Part III Mathematics, 2011

The First Incompleteness Theorem

Peter Smith
Faculty of Philosophy, University of Cambridge

Version 3.1

Contents

Preface

These notes were written to accompany four introductory lectures given in Easter 2011 as a supplement to Thomas Forster's earlier Part III Maths course on Computable Function Theory. These notes fill in some of the missing detail. There was going to be a fifth lecture, on the Second Incompleteness Theorem. But in the event that didn't happen; see e.g. Episode 10 of my *Gödel Without Tears*, downloadable from my website at http://www.logicmatters.net, for something of what I would have said.

Please let me know about typos/thinkos (email ps218 at cam.ac.uk). And if you have not downloaded these notes directly from my website, then you can go there to get the latest version. Versions are numbered $n.m$; I'll increment n for major changes.

Peter Smith

Assumed background

What do you need to bring to the party if you are to join in the fun? What background am I assuming in the following lectures?

From the theory of computable functions For the first lecture, all you really need is the notion of a primitive recursive function. Officially, you'll recall, the p.r. functions[1] form the smallest class of numerical functions which includes the trivial 'initial' functions (the successor function, the zero function and the 'projection' functions), and which is closed under definition by composition and primitive recursion. Unofficially, the p.r. functions are just the numerical functions which are effectively computable *without* any unbounded searches (which need be implemented by 'do until' loops): i.e. 'for' loops are enough to compute them.

For Lecture 4, which ties things back to Thomas Forster's lectures on computability, we'll need additionally to invoke four familiar and elementary results.

(i) If a non-empty set S is recursively enumerable (r.e.), then then there is a p.r. function which enumerates it, i.e. there is a primitive recursive function f such that $n \in S$ iff $\exists x\, f(x) = n$.

(ii) There are r.e. sets of numbers whose complements are not r.e.

(iii) The Halting Problem for Turing Machines is recursively unsolvable.

(iv) Kleene's Normal Formal Theorem: there is a three-place p.r. function C and a one-place p.r. function U such that any one-place partial recursive function can be given in the standard form

$$\varphi_e(n) =_{\text{def}} U(\mu z[C(e, n, z) = 0])$$

[1] Careful! In these notes, 'p.r.' (as quite often) abbreviates 'primitive recursive' not (as perhaps more often) 'partial recursive'.

for some value of the index e (μ of course is the least-number operator).

From mathematical logic We need the idea of a formalized language $\mathcal{L}$ – in particular, the idea of a first-order language with the usual bog-standard equipment of quantifiers, variables, connectives and identity as logical apparatus and where all quantifiers run over the same given domain of interpretation.

A formalized theory T built in some formalized language $\mathcal{L}$ has a determinate set of axioms and a deductive system for deriving theorems from the axioms. For a properly formalized theory, it is required that it is effectively decidable whether a purported proof is indeed a correctly constructed T-proof which starts from T-axioms and proceeds via correct moves in T's built-in deductive system. Moreover, without loss of generality, we can take the effective procedure for checking proofs for a properly set-up theory to involve no unbounded searches. In a phrase, we are going to be interested in p.r. axiomatized theories.

Formal arithmetics The basic language of first-order arithmetic, $\mathcal{L}_A$, has the non-logical vocabulary $\{\mathsf{0}, <, \mathsf{S}, +, \times\}$ whose intended interpretation is obvious. $\mathcal{L}_A$'s *standard numerals* are formed by iterating applications of the successor function, so that e.g. '$\mathsf{SSSS0}$' is the standard numeral for four.

The canonical theory of arithmetic built in $\mathcal{L}_A$ is *first-order Peano Arithmetic*, PA, which has the usual axioms for successor, the recursion equations for addition and multiplication, plus all instances of the induction schema. But we should also note Q, *Robinson Arithmetic*, which lacks the induction axioms, so just has the successor, addition and multiplication axioms plus the axiom $\forall \mathsf{x}(\mathsf{x} = \mathsf{0} \vee \exists \mathsf{y}(\mathsf{x} = \mathsf{Sy}))$ which says that every number other than zero is a successor number.

We are going to be interested too in richer theories T which extend PA. We'll assume that, even if T's native language $\mathcal{L}$ doesn't include $\mathcal{L}_A$, we can define surrogates for arithmetical expressions in $\mathcal{L}$. Recall, for example, how we implement arithmetic in set theory: we define '$\mathsf{0}$' as '$\emptyset$', use 'Sx' to abbreviate '$\mathsf{x} \cup \{\mathsf{x}\}$', etc. And then – to replicate the effect of $\mathcal{L}_A$'s quantifiers – we need to use restricted quantifers as in '$(\forall \mathsf{x} \in \mathbb{N})\varphi$', where '$\mathbb{N}$' abbreviates the definition for the set of finite ordinals.

Coding The second key notion from the elementary theory of computation that we need right from the outset is the idea of using numbers as

codes for, or indices of, syntactic objects like programs (e.g. for the set of tuples which constitute a program for a Turing machine). Key results in the theory of computation then arise from the dual role that numbers can play, as inputs to functions and as codes or indices for programs-defining-functions, and we exploit the interplay between these roles in 'diagonalization' arguments, by feeding the number e as input to the function φ_e inexed by e

We'll take for granted, then, the idea of a scheme of *Gödel-numbering* which systematically associates expressions and sequences of expressions of a formal language $\mathcal{L}$ with code numbers. We will require coding and decoding to be effective procedures that require no unbounded searches, so that any two acceptable schemes are primitively recursively inter-translatable. Assume some such scheme is fixed in any given context, so we can then talk without further ado about *the* Gödel number for an expression.

1

A generalized incompleteness theorem, Gödel-style

In this lecture, we prove a generalized version of the First Incompleteness Theorem. The core proof-idea is from Gödel.[1] As you'll see, once the definitional set-up is over, our proof is *very* quick indeed: so I'll need to say something in the next lecture about why Gödel's own proof of the original version of the Theorem is significantly longer: what additional work is he doing? what additional information do we get from that hard work?

But first we need to fix notation and terminology, and pin down a number of ideas – some of these ideas will already be very familiar, others are straightforwardly defined in terms of familiar notions.[2]

---

[1] Kurt Gödel, the greatest logician of the twentieth century, was born in what is now Brno in 1906. Educated and doing his early work in Vienna, Gödel left Austria for the USA in 1938, and spent rest of his life at the Institute of Advanced Studies at Princeton. Always a perfectionist, after the mid 1940s he more or less stopped publishing.

Gödel's doctoral dissertation, written when he was 23, established the *completeness* theorem for the first-order predicate calculus (i.e. a standard proof system for first-order logic indeed captures all the semantically valid inferences).

Later he would do immensely important and seminal work on set theory, as well as make contributions to proof theory and to the philosophy of mathematics. He even found models of General Relativity with closed time-like curves, allowing time-travel. Talk of 'Gödel's Theorems', however, typically refers to his two *incompleteness* theorems in an epoch-making 1931 paper, 'On formally undecidable propositions of *Principia Mathematica* and related systems I'. More on that title in due course.

For an overview of Gödel's work, see http://plato.stanford.edu/entries/goedel.

[2] The terminological situation is a bit messy. Some labels we'll be using are entirely standard; some are used in importantly different ways by different authors; while some natural ideas seem to have no commonly used labels at all. It might be helpful, then, if I star my own non-standard terminology when it is first defined. You can safely re-use unstarred jargon without comment; but if you adopt the starred coinages, they will probably need a word of explanation.

1.1 Notational conventions

A global convention The incompleteness theorems are going to tell us about the limitations of formal theories: very roughly, for any given arithmetically competent axiomatized theory T there will be truths of arithmetic it can't prove.

Evidently, if we are going to prove such results, it is going to crucial to be absolutely clear when we are proving theorems inside the official formalized theory T, and when we are – so to speak – standing outside the theory and doing ordinary informal arithmetic or doing informal reasoning about the theory T.

It's risky just to rely on context. So to keep things clear, we adopt a now quite widely used convention: *italic* symbols will belong to our informal mathematics, sans serif symbols belong to formal T-wffs.

Logic An $\mathcal{L}$-wff (well-formed formula) is closed – i.e. is a sentence – if it has no free variables; a wff is open if it has free variables. We'll use the slang 'k-place open wff' to mean a wff with k distinct free variables (though each variable may have more than one occurrence). If φ is a one-place open wff, and τ is some term, we will write $\varphi(\tau)$ for the result of substituting τ for each occurrence of the sole free variable in φ.

When there's a kosher proof of the sentence φ in the formalized theory T, we as usual write $T \vdash \varphi$.

Numerals and numerical quantifiers The standard numeral for the number n, recall, is '$\mathsf{SSS}\ldots\mathsf{S0}$' with n occurrences of 'S'. For convenience, we use '$\overline{\mathsf{n}}$' to abbreviate this standard numeral for n (overlining for formal numerals is, in fact, a pretty standard convention). And if '$\vec{n}$' is a tuple of natural numbers, then '$\vec{\overline{\mathsf{n}}}$' stands for the corresponding tuple of formal numerals.

Restricted quantifiers are defined as usual: thus '$(\forall \mathsf{x} \in \mathbb{N})\varphi$' is syntactic sugar for '$\forall \mathsf{x}(\mathsf{x} \in \mathbb{N} \to \varphi)$', and '$(\exists \mathsf{x} < \tau)\varphi$' similarly abbreviates '$\exists \mathsf{x}(\mathsf{x} < \tau \wedge \varphi)$'. In the general case, we are going to be interested in theories whose native language is richer than $\mathcal{L}_A$, and to get surrogates for the arithmetical quantifiers we'll need to strict the theories native quantifiers in the obvious way. Let's adopt the convention that *whatever theory T we are talking about, all quantifiers in displayed wffs are to be read as being restricted when necessary in order to serve as the theory's surrogate* arithmetical *quantifiers.*

Gödel numbers, Gödel numerals Take a scheme of Gödel-numbering expressions or sequences of expressions of the relevant language to have

been fixed. If e is an expression, or a sequence of expressions, then in informal contexts we use the expression $\ulcorner e \urcorner$ (meaning, of course, a left corner followed by the expression e followed by a right corner) to denote the Gödel number for e. Correspondingly, using overlining again, in formal contexts $\overline{\ulcorner e \urcorner}$ stands in for the standard numeral for that number. Thus,

(i) $\mathsf{x} = \mathsf{SS0}$ is a formal wff of $\mathcal{L}_A$.

(ii) $\ulcorner \mathsf{x} = \mathsf{SS0} \urcorner$ is the Gödel number for that wff in whatever scheme is in play – say $2^2 \cdot 3^{15} \cdot 5^{23} \cdot 7^{23} \cdot 11^{21}$ (in the scheme adopted in my book).

