OXFORD UNIVERSITY

Part C Maths, Maths & Phil, Maths & Comp Sci
M.Sc. in Math Sci, M.Sc. in Maths and Foundations of Comp Sci
B.Phil. and M.St. in Philosophy

16 lectures on

Gödel's Incompleteness Theorems

Hilary Term 2019

Daniel Isaacson
Faculty of Philosophy
Oxford University

6 March 2019

Contents

Lecture 1

Introduction: a weak form of Gödel's First Incompleteness Theorem; the symbols and expressions of a language for arithmetic $\mathcal{L}_A$; Gödel numbering of the expressions of $\mathcal{L}_A$

Monday, 14 January 2019

1.1 Introduction: a weak form of Gödel's First Incompleteness Theorem

The context of Kurt Gödel's discovery of the phenomenon of formal incompleteness, published in 1931 as "Über formal unentscheidbare Sätze der *Principia mathematica* und verwandter Systeme I" ("On formally undecidable propostions of *Principia mathemataica* and related systems I"), is David Hilbert's programme for justifying the use of the axiomatic method in mathematics by establishing the consistency of systems formalizing the various branches of mathematics.

Over the course of the nineteenth century, mathematicians had established the consistency of non-Euclidean geometry relative to the consistency of Euclidean geometry, e.g. by interpreting a two-dimensional non-Euclidean geometry as the geometry of a curved surface in three-dimensional Euclidean geometry. What of the consistency of Euclidean geometry? A basis for conviction that Euclidean geometry is consistent is its interpretation in the theory of real numbers, i.e. as Cartesian geometry, insofar as we are convinced that the theory of real numbers is consistent. But then on what basis can we be convinced that the theory of real numbers is consistent? The theory of real numbers can be interpreted in set theory. That interpretation has lots of virtues, but giving us a basis for confidence that the theory of real numbers is consistent probably isn't one of them, since the consistency of set theory has been a real issue, while the consistency of the theory of real numbers isn't as much of a worry, though it's still an issue. In the published text from his address to the World Congress of Mathematicians in Paris in 1900, Hilbert set out twenty-three problems whose solution he considered to be of the greatest importance to the development of mathematics at that time. Problem number 2 as "to find a direct proof for the consistency of the arithmetical [he meant the arithmetic of the real numbers] by means of a careful study and suitable modification of the known methods of reasoning in the theory of irrational numbers." [Hilbert [6], p. 1104]

In 1918 he declared, in his paper "Axiomatisches Denken", that

> we must... make the concept of specifically mathematical proof itself into an object of investigation (Hilbert [7], p. 1115).

Hilbert formulated the distinction between finitary and infintary mathematics as a tool by which to justify and develop his use of the axiomatic method in mathematiacs, and thereby his philosophy of mathematics. The paradigm of finitary mathematics is arithmetical calculation. Finitary mathematics is mathematical bedrock, corresponding to observation statements in science. A calculation such as $2^7 = 128$ is finitary, but the claim that exponentiation to the power 2 always yields a value, i.e. $\forall x \exists y(2^x = y)$ is infinitary, and more generally, quantification over the infinite domain of natural numbers is infinitary. However, quantification over a bounded, i.e. initial, and thereby finite, segment of the natural numbers belongs to finitary mathematics.

Hilbert's deep insight was to recognize that the formal manipulation of all symbols, not just the symbols for numbers, i.e. numerals and terms built up from numerals and symbols for arithmetical operations, belongs to finitary mathematics. In particular,

> a formalized proof, like a numeral, is a concrete and surveyable object. ([8], p. 383 and also in [9], p. 471.)

Hilbert recognized two sorts of finitary statements, general and particular (though he did not introduce terminology for this distinction). Particular finitary statements are decided by computations, e.g. $7 \times 5 = 35$, and $2^{10} = 1024$ and truth functional combinations of them (the truth values of such combinations being computable from the truth values of the component statements). General finitary statements contain free variables, and can be thought of as a template for particular finitary statements that result by substitution of numerals for the free variables, for example $x + y = y + x$, and $(n > 2 \supset x^n + y^n \neq z^n)$. On the other hand, $\forall x \forall y\, x + y = y + x$ and $\forall n \forall x \forall y \forall z (n > 2 \supset x^n + y^n \neq z^n)$ are infinitary.

For $F(v_1)$ a general finitary statement with free variable v_1, *bounded* quantification on the variable v_1, which is finitary, is expressible using (apparently) unbounded quantification by, in the case of universal quantification, $\forall v_1(v_1 \leq t \supset F(v_1))$, for t a term in the language of arithmetic, which we abbreviate as $(\forall v_1 \leq t)F(v_1)$, and in the case of existential quantification, $\exists v_1(v_1 \leq t \wedge F(v_1))$, which we abbreviate $(\exists v_1 \leq v_2)F(v_1)$. For t a term that denotes a number (either a numeral or a computable function applied to a numeral), $(\exists v_1 \leq t)F(v_1)$ and $(\forall v_1 \leq t)F(v_1)$ are particular finitary statements if v_1 is the only free variable in $F(v_1)$. If t is a free variable or an arithmetical function applied to one or more variables, $(\exists v_1 \leq t)F(v_1)$ and $(\forall v_1 \leq t)F(v_1)$ are general finitary statements.

Hilbert noted that general finitary statements are not closed under negation, i.e. the negation of a general finitary statement cannot be expressed as a general finitary statement. For example, Fermat's Last Theorem is expressible as a general finitary statement, $(n > 2 \supset x^n + y^n \neq z^n)$, but to say that Fermat's Last Theorem is false requires existential quantification, $\exists n \exists x \exists y \exists z (n > 2 \wedge x^n + y^n = z^n)$. On the other hand, the statement that a specific quadruple of numbers a, b, c, d is a counterexample, i.e. $(a > 2 \wedge a^n + b^n = c^n)$, is a particular finitary statement.

The statement that particular formal derivation is a proof of a particular formula in a specified formal system is, as Hilbert recognized, finitary, i.e. effectively decidable. However, Hilbert missed something about this insight of his which Gödel realized, namely that formal proofs can be *literally* identified with natural numbers, i.e. they could be taken to be numerical expressions, rather than merely like them. As Gödel put this point in (1931),

> Of course, for metamathematical considerations it does not matter what objects are chosen as primitive signs, and we shall assign natural numbers to this use, that is, we map the primitive signs one-to-one onto some natural numbers.

Numbers assigned to formulas of a formal language in this way are called Gödel numbers. There are, of course, an uncountable infinity of assignments of numbers to formulas, but also there are a countable infinity of assignments for which the coding of a formula by a number, and decoding of a number as a formula can be carried out effectively. The one we shall use here, which is not the one used by Gödel, is due to W.V. Quine (and is used by Raymond Smullyan in his book *Gödel's Incompleteness Theorems*, which is the starting point of these notes).

Definition 1 (notation for the Gödel number of an expression) *For a given assignment of Gödel numbers,* $\ulcorner E \urcorner =_{df}$ *the Gödel number of expression* E*.*

To carry out the arithmetization of syntax, the system must be able to "talk" about numbers, i.e. there must be for each natural number an expression in the language of the system that denotes that number.

Definition 2 (numerals for numbers) *A set of expressions of a given formal language such that each natural number is denoted by a unique expression in that set will be called numerals.*

Definition 3 (notaton for the numeral of a number) *For formal languages that have a numeral for each natural number, we denote by* $\overline{n}$ *the numeral for the natural number* n*.*

1.1.1 Truth

Truth for a sentence of $\mathcal{L}_{AE}$ in the structure of the natural numbers (the intended interpretation) can be defined by recursion over the recursive generation of the sentence in the usual way. For none of the results in this course do we require a formal definition of truth, and I will take it as known informally what it means for a formula in the language of arithmetic to be true in the structure of the natural numbers. As part of our informal notation, i.e. the mathematical language in which we talk about denotation and truth of terms and sentences of the formal language $\mathcal{L}_{AE}$, we denote the set of natural numbers by $\mathbf{N}$.

Definition 4 (truth in a language of arithmetic) *Whenever we speak off a sentence in a language of arithmetic, i.e. a closed formula, as true, we mean that it is true when interpreted in the domain of natural numbers with the usual arithmetical functions and relations on the natural numbers (also known as the standard model). When truth in a non-standard model is meant, that will be specified. Also, we speak of a formula with free variables as true if for all substitution of numerals for variables results in a true sentence, so for example we say that* $v_1 + v_2 = v_2 + v_1$ *is true.*

Note that when $\mathcal{L}_{AE}$ is interpreted over the natural numbers, in the standard interpretation, every element of the domain of interpretation, the numbers, is designated by a term in the language, the formal numerals. On this interpretation a sentence of the form $\exists v_i F(v_i)$ is true if and only if there is a number n such that the setence $F(\overline{n})$

is true. As we shall see later, there are intepretations of $\mathcal{L}_{AE}$ in which all the axioms of a given formal system of arithmetic are true, but in which there are elements of the domain, non-standard numbers, in which a sentence of the form $\exists v_i F(v_i)$ will be true, but for no natural number n is the sentence $F(\overline{n})$ true. This will be the case, for example, when a formula $F(n)$ has the meaning, on the standard interpretation, that n is the Gödel number of a proof of a sentence which the system refutes e.g. $0 = 1$, i.e. is a proof that the sytem is inconsistent. Gödel's Second Incompleteness Theorem, as we shall see, tells us that no consistent theory in which syntax can be arithmetized can prove its own consistency. Hence a sentence asserting that there is a proof in the system of an inconsistency can be consistently added to the system, and since this extended theory is consistent, it will have a model, and in that model $\exists v_i F(v_i)$ will be true, which is to say that for some element a in the domain of that model, a satisfies the formula $F(v_i)$, which we may express by writing $F(a)$, but for no number n is there a true *sentence* of the form $F(\overline{n})$.

