RECURSIION THEORY

LECTURE XIII

25 January 2022

LECTIO ULTIMA

Gödel's Incompleteness Theorem GIT

Informally: There are true sentences that are not provable.

LST^+ ⟶ augmented language of set theory

AHFST Augmented Hereditary Finite Set Theory

$(HF, \in, \{x; x \in HF\})$ ⟵ STANDARD MODEL

Write $\mathcal{HF}$

$\mathcal{HF} \models \text{AHFST}$

TRUE: true in $\mathcal{HF}$

PROVABLE: provable in AHFST

$\text{Prov} \subseteq \Sigma^*$ $\quad \text{Prov} := \{w;\ \text{AHFST} \vdash \varphi_w\}$

$\text{True} \subseteq \Sigma^*$ $\quad \text{True} := \{w;\ \mathcal{HF} \models \varphi_w\}$

lecture XII

GIT: $\text{Prov} \neq \text{True}$.

We're going to show that by showing

- Prov is Σ_1 ✓
- True is Π_1-hard.

Reconstruction of computability in LST^+

Going back to the beginning of the course, we remember how we defined computability:

$$M = (\Sigma, \Phi, h) \quad \text{model of computation}$$

$$P \in \Sigma^* \text{ program; } w \in \Sigma^* \text{ input}$$

Based on P, w, we defined the M-computation of P with input w.

$$C_M(P, w, i) \quad \text{where } i \in \mathbb{N}$$

Each time instance ($i \in \mathbb{N}$) is a configuration of the machine.

We said that the computation halts, if there is $i \in \mathbb{N}$ s.t.

$$C_M(P, w, i) \text{ is in the halting state}$$

This means that the statement

(*) "the M-computation of P with input w halts"

is true iff there is a finite sequence $C_0, \dots, C_n$ s.t. each C_i is a configuration

$$C_i = C_M(P, w, i)$$

and C_n is in the halting state.

If $P, w \in HF$, then there is some formula $\varphi \in LST^+$ s.t. $\varphi(P, w) \Leftrightarrow$ (*).

FROM LECTURE XII

Why are we doing this?

Want an arbitrary Π_1 set $A \subseteq \Sigma^*$ and a total computable function h s.t.

$$w \in A \Leftrightarrow h(w) \in \text{True}$$

$$\Updownarrow$$

$\forall v \; wv \in C$

(*) where C is a computable set.

Goal needs to be: Translate (*) into a formula of LST^+.

Fix program P and word w. Aim to formalize
"$f_P(w)$ halts and produces a nonempty output".

Try to write down an LST^+ sentence:

$\exists x$	x is a natural number $\wedge$
$\exists C$	C is a function with domain $x+1$ $\wedge$
	$C(0)$ has tape content $\boxed{w}$ $\wedge$
Φ	$C(x)$ is the first configuration in the halting state $\wedge$
	$C(x)$ has nonempty output $\wedge$
	C is a sequence of configuration according to $\boxed{P.}$

[Even though this is written in natural language, all of the terms showing up in this description have precise mathematical meaning expressible in the language of set theory.]

Q: Is Φ an LST^+ sentence?

No: P & w are free variables!

Idea: Replace P by c_P and w by c_w and obtain a sentence.

$\Phi^{0}_{P,w}$ will be an LST^+ – sentence:

$\exists x$ (x is a natural number $\wedge$

$\exists C$ (C is a function with $\mathrm{dom}(C) = x+1$ $\wedge$

$C(0)$ has tape content c_w $\wedge$

$C(x)$ is the first configuration in the halting state $\wedge$

$C(x)$ has output nonempty $\wedge$

C is a sequence of configuration according to c_P))

We obtain immediately:

$$\Phi^{0}_{P,w} \text{ is true} \iff \boxed{f_P(w)\downarrow \neq \varepsilon}$$

We recovered "computability" in LST^+.

We want a Π_1 statement

$\forall v \; f_P(w*v)\downarrow \neq \varepsilon$

So: add the extra quantifier.

$\Phi^1_{P,w}$ sentence in LST^+.