(iii) $\overline{\ulcorner \mathsf{x} = \mathsf{SS0} \urcorner}$ is a term of $\mathcal{L}_A$, of the form $\mathsf{SSS}\ldots\mathsf{S0}$ with $2^2 \cdot 3^{15} \cdot 5^{23} \cdot 7^{23} \cdot 11^{21}$ occurrences of 'S'. (You can see why we want the syntactic sugar here: it really does help the pill to go down!)

1.2 Capturing p.r. functions and relations

We are going to be concerned with formal theories T that can do enough arithmetic to be interesting. Here 'doing enough', at the very least, should mean that we can replicate inside the theory the sort of calculations that we can already do informally.

To put that a bit more carefully, let's first say

Definition 1 *A theory T* captures* *(many say* represents*) the k-place function $f\vec{x}$ iff there is an $(k+1)$-place open wff $\mathsf{F}(\vec{\mathsf{x}}, \mathsf{y})$ of T's language such that for all k-tuples $\vec{m}$ of natural numbers, and for all n, if $f\vec{m} = n$, then $T \vdash \forall \mathsf{y}(\mathsf{F}(\vec{\mathsf{m}}, \mathsf{y}) \leftrightarrow \mathsf{y} = \overline{\mathsf{n}})$.*

Definition 2 *A theory T* captures* *the k-place relation R iff there is a k-place wff $\mathsf{R}(\vec{\mathsf{x}})$ of T's language such that for all $\vec{m}$,*

if $R\vec{m}$, then $T \vdash \mathsf{R}(\vec{\mathsf{m}})$,
if not-$R\vec{m}$, then $T \vdash \neg\mathsf{R}(\vec{\mathsf{m}})$.

'Capturing' is here intended to be helpfully mnemonic for 'case by case prove': for note that capturing/representing a function is basically a matter of being able to prove, case by case, that a function takes the right values (and no others) for given particular arguments. Likewise for capturing relations. And note, by the way, that our definitions of capturing for functions and relations fit together as you might hope.

That is to say, if f is the characteristic function of the relation R, then T captures f if and only if it captures R (assuming that $T \vdash \overline{0} \neq \overline{1}$).[3]

It is very natural, then, to want T to be able to capture any primitive recursive function. For if *we* can informally define the k-place p.r. function f and compute its value for the arguments $\vec{m}$ to show that $f\vec{m} = n$ (and any p.r. function has a definition we could in principle use to compute it), then we'd like to be able to replicate the computation inside the formal theory T and prove a corresponding sentence $\mathsf{F}(\vec{\mathsf{m}}, \overline{\mathsf{n}})$ (and be able to prove $\neg\mathsf{F}(\vec{\mathsf{m}}, \overline{\mathsf{o}})$ for $n \neq o$). After all, going formal surely shouldn't stop us doing the p.r. computations that we can already do informally.

That motivates the following

Definition 3 *A theory T is* p.r. adequate* *iff $T \vdash \overline{0} \neq \overline{1}$ and T captures all p.r. functions.*

(The first clause is just to ensure that T handles characteristic functions sensibly!) Our interest henceforth is going to be in theories T which are p.r. adequate.

'Well it might be highly desirable for a theory to be p.r. adequate. But that's all a bit abstract: what must a theory look like in order actually to have the property?' Good question. We answer it in the next lecture.

1.3 Arithmetizing syntax

The arithmetized proof relation Without significant loss of generality, we can take it that proofs in the formal theory T are linear sequences of wffs.[4] We can then make the following stipulation:

Definition 4 *The relation $Prf_T(m, n)$ holds when m is the Gödel number of a sequence of wffs that is a T-proof, and n is the Gödel number of the wff proved.*

---

[3] Why? Well, suppose T captures f, so that there is an open wff $\mathsf{F}(\vec{\mathsf{x}}, \mathsf{y})$ such that if $f(\vec{m}) = n$, then $T \vdash \forall\mathsf{y}(\mathsf{F}(\vec{\mathsf{m}}, \mathsf{y}) \leftrightarrow \mathsf{y} = \overline{\mathsf{n}})$. It is easily checked that $\mathsf{F}(\vec{\mathsf{x}}, \overline{\mathsf{1}})$ captures R. And conversely, suppose T captures R by the wff $\mathsf{R}(\vec{\mathsf{x}})$. It is easily checked that $\mathsf{F}(\vec{\mathsf{x}}, \mathsf{y}) =_{\text{def}} (\mathsf{R}(\vec{\mathsf{x}}) \wedge \mathsf{y} = \overline{\mathsf{1}}) \vee (\neg\mathsf{R}(\vec{\mathsf{x}}) \wedge \mathsf{y} = \overline{\mathsf{0}})$ captures f.

[4] If proofs aren't linear in T's native form, we can do one of two things. Either (i) we just replace T's deductive system using trees, or whatever, with a linear version, and think about the equivalent theory T' instead. Or it may be less hassle if (ii) we instead extend our scheme for Gödel-numbering so that we can number not only wffs and sequences of wffs but also T's proof arrays (whether trees or other arrangements). Let's not fuss about this.

Then $Prf_T(m, n)$ is a numerical relation. But we can say more. Think about the business of checking whether it holds of a given pair of numbers. You mechanically decode m when treated as a Gödel number; you then mechanically check to see whether the output is garbage or is a well-constructed T-proof according to principles defining the formal system; if the output is a T-proof, mechanically check to see whether the conclusion has Gödel number n; if it does, the Prf_T relation holds, and not otherwise. So we can, in short, mechanically check whether $Prf_T(m, n)$ holds. Moreover, if T is a properly formalized theory, we can do this without unbounded searches – after all, what's the point of building a formal system where you can't check whether a purported proof is a proof without twiddling your thumbs while waiting upon the results of open-ended searches? For any sensible T, $Prf_T(m, n)$ *is a p.r. relation.*

Diagonalization A key syntactic construction in what follows involves putting the Gödel number for a one-place open wff – or more accurately, the standard numeral for that Gödel number – into free variable places in that formula. If we think of the Gödel number for an open formula as a way of indexing the formula in an enumeration of open formulae, this is rather like the familiar diagonalization construction, where we plug the enumerating index for a function into the function as argument. Hence we'll say

Definition 5 *The* diagonalization *of a one-free-variable open wff φ is $\varphi(\overline{\ulcorner\varphi\urcorner})$.*

Corresponding to the formal syntactic operation of diagonalization, there's an arithmetic function defined as follows:

Definition 6 *$diag_T(m) = n$ if m Gödel-numbers some one-place open wff φ of T's language and n Gödel-numbers its diagonalization $\varphi(\overline{\ulcorner\varphi\urcorner})$, or else $diag_T(m) = m$.*

And because the syntactic operation is a simple mechanical business, this corresponding arithmetic function is primitive recursive.

1.4 'T is p.r. formalized' and 'niceness'

We said that, for a sensibly formalized theory T, Prf_T will be p.r.; and with the sort of theory we are going be interested in (which has standard numerals), we can do diagonalization and so there'll be a corresponding p.r. diagonalization function $diag_T$. So let's just stipulate that

Definition 7 *T is a* p.r. formalized* *theory iff Prf_T and $diag_T$ are primitive recursive.*

Given our remarks so far, then, we are interested in those theories which are both p.r. formalized and p.r. adequate, and we'll assume that they have enough logic to make them interesting. In a single word, we want 'nice' theories, where

Definition 8 *T is a* nice* *theory iff it is p.r. adequate, p.r. formalized, and contains at least first-order logic.*[5]

Any nice theory T, since it captures every p.r. relation, will in particular be able to capture Prf_T: we'll write $\mathsf{Prf}_T(\mathsf{x}, \mathsf{y})$ for any two-place wff that can do the trick. So we have

Definition 9 $\mathsf{Prf}_T(\mathsf{x}, \mathsf{y})$ *stands in for any T-wff (and there will be many if T is nice!) such that, for any m, n,*

if $Prf_T(m, n)$, then $T \vdash \mathsf{Prf}_T(\overline{\mathsf{m}}, \overline{\mathsf{n}})$,
if not-$Prf_T(m, n)$, then $T \vdash \neg\mathsf{Prf}_T(\overline{\mathsf{m}}, \overline{\mathsf{n}})$.

And we'll immediately add a useful abbreviation whose motivation should be obvous:

Definition 10 *We put* $\mathsf{Prov}_T(\mathsf{y}) =_{\text{def}} \exists\mathsf{x}\,\mathsf{Prf}_T(\mathsf{x}, \mathsf{y})$,

We can naturally call such an expression *a (generic) provability predicate* for T.

If T is nice, it will also be able to capture the p.r. function $diag_T$: we'll write $\mathsf{Diag}_T(\mathsf{x}, \mathsf{y})$ for any wff that can do the trick. So

Definition 11 $\mathsf{Diag}_T(\mathsf{x}, \mathsf{y})$ *stands in for any T-wff such that, for any m, n, if $diag_T(m) = n$, then $T \vdash \forall\mathsf{y}(\mathsf{Diag}_T(\overline{\mathsf{m}}, \mathsf{y}) \leftrightarrow \mathsf{y} = \overline{\mathsf{n}})$.*

1.5 ω-consistency, ω-completeness

Finally by way of our scene-setting preliminaries, we need reminders about two standard notions about theories, and definitions of two perhaps new ideas:

[5] Actually we won't presuppose anything like full classical logic: intuitionistic logic is already more than enough. But let's not fuss about this.

Definition 12 *Suppose T is a theory:*

(i) *T is* negation-complete *if for every sentence φ of its language, either $T \vdash \varphi$ or $T \vdash \neg\varphi$. If T is negation-incomplete, a sentence φ that can't be proved or refuted is said to be* undecidable.[6]

(ii) *T is* inconsistent *if from some φ, $T \vdash \varphi$ and $T \vdash \neg\varphi$.*

(iii) *T is* ω-inconsistent *iff, for some wff $\varphi(\mathsf{x})$, $T \vdash \exists\mathsf{x}\,\varphi(\mathsf{x})$, yet for every m, $T \vdash \neg\varphi(\overline{\mathsf{m}})$.*

(iv) *T is* ω-incomplete *iff, for some open wff with one free variable $\varphi(\mathsf{x})$, (i) for each m, $T \vdash \varphi(\overline{\mathsf{m}})$, though (ii) $T \nvdash \forall\mathsf{x}\varphi(\mathsf{x})$.*

We'd obviously like negation-complete theories when we can get them (for they in principle settle one way or the other all claims in the relevant vocabulary). And equally obviously, inconsistency is a bad thing (certainly in a classical framework, where an inconsistent theory indiscriminately entails every sentence).