Gödel showed that the property of being the Gödel number of a provable formula is expressible within any system which can express basic arithmetic, i.e. there is a formula $Pr(v_1)$ with one free variable, in the language of a formal system for arithmetic, S, such that for every formula X in the language of S, $S \vdash X$ if and only if $Pr(\overline{\ulcorner X \urcorner})$ is true. And thereby unprovability is expressed by $\sim Pr(\overline{\ulcorner X \urcorner})$. We shall establish the existence of such formulas for a particular formal system of arithmetic in Lecture 4.

Gödel also showed (in effect, though it was Carnap who first explicitly stated this general result) that for any formula with one free variable $F(v_1)$ (in particular a formula that expresses the property of being the Gödel number of an unprovable formula), there is a sentence D such that the equivalence $(D \equiv F(\overline{\ulcorner D \urcorner}))$ is true. D is called a diagonal sentence for $F(v_1)$. We shall establish this result in Lecture 3.

From these results and on the assumption that everything provable in a given system S is true (in the sense of Definition 4) (a very strong assumption, much stronger than is needed to establish incompleteness, but it is illuminating to consider this simple case), it is easy to see that for G such that $(G \equiv \sim Pr(\overline{\ulcorner G \urcorner}))$, $S \nvdash G$, G is true, and $S \nvdash \sim G$, as follows.

Theorem 1 (weak form of Gödel's first incompleteness theorem) *Let S be a theory such that for each natural number n there is a numeral $\overline{n}$ in the language of S, and assume that:*

(i) there is a mapping $\ulcorner\urcorner$ of the expressions of the language of S to natural numbers, and a formula $Pr(v_1)$ in the language of S, such that for each formula X, $S \vdash X$ if and only if the sentence $Pr(\overline{\ulcorner X \urcorner})$ is true;

(ii) there is a sentence G such that the sentence $(G \equiv \sim Pr(\overline{\ulcorner G \urcorner}))$ is true;

(iii) every theorem of S is true.

Then $S \nvdash G$, G is true, and $S \nvdash \sim G$.

Proof. (1) Suppose that $S \vdash G$.

(2) By (1) and (i), $Pr(\overline{\ulcorner G \urcorner})$ is true.

(3) From (2) and (ii), G is false.

(4) From (3) and (iii), $S \nvdash G$.

(5) Since (4) contradicts (1), we have by *reductio ad absurdum* that $S \nvdash G$.

(6) From (5) and (i), $Pr(\overline{\ulcorner G \urcorner})$ is false.

(7) From (6) and (ii), G is true.

(8) From (7) and (iii), $S \nvdash \sim G$. ▲

Remarks about this result:

This version of the first Gödel incompleteness theorem is weak since, while assumptions (i) and (ii) can be established, which we shall do, assumption (iii), soundness of the system with respect to truth in arithmetic, is a highly non-finitistic assumption which is much stronger than necessary. The unprovability of the Gödel sentence holds from the assumption that S is consistent, which is finitistic is also the minimal, i.e. necessary condition, since an inconsistent theory proves everything, i.e.

Proposition 2 *For any system S and sentence X, a proof that $S \nvdash X$ from the assumption of consistency is best possible.*

Proof. If S is inconsistent, it proves everything, since $((A \wedge \sim A) \supset B)$ is logically valid, so in particular $S \vdash X$. ▲

Corollary 3 *If there is a formula X such that $S \nvdash X$, then S is consistent.*

Proof. By contraposition of the proof of Proposition 2. ▲

Gödel sketches the proof of this weak form of the First Incompleteness Theorem in section 1 of his 1931 paper, and notes that "The purpose of carrying out the above proof with full precision in what follows is, among other things, to replace the second of the assumptions just mentioned [every provable formula is true in the interpretation considered] by a purely formal and much weaker one." [Gödel [5], p. 176 (151)].

In his introductory section Gödel notes that this argument is "closely related" to the argument for the "Liar" paradox. The argument does not lead to a contradiction since it starts from the assumption that G is provable, and so by *reductio ad absurdum* establishes that G is not provable. Use of the Liar paradox also shows, as we shall see, that unlike provability in a formal system, truth in a language of arithmetic cannot be expressed in the language.

We shall establish how much basic arithmetic is required for arithmetization of syntax, i.e. to prove assumptions (i) and (ii), which will be made precise by the notion of Σ_0-arithmetic, which essentially consists of truth functional combinations of computations with addition and multiplication. Exponentiation is not needed. This shows that arithmetized syntax is a proper sub-part of what Hilbert meant by finitist mathematics. Hilbert never gave a precise characterization of finitist mathematics, but it is clear that it includes all primitive recursive functions, so plus and times, but also exponentiation and beyond. On the other had, both addition and multiplication are needed for incompleteness, as shown by the fact that Presburger Arithmetic, which is the theory of zero, successor, and addition, is complete. This completeness is no consolation for anyone who hoped for a complete system of arithmetic, since the properties of numbers that can be expressed using only successor and addition is extremely limited.

Gödel's proof of the independence of the Gödel sentence from formal arithmetic was unprecedented, in two crucial ways. One is in the means by which the proof is established. In the previous hundred years the independence of Euclid's fifth postulate from the other postulates of geometry had been established by the construction of a model in which the first four postulates of Euclidean geometry hold, and in which the fifth postulate is false. By contrast, Gödel's result is purely syntactic (exploiting Hilbert's insight). The other difference is that the fifth postulate is neither true nor false, per se. It is true in Euclidean geometry and false in non-Euclidean geometries. The Gödel sentence is demonstrably true, though not demonstrable in the system for which it is constructed.

Proving that $S \nvdash \sim G$ from a weaker condition than soundness of S. That the Gödel sentence G for a system S is not refutable, i.e. $\sim G$ is not provable, requires a stronger condition on S than consistency, though a condition much weaker than the soundness of S is sufficient. This situation will be analyzed carefully in later lectures.

Incompleteness from consistency. In 1936 J. Barkley Rosser established incompleteness just on the minimal condition of consistency, i.e. he constructed a sentence R for system S such that if S is consistent, $S \nvdash R$, and $S \nvdash \sim R$, and I will prove Rosser's Theorem in Lecture 9. Rosser's Theorem is not a strengthening of Gödel's Theorem, i.e. Rosser has not proved on a weaker assumption than in Gödel's proof that the Gödel sentence is undecidabale by a consistent system, and we shall see that there are consistent systems S such that for G the Gödel sentence for S, $S \vdash {\sim}G$.

Sketch of Gödel's second incompleteness theorem. Given that provability in a system S is expressible by a formula $Pr(v_1)$ in the language of S, then by Corollary 3, the consistency of S can be expressed in the language of S by ${\sim}Pr(\overline{\ulcorner X \urcorner})$, for any sentence X such that $S \vdash {\sim}X$. Then the first half of Gödel's first incompleteness theorem for a system S can be expressed in the language of S by the sentence $({\sim}Pr(\overline{\ulcorner X \urcorner}) \supset {\sim}Pr(\overline{\ulcorner G \urcorner}))$. By dint of considerable hard work, we are able to show that this sentence is provable in S, i.e. $S \vdash ({\sim}Pr(\overline{\ulcorner X \urcorner}) \supset {\sim}Pr(\overline{\ulcorner G \urcorner}))$. Hence if $S \vdash {\sim}Pr(\overline{\ulcorner X \urcorner})$, $S \vdash {\sim}Pr(\overline{\ulcorner G \urcorner})$. But also by dint of considerable hard work we can show that $S \vdash (G \equiv {\sim}Pr(\overline{\ulcorner G \urcorner}))$, so then $S \vdash G$. But we will have established that if S is consistent, $S \nvdash G$, so we have S is consistent. Hence we have shown that if $S \vdash {\sim}Pr(\overline{\ulcorner X \urcorner})$, S is inconsistent. By contraposition, if S is consistent, $S \nvdash {\sim}Pr(\overline{\ulcorner X \urcorner})$, i.e. a consistent theory cannot prove its own consistency. This is a deep result, from which a great deal else follows, as we shall see.

1.2 The symbols and expressions of a language for arithmetic $\mathcal{L}_{AE}$

A formal system is constructed from a formal language, and a formal language consists of terms and formulas specified from among the expressions generated by concatenation from a finite alphabet of symbols. Following Smullyan [16] we will use an alphabet of 13 symbols. We shall use these symbols to generate two different languages, $\mathcal{L}_{AE}$, and a sublanguage of $\mathcal{L}_{AE}$, $\mathcal{L}_A$. We begin with $\mathcal{L}_{AE}$. The heuristic meaning of $\mathcal{L}_{AE}$ is a language of arithmetic with exponentiation (as a primitive notion in the language), and correspondingly $\mathcal{L}_A$ is a language of arithmetic without exponentiation as primitive.

