar Paradox: if truth were definable, then the Diagonal Lemma would allow us to build the Liar Sentence. Since the Diagonal Lemma has a Π_1 variant, so does Tarski's theorem, as we saw in Assignment 5.3. This leads to a version of the First Incompleteness Theorem which appears frequently in the literature. Alternatively, one can use universal programs to prove this.

A weak form the of the First Incompleteness Theorem. The set

$$\Pi_1\text{-Th}(\mathbb{N}) := \{\sigma : \sigma \text{ is a } \Pi_1 \text{ sentence and } \mathbb{N} \models \sigma\}$$

is not recursive.

Proof. Assignment 5.3 applied to $M = \mathbb{N}$ tells us that Π_1-Th($\mathbb{N}$) is not r.e. □

Gödel's proof of his Incompleteness Theorem (1931) uses a sentence which expresses

This sentence is not provable.

This again originated from the Liar Paradox. To show that the negation of Gödel's sentence is not provable, he invoked a stronger consistency condition called ω-consistency.

Definition. An $\mathcal{L}_\mathrm{A}(\exp)$ theory T is *ω-consistent* if for every $\mathcal{L}_\mathrm{A}(\exp)$ formula $\theta(x)$,

$$\forall n{\in}\mathbb{N}\ \ T \vdash \theta(\underline{n}) \quad \Rightarrow \quad T + \forall x\ \theta(x) \nvdash \bot.$$

It is clear that all ω-consistent theories are consistent. As the answer to Question 13.8 below shows, the negation of Gödel's sentence may actually be provable if the theory involved is merely assumed to be consistent. In particular, there are consistent theories that fail to be ω-consistent. Rosser (1936) saw how to reduce ω-consistency to plain consistency using the witness comparison trick: instead of Gödel's sentence, use a sentence which expresses

If there is a proof of this sentence, then there is a smaller proof of the negation of this sentence.

Gödel's version of the First Incompleteness Theorem (1931) was stated for axiomatic systems extending Peano's axioms for arithmetic (and related systems). Raphael Robinson (1950) reduced Peano's axioms here to $\mathrm{Q} := \mathrm{Q}(\exp) \setminus \{(\mathrm{Qexp}_0), (\mathrm{Qexp}_1)\}$ and $\mathrm{R} := \mathrm{R}(\exp) \setminus (\mathrm{Rexp})$. This was further improved by Mostowksi, R. Robinson, and Tarski (1953) to all consistent theories over which all recursive functions are represented. Therefore, the representability of recursive functions is *the* key property which makes a theory incomplete in the First Incompleteness Theorem. As shown by Jeřábek ($\geqslant$ 2018), this improvement by Mostowksi, Robinson, and Tarski is strict, in the sense that there are theories strictly weaker than R (and hence Q) over which all recursive functions are represented. The aforementioned result by Mostowksi, Robinson, and Tarski is often stated in terms of decidability, but it is not hard to see the connections with (in)completeness.

Definition. An $\mathcal{L}_\mathrm{A}(\exp)$ theory T is *decidable* if

$$\{\sigma : \sigma \text{ is an } \mathcal{L}_\mathrm{A}(\exp) \text{ sentence and } T \vdash \sigma\}$$

is recursive.

It does not matter whether we use formulas or sentences in the definition of decidability because for every formula $\theta(\bar{x})$,

$$T \vdash \theta(\bar{x}) \quad \Leftrightarrow \quad T \vdash \forall \bar{x}\ \theta(\bar{x}),$$

as the reader can readily verify using the fact that no variable appears free in the theory T.

Proposition 13.4. All complete recursive $\mathcal{L}_A(\exp)$ theories T are decidable.

Proof. If $T \vdash \bot$, then T proves all $\mathcal{L}_A(\exp)$ sentences by (RAA′), and thus T is decidable because one can check algorithmically whether a natural number is the Gödel number of an $\mathcal{L}_A(\exp)$ sentence or not. So suppose $T \nvdash \bot$.

In view of the Church–Turing Thesis, it suffices to produce an algorithm which checks whether or not a given input $n \in \mathbb{N}$ is the Gödel number of an $\mathcal{L}_A(\exp)$ sentence σ satisfying $T \vdash \sigma$. The following is a description of such an algorithm $\mathbb{A}$. Step (3) below is based on Assignment 8.2, which gives an algorithm for checking provability in T.

(1) If n is not the Gödel number of an $\mathcal{L}_A(\exp)$ sentence, then return `false` and stop.