Evidently, being ω-incomplete is a rather sad short-coming for a theory; it fails to be able to prove $\forall\mathsf{x}\varphi(\mathsf{x})$ when it 'ought' to be able to, since it can prove each $\varphi(\overline{\mathsf{m}})$. But being ω-inconsistent is an outright disaster assuming we want T to be interpretable as doing arithmetic: for the theorem $\exists\mathsf{x}\,\varphi(\mathsf{x})$, with a numerical quantifier, can't be true on the intended arithmetical interpretation if each of $\neg\varphi(\overline{\mathsf{m}})$ is true too. We therefore want formal theories that cover arithmetic to be ω-consistent.

Note, finally, that since ω-consistency is a matter of *not* being able to prove a certain unwanted combination of wffs, ω-consistency entails plain consistency (since inconsistent theories prove anything).

1.6 The Fixed-point Lemma

So at last, we get down to proving our key theorems for this lecture.

By abuse of jargon, relying on the superficial analogy with the likes of $f(a) = a$, we'll say that

Definition 13 *If $\varphi(\mathsf{x})$ is a with one free variable, then γ is a* fixed point *for $\varphi(\mathsf{x})$ in theory T iff $T \vdash \gamma \leftrightarrow \varphi(\overline{\ulcorner\gamma\urcorner})$.*

Then we have the following extremely simple but surprisingly fruitful result:

[6] Careful: don't muddle the notions of (i) T's being undecidable – there is no algorithm to setting whether a given sentence φ has a proof – and (ii) a T-sentence φ's being undecidable.

Theorem 1 *If T is nice, then any T-wff $\varphi(\mathsf{x})$ with one free variable has a fixed point.*

Proof Put $\delta =_{\text{def}} \forall \mathsf{y}(\mathsf{Diag}_T(\mathsf{x},\mathsf{y}) \to \varphi(\mathsf{y}))$, and $\gamma =_{\text{def}} \delta(\overline{\ulcorner \delta \urcorner})$. We show that γ is a fixed point for φ.

Since diagonalizing δ yields γ, we have $diag(\ulcorner \delta \urcorner) = \ulcorner \gamma \urcorner$, and hence $T \vdash \forall \mathsf{y}(\mathsf{Diag}_T(\overline{\ulcorner \delta \urcorner},\mathsf{y}) \leftrightarrow \mathsf{y} = \overline{\ulcorner \gamma \urcorner})$ since T is p.r. adequate. But just by the definition of γ, $T \vdash \gamma \leftrightarrow \forall \mathsf{y}(\mathsf{Diag}_T(\overline{\ulcorner \delta \urcorner},\mathsf{y}) \to \varphi(\mathsf{y}))$. Hence, substituting provable equivalents, we have $T \vdash \gamma \leftrightarrow \forall \mathsf{y}(\mathsf{y} = \overline{\ulcorner \gamma \urcorner} \to \varphi(\mathsf{y}))$, and therefore $T \vdash \gamma \leftrightarrow \varphi(\overline{\ulcorner \gamma \urcorner})$. □

Three comments. (a) The fixed point sentence is arithmetical. (b) It would have done just as well to put $\delta' =_{\text{def}} \exists \mathsf{y}(\mathsf{Diag}_T(\mathsf{x},\mathsf{y}) \wedge \varphi(\mathsf{y}))$, and set $\gamma =_{\text{def}} \delta'(\overline{\ulcorner \delta' \urcorner})$, with the rest of the proof much as before. (c) The Fixed Point Lemma is equally often called (by another slight abuse) the Diagonalization Lemma.

1.7 A general version of Gödel's First Theorem

We can now prove the following key result:

Theorem 2 *Suppose T is nice. Then there is a sentence G_T such that (i) if T is consistent, $T \nvdash \mathsf{G}_T$. And (ii) if T is ω-consistent, $T \nvdash \neg\mathsf{G}_T$.*

Proof Take a generic proof predicate for T, $\mathsf{Prov}(\mathsf{y}) =_{\text{def}} \exists \mathsf{x}\, \mathsf{Prf}(\mathsf{x},\mathsf{y})$ (see Defn. 10; we drop subscript T's for reasons of aesthetics!). By the Fixed Point Theorem, there is a sentence G such that $T \vdash \mathsf{G} \leftrightarrow \neg\mathsf{Prov}(\overline{\ulcorner \mathsf{G} \urcorner})$.

(i) Suppose $T \vdash \mathsf{G}$. Then $T \vdash \neg\mathsf{Prov}(\overline{\ulcorner \mathsf{G} \urcorner})$. But if there *is* a proof of G, then for some m, $Prf(m, \ulcorner \mathsf{G} \urcorner)$, so $T \vdash \mathsf{Prf}(\overline{m}, \overline{\ulcorner \mathsf{G} \urcorner})$, since T captures Prf by Prf. Hence $T \vdash \exists \mathsf{x}\, \mathsf{Prf}(\mathsf{x}, \overline{\ulcorner \mathsf{G} \urcorner})$, i.e. we also have $T \vdash \mathsf{Prov}(\overline{\ulcorner \mathsf{G} \urcorner})$, making T inconsistent. So if T is consistent, $T \nvdash \mathsf{G}$.

(ii) Suppose $T \vdash \neg\mathsf{G}$. Then $T \vdash \mathsf{Prov}(\overline{\ulcorner \mathsf{G} \urcorner})$, i.e. $T \vdash \exists \mathsf{x}\, \mathsf{Prf}(\mathsf{x}, \overline{\ulcorner \mathsf{G} \urcorner})$. But given T is consistent, if T proves $\neg\mathsf{G}$, there is no proof of G, i.e. for every m, not-$Prf(m, \ulcorner \mathsf{G} \urcorner)$, whence for every m, $T \vdash \neg\mathsf{Prf}(\overline{m}, \overline{\ulcorner \mathsf{G} \urcorner})$. So we have a φ such that T proves $\exists \mathsf{x} \varphi(\mathsf{x})$ while it refutes each instance $\varphi(\overline{m})$, which makes T ω-inconsistent. So if T is ω-consistent, $T \nvdash \neg\mathsf{G}$. □

It immediately follows, of course, that

Theorem 3 *If T is nice and ω-consistent, it is negation-incomplete.*

NB, we *haven't* shown that there are some elusive arithmetical truths that are forever unprovable, no matter what our resources – i.e. we haven't shown that, for some true φ, every nice T fails to prove φ. What we've shown has the quantifiers the other way about: for every suitable T, there is some true φ (whichever of G and $\neg\mathsf{G}$ is true) which T doesn't prove.

Our theorem is a generic version of Gödel's First Incompleteness Theorem. Or, as it sometimes better called, the *Incompletability* Theorem. For suppose that T is nice. How could we try to 'complete' T so it decides every arithmetical sentence one way or the other? We could try throwing in more axioms. For example, we could for a start add G_T as a new axiom. But this augmented theory T' will remain p.r. adequate and it will be logically at least as strong. So our theorem tells us that completeness would come at an exhorbitant prices. Either (i) T' would no longer being p.r. axiomatized (we can no longer decide what's an axiom in a p.r. way[7]), or (ii) T would no longer be interpretable as, in part, about the natural numbers because it is ω-inconsistent. So, assuming we want a properly axiomatized theory, the arithmetical part of which is still interpretable as being about the natural numbers, T must remain forever incomplete.

And here's a corollary of our proof of Theorem 2:

Theorem 4 *If T is nice and consistent, it is ω-incomplete.*

Proof We've proved that, given niceness and consistency, $T \nvdash \mathsf{G}$, hence $T \nvdash \neg\mathsf{Prov}(\overline{\ulcorner\mathsf{G}\urcorner})$, i.e. $T \nvdash \forall\mathsf{x}\,\neg\mathsf{Prf}(\mathsf{x}, \overline{\ulcorner\mathsf{G}\urcorner})$.

But, since T doesn't prove G, that means for every m, not-$Prf(m, \ulcorner\mathsf{G}\urcorner)$, whence – by T's p.r. adequacy – $T \vdash \neg\mathsf{Prf}(\overline{\mathsf{m}}, \overline{\ulcorner\mathsf{G}\urcorner})$.

So T is indeed ω-incomplete. □

[7] Or in any other way, because if a theory is effectively axiomatizable at all, it is re-axiomatizable in a p.r. way.

2

Giving the incompleteness theorem more bite

As noted before, in computable function theory, we're familiar with a dual use of numbers – as 'data' to be fed to functions and yielded as output, and as 'indices' or 'codes' for (programs for) the functions themselves. In particular, we're familiar with the 'diagonalization' trick, where we take the number n and feed it as input data to the function φ_n determined by the program with code number n. (Standard example: the proof of the unsolvability of Halting Problem).

This 'diagonalization' trick has its roots Gödel's 1931 paper. Here he (i) uses numbers as indices/codes for formulae, and then (ii) feeds a number to the property expressed by the formula coded by that number as index. We saw this simple construction at work in proving the Fixed Point Lemma, Theorem 1. Then this very speedily – and by an elementary argument – yielded a general version of the Incompleteness Theorem.[1]

But how have we got here quite so fast, when it took Gödel twenty-five dense pages? Well, we've just so far *assumed* that we are dealing with a 'nice' theory (i.e. a p.r. formalized, p.r. adequate theory), whatever one of those might look like. So we've sidestepped all the rather tedious work needed to establish the niceness of this or that particular theory (such as the cut-down version of *Principia*'s type-theory which Gödel officially discusses).

OK, then: how *do* we establish that a particular given theory is indeed

[1] Indeed, it was of some conceptual importance to Gödel that the first incompleteness theorem *is* quick and easy – once you see the basic idea, of course. The reason that no one had got the result earlier, he thought, was not lack of mathematical facility, but lack of conceptual clarity about the idea of a formally disciplined theory, and the distinction that now seems so plain to us – thanks in part to Gödel! – between syntax and semantics.

nice? How can we do this e.g. for the canonical arithmetical theory PA, first-order Peano Arithmetic? This theory certainly contains first-order logic. So (1) we have to confirm that the function $diag_{\mathsf{PA}}$ and the relation Prf_{PA} are indeed primitive recursive. And (2) we need to show that PA is p.r. adequate. We'll consider these two major steps in turn.