Definition 5 (the primitive symbols of $\mathcal{L}_{AE}$) *The primitive symbols of the language $\mathcal{L}_{AE}$ are the following:*

$0 \ {}' \ (\ ) \ f \ {}_{,} \ v \ \sim \ \supset \ \forall \ = \ \leq \ \sharp$

These formal symbols will be used with the following intended meanings:

The symbol 0 denotes the natural number zero[1].

[1]Note that in this sentence I am being casual about the distinction between use and mention. That distinction is easily but cumbersomely dealt with by using quotation marks, in which case this given sentence would read: "The symbol '0' denotes the natural number zero", which is fine, though fussy, but the next sentence would become: "The symbol '′' denotes the successor function",

The symbol $'$ denotes the successor function.

The symbols (and) are left and right brackets.

The symbols f and v are for functions and variables, to which numerical subscripts in tally notation, i.e. iterations of the subscript $_{\prime}$ are attached. The strings of symbols $f_{\prime}, f_{\prime\prime}, f_{\prime\prime\prime}$ will denote the functions addition, multiplication, and exponentiation, respectively, which we will write informally as $+$, $\cdot$, and *exp* or x^y in the usual notation. There are an infinity of variables $v_{\prime}, v_{\prime\prime}, v_{\prime\prime\prime}, \ldots$, which we will usually write as $v_1, v_2, v_3, \ldots$. If we want to signify a variable without specifying which variable it is, we will write v_i, v_j etc or use informal variable letters x, y, z, u, v, w.

The symbol for the propositional connectives negation and implication are $\sim$ and $\supset$. The symbol for the universal quantifier is $\forall$.

The symbols $=$ and $\leq$ are for the two-place relations of equality and less than or equals.

The symbol $\sharp$ will be used to mark breaks between strings of symbols that are terms and formulas of the language when these strings of formulas occur in sequences of terms and formulas (to be defined in the next lecture).

An *expression* in the language is (almost) any finite string of these symbols. For a technical reason (to do with our choice of Gödel numbering) we exclude from the class of expressions strings of more than one symbol that begin with the symbol $'$. The set of expressions for the language $\mathcal{L}_{AE}$ is specified by the following recursive definition.

Definition 6 (expressions of $\mathcal{L}_{AE}$) *basis: Each one of the symbols* $0 \ ' \ (\ ) \ f \ _{\prime} \ v \ \sim \ \supset \ \forall \ = \ \leq \ \sharp$ *is an expression.*

recursion: If E_i and E_j are expressions, and E_i is not the symbol $'$, then the concatenation of E_i and E_j, i.e. the result of writing E_i directly followed by E_j, is an expression, which we sometimes symbolize as $E_i{}^\frown E_j$, or more often simply as E_iE_j.

Remark: Because we have by this definition excluded from the class of expressions strings of more than one symbol that begin with the symbol $'$, the expression $0''$ exists as $0'{}^\frown{}'$, but not as $0{}^\frown{}''$.

1.3 Gödel numbering of the expressions of $\mathcal{L}_{AE}$

.

which is difficult to read and looks silly.

We assign Gödel numbers to the expressions of $\mathcal{L}_{AE}$. This can be done in infinitely many ways. The way we shall do it, following Smullyan following Quine, makes the link between Gödel numbering of expressions as strings of formal symbols particularly transparent. Gödel's original method involved coding by exponents of prime factors. On our method each number is the Gödel number of an expression, while on Gödel's method not every number is a Gödel number. Having every number be a Gödel number makes the formulation of some results a little simpler but is not essential.

Definition 7 (notation for an expression in term so its Gödel number) $E_n =_{df}$ *the expression with Gödel number n.*

Corollary 4 (of Definitions 1 and 7) $\ulcorner E_n \urcorner = n$.

Proof. By Definition 1, $\ulcorner E_n \urcorner$ is the Gödel number of E_n. By Definition 7, the Gödel number of E_n is n. ▲

We are used to the idea that numbers are denoted by numerals and that numerals are not the same thing as numbers. The Roman numerals for the first five non-zero natural numbers are I, II, III, IV, V, while the Arabic numerals are 1, 2, 3, 4 , 5. The crucial property of the Arabic numerals is that they are constructed on a place-value system with a base of 10. That the system of numerals in common use is base 10 is presumably down to the contingent fact (it could have been otherwise) that human beings have 10 fingers. Any other number greater or equal to 2 gives a perfectly good numeral system with that base. Machine code for computers is in base 2, I take it. The number we write as 15 in base 10 we write as 1111 in base 2 (i.e. $15 = 8 + 4 + 2 + 1$) and as 13 in base 12, i.e. as $12 + 3$, taking the first 10 digits of base 12 notation to be the digits used in base 10 ; but to express 131 as given in base 10 in base 12, we need two more primitive symbols, say η and ϵ, so $131_{10} = \eta\epsilon_{12}$, i.e. $10 \times 12 + 11$.

We shall be using base 13 representation of numbers to generate Gödel numbers for $\mathcal{L}_{AE}$ by taking the Gödel number of each of the 13 primitive symbols to be the number given by a digit base 13. However, *within* $\mathcal{L}_{AE}$, our numerals for numbers will be given by a tally notation, rather than place values: the formal numeral for the number n is the expression $0\overbrace{'\cdots'}^{n}$, i.e. the result of n-many iterations of concatenating the symbol $'$ on the right starting with the symbol 0.

The following function plays a key role in our chosen system of Gödel numbering.

Definition 8 (concatenation of base b numerals) *For any natural numbers m and n and natural number $b \geq 2$, we denote by $m *_b n$ the number designated by the* *base b numeral that results from concatenating the base b numeral for m with the base b numeral for n.*

Note that $*_b$ is a function mapping each pair of natural numbers to a natural number. Natural numbers are not intrinsically in base b or any other base notation, and the role of b in this function is to specify a method for calculating the function. To calculate $m *_b n$, we express m and n in base b notation and concatenate the two expressions, in that order. By Definition 8, $m *_b n$ is the number whose base b notation is produced by that concatenation.

Examples: For $m = 673, n = 32$ (written in base 10), $m *_{10} n = 67332$ and $n *_{10} m = 32673$. For $m = 59, n = 0, m *_{10} n = 590$ and $n *_{10} m = 059 = 59$.

Remark: As illustrated by these examples, $*_b$ is not commutative. It is also not associative, e.g. $(17 *_b 0) *_b 59 = 17059 \neq 1759 = 17 *_b (0 *_b 59)$. Non-associativity only arises when the middle value is 0, but since we will include 0 as a Gödel number we cannot suppress parentheses in multiple computations with $*_b$ except by adopting a convention for reinstating them; we adopt the common convention of association to the left, i.e. $x *_b y *_b z = (x *_b y) *_b z$.

We assign Gödel numbers to expressions by first stipulating the Gödel numbers of the symbols. We assign to these thirteen symbols the numbers denoted by the thirteen digits of base 13 notation, where the digits for 10, 11, and 12 (as we write them in base 10) are taken to be η, ϵ, and δ, respectively.

Definition 9 (assignment of Gödel numbers to expressions) *By recursion over the recursive definition of expressions.*

Base case: The assignment of numbers to symbols is specified by

0	$'$	*(*	*)*	f	$\prime$	v	$\sim$	$\supset$	$\forall$	$=$	$\leq$	$\sharp$
1	*0*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	η	ϵ	δ

Recursion: For expressions X and Y, $\ulcorner X\frown Y \urcorner = \ulcorner X \urcorner *_{13} \ulcorner Y \urcorner$

It is in order for each expression to have a unique Gödel number that we stipulated in Definition 6 that the class of expressions does not contain strings of more than one symbol that begin with the prime symbol $'$. For example, if $'\forall$ were an expression, we would have $\ulcorner '\forall \urcorner = 09 = 9 = \ulcorner \forall \urcorner$, and if $''$ were an expression, we would have $\ulcorner ' \urcorner = 0 = \ulcorner '' \urcorner$.

There is a technical advantage in taking the base b to be a prime number but it is not essential. We can use base 10 and the operation $*_{10}$ even with thirteen symbols by, for example, assigning the thirteen symbols respectively the following numbers (written in base 10):

0	$'$	(	)	f	$,$	v	$\sim$	$\supset$	$\forall$	$=$	$\leq$	$\sharp$
1	0	2	3	4	5	6	7	89	899	8999	89999	899999

Of course on this assignment not every number is a Gödel number. But we can effectively tell the ones that are, i.e. we know that if an 8 or a 9 occurs in its base 10 notation, it must occur within a string of the form 89, 899, 8999, 89999, 899999, and we know which symbol is coded by counting the number of 9s in that string.

Lecture 2

Terms and formulas of the language $\mathcal{L}_{AE}$; Expressibility in $\mathcal{L}_{AE}$; Substitution and quasi-substitution of a numeral for a free variable in a formula

Wednesday 16 January 2019

2.1 Terms and formulas of the language $\mathcal{L}_{AE}$

Having specified the alphabet of primitive symbols for the language of arithmetic $\mathcal{L}_{AE}$ which we will be working in, and the notion of an expression in the language as any finite concatenation of these primitive symbols, except for a strong of more than one symbol that begins with a $\prime$, we now specify from among all expressions in $\mathcal{L}_{AE}$ those that we use in a language. These are of two kinds, terms and formulas. Terms and formulas are in turn of two kinds, closed (containing no free variables) and open (containing one or more free variables). Closed terms designate objects in the domain, under an interpretation of the language, and closed formulas are true or false in an interpretation of the language. Open terms can be transformed into closed terms by substituting a closed term for each free variable. Open formulas can be transformed into closed formulas by, for each free variable in the formula, substituting a closed terms for that free variable, or by prefixing a quantifier which binds that free variable. The terms and formulas of $\mathcal{L}_A$ are generated by the following definitions.