(2) Let $n = \ulcorner\sigma\urcorner$, where σ is an $\mathcal{L}_A(\exp)$ sentence.

(3) Run the two subalgorithms below in parallel.

   (a) If $T \vdash \sigma$, then return `true`.

   (b) If $T \vdash \neg\sigma$, then return `false`.

To verify that $\mathbb{A}$ works as claimed, pick any $\mathcal{L}_A(\exp)$ sentence σ. Since T is complete, either $T \vdash \sigma$ or $T \vdash \neg\sigma$. So $\mathbb{A}$ must return either `true` or `false`. If $\mathbb{A}$ returns `true` on input $\ulcorner\sigma\urcorner$, then clearly $T \vdash \sigma$. If $\mathbb{A}$ returns `false` on input $\ulcorner\sigma\urcorner$, then $T \vdash \neg\sigma$ and so $T \nvdash \sigma$ by ($\bot$) because $T \nvdash \bot$. □

The sharp-eyed reader may have noticed that Assignment 8.2 and hence Proposition 13.4 remain true when one relaxes the recursiveness condition on the theory to r.e.-ness. This is a particular case of a more general phenomenon. We skip the proof, although it is not long.

Craig's Trick (1953)**.** Every r.e. set of $\mathcal{L}_A(\exp)$ formulas is equivalent to a recursive set of $\mathcal{L}_A(\exp)$ formulas. □

In view of bounded $\Delta_0(\exp)$ comprehension (from Lecture 11, pages 44f.), we know $I\Delta_0(\exp)$ can carry out constructions that

(A) are defined by $\Delta_0(\exp)$ formulas; and

(B) do not involve objects $g(x)$ that grow faster than $\exp^{(n)}(x)$ for all $n \in \mathbb{N}$.

In fact, these are the *only* constructions $I\Delta_0(\exp)$ can carry out.

Theorem 13.5 (essentially Parikh 1971)**.** Suppose $I\Delta_0(\exp) \vdash \forall x\, \exists y\, \theta(x, y)$, where $\theta \in \Delta_0(\exp)$. Then there is $n \in \mathbb{N}$ such that

$$I\Delta_0(\exp) \vdash \forall x\, \exists y{<}\exp^{(n)}(x)\ \theta(x, y). \qquad \square$$

An important kind of objects one can define in $I\Delta_0(\exp)$ are sequences. (See our definition of comma-separated lists in Lecture 10.) Sequences are key components of universal programs/formulas. The coding of sequences using natural numbers goes back to Gödel's original paper (1931). Traditionally, the formula used to code sequences is called β. So the lemma asserting the existence of such a formula is usually called the β Lemma. Here we state the β Lemma for $I\Delta_0$, which is essentially $I\Delta_0(\exp)$ minus all the sentences that mention the symbol exp. The formula $(s)_i = x$ is intended to mean 'the ith element in the sequence coded by s is x'.

Gödel's β Lemma. There is a Δ_0 formula $(s)_i = x$ such that $I\Delta_0$ proves

(1) $\forall s, i\ \exists! x\ (s)_i = x$;

(2) $\forall s, i, x\ \big((s)_i = x \to x \leqslant s\big)$; and

(3) $\forall s, \ell, x\ \exists s'\ \big(\forall i{<}\ell\ (s')_i = (s)_i \wedge (s')_\ell = x\big)$. □

Gödel proved his β Lemma using the Chinese Remainder Theorem. By tweaking this well-known proof in a skillful way, Jeřábek (2012) managed to find a (not necessarily Δ_0) formula $(s)_i = x$ such that the purely algebraic theory PA^- from Lemma 11.1 proves (1) and (3) in the β Lemma. This is surprising because PA^- contains no induction axiom at all. As Visser (2008) showed, no such formula exists when PA^- is weakened to the theory Q in Proposition 4.5.