2.1 Confirming that PA is p.r. formalized

To evaluate $diag_{\mathsf{PA}}(m)$ we need to check whether m is the Gödel code of a wff φ with one free variable. If it is, we form the wff $\varphi(\overline{\mathsf{m}})$, and work out its Gödel number $\ulcorner\varphi(\overline{\mathsf{m}})\urcorner$: otherwise we set the output of the function to m. That computation can evidently be done without unbounded searches – for acceptable forms of Gödel coding and decoding (by stipulation!) are effective and can be executed with bounded procedures, and the substitution operation doesn't involve anything unbounded. So $diag_{\mathsf{PA}}(m)$ is primitive recursive.

Similarly, to repeat the line of argument in §1.3, we can mechanically decide whether $Prf_{\mathsf{PA}}(m, n)$ without open-ended searches. That's because to check whether, after decoding m, we get a string of symbols which constitutes a kosher PA proof of the wff numbered n we only need bounded searches along that string. So, again, Prf_{PA} is a p.r. relation.

Now, if you are already familiar with p.r. functions – and if in particular you are happy with the claim that 'no unbounded searches in computing f means that f is p.r.' – then those quick-and-dirty arguments should be convincing. But if you aren't so happy (perhaps because you are time-shifted back to 1931, before the general development of the theory of computation, and when all this was just getting started), you would reasonably insist on an official explicit definition for $diag_{\mathsf{PA}}$ and for (the characteristic function of) Prf_{PA}.

You'll then have go through the sort of extended palaver that Gödel went through in his 1931 paper to show that the cut-down type-theory which he is discussing there is indeed p.r. axiomatized. Essentially, you have to construct a long list of ever-more-complex functions, with each one defined by composition or recursion from earlier functions on the list. Doing this is, in effect, a laborious but routine programming exercise whose fine details will depend on the kind of Gödel-numbering scheme you have adopted – fun if you like that kind of thing, and not if not: we can't go into more details here, but any mathematical logic book will give you some of the gory details. And do it for one theory like PA, with one Gödel-numbering scheme, and you'll be readily convinced that

it can be done for any other sensibly presented axiomatized theory with any other sensible numbering scheme.

2.2 Confirming that PA is p.r. adequate

The task of giving an adequacy proof is conceptually more interesting, and the proof also yields a lot of additional information, giving us a specification for a very large swathe of the theories to which the First Theorem applies and also key news about just how logically simple undecidable Gödel sentences can be.

The general strategy for proving that PA is p.r. adequate is obvious. We need to show that (i) PA can capture the so-called 'initial' functions (the zero function, projection functions, and successor). Then we show that (ii) if PA can capture g and h, it can capture their composition $f = g \circ h$. Finally we prove (iii) if PA can capture g and h, it can capture the function f defined from g and h by primitive recursion – i.e., the function such that $f(\vec{x}, 0) = g(\vec{x})$, and $f(\vec{x}, Sy) = h(\vec{x}, y, f(\vec{x}, y))$. Then, since any p.r. function is derivable from the initial functions via repeated definitions by composition and/or primitive recursion, it follows by induction on the length of the definitional chain that PA can capture every p.r. function.

Step (ii) is easy. Suppose g and h are one-place functions, captured by the wffs $\mathsf{G(x, y)}$ and $\mathsf{H(x, y)}$ respectively. Then, the function $f(x) = h(g(x))$ is captured by the wff $\exists\mathsf{z}(\mathsf{G(x, z)} \wedge \mathsf{H(z, y)})$. Other cases where g and/or h are multi-place functions can be handled similarly. We needn't delay over this.

The tricky step in the proof is showing (iii). Take the very simplest sort of case, where we have $f(0) = k$, and $f(Sy) = h(f(y))$. Then $f(m) = n$ just if there is a sequence of numbers $a_0, a_1, a_2, \ldots a_m$, where $a_0 = k$, $a_{a+1} = h(a_i)$, and $a_m = n$. So we might hope to capture f in PA if we can in effect talk not just about numbers but about *sequences* of numbers from inside PA. But how on earth can we do that?

The β-function trick is the way to pull this off – for a β-function in effect takes a code number and spits out a finite sequence. The root idea is that, corresponding to $a_0, a_1, a_2, \ldots a_m$ there will be a code number c, and the decoding function is such that $\beta(c, i) = a_i$ for $0 \leq i \leq m$. With this sort of apparatus in place, generalizing over codes enables us to say, in effect, 'there is a sequence of numbers such that ...'. For example, we say that $f(m) = n$ by saying that there's a c such that $\beta(c, 0) = k$, and for all $0 \leq i \leq m$, $\beta(c, i + 1) = h(\beta(c, i))$, and $\beta(c, m) = n$.

Now, it would be obvious how to do the trick if only exponentiation were available! We could use powers of primes to code up a sequence, i.e. put $c = 2^{a_0} \cdot 3^{a_1} \cdot 5^{a_2} \cdot \ldots \cdot \pi_m^{a_m}$, where π_m is the m-th prime (counting from zero!), and the p.r. function $exp(c, i)$ which outputs the exponent of i in the prime factorization of c will then serve very nicely as a β-function. But of course we don't have exponentiation built into $\mathcal{L}_A$, the language of PA. Kudos, then, to Gödel for spotting how to do the β-function trick even with the more limited resources currently at our disposal.

Think of his β-function as taking *two* code numbers c, d: then he defines

$\beta(c, d, i) =_{\text{def}}$ the remainder left when c is divided by $d(i + 1) + 1$.

And we can use the Chinese Remainder Theorem to prove that, given any sequence $a_0, a_1, a_2, \ldots a_m$, we can indeed find a suitable pair of numbers c, d such that for $0 \leq i \leq m$, $\beta(c, d, i) = a_i$. Since PA knows about multiplication and division, it knows about remainders-on-division too, and can capture Gödel's three-place β-function with a suitable four-place wff B.

Take the simplest sort of definition by recursion again, where f is defined by saying $f(0) = k$ and $f(Sy) = h(f(y))$, so as we noted before – but now using a three-place β function, and changing variables, $f(x) = y$ just in case

(i) There is a pair of code numbers c, d such that: $\beta(c, d, 0) = k$, and if $u < x$ then $\beta(c, d, Su) = h(\beta(c, d, u))$, and $\beta(c, d, x) = y$.

Suppose h is captured by H; then we can render that claim formally as follows

(ii) $\exists \mathsf{c} \exists \mathsf{d} \{ \mathsf{B}(\mathsf{c}, \mathsf{d}, 0, \overline{\mathsf{k}}) \wedge$
$(\forall \mathsf{u} < \mathsf{x})[\exists \mathsf{v} \exists \mathsf{w} \{ (\mathsf{B}(\mathsf{c}, \mathsf{d}, \mathsf{u}, \mathsf{v}) \wedge \mathsf{B}(\mathsf{c}, \mathsf{d}, \mathsf{Su}, \mathsf{w})) \wedge \mathsf{H}(\mathsf{v}, \mathsf{w}) \}] \wedge$
$\mathsf{B}(\mathsf{c}, \mathsf{d}, \mathsf{x}, \mathsf{y}) \}$.

Abbreviate all that by '$\mathsf{F}(\mathsf{x}, \mathsf{y})$', and we've arrived at a wff that captures f (as can be checked). And the construction evidently generalizes to cover other functions defined by primitive recursions from capturable functions.

2.3 Mining the adequacy proof

If we dig inside the details of this adequacy proof we find two important things. First, inspection reveals that we don't need the full strength of

PA to get a p.r. adequate theory. In fact, the induction-free Robinson Arithmetic Q has all we need. (Indeed, Q was introduced into the literature precisely to serve the role as a neatly formulated but very weak arithmetic which has what it takes.)

Suppose, then, we add another definition:

Definition 14 *A* normal* *theory is one that p.r. formalized, and which includes Robinson Arithmetic* Q*.*

'Normal' theories deserve the name! As we've said before, a properly formalized theory will be p.r. formalized, and if you aim to cover any interesting amount of arithmetic beyond what can be given by a pocket calculator, which can express no quantified truths, your theory will include Q. We then have the memorable slogan

Theorem 5 *Any normal theory is nice.*

and so the incompleteness theorem will apply.

But second, we can also extract additional information from the adequacy proof, this time about the kinds of expressions we need to do the capturing job. We will be interested in categorizing wffs of $\mathcal{L}_A$ by *quantifier complexity*. You may have met before the standard definitions for the initial levels of the arithmetical hierarchy:

(i) Δ_0 (a.k.a. Σ_0) wffs: these are wffs with no unbounded quantifiers, i.e. any quantifier is either of the form $(\forall \mathsf{x} < \tau)$ or $(\exists \mathsf{x} < \tau)$ for some variable and some term τ. (Evidently we can determine the truth value of any *closed* Δ_0 wff on the intended interpretation by a mechanical computation.)

(ii) Σ_1 wffs: these are logically equivalent to wffs of the form $\exists \vec{\mathsf{x}} \varphi$ where φ is Δ_0 (where of course $\exists \vec{\mathsf{x}}$ is a block of existential quantifiers).

(iii) Π_1 wffs: these are logically equivalent to wffs of the form $\forall \vec{\mathsf{x}} \varphi$ where φ is Δ_0.

It then turns out that we have

Theorem 6 Q*, and hence any normal theory, can in fact capture any p.r. function by using just a* Σ_1 *formula.*

Why so? Well, we only need a Δ_0 wff to capture the initial functions

and to capture Gödel's β-function too, for we can put

$$\mathsf{B}(\mathsf{c},\mathsf{d},\mathsf{i},\mathsf{y}) =_{\text{def}} (\exists \mathsf{u} \leq \mathsf{c})[\mathsf{c} = \{\mathsf{S}(\mathsf{d} \times \mathsf{Si}) \times \mathsf{u}\} + \mathsf{y} \ \wedge \ \mathsf{y} \leq (\mathsf{d} \times \mathsf{Si})].$$

So as we build up a wff F to reflect f's p.r. definition, we can do all the construction in such a way that the only unbounded quantifiers we introduce are initial existential ones. These appear when we compose functions (in effect saying there's a number which is the output of one and input to the other), and when we mirror definitions by primitive recursion (in effect saying there are a couple of code numbers which code a certain sequence).

Note an immediate corollary of this. A theory T containing Q can capture, in particular, $diag_T$ and Prf_T by Σ_1 wffs Diag_T and Prf_T. In this case, the corresponding $\mathsf{Prov}_T(\mathsf{y}) =_{\text{def}} \exists \mathsf{x}\, \mathsf{Prf}_T(\mathsf{x},\mathsf{y})$ is also Σ_1.