$\Phi^0_{P,w} \iff f_P(w)\downarrow \neq \varepsilon$

Want: $\forall v\, f_P(w * v)\downarrow \neq \varepsilon$

$$\forall z \left(z \in \Sigma^* \longrightarrow \exists y \left(y = c_w * z \wedge \Phi^0_{P,y} \right) \right)$$

This has the property:

$$\Phi^1_{P,w} \text{ is true} \iff \forall v\, f_P(w * v)\downarrow \neq \varepsilon.$$

(*)

<u>Remark</u> There is nothing special here about Π_1. If we wanted to do the same with Π_2, we could just add another quantifier and obtain a formula

$$\Phi^2_{P,w} \iff \forall v \exists u\, f_P((w * v) * u)\downarrow \neq \varepsilon.$$

And so on: $\Phi^n_{P,w}$ for Π_n-formulas.

Remark 2 (very important).

The process that produces $\Phi^0_{P,w}$ or $\Phi^1_{P,w}$ (or $\Phi^n_{P,w}$) is computable. I.e., if P is fixed, there is a computable function

$$h_P : \begin{array}{ccc} w & \longmapsto & \Phi^1_{P,w} \\ \Sigma^* & \longrightarrow & \Sigma^* \end{array}$$

interpreted as element of Σ^*

Theorem True $:= \{ w \,;\, \mathfrak{N} \models \varphi_w \}$ is Π_1-hard.

Proof. Let $A \subseteq \Sigma^*$ be Π_1. So there is $C \subseteq \Sigma^*$ computable s.t.

$$w \in A \iff \forall v \;\; w * v \in C.$$

Let P be a program for the characteristic function of C.

* [from last page]

$$w \in A \iff \underline{\forall v} \;\; f_P\underline{(w*v)\downarrow \neq \varepsilon}.$$

$$\iff h_P(w) = \Phi^1_{P,w} \text{ is true}$$

So, this means that

h_p is a computable reduction of A to True,

so $A \leq_m \text{True}$.

q.e.d.

<u>Remark</u> Since there was nothing special about Π_1 in the construction of the formulas $\Phi^1_{P,w}$ (and thus the reductions h_p), the same proof with $\Phi^n_{P,w}$ instead of $\Phi^1_{P,w}$ will show that True is Π_n-hard.

So, True is much more complex than anything we've seen in our analysis of degrees of unsolvability.

COROLLARY Gödel's First Incompleteness Theorem

If T is any computable set of LST^+-sentences such that

$$\mathfrak{HF} \models T,$$

then there are LST^+ sentences φ s.t.

$$\mathfrak{HF} \models \varphi$$

but

$$T \not\vdash \varphi.$$

[Metamathematical remark: We do not need to say that T is consistent since the assumption $\mathfrak{HF} \models T$ implies that.]

Emphasise that this is more than just saying "AHFST is not complete". As mentioned before, it's called ESSENTIAL INCOMPLETENESS.

Question & Remark

Already completeness / compactness gives us limitations on axiomatisation of the standard model:

If $\mathcal{HF} \models T$, then by compactness, there are models M s.t.

$$M \not\cong \mathcal{HF}$$

and $M \models T$.

[First-order theories with infinite models can never be categorical.]

<u>Remember</u>: Notions of ISOMORPHISM

$$M \cong N$$

and ELEMENTARY EQUIVALENCE

$$M \equiv N$$

[for all sentences φ

$$M \models \varphi \iff N \models \varphi \,]$$

<u>MLML</u>: Isomorphism lemma shows that

$$M \cong N \implies M \equiv N.$$

Converse is not true in general.

So: Completeness / Compactness gives for each theory T NONISOMORPHIC MODELS OF T

whereas Incompleteness gives NON-ELEMENTARILY EQUIVALENT MODELS OF T.

Gödel's Second Incompleteness Theorem

gives a concrete example of a true, but unprovable sentence.

Usually described as:

No system can prove its own consistency.

Using our description of provability, a formula $\Phi(n) \Longleftrightarrow T \vdash \varphi_n$.

fix some n_0 s.t. φ_{n_0} is inconsistent

[e.g., $0=1$]

then "T is consistent" is expressed by $\neg \Phi(n_0) \longleftarrow \mathrm{Cons}_T$

Then Gödel's Second Incompleteness Thm states:

If T is any computable set of LST^+-sentences s.t. $\mathfrak{H}\mathfrak{F} \models T$, then Cons_T is not provable from T.

MÜNDLICHE PRÜFUNG (entweder online oder nicht):
zwischen 22.3. und 1.4.