In a rather standard way, one can make recursive definitions using sequences. Every recursively defined object is built up from certain initial objects using finitely many applications of certain construction rules. For example,

$$\mathcal{L}_{\mathrm{A}}(\exp) \text{ terms} = \underbrace{\mathsf{v}_i \mid 0 \mid 1}_{\text{initial objects}} \mid \underbrace{t+s \mid t\times s \mid \exp(t)}_{\text{construction rules}}.$$

Therefore, the formula $\mathrm{term}(t)$ expressing 't is a term' has the form

$$\exists \text{ construction sequence } s \ \ \exists \ell \left(\begin{array}{l} \forall i{\leqslant}\ell \left(\begin{array}{l} (s)_i \text{ is an initial object} \\ \vee (s)_i \text{ is obtained from } (s)_0, (s)_1, \dots, (s)_{i-1} \text{ using a construction rule} \end{array} \right) \\ \wedge (s)_\ell = t \end{array} \right).$$

The key properties of a class of recursively defined objects are that it contains all the initial objects, and that it is closed under all the construction rules. The key properties of $\mathrm{term}(t)$ are thus:

- $\forall i \ \mathrm{term}(\mathsf{v}_i) \wedge \mathrm{term}(0) \wedge \mathrm{term}(1)$;
- $\forall t_1, t_2 \ \big(\mathrm{term}(t_1) \wedge \mathrm{term}(t_2) \to \mathrm{term}(t_1 + t_2)\big)$;
- $\forall t_1, t_2 \ \big(\mathrm{term}(t_1) \wedge \mathrm{term}(t_2) \to \mathrm{term}(t_1 \times t_2)\big)$; and
- $\forall t \ \big(\mathrm{term}(t) \to \mathrm{term}(\exp(t))\big)$.

All these properties about $\mathrm{term}(t)$ are provable in $\mathrm{I}\Delta_0(\exp)$. Notice that the variable s in the formula $\mathrm{term}(t)$ can be bounded above by $2^{p(\log(t))}$ where $p(X) \in \mathbb{N}[X]$ because the length of the sequence coded by t is roughly $\log(t)$. So $\mathrm{term}(t)$ is equivalent to a $\Delta_0(\exp)$ formula in $\mathrm{I}\Delta_0(\exp)$. The full strength of $\mathrm{I}\Delta_0(\exp)$ is not needed here because only one layer of exponential appears.

In our definition of the formula $\Box(y)$ on page 40, we used an arbitrary Σ_1 formula that defines in $\mathbb{N}$ the theory T in question. However, some of these formulas may not behave well, as the example below shows.

Example 13.6. Let $\delta(x)$ be a Σ_1 formula that defines a consistent $\mathcal{L}_{\mathrm{A}}(\exp)$ theory $T \supseteq \mathrm{I}\Delta_0(\exp)$ in $\mathbb{N}$. Notice $\mathbb{N} \models \neg\Box_T\bot$ because T is consistent. Thus $\delta'(x) := \delta(x) \vee \Box_T\bot$ also defines T in $\mathbb{N}$. Note $T \nvdash \neg\Box_T\bot$ by the Second Incompleteness Theorem. So the Completeness Theorem gives $M \models T + \Box_T\bot$. If one adopts $\delta'(x)$ as the definition of T in M, then *every* $\mathcal{L}_{\mathrm{A}}(\exp)$ formula is an element of T in the sense of M because $M \models \forall x\ \delta'(x)$.

The undesirable situation in the example above cannot happen if the defining formula $\delta'(x)$ *represents* T in T. Therefore, in the definition of the formula $\Box_T(y)$, it is better to use a Σ_1 formula that represents T in T, rather than one that merely defines T in $\mathbb{N}$. Carefully crafted provability predicates have positive applications too. Compare the following with the Second Incompleteness Theorem on page 52.

Theorem 13.7 (Pudlák 1985)**.** For every finite $\mathcal{L}_{\mathrm{A}}(\exp)$ theory $T \vdash \mathrm{PA}^-$, there is an $\mathcal{L}_{\mathrm{A}}(\exp)$ formula $\Box(y)$ such that $T \vdash \neg\Box\bot$ and

$$\{a \in \mathbb{N} : \mathbb{N} \models \Box(a)\} = \{\ulcorner\theta\urcorner : \theta \text{ is an } \mathcal{L}_{\mathrm{A}}(\exp) \text{ formula and } T \vdash \theta\}. \qquad \Box$$

The reason why the theorem above does not contradict the Incompleteness Theorems on page 33 is that Pudlák's $\Box$ does not satisfy Löb's derivability conditions over T. As we saw in Lecture 12, the main step in the proof of Löb's conditions is to show the Σ_1 completeness of T in T to get (N) and (IN). This is in stark contrast to the situation in the First Incompleteness Theorem, where

Mostowksi, Robinson and Tarski (1953) tell us that Σ_1 completeness is not key, although the Second Incompleteness Theorem can be viewed as a formalization of the First Incompleteness Theorem. The ($\Box$D) condition does not play an important role here because, as shown by Jeroslow (1973), Gödel's Second Incompleteness Theorem remains true when one omits ($\Box$D).