But we constructed the fixed point for $\neg\mathsf{Prov}$ by first setting

$$\delta =_{\text{def}} \forall \mathsf{y}(\mathsf{Diag}_T(\mathsf{x},\mathsf{y}) \rightarrow \neg \mathsf{Prov}_T(\mathsf{y}))$$

What's inside the bracket is a disjunction of two Π_1 wffs, which is Π_1; so δ as a whole is Π_1. And now we put $\mathsf{G} = \delta(\overline{\ulcorner \delta \urcorner})$. So this fixed point is also Π_1.

2.4 Refining the First Theorem

Putting everything together, then, from Theorems 2, 5 and our last remarks about quantifier complexity we get a sharper version of the First Incompleteness Theorem:

Theorem 7 *If T is normal, then there is a Π_1 sentence G such that (i) if T is consistent, $T \nvdash \mathsf{G}$, and (ii) if T is ω-consistent, $T \nvdash \neg\mathsf{G}$.*

No matter how fancy T is, then, if it is normal and ω-consistent it must have undecidable Π_1 sentences, which are purely arithmetic sentences with a minimal degree of quantifier complexity – i.e. universal quantifications of decidable conditions.

2.5 The 'syntactic' vs the 'semantic' theorems

Notice that we have proved the incompleteness theorem making only purely *syntactic* assumptions about T. Thus,

(i) Being p.r. formalized is a syntactic feature, obviously.

(ii) But being p.r. adequate is also syntactically defined, for it is a matter of what sentences are derivable when.
(iii) Being (syntactically) consistent is trivally a syntactic notion.
(iv) Being ω-consistent is also syntactically defined in terms of a certain combination of sentences not being derivable. (True, failure of ω-consistency in a theory has semantic implications – it constrains our interpretation of the arithmetic of the theory to be non-standard: but that doesn't affect the point that the notion of ω-consistency itself is defined syntactically.)

And the syntactic character of the assumptions here was of some significance to Gödel.

Why? Well, go back to the Vienna of the late 20s. On the one hand, there are positivists who are suspicious on quite general grounds about the very notion of truth (to take the key semantic notion) – after all, the notion is beset by paradox (the Liar and its coevals) and seems freighted with metaphysical ideas about 'correspondence with the world' which are notoriously difficult to make sense of. On the other hand there are the Hilbertians who have special worries about the idea of truth in mathematics (at least when mathematics purports to go beyond the primary school idealization of the 'concrete' arithmetic that can be done with counting blocks or an abacus). Now, it isn't that Gödel shared these doubts about the notion of truth: on the contrary. But it was very important to him that his incompleteness results were not thought to tainted by such suspicions, so he insists on using purely syntactic premisses – though, as we'll now see, the quickest argument for incompleteness does in fact start from semantically loaded premisses.

Thus suppose T is a *sound* theory (i.e. its axioms are true on the intended interpretation, its deductive system is truth-preserving, so all theorems are true). And suppose

(i) Diag_T is constructed so as to *faithfully express* the function $diag_T$, i.e. $\mathsf{Diag}_T(\overline{\mathsf{m}}, \overline{\mathsf{n}})$ is true (on the intended interpretation of $\mathcal{L}_A$) if and only if $diag(m) = n$, and
(ii) Prf_T faithfully expresses Prf_T: i.e. $\mathsf{Prf}_T(\overline{\mathsf{m}}, \overline{\mathsf{n}})$ is true on the intended interpretation if and only if $Prf_T(m, n)$.

(Actually, the same wffs as do the capturing job will do the trick, so we know that there are such wffs.[2] But we don't need to do the work of

[2] Suppose Prf_T captures Prf_T; then if $Prf_T(m, n)$ then $T \vdash \mathsf{Prf}_T(\overline{\mathsf{m}}, \overline{\mathsf{n}})$, so since T is sound, $\mathsf{Prf}_T(\overline{\mathsf{m}}, \overline{\mathsf{n}})$ is true; and similarly if not-$Prf_T(m, n)$, $\mathsf{Prf}_T(\overline{\mathsf{m}}, \overline{\mathsf{n}})$ is false.

proving capturing: it is enough to be able to construct wffs that say the right things on the intended interpretation.)

Now consider the fixed point you get by putting

$$\delta =_{\text{def}} \forall \mathsf{y}(\mathsf{Diag}_T(\mathsf{x}, \mathsf{y}) \to \neg\exists \mathsf{u}\mathsf{Prf}_T(\mathsf{u}, \mathsf{x}))$$

and then diagonalizing it to get

$$\mathsf{G} =_{\text{def}} \forall \mathsf{y}(\mathsf{Diag}_T(\overline{\ulcorner \delta \urcorner}, \mathsf{y}) \to \neg\exists \mathsf{u}\mathsf{Prf}_T(\mathsf{u}, \overline{\ulcorner \delta \urcorner})).$$

Then this diagonalization is true iff for any y, if y numbers the result of diagonalizing the wff with g.n. $\ulcorner \delta \urcorner$ then no number numbers a T-proof of it. In other words, G is true iff the result of diagonalizing the wff δ is unprovable in T, i.e. iff G is unprovable. In sum G is true iff it is itself unprovable in T.

And now the argument can go like this:

(i) Suppose $T \vdash \mathsf{G}$. Then since, by hypothesis, T is sound, G is true, and so it is unprovable. Contradiction. So, $T \nvdash \mathsf{G}$

(ii) Hence G is true. So $\neg\mathsf{G}$ is false. Hence, being sound, $T \nvdash \neg\mathsf{G}$.

NB, for this 'semantic argument' that T can't decide G one way or the other, we don't in fact need to do all the hard work of showing that T can capture *diag* and *Prf*. Nor do we explicitly need the notion of ω-consistency (and similarly for other arguments for T's incompleteness that depend on the soundness of T: they don't need us to invoke ω-consistency either). But more on getting rid of the assumption of ω-consistency in the next lecture.

3

Truth, Tarski, Rosser, and *Principia*

It is tempting to say – people very often *do* say – that our undecidable sentences G_T (defined as fixed points for the negation of a provability predicate for T) are constructed so as to 'say' of themselves that they are unprovable. And since they *are* unprovable, assuming the relevant theory T is consistent, they must be true. But, as we'll see, this general claim is false

We first demonstrate that (it's not a particularly deep point: but it shows we have to be careful in the informal commentary we give on Gödelian results). We then continue with the semantic theme for just a little more, and consider one version of Tarski's Theorem, which reveals what we can think of as the master thought behind the incompleteness results.

Next we turn back to the syntactic version of the incompleteness theorem and look at Rosser's trick for dropping the requirement of ω-consistency.

Finally in this lecture, a very quick word about the impact of all this on the ostensible target of Gödel's paper, the system of *Principia Mathematica*.

3.1 Generic Gödel sentences and arithmetic truth

Let's start by reminding ourselves of the trivial fact that

Theorem 8 *If φ is a true Σ_1 sentence, then $\mathsf{Q} \vdash \varphi$. In a word, Q is* Σ_1-complete.

Proof Suppose φ is equivalent to $\exists\vec{\mathsf{x}}\psi(\vec{\mathsf{x}})$, with ψ a Δ_0 wff. Then if φ is true, there must be some numbers $\vec{\mathsf{n}}$ which make $\psi(\vec{\mathsf{n}})$ true. But Q

can prove any true closed Δ_0 wff (for that's just equivalent to a true Boolean combination of simple equations), so Q can prove $\psi(\vec{\mathsf{n}})$. Hence it can prove its existential quantification. □

From this it is immediate that

Theorem 9 *If φ is a Π_1 sentence of a normal theory T such that $T \nvdash \neg\varphi$, then φ is true.*

Proof Suppose φ is Π_1 but false. Then $\neg\varphi$ is Σ_1 and true. Hence Q can prove $\neg\varphi$, and therefore so can any normal theory T. So if T *can't* prove $\neg\varphi$, $\neg\varphi$ isn't true. So φ *is* true. □

So suppose we now define terms as follows:

Definition 15 *A* Gödel sentence *for a theory T is any fixed point for the negation of any generic provability predicate Prov_T.*

Then, since the Gödel sentences of normal ω-consistent theories are unprovable by Theorem 2, it is immediate that

Theorem 10 *Any Π_1 Gödel sentence for an ω-consistent normal theory T is true.*

Note that we've already shown that normal theories will indeed have Π_1 Gödel sentences (see Theorem 7)), so these will be true, assuming ω-consistency. Note, *this* result about truth is established without requiring T to be sound overall.

Note, moreover, that we proved our last result without claiming that generic Gödel sentences 'say' that they are unprovable-in-T. Which is good, because that's not in general right.

For we can show

Theorem 11 *There are normal and ω-consistent theories with Gödel sentences which are* false.

Proof If $diag_T(m) = n$, then by definition $T \vdash \forall\mathsf{y}(\mathsf{Diag}(\mathsf{m},\mathsf{y}) \leftrightarrow \mathsf{y} = \mathsf{n})$. Let Θ be *any* closed wff which is a theorem of T. Plainly it follows that if $diag_T(m) = n$, we also have $T \vdash \forall\mathsf{y}((\mathsf{Diag}(\mathsf{m},\mathsf{y}) \wedge \Theta) \leftrightarrow \mathsf{y} = \mathsf{n})$. In other words, the wff $\mathsf{Diag}'(\mathsf{x},\mathsf{y}) =_{\text{def}} (\mathsf{Diag}(\mathsf{x},\mathsf{y}) \wedge \Theta)$ also captures $diag_T$.

So by the remark following the proof of Theorem 1, the wff $\mathsf{G}_\Theta =_{\text{def}} \exists\mathsf{y}(\mathsf{Diag}'(\overline{\mathsf{d}},\mathsf{y}) \wedge \neg\mathsf{Prov}(\mathsf{y}))$ is a fixed point for $\neg\mathsf{Prov}$, where d is a certain

Gödel number. Since $\mathsf{Diag}'(\mathsf{x}, \mathsf{y})$ contains Θ as a conjunct, when Θ is false so is G_Θ.

Suppose then that T is arithmetically *unsound*, i.e. entails some *false* sentence of $\mathcal{L}_A$. And suppose Θ in the argument above is such a false theorem. Then, assuming T is normal and ω-consistent, it will have an undecidable Gödel sentence G_Θ which is false. □

And since such a Gödel sentence G_Θ is *false*, it plainly cannot 'say' – in any sense, however stretched – what is true, namely that it is unprovable!

In fact, we can sharpen the last theorem:

Theorem 12 *There are normal and ω-consistent theories framed in $\mathcal{L}_A$ with Gödel sentences which are* false.