2.1.1 Terms

Definition 10 (Variables) *$v_,$ is a variable, and if the expression E is a variable then $E^\frown{}_,$, i.e. $E_,$, the concatenation of E and the subscript symbol '$_,$', is a variable.*

Remark: So formal variables in $\mathcal{L}_A$ are expressions of the form $v_,, v_{,,}, v_{,,,}, \ldots$. We will abbreviate the string of symbols consisting of the formal variable symbol v followed by n subscripts as v_n.

Definition 11 (Numerals) *The symbol 0 is a numeral. If the expression E is a numeral, then the expression $E^\frown{}'$, i.e. E', the concatenation of E with the prime symbol, is a numeral.*

So numerals in $\mathcal{L}_A$ are the expressions $0, 0', 0'', 0''', \ldots$. For each natural number n, the numeral for n in $\mathcal{L}_A$ is $0^{\overbrace{\prime\cdots\prime}^{n}}$, which by Definition 3 we abbreviate as $\overline{n}$. For example, $0'''''''$ is the formal numeral in $\mathcal{L}_A$ for the number 7, abbreviated as $\overline{7}$.

Corollary 5 (of the definition of numerals in $\mathcal{L}_A$) *Writing '$n+1$' as our informal notation for the next natural number after n, for any natural number n, $\overline{n+1}$ is $\overline{n}^\frown{}'$ (which we will usually write as $\overline{n}'$), i.e. the numeral for the number $n+1$ is the concatenation of the numeral for the number n and the symbol $'$.*

Proof. Concatenation of $'$ to a string of primes adds one prime to that string. Hence the numeral for $n+1$ is the concatenation of one $'$ to the numeral for n, i.e. $\overline{n+1} = \overline{n}^\frown{}'$, where this equation is of the form $t = t$. ▲

Definition 12 (Terms) *Among expressions of $\mathcal{L}_{AE}$, the class of* terms *is specified by the following recursive definition:*

Base clause: Each variable and each numeral is a term.

Induction clauses: If t is a term, then t' is a term. If t_1 and t_2 are terms, then $(t_1 f_, t_2), (t_1 f_{,,} t_2)$, and $(t_1 f_{,,,} t_2)$ are terms. As remarked in Lecture 1, the expressions $f_,, f_{,,}$, and $f_{,,,}$ in $\mathcal{L}_{AE}$ will be interpreted as addition, multiplication, and exponentiation, respectively.

Note that while the formal system for arithmetic with which we begin will have exponentiation as primitive, i.e. it will include axioms governing $f_{,,,}$, we shall show, following Gödel, that exponentiation can be expressed in terms of zero, successor, plus, and times, and we shall drop the generation of terms in the language from $f_{,,,}$.

Definition 13 (closed terms) *A term in which no variable occurs is called a closed term.*

2.1.2 Formulas

Definition 14 (Atomic formulas) *For t_1 and t_2 any terms of $\mathcal{L}_A$, an* atomic formula *is an expression of the form $t_1 = t_2$ or of the form $t_1 \leq t_2$.*

Definition 15 (Formulas) *The class of* formulas *is specified by the following recursive definition:*

Base clause: Every atomic formula is a formula.

Induction clauses: If F and G are formulas, then $\sim F$ and $(F \supset G)$ are formulas, and for every variable v_i, the expression $\forall v_i F$ is a formula. [Note that the formula $(F \supset G)$ is enclosed in brackets, but that the other two formation rules do not introduce new brackets.]

We will use the logical equivalences between conjunction, disjunction, and existential quantification and expressions in terms of $\sim$, $\supset$, and $\forall$ as abbreviations, i.e.

Definition 16 *For formulas A and B,*

$(A \wedge B) =_{df} \sim(A \supset \sim B)$;

$(A \vee B) =_{df} (\sim A \supset B)$;

$(A \equiv B) =_{df} ((A \supset B) \wedge (B \supset A)) =_{df} \sim((A \supset B) \supset \sim(B \supset A))$

$\exists v_i A =_{df} \sim\forall v_i \sim A$.

2.1.3 Free and bound variables; open formulas and sentences

Definition 17 (a variable occurs free in a formula) *(i) If F is an atomic formula, all occurrences of variables in F are free.*

(ii) If F is a formula and variable v_i occurs free in F, then variable v_i occurs free in $\sim F$.

(iii) For F and G are formulas and variable v_i occurs free in F or in G , then v_i occurs free in $(F \supset G)$.

(iv) For any formula F, if v_i occurs free in F, v_i is free in $\forall v_j F$ iff $j \neq i$.

Note that clause (iii) of this definition includes the case where v_i occurs free both in F and in G, since we take 'or' with its inclusive meaning, as in the truth function for $\vee$.]

Definition 18 (bound variables) *A variable occurs bound in a formula F iff it occurs in F and does not occur free in F.*

Definition 19 (open formulas) *A formula with one or more free variables is an open formula*

Definition 20 (closed formulas) *A formula with no free variables is a closed formula, also called a sentence.*

Notational convention: We write $F(v_i)$ to signify a formula in which the variable v_i occurs free. Other unspecified variables may occur free as well, unless we stipulate that v_i is the only variable free in $F(v_i)$. In the latter case we say that $F(v_i)$ is a one-place formula. Similarly we write $F(v_{i_1}, \ldots v_{i_k})$ for a formula in which variables $v_{i_1}, \ldots v_{i_k}$ occur free, possibly with other free variables, unless we stipulate that these are the only variables occurring free, in which case we say that F is a k-place formula.

Note that this convention on possible occurrence of free variables other than those explicitly shown is different from Smullyan, who says "We write $F(v_{i_1}, \ldots v_{i_k})$ for any formula in which $v_{i_1}, \ldots v_{i_k}$ are the only free variables." (p. 16). Since there are situations, e.g. in stating the Induction axioms, in which we need to allow for the possibility of other free variables than those explicitly shown, Smullyan's convention has in those situations to be violated, e.g. "$F(v_1)$ is to be any formula at all (it may contain free variables other than v_1)" (Smullyan, p. 29). It seems to me more coherent to allow unspecified other variablesin in general, and then to stipulate in any particula case, where necessary, that there are no other free variables.

Definition 21 (regular open formulas) *An open formula is* regular *if its k-many free variables are the first k variables.*

2.2 Denotation of closed terms in $\mathcal{L}_{AE}$, truth of sentences of $\mathcal{L}_{AE}$, expressibility of sets and relations of natural numbers by formulas of $\mathcal{L}_{AE}$

2.2.1 Denotation of closed terms

Proposition 6 *On the intended interpretation of the formal language $\mathcal{L}_{AE}$, each closed term denotes (designates, refers to) a particular natural number.*

Proof By induction over the recursive definition of terms:

(i) the numeral 0 denotes the natural number zero.

(ii) If the closed term c denotes n, then the closed term c' denotes the successor of n; if the closed terms c_1 and c_2 denote n_1 and n_2 respectively, then $(c_1 f_{\prime} c_2)$ denotes the sum of n_1 and n_2, $(c_1 f_{\prime\prime} c_2)$ denotes the product of n_1 and n_2, and $(c_1 f_{\prime\prime\prime} c_2)$ denotes n_1 raised to the power n_2. ▲

2.2.2 Expressibility

A very important notion in the arithmetization of syntax (and elsewhere in mathematical logic and mathematics more generally) is that of a formula in a given language *expressing* a property of objects, or a relation between objects, in a domain of interpretation of the language. For example, the first hypothesis in the weak form of Gödel's first incompleteness theorem (Theorem 1) in Lecture 1 is that the property of being the Gödel number of a formula provable in the given axiomatic theory is expressible in the language of that theory. We define this notion with respect to truth in the natural numbers (Definition 4).

Definition 22 (expressibility of relations) *A formula $F(v_1, \ldots, v_k)$ in $\mathcal{L}_{AE}$ with k-many free variables is said to* express *a relation $R \subseteq \mathbf{N}^k$ iff for every k-tuple $< n_1, \ldots, n_k >$ of natural numbers, the sentence $F(\overline{n}_1, \ldots, \overline{n}_k)$ is true iff $< n_1, \ldots, n_k > \in R$, in which case the relation R is said to be* expressible *in $\mathcal{L}_{AE}$.*

Definition 23 (expressibility of functions) *A function $f : \mathbf{N}^k \to \mathbf{N}$ is* expressible *in $\mathcal{L}_{AE}$ iff the relation $f(n_1, \ldots, n_k) = m$ is expressible in $\mathcal{L}_{AE}$.*

2.3 Concatenation of numbers in a given base notation is expressible in $\mathcal{L}_{AE}$.

The two-place function $f(m,n) = m *_b n$ introduced by Definition 8 in the previous lecture is expressible in $\mathcal{L}_{AE}$, as follows:

Lemma 7 *For a fixed number $b \geq 2$, the condition that v_1 is a power of b, which we abbreviate as $Pow_b(v_1)$, is expressible in $\mathcal{L}_{AE}$.*

Proof. $Pow_b(v_1)$ iff $\exists v_2(v_1 = b^{v_2})$, or more formally

$Pow_b(v_,)$ iff $\sim \forall v_{,,} \sim v_, = (\bar{b} f_{,,,} v_{,,})$. ▲

Lemma 8 *For $\ell_b(n)$ the length of the base b notation for n, i.e. the number of digits in the base b notation of n, the two-place relation $b^{\ell_b(v_1)} = v_2$ is expressible in $\mathcal{L}_{AE}$*

Proof. In all but the case where $v_1 = 0$, this is the condition that v_2 is the least power of b greater than v_1, e.g. for $b = 10$ and $v_1 = 935$, $\ell_{10}(935) = 3$, and 10^3 is the least power of 10 greater than 935. In the case of $v_1 = 0$, $\ell_b(0) = 1$, so $v_2 = b^1 = b$, but the least power of b greater than 0 is $b^0 = 1$, so for $v_1 = 0$, $v_2 = b$ (which is $\neq 1$ since b is a base and hence ≥ 2).