The following notation is common in the literature.

Notation. Let T be a recursive $\mathcal{L}_\mathrm{A}(\exp)$ theory, and $\Box_T$ be a provability predicate for T with respect to a fixed Σ_1 formula representing T in $\mathrm{R}(\exp)$. Then one often writes $\neg\Box_T\bot$ as $\mathrm{Con}(T)$ or Con_T, although T is strictly speaking not a number.

The Second Incompleteness Theorem asserts the unprovability of consistency. It is worth noting that it does not say anything about the unprovability of *in*consistency.

Question 13.8. Is there a recursive consistent $\mathcal{L}_\mathrm{A}(\exp)$ theory $T \supseteq \mathrm{I}\Delta_0(\exp)$ such that $T \vdash \neg\mathrm{Con}(T)$?

Answer. Yes: take $T = \mathrm{I}\Delta_0(\exp) + \neg\mathrm{Con}(\mathrm{I}\Delta_0(\exp))$.

- T is consistent by the Second Incompleteness Theorem.
- T proves $\neg\mathrm{Con}(\mathrm{I}\Delta_0(\exp))$ and thus also $\neg\mathrm{Con}(T)$ because $T \supseteq \mathrm{I}\Delta_0(\exp)$. $\Box$

It is also worth noting that the Incompleteness Theorems can be adapted in a straightforward way to apply to any theory (e.g., set theory) that *interprets* sufficiently strong arithmetic. We do not include a precise definition of interpretations here because this notion is intuitive but rather technical when defined precisely. For example, most set theories can define arithmetic via ω.

One the one hand, our weak form of the First Incompleteness Theorem on page 56 tells us that the set $\mathrm{Th}(\mathbb{N}, 0, 1, +, \times, \exp, <)$ of all $\mathcal{L}_\mathrm{A}(\exp)$ sentences true in $\mathbb{N}$ is not recursive. This implies

$$\mathrm{Th}(\mathbb{N}, +, \times, <) \quad \text{and} \quad \mathrm{Th}(\mathbb{Q}, +, \times)$$

are not recursive because $\exp\colon x \mapsto 2^x$ is definable in $(\mathbb{N}, +, \times)$ and $\mathbb{N}$ is definable in $(\mathbb{Q}, +, \times)$. The $\mathrm{Th}(\mathbb{N}, +, \times, <)$ case was already shown by Gödel (1931), while the $\mathrm{Th}(\mathbb{Q}, +, \times)$ case was proved by J. Robinson (1949).

On the other hand, the following theories are all recursive:

$$\mathrm{Th}(\mathbb{N}, +), \quad \mathrm{Th}(\mathbb{N}, \times), \quad \mathrm{Th}(\mathbb{R}, +, \times, <), \quad \text{and} \quad \mathrm{Th}(\mathbb{C}, +, \times),$$

as proved by Presburger (1929), Skolem (1930), Tarski (1948), and Tarski (1948) respectively. The third and the fourth examples may be counter-intuitive at first glance because it is generally felt that $\mathbb{R}$ and $\mathbb{C}$ are more complicated than $\mathbb{N}$. Nevertheless, the first-order language of (ordered) rings fails to capture this complexity. The next two lectures will be devoted to Presburger's result that $\mathrm{Th}(\mathbb{N}, +)$ is recursive.

We conclude this lecture with a paradox in natural languages which is analogous to the original proof of Löb's theorem (on page 34).

Henkin–Löb Paradox. Let $\spadesuit$ be any sentence. We show that $\spadesuit$ is true.

Let σ be the sentence $\sigma \to \spadesuit$, i.e.,

if this sentence is true, then $\spadesuit$ is true.

Assume σ is true. Then

	$\sigma \to \spadesuit$ is true	by the choice of σ;
$\therefore$	$\spadesuit$ is true	by *modus ponens* and the assumption.

The previous paragraph shows $\sigma \to \spadesuit$ is true. Thus

	σ is true	by the choice of σ;
$\therefore$	$\spadesuit$ is true	by *modus ponens*, since both $\sigma \to \spadesuit$ and σ are true. $\Box$

Arithmetic is unable to produce such a self-referential sentence because it involves the notion of truth, which we know is not definable by Tarski's theorem.