Here's the proof idea. There are theories T in the language $\mathcal{L}_A$ which are ω-consistent but unsound – invoke such a T in the proof of the last theorem and we evidently get the sharpened result. And to construct such a T, consider e.g. the theory $\mathsf{PA}+\Omega$, where Ω is carefully constructed – via Gödel coding – to be true if and only if Ω is ω-inconsistent with PA. Now, assuming PA is sound (on the intended standard interpretation), any *true* sentence will be ω-consistent with it. So Ω can't be true. So $\mathsf{PA} + \Omega$ is unsound. But since Ω is false, the theory is ω-consistent!

It is worth stressing Theorem 12, for it certainly isn't unknown for good authors to forget that they've officially introduced Gödel sentences in the generic way, but then give an informal gloss about what's going on which only applies to the Π_1 cases. To take just one example – selected to show that this can happen even in the best books – Mendelson in his classic text *Mathematical Logic* in effect characterizes a Gödel sentence G for a theory T in language $\mathcal{L}$ as one where $T \vdash \mathsf{G} \leftrightarrow \neg\mathsf{Prov}(\ulcorner \mathsf{G} \urcorner)$, with Prov a generic proof predicate, defined as we have defined it. Changing his notation, he then writes

> G is equivalent in T to asserting that there is no proof in T of G. Hence, G is equivalent in T to an assertion that G is unprovable in T. In other words, G says 'I am not provable in T.' ... Under the standard interpretation, G asserts its own unprovability in T. Therefore, G is true for the standard interpretation.

But not so. For a start, that forgets the possibility that T might be unsound, and the biconditional theorem $\mathsf{G} \leftrightarrow \neg\mathsf{Prov}(\ulcorner \mathsf{G} \urcorner)$ could be an unreliable guide to the truth!

3.2 Truth-predicates and Tarski's Theorem

'Snow is white' is true iff snow *is* white. 'Grass is blue' is true iff grass *is* blue. 'Fire burns' is true iff fire burns. And so it goes, for every other biconditional stamped out of the same template. That's because of the meaning of the informal truth-predicate 'is true'. (Indeed, some would say that the fact that all the instances of the bi-conditional template hold is more or less all there is to be said about the truth-predicate: it just serves as a 'disquotation' operator.)

Let's transpose that thought into a formal key. In most formal languages, however, we can't refer to sentences using quotation; but if the language contains apparatus for arithmetic, we can use Gödel numbering instead to tag sentences, as we in fact have already been doing. So let's say

Definition 16 *The one-place open wff* T *is a truth-predicate for the language* $\mathcal{L}$ *iff the biconditional* $\mathsf{T}(\overline{\ulcorner\varphi\urcorner}) \leftrightarrow \varphi$ *is true for every* $\mathcal{L}$*-wff* φ*,*

where we mean, of course, true relative to the standard interpretation built into $\mathcal{L}$ (together with the interpretation for the predicate T).

Now, on that definition, a truth-predicate T for $\mathcal{L}$ needn't already belong to $\mathcal{L}$; but we haven't yet ruled out that it does belong to $\mathcal{L}$. Let's say that

Definition 17 *A language* $\mathcal{L}$ *contains its own truth-predicate when there is a one-place open wff* T *of* $\mathcal{L}$ *which is a truth-predicate for* $\mathcal{L}$*.*

So now suppose that we are dealing with a normal theory T built in the language $\mathcal{L}$ which extends $\mathcal{L}_A$ and contains its own truth-predicate. Then,

(i) A version of the Fixed Point Lemma tells us that there is a sentence λ such that $\mathsf{Q} \vdash \lambda \leftrightarrow \neg\mathsf{T}(\overline{\ulcorner\lambda\urcorner})$.[1] And since we believe what Q tells us, that theorem must be true. So we have a λ such that $\lambda \leftrightarrow \neg\mathsf{T}(\overline{\ulcorner\lambda\urcorner})$ is true.

[1] Why? It's the same argument as for Theorem 1, mutatis mutandis. Since the T is normal, the function $diag_T$ is p.r.; and hence, since even Q is p.r. adequate, it can capture that function with an $\mathcal{L}_A$ wff Diag_T.

Now put $\delta =_{\text{def}} \forall\mathsf{y}(\mathsf{Diag}_T(\mathsf{x},\mathsf{y}) \rightarrow \neg\mathsf{T}(\mathsf{y}))$, and let $\lambda =_{\text{def}} \delta(\overline{\ulcorner\delta\urcorner})$. Since diagonalizing δ yields λ, we of course have $diag(\ulcorner\delta\urcorner) = \ulcorner\lambda\urcorner$, and hence $\mathsf{Q} \vdash \forall\mathsf{y}(\mathsf{Diag}_T(\overline{\ulcorner\delta\urcorner},\mathsf{y}) \leftrightarrow \mathsf{y} = \overline{\ulcorner\gamma\urcorner})$.

But, trivially, $\mathsf{Q} \vdash \gamma \leftrightarrow \forall\mathsf{y}(\mathsf{Diag}_T(\overline{\ulcorner\delta\urcorner},\mathsf{y}) \rightarrow \neg\mathsf{T}(\mathsf{y}))$. Whence, immediately, $\mathsf{Q} \vdash \lambda \leftrightarrow \neg\mathsf{T}(\overline{\ulcorner\lambda\urcorner})$.

(ii) But because T by hypothesis is a truth-predicate for $\mathcal{L}$ we also have $\lambda \leftrightarrow \mathsf{T}(\overline{\ulcorner \lambda \urcorner})$.

But we can't consistently have both (1) and (2). Hence we get a version of Tarski's theorem:

Theorem 13 *No normal theory T can have a language which contains its own truth-predicate.*

3.3 The Master Argument for incompleteness

Tarksi's result about the non-expressibility of truth gives us a particularly illuminating take on the argument for incompleteness.

Truth-in-L_A isn't provability-in-PA, because while PA-provability *is* faithfully expressible in L_A (using a sensibility constructed provability predicate), truth-in-L_A *isn't* (PA can't contain the truth-predicate for its own language). So assuming that PA is sound so that everything provable in it is true, this means that there must be truths of L_A which it can't prove. Similarly, of course, for other normal theories.

And in a way, we might well take this to be *the* Master Argument for incompleteness, revealing the true roots of the phenomenon. Gödel himself wrote (in response to a query)

> I think the theorem of mine that von Neumann refers to is ... that a complete epistemological description of a language A cannot be given in the same language A, because the concept of truth of sentences in A cannot be defined in A. *It is this theorem which is the true reason for the existence of undecidable propositions in the formal systems containing arithmetic.* I did not, however, formulate it explicitly in my paper of 1931 but only in my Princeton lectures of 1934. The same theorem was proved by Tarski in his paper on the concept of truth.

In sum, as we emphasized before, arithmetical truth and provability in this or that formal system must peel apart.

3.4 Rosser's Theorem

Back to the syntactic First Theorem. Currently, one half requires the assumption that we are dealing with a theory T which is not only consistent but is ω-consistent. But we can improve on this: following Barkley Rosser, we can construct a *different* and more complex sentence R_T such that we only need to assume T is plain consistent in order to show that R_T is formally undecidable.

What's the trick? Well, putting it roughly, where Gödel constructs a sentence G_T that (in good cases) indirectly says 'I am unprovable in T', Rosser constructs a 'Rosser sentence' R_T which indirectly says 'if I am provable in T, then my negation is already provable' (i.e. it says that if there is a proof of R_T with super g.n. n, then there is a proof of $\neg\mathsf{R}_T$ with a smaller code number).

Here's how to implement the trick. Consider the relation $\widetilde{Prf}_T(m, n)$ which holds when m numbers a T-proof of the *negation* of the wff with number n. This relation is obviously p.r. given that Prf_T is; so assuming T is has the usual properties it will be captured by a wff $\widetilde{\mathsf{Prf}}_T(\mathsf{x}, \mathsf{y})$. So let's consider *the Rosser provability predicate* defined as follows:

Definition 18 $\mathsf{RProv}_T(\mathsf{x}) =_{\text{def}} \exists\mathsf{v}(\mathsf{Prf}_T(\mathsf{v}, \mathsf{x}) \wedge (\forall\mathsf{w} \leq \mathsf{v})\neg\widetilde{\mathsf{Prf}}_T(\mathsf{w}, \mathsf{x}))$.

Then a sentence is Rosser-provable in T – its g.n. satisfies the Rosser provability predicate – if it has a proof (in the ordinary sense) and there's no 'smaller' proof of its negation.

Now we apply the Diagonalization Lemma, not to the negation of a regular provability predicate (which is what we just did to get Gödel's First Theorem again), but to the negation of the Rosser provability predicate. The Lemma then tells us,

Theorem 14 *Given that T is normal, then there is a sentence R_T such that $T \vdash \mathsf{R}_T \leftrightarrow \neg\mathsf{RProv}_T(\overline{\ulcorner\mathsf{R}_T\urcorner})$.*

We call such a sentence R_T a Rosser sentence for T. We then can show

Theorem 15 *If T is nice and consistent, then $T \nvdash \mathsf{R}_T$ and $T \nvdash \neg\mathsf{R}_T$.*

Sadly, however – and there's no getting away from it – the proof of this theorem is rather unpretty.[2] We then have to do more work to beef up

[2] For masochists. Suppose φ is any theorem. Then for some m, $Prf(m, \ulcorner\varphi\urcorner)$. Since Prf captures Prf, $T \vdash \mathsf{Prf}(\overline{\mathsf{m}}, \overline{\ulcorner\varphi\urcorner})$.

Also, since T is consistent, $\neg\varphi$ is unprovable, so for all n, not-$\widetilde{Prf}(\overline{n}, \ulcorner\varphi\urcorner)$. Since $\widetilde{\mathsf{Prf}}$ captures $\widetilde{Prf}$, then for each $n \leq m$ in particular, $T \vdash \neg\widetilde{\mathsf{Prf}}(\overline{\mathsf{n}}, \overline{\ulcorner\varphi\urcorner})$. Whence, after some work, we can get $T \vdash (\forall\mathsf{w} \leq \overline{\mathsf{m}})\neg\widetilde{\mathsf{Prf}}(\mathsf{w}, \overline{\ulcorner\varphi\urcorner})$.

So: $T \vdash \mathsf{Prf}(\overline{\mathsf{m}}, \overline{\ulcorner\varphi\urcorner}) \wedge (\forall\mathsf{w} \leq \overline{\mathsf{m}})\neg\widetilde{\mathsf{Prf}}(\mathsf{w}, \overline{\ulcorner\varphi\urcorner})$. Existentially quantifying, $T \vdash \mathsf{RProv}(\overline{\ulcorner\varphi\urcorner})$.