This relation is thus expressed by the following condition on v_1 and v_2.

$$((v_1 = 0 \wedge v_2 = b) \vee (v_1 \neq 0 \wedge Pow_b(v_2) \wedge v_1 < v_2 \wedge \forall v_3((Pow_b(v_3) \wedge v_1 < v_3) \supset v_2 \leq v_3))).$$

The above condition is expressible in $\mathcal{L}_{AE}$ by Lemma 7 and the fact that $v_1 < v_2$ is equivalent to $(v_1 \leq v_2 \wedge \sim v_1 = v_2)$. ▲

Theorem 9 *For any number $b \geq 2$, the relation $v_1 *_b v_2 = v_3$ is expressible in $\mathcal{L}_{AE}$.*

Proof. The relation $v_1 *_b v_2 = v_3$ is expressed by the condition that

$$v_1 \cdot b^{\ell_b(v_2)} + v_2 = v_3.$$

For example, $1570 *_{10} 365 = 1570365 = 1570000 + 365 = 1570 \cdot 10^3 + 365 = 1570 \cdot 10^{\ell_{10}(365)} + 365$.

This condition is equivalent to

$$\exists v_4(b^{\ell_b(v_2)} = v_4 \wedge ((v_1 \cdot v_4) + v_2) = v_3).$$

By Lemma 8, this relation is expressible in $\mathcal{L}_{AE}$. ▲

2.4 Substitution and quasi-substitution

The operation of substituting a numeral for a free variable in a formula lies at the heart of constructing a 'self referential' sentence such as one that 'says' 'This sentence is not provable in the given formal system'. We introduce notation for such substitutions in the more general case of substitution of a term for a variable, but having introduced that notation, the case that will concern us is substitution of terms that are numerals.

Notation for substitution of a term for a free variable in a term or a formula: We write $t_1(v_i/t_2)$ to signify the result of substituting the term t_2 for all free occurrence of v_i in the open term $t_1(v_i)$ and $F(v_i/t)$ to signify the result of substituting the term t for all (unless in a specified situation we allow the possibility of some but not all) free occurrence of v_i in the formula $F(v_i)$. In this notation we allow the possibility that v_i does not occur in t_1 or in $F(v_i)$, in which case $t_1(v_i/t_2)$ is t_1 and $F(v_i/t)$ is F. If the context makes it clear that the variable to be substituted for is v_i, we may simply write $t_1(t_2)$ or $F(t)$. If v_i is the only free variable in t_1 or in $F(v_i)$, and t is a closed term, then $t_1(v_i/t)$ is a closed term, and $F(t)$ is a closed formula (sentence).

If $v_{i_1}, \dots v_{i_k}$ are some (though not necessarily all) free variables occurring in the formula F, we write, for terms $t_1, \dots, t_k$, $F(v_{i_i}/t_1, \dots, v_{i_k}/t_k)$ for the result of substituting the terms $t_1, \dots, t_k$ for all (unless in a specified situation we allow the possibility of some but not all) free occurrences of $v_{i_1}, \dots v_{i_k}$ in $F(v_{i_1}, \dots v_{i_k})$, respectively. If it's clear from the context which term is to be substituted for which variable, we simply write $F(t_1, \dots, t_k)$. If $t_1, \dots, t_k$ are closed term and if $v_{i_1}, \dots v_{i_k}$ are the only variables that occur free in $F(v_{i_1}, \dots v_{i_k})$, and $t_1, \dots, t_k$ are substituted for all occurrences of $v_{i_1}, \dots v_{i_k}$ in F, then $F(v_{1_i}/t_1, \dots, v_{i_k}/t_k)$ is a closed formula (sentence). Similarly for substitutions into a term.

For $F(v_1, \dots, v_k)$ a regular open formula with k-many free variables (Definition 21) and $< t_1, \dots, t_k >$ a k-tuple of terms, the expression $F(t_1, \dots, t_k)$ is unambiguous, i.e. we don't need to stipulate which term is substituted for which variable. Again similarly for terms.

In the following two definitions and subsequently, I use $\doteq$ to signify that the two expressions joined by it are identical, and square brackets to indicate in which term or formula the substitution is being made.

Definition 24 (substitution of a numeral for a variable in a term)

Base: $[v_i](v_i/\overline{n}) \doteq \overline{n}$

For $i \neq j$, $[v_j](v_i/\overline{n}) \doteq v_j$

$[0](v_i/\overline{n}) \doteq 0$

Recursion:

For any term t, $[t'](v_i/\overline{n}) \doteq t(v_i/\overline{n})'$

For any terms t_1 and t_2, $[(t_1 f, t_2)](v_i/\overline{n}) \doteq (t_1(v_1/\overline{n}) f, t_2(v_1/\overline{n}))$, *and similarly for $f_{\prime\prime}$ and $f_{\prime\prime\prime}$.*

Definition 25 (substitution of a numeral for a free variable in a formula) *Base:*

$[t_1 = t_2](v_i/\overline{n}) \doteq t_1(v_1/\overline{n}) = t_2(v_i/\overline{n})$

$[t_1 \leq t_2](v_i/\overline{n}) \doteq t_1(v_1/\overline{n}) \leq t_2(v_i/\overline{n})$

Recursion:

$[\sim F(v_i)](v_i/\overline{n}) \doteq \sim F(v_i/\overline{n})$

$[(F \supset G)](v_i/\overline{n}) \doteq (F(v_i/\overline{n}) \supset G(v_i/on))$

If $i \neq j$, $[\forall v_j F](v_i/\overline{n}) \doteq \forall v_j F(v_i/\overline{n})$

$[\forall v_i F](v_i/\overline{n}) \doteq \forall v_i F(v_i))$

Note that by Definitions 24 and 25, if v_1 does not occur in term t, then $t(v_i/\overline{n}) \doteq t$, and if v_i does not occur free in formula F, then $F(v_i/\overline{n}) \doteq F$.

To express substitution of a numeral for a free variable in a formula in arithmetized syntax, we have to express the function $s(x, y, z) = \ulcorner E_x(v_z/\overline{y})\urcorner$ in $\mathcal{L}_{\mathcal{AE}}$. This can be done, but doing so requires decoding the formula $E_x(v_z/\overline{y})\urcorner$ from x, y, z by the recursive Definitions 24 and 25, which is complicated and cumbersome. Tarski has shown how to avoid these complications by arithmetizing a formula, which we will call quasi-substitution, that is logically equivalent to substitution of a numeral in a formula, but whose Gödel number is generated directly from the Gödel number of the formula into which a numeral is being substituted, without going through the recursive generation of the formula.

Definition 26 (quasi-substitution) *For $F(v_i)$ a formula of $\mathcal{L}_{AE}$ and $\overline{n}$ a numeral,* $F[\overline{n}] =_{df} \forall v_i(v_i = \overline{n} \supset F(v_i))$.

Lemma 10 (quasi-substitution logically equivalent to substitution) *For $F(v_i)$ a formula of $\mathcal{L}_{AE}$ and $\overline{n}$ a numeral,* $(\forall v_i(v_i = \overline{n} \supset F(v_1)) \equiv F(\overline{n}))$ *is logically valid.*

Proof. (i) Suppose $\forall v_i(v_i = \overline{n} \supset F(v_i))$. Then by universal instantiation, $(\overline{n} = \overline{n} \supset F(\overline{n}))$. Since $\overline{n} = \overline{n}$ is logically valid, by modus ponens, $F(\overline{n})$.

(ii) Suppose $F(\overline{n})$. Then by substitutivity of identity, $(v_i = \overline{n} \supset F(v_i))$. By universal generalization, $\forall v_i(v_i = \overline{n} \supset F(v_i))$. ▲

Note We could also have defined $F[\overline{n}]$ as $\exists v_i(v_i = \overline{n} \wedge F(v_i))$ since:

Lemma 11 $(\forall v_i(v_i = \overline{n} \supset F(v_i)) \equiv \exists v_i(v_i = \overline{n} \wedge F(v_i)))$ *is logically valid.*

Proof. Exercise. ▲

Lecture 3

Arithmetization of quasi-substitution and diagonal quasi-substitution; The Diagonal Lemma; expressibility of properties of sequence numbers

Monday, 21 January 2019

3.1 Arithmetization of quasi-substitution and diagonal quasi-substitution

Our first step in the arithmetization of syntax, i.e. showing that syntactic operations on expressions of $\mathcal{L}_{AE}$ correspond with operations on their Gödel numbers that are expressible in $\mathcal{L}_{AE}$, is to show that the function $s(m,n)$ that gives the Gödel number of the formula that results from the quasi-substitution into E_m of the numeral $\overline{n}$, i.e. $E_m[\overline{n}]$, is expressible in $\mathcal{L}_{AE}$.

