RECURSION THEORY XI

13 January 2022

UNESCO WORLD LOGIC DAY
#WLD 2022
14 JANUARY 2022

LECTURE XII
Wednesday
19 Jan 2022
14–16

Two improvements on Lecture X:

1. Proof that the function S is not a computable growth fn
2. Application of the Rec. Thm. for constant fns

Theorem There is no total computable f s.t. $\gamma_f \geq S$.

$$S : \mathbb{N} \longrightarrow \mathbb{N}$$
$$n \longmapsto \max\{i ; \exists P \ |P| \leq n \ \& \ f_P(\varepsilon)\downarrow \text{ after exactly } i \text{ steps}\}$$

Proof of Thm. We are going to show: if such f exists, then the halting problem is computable. $\longrightarrow$ Contradiction!

As yesterday, we have a computable operation

$$w \longmapsto P_w$$

s.t. $f_{P_w}(\varepsilon) = f_w(w)$

We're going to describe a decision algorithm for $\mathbb{K}$, using our computable function f with $\gamma_f > S$.

Given $w \in \Sigma^*$. First produce P_w.

Calculate $\gamma_f(|P_w|)$:

There are finitely many v s.t. $|v| \leq |P_w|$. Run f on all of them. Since f was total, we get $f(v)$ for each of them. Then $\gamma_f(|P_w|) = \max\{|f(v)|;\ v \text{ is one of them}\}$.

By assumption, we know that

$$\boxed{N_w := \gamma_f(|P_w|) \geq S(|P_w|).} \quad (*)$$

Now run f_{P_w} on input ε for N_w many steps.

Case 1. This has halted
$\longrightarrow$ Output YES!

Case 2. It hasn't.
$\longrightarrow$ Output NO!

Claim This is a correct decision algorithm for $\mathbb{K}$ (i.e., YES $\Longleftrightarrow w \in \mathbb{K}$).

If we're in Case 1, $f_{P_w}(\varepsilon)$ has halted,
$$f_{P_w}(\varepsilon) = f_w(w)$$
so $w \in \mathbb{K}$.

If we're in Case 2, then $f_{P_w}(\varepsilon)$ has not halted before step N_w.

Towards a contradiction, assume $w \in \mathbb{K}$,
i.e., $f_w(w)\downarrow \Longleftrightarrow f_{P_w}(\varepsilon)\downarrow$.

But then $f_{P_w}(\varepsilon)$ halts before $S(|P_w|) \leq N_w$

So, we're not in Case 2. Contradiction,
$\Longrightarrow w \notin \mathbb{K}$.

q.e.d.

2. Second improvement:
constant functions.

Remember the application for Rec. Thm. that got P with $W_P = \{P\}$.

Another application:

$$g(w, v) = w.$$

Clearly computable, by s-m-n h total computable s.t.

$$\boxed{f_{h(w)}(v) = g(w, v) = w.}$$

By the Rec. Thm., we get P s.t.

$$f_{h(P)} = f_P.$$

$f_{h(w)}$ is the constant function with value w.

$f_{h(P)}$ is the constant function with value P

get P s.t. f_P is the constant function with value P.

GÖDEL'S INCOMPLETENESS THM

Usually phrased as:

If PA is Peano arithmetic, then
if PA is consistent, then there are
true sentences φ s.t. $PA \nvdash \varphi$.

Generalised to:

If $T \supseteq PA$ is any computable
theory that is consistent, then
there are true sentences φ s.t.
$T \nvdash \varphi$.

ESSENTIAL INCOMPLETENESS

! What does it mean for a theory to be
computable?

Standard idea of a proof:

Consider the statement
"this sentence is not provable" φ

There is an informal argument for the truth of such
a sentence:
If φ is false, then "φ is provable"
is true, but false sentences cannot be
provable.
So φ is true.

How do we even formalize something like "this sentence"

Ordinary first-order logic does not have reference to sentences.

The classical Gödel proof will have to overcome these technical obstacles.

We're going a different route:

Sometimes, logicians say that Gödel's Incompleteness Theorem is just

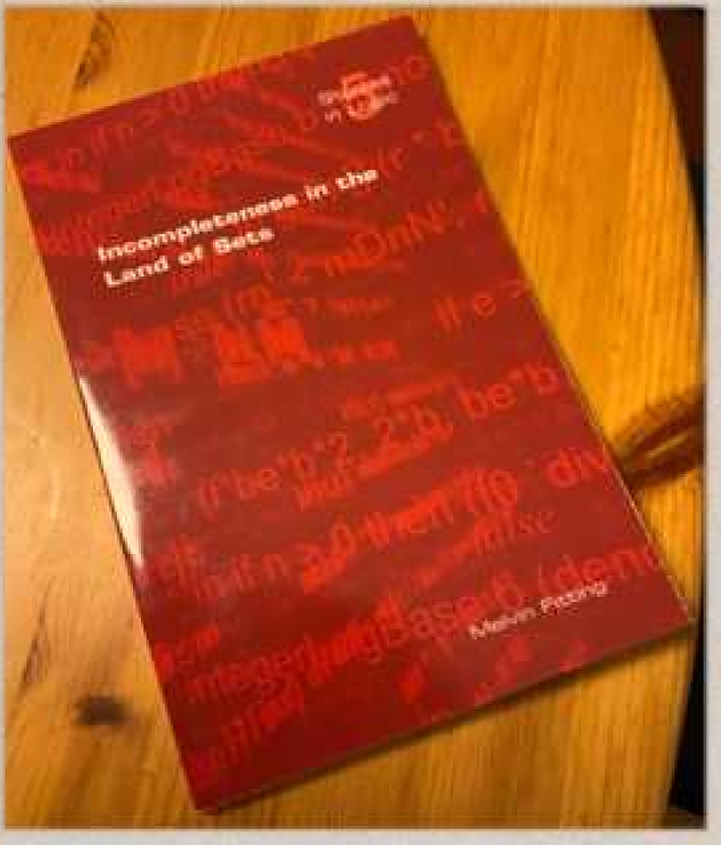
$$\Sigma_1 \neq \Pi_1.$$

[This is a theorem we already proved.]

What's its relationship to incompleteness?

ANSWER: We'll show that the set of provable formulas is Σ_1, but the set of true formulas is Π_1-hard.

Thus, since $\Sigma_1 \neq \Pi_1$, these two sets cannot be the same.



MELVIN FITTING,
Incompleteness in the
Land of Sets.

We are going to prove this not for PA, but for some simple set theory.

Reason "Provability" talks about sequence of sequents, whose sequents are sequences of formulas and formulas are sequences of symbol.

So, the objects we're interested in (proofs) are sequences of sequences of sequences of symbols. All of this has to be encoded in arithmetic.

In set theory, these sequences are objects in their own right.

Some basic set theory

The finite levels of the von Neumann hierarchy:

$$V_0 := \emptyset$$

$$V_{n+1} := \mathcal{P}(V_n)$$

$$HF := \bigcup_{n\in\mathbb{N}} V_n$$

A set X is called <u>transitive</u> if for all z, y $z\in y\in X \rightarrow z\in X$ [Equivalently, if $y\in X \rightarrow y\subseteq X$]

HEREDITARILY FINITE
This means finite and all elements and iterated elements are finite.

<u>Properties</u> ① Every V_n is transitive.

[Proof by induction. V_0 trivial. But transitivity is preserved by power set:

X is transitive $\Longrightarrow \mathcal{P}(X)$ transitive

$$z\in y\in \mathcal{P}(X)$$
$$\Longrightarrow y\subseteq X$$
$$\Longrightarrow z\in X$$

By transitivity of X, $z\subseteq X$, and thus $z\in \mathcal{P}(X)$.]

② HF is transitive as a union of transitive sets.

③ $n \leq m \implies V_n \subseteq V_m$

[Once more by induction. Only need to show that $V_n \subseteq V_{n+1}$. That's just transitivity of V_n: if $x \in V_n$, $x \subseteq V_n$, thus $x \in \mathcal{P}(V_n) = V_{n+1}$.]

④ All elements of HF are finite.

⑤ HF is countable.

⑥ $V_n \cap \mathbb{N} = n$.

[Proof by induction.

$V_0 \cap \mathbb{N} = \emptyset \cap \mathbb{N} = \emptyset = 0$.

Suppose $V_n \cap \mathbb{N} = n$. (*)

$\implies n \subseteq V_n$, so $n \in \mathcal{P}(V_n) = V_{n+1}$.

$n+1 = n \cup \{n\} \subseteq V_{n+1}$.

This leaves to show that $n+1 \notin V_{n+1}$.

Towards a contradiction, suppose it is:

$n+1 \in V_{n+1} = \mathcal{P}(V_n)$

$n \in n+1 \subseteq V_n$

$\implies n \in V_n$. Contradiction! to (*)]

(7) $\mathbb{N} \subseteq HF$.

Consider $(HF, \in)$ and consider which axioms of set theory hold in this structure, we observe:

FST

FINITE SET THEORY

- Extensionality
- Union
- Power set:

$x \in V_n$ and $A \subseteq x$

$\Downarrow$ $\qquad\qquad$ $\Downarrow$

$x \subseteq V_n$ $\qquad$ $A \subseteq V_n \Rightarrow A \in V_{n+1}$.

$\Longrightarrow \wp(x) \subseteq V_{n+1}$

$\Longrightarrow \wp(x) \in V_{n+2} = \wp(V_{n+1})$.

- Pairing: $x, y \in V_n$, $\{x,y\} \subseteq V_n$

$\Longrightarrow \{x,y\} \in V_{n+1}$.

- Separation scheme

$x \in V_n$ φ formula

$\{z \in x; \varphi(z)\} \subseteq V_n$

each of these is in V_n $\Longrightarrow$

$\{z \in x; \varphi(x)\} \in V_{n+1}$.

As in the lecture course MLML, we know that in models of FST, the operations

$$X, Y \longmapsto X \times Y$$

$$X, Y \longmapsto \{R \,;\, R \text{ is a relation between } X \text{ and } Y\} = \mathcal{P}(X \times Y)$$

$$X, Y \longmapsto \{f \,;\, f \text{ is a function from } X \text{ to } Y\}$$

well-defined.