Hence if, in particular, $T \vdash \mathsf{R}$, then $T \vdash \mathsf{RProv}(\overline{\ulcorner\mathsf{R}\urcorner})$. But – by the definition of R – if $T \vdash \mathsf{R}$, then $T \vdash \neg\mathsf{RProv}(\overline{\ulcorner\mathsf{R}\urcorner})$. Contradiction. Hence $T \nvdash \mathsf{R}$.

Suppose now that $\neg\varphi$ is a theorem. Then for some m, $\widetilde{Prf}(m, \ulcorner\varphi\urcorner)$, so $T \vdash \widetilde{\mathsf{Prf}}(\overline{\mathsf{m}}, \overline{\ulcorner\varphi\urcorner})$.

Since T is consistent, φ is unprovable, so for all n, not-$Prf(n, \ulcorner\varphi\urcorner)$. Whence,

that proof idea to show that in fact (as with Gödel's original proof) we can find a Π_1 sentence which is undecidable so long as T is consistent (that work is done in my book on p. 179). Summarizing, we then get a version of Rosser's Theorem:

Theorem 16 *theorem If T is normal, then there is Π_1 sentence φ such that, if T is consistent then $T \nvdash \varphi$ and $T \nvdash \neg\varphi$.*

3.5 The First Theorem and *Principia*

Let's just pause here to think a bit about some broader implications of our incompleteness theorem.

The elementary arithmetic of successor, addition, and multiplication is child's play! It is – or at least, *was* – highly plausible to suppose that, whether the answers are readily available to us or not, questions posed in $\mathcal{L}_A$ have entirely determinate answers which are 'fixed' by (a) the fundamental zero-and-its-successors structure of the natural number series plus (b) the nature of addition and multiplication as given by the school-room explanations.

So it is (or was) surely very plausible to suppose that we should be able lay down a bunch of axioms which characterize the number series, addition and multiplication (which codify what we teach the kids), and that these axioms should in principle settle every truth of $\mathcal{L}_A$, in the sense that every such truth is logically provable from these axioms. For want of a standard label, call this view *deductivism* about basic arithmetic.

What could be the status of the axioms? I suppose you might, for example, be a Kantian deductivist who holds that the axioms encapsulate 'intuitions' in which we grasp the fundamental structure of the numbers and the nature of addition and multiplication, where these 'intuitions' are a special cognitive achievement in which we somehow represent to ourselves the arithmetical world.

But talk of intuition is very puzzling and problematic (at least, once we go beyond the most elementary cases). So we might well be tempted instead by Frege's view that the axioms are *analytic*, i.e. are truths

---

after some work, $T \vdash (\forall \mathsf{v} \leq \overline{\mathsf{m}})\neg\mathsf{Prf}(\mathsf{v}, \overline{\ulcorner\varphi\urcorner})$. Whence we get $T \vdash \forall \mathsf{v}(\mathsf{Prf}(\mathsf{v}, \overline{\ulcorner\varphi\urcorner}) \to \overline{\mathsf{m}} \leq \mathsf{v})$.

So $T \vdash \forall \mathsf{v}(\mathsf{Prf}(\mathsf{v}, \ulcorner\varphi\urcorner) \to (\overline{\mathsf{m}} \leq \mathsf{v} \wedge \widetilde{\mathsf{Prf}}(\overline{\mathsf{m}}, \ulcorner\varphi\urcorner)))$. That implies $T \vdash \forall \mathsf{v}(\mathsf{Prf}(\mathsf{v}, \overline{\ulcorner\varphi\urcorner}) \to (\exists \mathsf{w} \leq \mathsf{v})\widetilde{\mathsf{Prf}}(\mathsf{w}, \overline{\ulcorner\varphi\urcorner}))$. So given our definition, $T \vdash \neg\mathsf{RProv}(\overline{\ulcorner\varphi\urcorner})$.

Hence if, in particular, $T \vdash \neg\mathsf{R}$, then $T \vdash \neg\mathsf{RProv}(\overline{\ulcorner\mathsf{R}\urcorner})$. But – by the definition of R – if $T \vdash \neg\mathsf{R}$, then $T \vdash \mathsf{RProv}(\overline{\ulcorner\mathsf{R}\urcorner})$. Contradiction. Hence $T \nvdash \neg\mathsf{R}$.

of logic-plus-definitions. On this view, we don't need 'intuitions' going beyond logic: reasoning from definitions is enough. The Fregean brand of deductivism is standardly dubbed 'logicism'.

Famously, Frege's attempt to be a logicist deductivist about arithmetic (and indeed about a lot more than basic arithmetic) hit the rocks, because – as Russell showed – his logical system is in fact inconsistent in a pretty elementary way (it is beset by Russell's Paradox). That devastated Frege, but Russell was undaunted, and still gripped by deductivist ambitions he wrote:

> All mathematics [yep! – *all* mathematics] deals exclusively with concepts definable in terms of a very small number of logical concepts, and ... all its propositions are deducible from a very small number of fundamental logical principles.

That's a big promisory note in Russell's *The Principles of Mathematics* (1903). And *Principia Mathematica* (1910–13) is Russell's attempt with Whitehead to make good on that promise. The project is to set down some logical axioms and definitions and deduce the laws of basic arithmetic (and then more) from them. Famously, they eventually get to prove that $1 + 1 = 2$ at *110.643 (Volume II, page 86), accompanied by the wry comment, 'The above proposition is occasionally useful'.

Now, *Principia*, frankly, is a bit of a mess – in terms of clarity and presentational rigour, it's quite a step backwards from Frege. And there are technical complications which mean that not all *Principia*'s axioms are clearly 'logical' even in a stretched sense.[3]. But leave those worries aside – they pale into insignificance compared with the bomb exploded by Gödel in 'On formally undecidable propositions of *Principia Mathematica*...'. For the First Incompleteness Theorem shows that any form of deductivism about even just basic arithmetic – despite the intuitive pull – is in fatal trouble. The proponent of deductivism about basic arithmetic (logicist or otherwise) wants to pin down first-order arithmetical truths about successor/addition/multiplication, without leaving any out, for he wants a complete story about the roots of arithmetic truth. So he will want to find a negation-complete set of axioms. We now know that there can't be such a set of axioms, at least if we want to be able to specify the axioms in a suitably controlled way, include at

[3] In particular, there's an appeal to a brute-force *Axiom of Infinity* which in effect states that there is an infinite number of objects; and then there is the notoriously dodgy *Axiom of Reducibility*. For more on this see http://plato.stanford.edu/entries/principia-mathematica/

least Q, and stay ω-consistent (and hence interpretable as, in part, still doing arithmetic).

So varieties of deductivism about arithmetic, and logicism in particular (despite its intuitive attractions), must necessarily fail.

4
The Incompleteness Theorems: Lecture 4

So far, we've looked at what was essentially Gödel's 1931 way of proving a theory T incomplete, souped up in just two ways.

(i) In Lecture 2, we noted Robinson's 1951 way of sharpening the notion of 'niceness' (i.e. of being p.r. axiomatized, and p.r. adequate). It was clear from the start that containing PA is enough for a theory to be p.r. adequate: Robinson observed that containing the induction-free Q in fact suffices.

(ii) In Lecture 3, we noted Rosser's 1936 way of dropping the assumption that T is ω-consistent – using Rosser's construction, we need only assume T is nice and consistent.

But note, none of this "classic" route to incompleteness depends on a developed general theory of computation (all we need is the idea of a p.r. function, and the idea of coding). In this lecture, we are going to look at three arguments for incompleteness that become available once we have some basic computability theory in place.

4.1 A proof using facts about r.e. sets of numbers

First, recall two facts about recursively enumerable sets:

(i) If the set S is non-empty and recursively enumerable, then there is a p.r. function which enumerates it, i.e. there is a primitive recursive function f such that $n \in S$ iff $\exists x\, f(x) = n$. (Why? If S is r.e., it is the range of some partial computable function φ. Now, map input x to the x-th pair of numbers $\langle s, t$, and now compute s stages of $\varphi(t)$. If that computation has halted with output n, put $f(x) = n$, otherwise set $f(x)$ to some default member of S.

Evidently, for every $n \in S$, $\exists x\, f(x) = n$. But the computation of f involves no unbounded searches, so f is primitive recursive).

(ii) There is a non-empty set of numbers K which is recursively enumerable, but whose complement is not r.e. (Just put $K = \{e \mid e \in W_e\}$, where W_e is set of $\{n \mid \varphi_e(n)\downarrow\}$, i.e. the set of n such that the e-th partial computable function φ_e yields a value for input n. A diagonal argument shows that the complement $\overline{K}$ is not r.e.

We'll say a set of sentences is r.e. if – under some sensible coding – the set of Gödel numbers for the sentences is r.e. (and requiring 'sensible' coding schemes to be p.r. intertranslatable ensures this definition is scheme-invariant).

Definition 19 *A language $\mathcal{L}$ is* sufficiently expressive *iff (i) $\mathcal{L}$ has numerical quantifiers, and (ii) for any (one place) p.r. function f, there is an $\mathcal{L}$-wff $\mathsf{F}(\mathsf{x},\mathsf{y})$ such that $\mathsf{F}(\overline{\mathsf{m}},\overline{\mathsf{n}})$ is a true sentence iff $f(m) = n$.*

In other words, $\mathcal{L}$ can faithfully express any p.r. function.

Now, take k to be a p.r. function which enumerates K (a set which is r.e. but whose complement isn't). If $\mathcal{L}$ is sufficiently expressive, there is a wff $\mathsf{K}(\mathsf{x},\mathsf{y})$ which expresses k, so such that $n \in K$ iff $\exists\mathsf{x}\,\mathsf{K}(\mathsf{x},\overline{\mathsf{n}})$ is true, and $n \in \overline{K}$ iff $\neg\exists\mathsf{x}\,\mathsf{K}(\mathsf{x},\overline{\mathsf{n}})$ is true.

But if we could recursively enumerate the truths of $\mathcal{L}$, we could recursively enumerate in particular the truths of the form $\neg\exists\mathsf{x}\,\mathsf{K}(\mathsf{x},\overline{\mathsf{n}})$ (where that's a numerical quantifier!), which would give us a recursive enumeration of the members of the complement $\overline{K}$. But that's impossible, given the definition of K. So

Theorem 17 *The truths of a sufficiently expressive $\mathcal{L}$ are not r.e.*

However,

Theorem 18 *The theorems of a p.r. axiomatized theory T are recursively enumerable.*

Just zig-zag through m, n pairs checking whether $Prf_T(m,n)$ holds; spit out n when it does.

So suppose T is p.r. axiomatized and has a sufficiently expressive language $\mathcal{L}$. Then T's theorems are r.e., but the truths of T's language are not r.e. : so the T-theorems and the truths are distinct sets. Suppose T is also sound. Then it can't overshoot by proving theorems which

aren't true. So T must undershoot. There is some true φ it can't prove. And since then $\neg\varphi$ is false, T can't prove that either.