Definition 27 (the Gödel number of a quasi-substition)

$$s(x,y,z) =_{df} \ulcorner \forall v_z(v_z = \overline{y} \supset E_x)\urcorner$$

Note that for many values of x, E_x will not be a formula in which the variable v_z occurs free, or indeed will not be a formula at all. These are 'don't care' cases.

We could rule them out by stipulating that $s(x, y, z)$ has a value only if E_x is a formula in which v_z occurs free, meaning that $s(x, y, z)$ is a partial function. For our purposes it's a bit simpler to let all the functions we deal with be total.

Before we can establish that the three-place relation $s(m, n) = r$ is expressible in $\mathcal{L}_{AE}$, we must calculate the Gödel numbers of the numerals.

Lemma 12 *The Gödel number of $\overline{n}$, the numeral of the number n, is 13^n (where thirteen has been written in base 10, but the calculation of the Gödel number uses the concatenation of base 13 numerals).*

Proof. The numeral of the number n is the expression $0\overbrace{{}' \cdots {}'}^{n}$. The symbol 0 is assigned Gödel number 1, the symbol $'$ is assigned the number 0, so that whole expression is, by concatenation, assigned the number written in base-13 notation as $1\overbrace{0 \ldots 0}^{n}$, which is the number 13^n, with thirteen written in base 10 notation. ▲

Theorem 13 (expressibility of the quasi-substitution function) *There is a three-place formula in $\mathcal{L}_A$, call it $S(v_1, v_2, v_3)$, such that for each triple of natural numbers $< n_1, n_2, n_3 >$, $S(\overline{n}_1, \overline{n}_2, \overline{n}_3)$ is true if and only if $\ulcorner \forall v_1(v_1 = \overline{n}_2 \supset E_{n_1})\urcorner = n_3$, i.e. by Definitions 22 and 23, $S(v_1, v_2, v_3)$ expresses the function $s(m, n) = r$ specified in Definition 27.*

Proof. Let $k = \ulcorner \forall v_1(v_1 =\urcorner$, a particular number whose base 13 notation, given our assignment of base 13 digits to the symbols of our language, is 965265η (or if we use the base 10 Gödel numbering also given in Lecture 1, whose base 10 notation is 899652658999).

Going with our base 13 Gödel numbering, we have that the required relationship between v_1, v_2, v_3 is

$$v_3 = \overline{k} *_{13} (\overline{13} f_{'''} v_2) *_{13} \overline{8} *_{13} v_1 *_{13} \overline{3},$$

but as it stands this does not give us expressibility in $\mathcal{L}_{AE}$, since the expressibility in $\mathcal{L}_{AE}$ of $*_{13}$ as a two place function is given by a three-place formula in $\mathcal{L}_{AE}$, call it $C_{13}(v_1, v_2, v_3)$ (established in Theorem 9), so the required relationship between v_1, v_2, v_3 is expressed in $\mathcal{L}_{AE}$ (omitting the subscript 13 to reduce clutter) as

$$\exists v_4 \exists v_5 \exists v_6(((C(\overline{k}, (\overline{13} f_{'''} v_2), v_4) \wedge C(v_4, \overline{8}, v_5)) \wedge C(v_5, v_1, v_6)) \wedge C(v_6, \overline{3}, v_3))$$

▲

Definition 28 (diagonal quasi-substitution) *The diagonal quasi-substitution function $d(n) =_{df} s(n, n)$, i.e. $d(n) = \ulcorner \forall v_1(v_1 = \overline{n} \supset E_n)\urcorner$.*

Remark. By Definition 26, $\forall v_1(v_1 = \overline{n} \supset E_n) =_{df} E_n[\overline{n}]$, so Definition 28 means that $d(n) = \ulcorner E_n[\overline{n}]\urcorner$.

Corollary 14 (expressibility of diagonal quasi-substitution) *There is a two-place formula in $\mathcal{L}_{AE}$, call it $D(v_1, v_2)$, such that for each pair of natural numbers $< m, n >$, $D(\overline{m}, \overline{n})$ is true if and only if $n = \ulcorner\forall v_1(v_1 = \overline{m} \supset E_m)\urcorner$, i.e. the one-place function that yields the Gödel number of a quasi-substution formula as a function of the number whose numeral is being quasi-substituted into the formula which has that number as its Gödel number, $d(n) = \ulcorner\forall v_1(v_1 = \overline{n} \supset E_n)\urcorner$, is expressible in $\mathcal{L}_{AE}$.*

Proof. Let $D(v_1, v_2)$ be the formula that results by substituting v_1 for v_2 and v_2 for v_3 in $S(v_1, v_2, v_3)$. For all numbers m, n, $S(\overline{m}, \overline{m}, \overline{n})$ is true iff $s(m, m) = n$. By the definition of $d(m)$, $s(m, m) = d(m)$. So $D(\overline{m}, \overline{n})$ is true iff $d(m) = n$. ▲

3.2 The Diagonal Lemma

The existence of a sentence G such that the equivalence $(G \equiv \sim Pr(\overline{\ulcorner G\urcorner}))$ is true, i.e. condition (ii) of Theorem 1, is proved by the Diagonal Lemma, which establishes the existence of a diagonal sentence for any formula $F(v_i)$ with one free variable. Proof of the Diagonal Lemma is by a kind of double substitution into $F(v_i)$, first substitution of an expression for the diagonal function in place of the variable v_i, and then substitution of the numeral for the Gödel number of the formula that results from that first substitution in place of the free variable of the diagonal function as substituted into $F(v_i)$.

Theorem 15 (Diagonal Lemma) *For any open formula $F(v_i)$ in $\mathcal{L}_{AE}$ with one free variable v_i, there exists a sentence C in $\mathcal{L}_{AE}$ such that the equivalence $(C \equiv F(\overline{\ulcorner C\urcorner}))$ is true.*

Proof In this derivation $D(v_1, v_2)$ is the formula in $\mathcal{L}_{AE}$ proved to exist in Corollary 14.

(1)	(1) $k = \ulcorner\forall v_2(D(v_1, v_2) \supset F(v_2))\urcorner$	Assumption(*)
(1)(2)	(2) $C \doteq \forall v_1(v_1 = \overline{k} \supset \forall v_2(D(v_1, v_2) \supset F(v_2)))$	Assumption
(2)	(3) $(C \equiv \forall v_2(D(\overline{k}, v_2) \supset F(v_2))$	(2) and Lemma 10
	(4) $D(\overline{k}, \overline{d(k)})$	Corollary 14
	(5) $\forall v_2(D(\overline{k}, v_2) \supset F(v_2)) \supset (D(\overline{k}, \overline{d(k)}) \supset F(\overline{d(k)}))$	logically valid
	(6) $\forall v_2(D(\overline{k}, v_2) \supset F(v_2)) \supset F(\overline{d(k)})$	(4)(5) logical inference
(2)	(7) $(C \supset F(\overline{d(k)}))$	(3)(6) logic
(8)	(8) $F(\overline{d(k)})$	Assumption
(8)	(9) $(v_2 = \overline{d(k)} \supset F(v_2))$	(8) substitutivity of =

	(10) $\forall v_2(D(\overline{k}, v_2) \supset v_2 = \overline{d(k)})$	Corollary 14 and the fact that $d(x)$ is a function
	(11) $(D(\overline{k}, v_2) \supset v_2 = \overline{d(k)})$	(10) $\forall$-Elimination
(8)	(12) $(D(\overline{k}, v_2) \supset F(v_2))$	(11)(9) transitivity of $\supset$
(8)	(13) $\forall v_2(D(\overline{k}, v_2) \supset F(v_2))$	(12) $\forall$-Introduction(**)
(8)	(14) $\forall v_1(v_1 = \overline{k} \supset \forall v_2(D(v_1, v_2) \supset F(v_2)))$	(13) and Lemma 10
(8)(2)	(15) C	(14)(2)
(2)	(16) $(F(\overline{d(k)}) \supset C)$	(8)(15) $\supset$-Introduction
(2)	(17) $(C \equiv F(\overline{d(k)}))$	(7)(16) $\equiv$-Introduction
(1)(2)	(18) $d(k) = \ulcorner C \urcorner$	(1)(2) (***)
(1)(2)	(19) $(C \equiv F(\overline{\ulcorner C \urcorner}))$	(17)(18) substitutivity of $=$

(*)The formula $F(v_2)$ is $F(v_i/v_2)$, subject to the proviso that v_i is free for v_2 in F, meaning that in F, v_i does not occur within the scope of a quantifier which binds the variable v_2. If v_i occurs within the scope of $\forall v_2$ in F, change the variable of quantification and the occurrence of v_2 which it binds to a variable v_j which doesn't occur in F, which results in a formula logically equivalent to $F(v_i)$.

(**)Justified since v_2 is not free in the assumption $F(\overline{d(k)})$

(***)The formula C is the quasi-substitution of the numeral for the Gödel number of the formula into which it is being quasi-substituted, k, by (1) and (2), so by Definition 28, $\ulcorner C \urcorner$ is the value of the function $d(v_1)$ applied to k. ▲

3.3 Properties of sequences of digits

The first tool we need in order to code sequences of numbers in $\mathcal{L}_{AE}$ is to show that we can express in $\mathcal{L}_{AE}$ the relations that the base b notation of a number m *begins*, or *ends*, or *is part of* the base b notation of a number n.