Therefore: if $x, y \in HF$, then the set of functions from x to y is in HF.

Since for each $n \in \mathbb{N}$, $n \in HF$, this means that the set of n-tuples of x, i.e., x^n is in HF for every $x \in HF$.

If I form $x^* := \bigcup_{n \in \mathbb{N}} x^n \subseteq HF$.

<u>Reminder</u> FST does not imply "everything is finite". In particular,

$$Z = FST + Inf$$

[where Inf is the axiom of infinity] and Z refutes "everything is finite".

In HF, everything <u>is</u> finite, so we'd like to have some expression that states this:

$$\text{Finite} = \forall x \,(x \text{ is an ordinal, then } x = \emptyset \text{ or } \exists y \; x = y \cup \{y\})$$

$$HFST := FST + \text{Finite}$$

and

$$(HF, \in) \models HFST.$$

<u>Note</u> HFST cannot really describe finiteness in the same sense in which first order theories with infinite models can never do that.

In HFST, we can now talk about computation. If Σ is a finite alphabet, identify the elements of Σ with elements of HF:

$$k : \Sigma \longrightarrow HF \quad \text{injection.}$$

Then $\mathrm{ran}(k) \in HF$ and we can identify Σ^* with $(\mathrm{ran}(k))^* = \bigcup_{n \in \mathbb{N}} (\mathrm{ran}(k))^n$.

Thus Σ^* is a subset of HF (via k).

We're augmenting our set theoretic language LST with countably many constant symbols that represent the elements of HF:

$x \in HF$, introduce constant symbol c_x

We need to add axioms that guarantee that c_x really behaves like x, i.e., c_x can only be interpreted by x.

We wish to have formula φ_x s.t. $\boxed{\varphi_x(z) \iff z = x}$ (*)

Then we add the axioms

$$\{\varphi_x(c_x) \,;\, x \in HF\}$$

Then $\underline{AHFST} := HFST \cup \{\varphi_x(c_x) \,;\, x \in HF\}$

AUGMENTED HEREDITARILY FINITE SET THEORY

Why does φ_x exist?

By the structure of HF, we can define formulas by recursion:

$$\varphi_\emptyset(z) :\iff \forall y (y \notin z)$$

Suppose for all elements $x \in V_n$, there is a formula φ_x satisfying (*).

Let $y \in V_{n+1} = \mathcal{P}(V_n)$

$\implies y \subseteq V_n$

$\quad \quad \quad$ "

$\{x_1, \dots, x_k\}$ with $x_i \in V_n$

so by assumption, we have fmlas φ_{x_i} for $1 \leq i \leq k$.

$$\varphi_y(z) :\iff \forall v \left(v \in z \iff \varphi_{x_1}(v) \vee \dots \vee \varphi_{x_k}(v) \right)$$

Then $\varphi_y(z) \iff z = y$.

ASIDE. AHFST looks as if it describes HF precisely. But it has non-standard models: by the standard compactness argument.

Add another constant symbol c to the language and consider

$$T = \text{AHFST} + \{ \underline{\underline{c \neq c_x}} \,;\, x \in \text{HF} \}$$

Every finite subset of T is satisfiable: if $T_0 \subseteq T$ finite, there is some $x \in \text{HF}$ s.t. $c \neq c_x \notin T_0$. Thus, interpret c as x and c_y as y in HF, then this is a model of T_0. So, by compactness, T is satisfiable.

If $M \models T$, then there is a subset of M isomorphic to HF, but c is interpreted as something different.

Next goal: Define notions of formulas and provability in the setting of AHFST.

Fix Σ the alphabet used in computation and identify it with and element of HF. For notational simplicity, ignore the injection k and assume $\Sigma \in HF$, but $\Sigma^* \subseteq HF$.

LST : language of set theory

LST$^+$: augmented language of set theory with extra constant symbols c_x

Formulas in LST and LST$^+$ can be encoded in some reasonable way in Σ^*.

We'll assume that syntactic operations on formulas can be done computably on their codes in Σ^*.

$$\begin{array}{ccc} w & \longmapsto & \varphi_w \\ \cap & & \text{LST}^+\text{-formula} \\ \Sigma^* & & \\ \varphi & \longmapsto & k(\varphi) \in \Sigma^* \end{array}$$

with $k(\varphi_w) = w$

$$\varphi_{k(\varphi)} = \varphi$$

such that any syntactic operations on the formula $\varphi \longmapsto \psi$ correspond to a function $f : \Sigma^* \longrightarrow \Sigma^*$

$$\begin{array}{ccc} w & \longmapsto & k(\psi) \\ \downarrow & & \uparrow \\ \varphi & \longmapsto & \psi \end{array}$$

that is computable.

MAIN OBSERVATION

Once that all formulas of LST^+ are in HF, then so are sequences of formulas (sequents) and sequences of sequents ($\rightsquigarrow$ proofs).

Next lecture: Wednesday 19 January 2022, 14^{15}-15^{45}.