Whence:

Theorem 19 *Any sound p.r. axiomatized theory with a sufficiently expressive language is negation-incomplete.*

So that gives us an incompleteness theorem using elementary results from computability theory, together with a semantic premiss about T. The overall proof involves diagonalization in the background, in proving $\overline{K}$ is not r.e.; and it doesn't explicitly deliver witness to incompleteness. But we do know some truth of the form $\neg\exists\mathsf{x}\,\mathsf{K}(\mathsf{x},\overline{\mathsf{n}})$ must be unprovable, with K as above (or else we could recursively enumerate $\overline{K}$ by listing the n such that $\neg\exists\mathsf{x}\,\mathsf{K}(\mathsf{x},\overline{\mathsf{n}})$ is a theorem). Add the info that K can be Σ_1 and it follows that there is a Π_1 undecidable sentence.

4.2 A proof using unsolvability of Halting Problem

Start with a natural definition:

Definition 20 *Put $h(e,n,j) = 1$ iff the computation of Turing machine e on input n halts within j steps, and $h(e,n,j) = 0$ otherwise.*

We can call h the halting characteristic function, and evidently h is p.r. – for we can determine the value of $h(e,n,j)$ in j steps, i.e. without an unbounded search.

Now, if $\mathcal{L}$ is sufficiently expressive, then there is an $\mathcal{L}$-wff H such that $\mathsf{H}(\overline{\mathsf{e}},\overline{\mathsf{n}},\overline{\mathsf{j}},\overline{\mathsf{1}})$ is true iff $h(e,n,j) = 1$. In particular, then, $\mathsf{S}(\overline{\mathsf{e}}) =_{\text{def}} \exists\mathsf{x}\,\mathsf{H}(\overline{\mathsf{e}},\overline{\mathsf{e}},\mathsf{x},\overline{\mathsf{1}})$ is true iff the Turing machine number e halts on input e. (NB, for future reference, can be Σ_1.)

Suppose again that we could recursively enumerate the truths of the sufficiently expressive $\mathcal{L}$. Start enumerating, and eventually either $\mathsf{S}(\overline{\mathsf{e}})$ or $\neg\mathsf{S}(\overline{\mathsf{e}})$ must turn up. So that would effectively decide whether Turing machine number e halts on input e. But we know from the unsolvability of the self-halting problem that that can't the effectively decided. So we cannot recursively enumerate the truths of a sufficiently expressive $\mathcal{L}$, which re-proves Theorem 17 again. Then the argument can proceed as before to a semantic incompleteness theorem.

However, we can also give another argument from unsolvability of the self-halting problem to an incompleteness result, this time relying only on the syntatic assumption of ω-consistency.

For suppose T is a normal theory. It can then capture h by a wff H. Define as before $\mathsf{S}(\overline{\mathsf{e}}) =_{\text{def}} \exists \mathsf{x}\, \mathsf{H}(\overline{\mathsf{e}}, \overline{\mathsf{e}}, \mathsf{x}, \overline{1})$.

Now consider a Turing machine that, given input e, runs through values of m checking whether $Prf_T(m, \ulcorner \neg \mathsf{S}(\overline{\mathsf{e}}) \urcorner)$ holds, and halts if it finds such an m, and trundles on for ever otherwise. Clearly there is such a machine S (which tries to see if T can prove that machine e fails to halt on its own index as input), even if it would be a pain to program it.

Let machine S have index s. We now ask: does it halt on input s? If it does, then for some m, $Prf_T(m, \ulcorner \neg \mathsf{S}(\overline{\mathsf{s}}) \urcorner)$, so $T \vdash \neg \mathsf{S}(\overline{\mathsf{s}})$. But also, if machine index s halts on input s we have $h(e, e, m) = 1$ for some m, hence $T \vdash \mathsf{H}(\overline{\mathsf{e}}, \overline{\mathsf{e}}, \overline{\mathsf{m}}, \overline{1})$, hence $T \vdash \mathsf{S}(\overline{\mathsf{s}})$. So assuming T is consistent S doesn't halt on input s.

Suppose now $T \vdash \neg \mathsf{S}(\overline{\mathsf{s}})$. Then for some m, $Prf_T(m, \ulcorner \neg \mathsf{S}(\overline{\mathsf{s}}) \urcorner)$, which (by definition of S) entails that S halts on input s.

But we've just seen that, assuming T is consistent, S *doesn't* halt on input s. So if T is consistent, $T \nvdash \neg \mathsf{S}(\overline{\mathsf{s}})$.

Suppose alternatively that $T \vdash \neg\neg \mathsf{S}(\overline{\mathsf{s}})$, i.e. $T \vdash \exists \mathsf{x}\, \mathsf{H}(\overline{\mathsf{s}}, \overline{\mathsf{s}}, \mathsf{x}, \overline{1})$. Assuming T's consistency again, S doesn't halt on input s by the m-step of the computation, for any m. So for every m, not-$h(s, s, m, 1)$, hence for every m, $T \vdash \mathsf{H}(\overline{\mathsf{s}}, \overline{\mathsf{s}}, \overline{\mathsf{m}}, \overline{1})$, making T ω-inconsistent.

If we now put $\mathsf{G} =_{\text{def}} \neg \mathsf{S}(\overline{\mathsf{s}})$, which can be Π_1, this all goes to reprove

Theorem 7 *If T is normal, then there is a Π_1 sentence G such that (i) if T is consistent, $T \nvdash \mathsf{G}$, and (ii) if T is ω-consistent, $T \nvdash \neg \mathsf{G}$.*

4.3 For fun! – another proof using Kleene's Theorem

Recall Kleene's Normal Form Theorem:

Theorem 20 *There is a three-place p.r. function C and a one-place p.r. function U such that any one-place partial recursive function can be given in the standard form*

$$\varphi_e(n) =_{\text{def}} U(\mu z[C(e, n, z) = 0])$$

for some value of e.

We'll use this first to prove a result about 'completions' of partial functions. We say

Definition 21 *The function f is a* completion *of a partial function φ if f is total and for all n where $\varphi(n)$ is defined, $f(n) = \varphi(n)$.*

We then have:

Theorem 21 *Not every partial recursive function has a recursive completion.*

Proof Put $\delta(n) \simeq U(\mu z[C(n, n, z) = 0]) + 1$. So $\delta(n) = \varphi_n(n) + 1$ when that is defined and is undefined otherwise.

$\delta(n)$ is by construction partial recursive. Suppose f completes it. Then for some c, then, f is the total function φ_c – remember, the partial recursive functions include the total recursive functions. So in particular, (i) $f(c) = \varphi_c(c)$.

But then $\delta(c) = \varphi_c(c) + 1$ is defined and, by definition of f, (ii) $f(c) = \delta(c) = \varphi_c(c) + 1$.

However, (i) contradicts (ii): so there can be no such completion of δ. □

We'll now assume for reductio that T is (i) p.r. axiomatized, (ii) p.r. adequate, and (iii) ω-consistent (and hence consistent), yet is (iv) negation complete.

Since T is p.r. adequate, there will be a four-place wff C by which it can represent the p.r. function C that appears in Kleene's Normal Form theorem. So consider the following definition,

$$f_e(n) = \begin{cases} U(\mu z[C(e, n, z) = 0]) & \text{if } \exists z[C(e, n, z) = 0] \\ 0 & \text{if } T \vdash \forall \mathsf{z} \neg \mathsf{C}(\overline{\mathsf{e}}, \overline{\mathsf{n}}, \mathsf{z}, \mathsf{0}) \end{cases}$$

We show that – given our assumptions about T – the conditions are exclusive and exhaustive, and so well-define a recursive total function for any e. But that would make any $\varphi_e(n) (= U(\mu z[C(e, n, z) = 0]))$ have a recursive completion, contradicting the last theorem.

Hence, if T satisfies (i), (ii) and (iii), it can't satisfy (iv), i.e. it is not negation complete.

So to complete the argument we just need to show that if T satisfies (i) to (iv), the two definitional conditions are (A) mutually exclusive, and (B) exhaustive, and then (C) f_e is recursive.

Proof for (A) Assume (a) $C(e, n, k) = 0$ for some number k. Then $T \vdash \mathsf{C}(\overline{\mathsf{e}}, \overline{\mathsf{n}}, \overline{\mathsf{k}}, \mathsf{0})$. So we can't have $T \vdash \forall \mathsf{z} \neg \mathsf{C}(\overline{\mathsf{e}}, \overline{\mathsf{n}}, \mathsf{z}, \mathsf{0})$ given that T is consistent.

Proof for (B) Suppose the first condition in the definition of f_e doesn't hold. Then for every k, it isn't the case that $C(e, n, k) = 0$. So since T represents C, for every k, $T \vdash \neg\mathsf{C}(\overline{\mathsf{e}}, \overline{\mathsf{n}}, \mathsf{k}, \mathsf{0})$. By hypothesis T is ω-consistent, so $T \nvdash \exists\mathsf{z}\,\mathsf{C}(\overline{\mathsf{e}}, \overline{\mathsf{n}}, \mathsf{z}, \mathsf{0})$. Hence since T is negation complete and it can't prove $\exists\mathsf{z}\,\mathsf{C}(\overline{\mathsf{e}}, \overline{\mathsf{n}}, \mathsf{z}, \mathsf{0})$, it proves its negation, i.e. $T \vdash \forall\mathsf{z}\,\neg\mathsf{C}(\overline{\mathsf{e}}, \overline{\mathsf{n}}, \mathsf{z}, \mathsf{0})$. So the second condition holds

Proof for (C) It remains to show – still assuming those conditions on T – that f_e is recursive.

For input n, zig-zag between two searches. One search runs through $k = 0, 1, 2, \ldots$ and looks to see if k such that $C(e, n, k) = 0$ (and if and when we first find one, then we put $f_e(n) = U(\mu z[C(e, n, z) = 0])$, as instructed). The other search runs through the proofs of T, looking to see if $\forall\mathsf{z}\neg\mathsf{C}(\overline{\mathsf{e}}, \overline{\mathsf{n}}, \mathsf{z}, \mathsf{0})$ gets proved (and then we put $f_e(n) = 0$). Each of those searches can be effectively pursued. And eventually one of the searches must terminate since the alternatives are exhaustive, and give us a value for $f_e(n)$.

That shows f_e is effectively computable. Apply Church's Thesis in labour-saving mode to conclude it is recursive.

Which completes the proof. Exercise: mine the proof to extract the further information that T will be incomplete even for Π_1 sentences.