Definition 29 (x begins y) x begins *y in base b notation iff the base b notation of x is a (not necessarily proper) initial segment of the base b notation of y.* We write this as xB_by.

Examples. In base 10, 2 begins 20, but note that in base 13, 2 (as it is written in base 10, and in base 13) does not begin 20, since 20 base 10 is written 17 base 13. Other examples (base 10): The numbers which written in base 10 are 7, 76, 760, 7600, 76007, 760074, and 7600748 all *begin* 7600748 in base 10. The last of these examples points up the fact that every number begins itself, i.e. an initial segment need not be a *proper* initial segment. **Note** that the number 0 does not begin any number except itself, i.e. we don't say that 0 begins 760748, even though 0760748 = 760748.

Definition 30 (x ends y) *x ends y in base b notation, which we write as xE_by, if the base b notation of x is an end segment (not necessarily proper) of the base b notation of y.*

Examples. In base 10, the following numbers all end 7600748: 7600748, 600748, 748, 48, 8.

Given the notion of one number beginning another in a given base representation and the notion of one number ending another in a given base, we can define the notion of a number being *part* of another in a given base in terms of these two notions:

Definition 31 (x is part of y) *x is* part *of y, in base b notation, which we write as xP_by, if x ends some number that begins y.*

Remark. Every number is a base b part of itself. Given a base b notation for x, every proper sub-segment of the base b notation that does not begin with a 0 is the base b notation of a number y that is a base b part of x. In base 10, the parts of 2600748 are all the numbers that begin or that end it, and 60074, and all the numbers that begin or end it, and 7.

Theorem 16 *For any $b \geq 2$ the following relations are expressible in $\mathcal{L}_{AE}$: (1) xB_by, (2) xE_by, (3) xP_by and, for any natural number $n \geq 2$, (4) $x_1 *_b \ldots *_b x_nP_by$*

Proof. We will prove a stronger result, which we need later, that these relations not only are expressible in $\mathcal{L}_{AE}$, but also that this can be done using only bounded quantifiers, i.e. that these are finitary properties of numbers.

1. If 0 does not occur in the base b numeral for y, then x begins y just in case there exists z such that $x *_b z = y$. However, if a zero or a string of zeros occurs in the base b numeral for y and the base b numeral for x is an initial segment of the base b numeral of y which ends just before the 0 or string of 0s in the base b numeral for y or includes some but not all of those 0s, then the numeral of x has to be extended by the remaining 0s before it can be concatenated with a numeral to result in the numeral for y. The extension of the base b numeral for x by the required number of 0s is accomplished by multiplying x by b raised to the power of how many 0s need to be appended. This condition can be expressed in terms of the previously expressed notions $Pow_b(w)$ and $x *_b z = y$, as follows:

$$xB_by \text{ iff } (x = y \lor (x \neq 0 \land (\exists z \leq y)(\exists w \leq y)(Pow_b(w) \land (x \cdot w) *_b z = y)))$$

The bounds on the quantifiers hold from the fact that if z is part of y, then $z \leq y$, and any number of the form 10...0 in base b with a string of 0s in y is $\leq y$.

2. xE_by iff $(x = y \vee (\exists z \leq y)(z *_b x = y)$

For this case we don't have any complications from the occurrence of zeros.

3. xP_by iff $(\exists z \leq y)(xE_bz \wedge zB_by)$

4. $x_1 *_b \ldots *_b x_nP_by$ iff $(\exists z \leq y)(x_1 *_b \ldots *_b x_n = z \wedge zP_by)$. ▲

Lecture 4

Expressibility of properties of sequence numbers in the language $\mathcal{L}_A$; A formal system $\mathbf{PA}_E$ for arithmetic; an arithmetized proof predicate for $\mathbf{PA}_E$; the weak form of Gödel's First Incompleteness Theorem for $\mathbf{PA}_E$ from Lecture 1 now proved; a weaker form of incompleteness of $\mathbf{PA}_E$ provd from the undefinability of truth for $\mathcal{L}_A$ in $\mathcal{L}_A$

Wednesday 23 January 2019

4.1 Expressibility of properties of sequence numbers in the language $\mathcal{L}_{AE}$

4.1.1 Sequence numbers

Treating sequences of expressions as expressions with Gödel numbers is very convenient. It requires making sequences of expressions into single expressions, which is done by expanding the langauge of arithmetic by introducing a symbol to mark the boundary between two successive expressions, for which we use $\sharp$. We included $\sharp$ among the primitive symbols of $\mathcal{L}_{AE}$ (Lecture 1), which did not enter into the rules for the formation of terms and formulas of $\mathcal{L}_{AE}$. When a sequence of expressions has been concatenated with $\sharp$, the whole sequence of expressions will itself be an expression, which has a Gödel number.

Definition 32 (sequence number) *x is a* sequence number *if it is the Gödel number of an expression of the form $\sharp E_{i_1}\sharp E_{i_2}\sharp \ldots E_{i_k}\sharp$ in $\mathcal{L}_{AE}$ where each expression E_{i_j} does not contain the symbol $\sharp$,*

4.1.2 Coding of finite sequences of Gödel numbers

Recall the assignment of the first 12 digits of base 13 representation of natural numbers to the 12 symbols that enter into formulas of the language $\mathcal{L}_{AE}$:

0	$'$	$($	$)$	f	$,$	v	$\sim$	$\supset$	$\forall$	$=$	$\leq$
1	0	2	3	4	5	6	7	8	9	η	ϵ

Thus if a number is the Gödel number of a formula in $\mathcal{L}_{AE}$ on the particular Gödel numbering we have adopted, then the 13th digit, δ, will not occur in its base 13 representation. Call the class of such numbers $N_{\overline{\delta}}$.

A formal proof is a finite sequence of formulas, so to code a proof by a number it suffices to find a way of coding finite sequences of numbers in $N_{\overline{\delta}}$. We code such a sequence $(a_1, \ldots, a_n)$ by the number $\delta *_{13} a_1 *_{13} \delta *_{13} a_2 *_{13} \delta *_{13} \ldots *_{13} \delta *_{13} a_n *_{13} \delta$. In future I shall mostly suppress the explicit notation of base 13 (or more generally base b for any $b \geq 2$) concatenation and write $v_1 = v_2 v_3$ for $v_1 = v_2 *_b v_3$, i.e. symbolize the concatenation relation by concatenation itself.

There are several points about the concatenation relation that need to be borne in mind. (1) It is a three place relation and not a two-place function. (2) It is a relation between numbers, and numbers are expressible in base b notation for all $b \geq 2$ but are not *in* base b notation. The situation is similar to what it is in number theory generally. When we compute with natural numbers we do so using their base 10 notation (or in the case of computers, base 2 notation). But when we prove something about numbers what we prove is proved using properties of numbers that are not specific to decimal notation, even if what is being proved specifically refers to decimal notation, as in "if the digits of a number in its base 10 notation add up to 9 then the number is divisible by 9", which is proved from general properties of the congruence relation and the fact that $10 \equiv 1 \pmod 9$.

Proposition 17 (sequence numbers) *A natural number n is a* sequence number *iff $n = \delta a_1 \delta a_2 \delta \ldots \delta a_n \delta$ for $a_i \in N_{\overline{\delta}}$.*

Proof. Immediate from the definition of sequence number above.

Proposition 18 *The property of being a sequence number, $Seq(v_1)$, is expressible in $\mathcal{L}_{AE}$.*

Proof. The property of being a sequence number is expressible by the following formula:

$$(\delta B v_1 \wedge \delta E v_1 \wedge \delta \neq v_1 \wedge \sim \delta\delta P v_1 \wedge (\forall v_2 \leq v_1)(\delta 0 v_2 P v_1 \supset \delta B v_2))$$

The first four conjuncts characterize the required occurrences of the digit δ in the base 13 representation of v_1. The last conjunct rules out the occurrence of a string of zeros of length greater than one as a number in sequence coded by a sequnce number. The reason for this requirement is we want each sequences of numbers in $N_{\overline{\delta}}$ to be coded by a unique sequence number. The sequence (0) is coded by $\delta 0 \delta$. But since $00 = 0$, it would also be coded by $\delta 00 \delta$, and $\delta 00 \delta \neq \delta 0 \delta$. ▲

Definition 33 *For v_1 a sequence number, we say that v_2 is* in *v_1, symbolized as $v_2 \in v_1$, iff v_2 is one of the numbers coded by v_1*

Proposition 19 *$v_2 \in v_1$ is expressible in $\mathcal{L}_{AE}$.*

Proof. $v_2 \in v_1$ iff $(Seq(v_1) \wedge \delta v_2 \delta P v_1 \wedge \sim \delta P v_2)$. It is a necessary condition for $v_2 \in v_1$ that $\delta v_2 \delta P v_1$ but not sufficient since numbers of the form $\delta a_1 \delta a_2 \delta$ satisfy it; the condition $\sim \delta P v_2$ rules out those cases. ▲

In expressing the condition that a number is the Gödel number of a proof in a formal system we need to be able to express the condition that one part of a sequence number occurs earlier in the sequence than another. We do this as follows.

Definition 34 *$v_2 \underset{v_1}{\prec} v_3$ iff v_1 is the sequence number of a sequence in which v_2 and v_3 occur and the first occurrence of v_2 in the sequence is earlier that the first occurrence of v_3 in the sequence.*

Proposition 20 *The three-place relation* $v_2 \underset{v_1}{\prec} v_3$ *is expressible in* $\mathcal{L}_{AE}$.

Proof. $v_2 \underset{v_1}{\prec} v_3$ iff $(v_2 \in v_1 \wedge v_3 \in v_1 \wedge (\exists v_4 \leq v_1)(v_4 B v_1 \wedge v_2 \in v_4 \wedge \sim v_3 \in v_4))$

Note that the formulas $v_2 \in v_1$ and $v_3 \in v_1$ each contains the condition $Seq(v_1)$, so we don't need a separate conjunct $Seq(v_1)$. ▲

4.2 The formal system PA$_E$ for arithmetic

4.2.1 Formalization of first-order logic for PA$_E$

We now begin the investigation of formal systems of arithmetic. We shall adopt the system PA$_E$ used by Smullyan in [16]. PA stands for Peano Arithmetic, a standard misnomer[1]. The subscript E signifies that in this system exponentiation is taken as primitive, i.e. the term $f_{'''}$ is governed by its own axioms. In Lecture 5 we shall see that exponentiation need not be taken as primitive, and that via coding of ordered pairs of natural numbers the relation $x^y = z$ can be expressed in terms of zero, successor, addition, and multiplication.

A formal language is first-order if its quantifiers range only over the objects in its domain of interpretation and not over collections (pluralities) of those objects. A formal language is second-order if it has quantifiers that range over pluralities of objects in its domain of interpretation, possibly also over relations between objects and/or functions from objects to objects. A theory is first-order if its formal language is first-order. Formal systems of first-order logic are complete (which Gödel proved in 1930, in his doctoral thesis, published as Gödel [4]–the incompleteness theorem was his Habilitation thesis). There is no complete axiomatization of full second-order logic. Accordingly, where we are interested in properties of what can, or cannot, be proved in formal systems, we shall be concerned only with first-order systems. In this course we will investigate properties of a number of different formal systems for arithmetic.

Stipulation 1 (Completeness with respect to logical validity) *All formal systems considered in this course are assumed to contain a complete axiomatization of (classical) first-order logic with identity*

[1]It was Dedekind who established the first axiomatization of arithmetic, in 1888, which Peano took over in his publication a year later. Peano cites Dedekind 1888 as the source of his axioms. It seems to have been Russell who introduced the misnomer Peano Arithmetic.

I now set out the logical axioms and rules of inference for our system PA_E. The axioms fall into two groups, for propositional logic and for predicate logic.

Group I Axioms for propositional logic. All instances of the following schemata:

L_1 $(F \supset (G \supset F))$

L_2 $((F \supset (G \supset H)) \supset ((F \supset G) \supset (F \supset H)))$

L_3 $((\sim F \supset \sim G) \supset (G \supset F))$

These three axiom schema for propositional logic, are standard. Axioms L_1 and L_2 are precisely the axioms required to prove the Deduction Theorem for propositional logic, and with L_3 and Modus ponens (introduced below), the system is complete with respect to classical truth-functional validity.

Group II Axioms for predicate logic. All instances of the following schemata:

L_4 $(\forall v_i(F \supset G) \supset (\forall v_i F \supset \forall v_i G))$

L_5 $(F \supset \forall v_i F)$, provided v_i does not occur in F.

L_6 $\exists v_i\, v_i = t$, provided v_i does not occur in t.

L_7 $(v_i = t \supset (Y_1 \supset Y_2))$, where v_i is any variable, t is any term, Y_1 is any atomic formula in which v_i occurs, and Y_2 is obtained from Y_1 by replacing *one* occurrence of v_i in Y_1 by t;

The formulation of this axiom schema can be given a more primitively syntactic specification, as follows:

$(v_i = t \supset (X_1{}^\frown v_i{}^\frown X_2 \supset X_1{}^\frown t^\frown X_2))$, where v_i is any variable, t is any term, and X_1 and X_2 are any expressions such that $X_1{}^\frown v_i{}^\frown X_2$ is an *atomic* formula.

Rules of inference

R_1 From F and $(F \supset G)$, infer G. [Modus Ponens]

R_2 From F, infer $\forall v_i F$. [Generalization]

The axiomatization of first-order predicate logic by the Group II schemata is highly unnatural in terms of establishing formulas as logically valid. Its virtue for us is that it is easy to arithmetize, as we shall carry out in the next section, since it involves no substitution of terms for free variables, which more natural axiomatizations of predicate logic with identity do. Note that L_6 strictly should be written $\sim \forall v_i \sim$

$v_i = t$. To prove the following valid formulas from these schemata is non-trivial: $v_i = v_i$, $(v_i = v_j \supset v_j = v_i)$, $(\forall v_i F(v_i) \supset F(t))$ for t any term of $\mathcal{L}_{AE}$ not containing a variable that is quantified in $F(v_i)$ within the scope of which v_i occurs. For proofs of these formulas from these schemata see Donald Kalish and Richard Montague [10], Lemmas 2, 3, and 8, pp. 85-87. The Group II axioms are impractical for use by a person constructing a proof. For this reason, when it's necessary to establish that a particular formula is derivable in our system for arithmeitc, PA$_E$ and variants, I shall show that it is derivable using the system of natural deduction in the notes on First-Order Logic posted on the course webpage

4.2.2 The non-logical axioms of arithmetic for PA$_E$

The non-logical axiom of arithmetic for PA$_E$ are in two groups, one consisting of axioms characterising the non-logical symbols of $\mathcal{L}_{AE}$, the other the axiom schema of arithmetical induction. The axioms for the four arithmetical operation of successor, addition, multiplication, and exponentiation, and the ordering relation on the natural numbers are in pairs, corresponding to the base case and the inductive step of definitions by recursion, plus an axiom stipulating that $\leq$ is linear.

Group III Axioms specific to each of the primitive non-logical symbols of the language

N_1 $(v_1' = v_2' \supset v_1 = v_2)$

N_2 $\sim 0 = v_1'$

N_3 $(v_1 + 0) = v_1$

N_4 $(v_1 + v_2') = (v_1 + v_2)'$

N_5 $(v_1 \cdot 0) = 0$

N_6 $(v_1 \cdot v_2') = ((v_1 \cdot v_2) + v_1)$

N_7 $(v_1 \leq 0 \equiv v_1 = 0)$

N_8 $(v_1 \leq v_2' \equiv (v_1 \leq v_2 \vee v_1 = v_2'))$

N_9 $(v_1 \leq v_2 \vee v_2 \leq v_1)$

N_{10} $v_1^0 = 0'$

N_{11} $v_1^{v_2'} = (v_1^{v_2} \cdot v_1)$

Group IV Axiom schema of mathematical induction

Group IV is all instances of (a version of) the axiom schema of induction. A usual formulation of the induction schema is

$$(F(0) \supset (\forall v_1(F(v_1) \supset F(v_1')) \supset \forall v_1 F(v_1)))$$

Two formulas within this schema are generated by substitution, namely $F(0)$ and $F(v_1')$, and for ease of arithmetization we want to use quasi-substitution instead of substitution. We can't use quasi-substitution directly on $F(v_1)$ to express $F(v_1')$, since the quasi-substituted term cannot contain the quantified variable of the quasi substitution, i.e. in this case we would have $\forall v_1(v_1 = v_1' \supset F(v_1))$, which contains the refutable antecedent $v_1 = v_1'$, so it's provable outright, as well as which it's a sentence, i.e. v_1 does not occur free in it, so in no way equivalent to $F(v_1')$. We could change the variable in the auxiliary quantification to v_i for some v_i that does not occur free in $F(v_1)$, i.e. $\forall v_i(v_i = v_1' \supset F(v_i))$, which is logically equivalent to $F(v_1')$. But this involves substitution of the variable v_i for all free occurrences of v_1 in F, which would defeat the purpose of avoiding substitution. However, we can achieve the desired equivalence by double use of quasi-substitution. First we use quasi-substitution to obtain from $F(v_1)$ a formula logically equivalent to $F(v_i)$, namely $\forall v_1(v_1 = v_i \supset F(v_1))$ where v_i is any variable that does not occur in $F(v_1)$. Then we use another quasi-substitution to obtain a formula without any substitutions that is logically equivalent to $F(v_1')$, namely $\forall v_i(v_i = v_1' \supset \forall v_1(v_1 = v_i \supset F(v_1)))$. We abbreviate this formula as $F[[v_1']]$. This logical equivalence requires that v_i does not occur *free* in $F(v_1)$, but the sufficient condition that it does not occur at all in $F(v_1)$ is easier to express in arithmetized syntax, and that's the condition we take.

N_{12} $(F[0] \supset (\forall v_1(F(v_1) \supset F[[v_1']]) \supset \forall v_1 F(v_1)))$,

where, for v_i any variable that does not occur in $F(v_1)$, and $F[[v_1']]$ is $\forall v_i(v_i = v_1' \supset \forall v_1(v_1 = v_i \supset F(v_1)))$.

Recall that when we write a schematic formula $F(v_1)$, unless we stipulate otherwise, variables other than v_1 may occur free in it. These other free variables are referred to as *parameters.*

Definition 35 (the system PA$_E$) *The system PA_E consists of the logical axioms of Group I and Group II, the logical rules of inference R_1 and R_2, and the non-logical axioms of Group III and Group IV.*

Definition 36 (a proof in PA$_E$) *A proof in PA_E is a sequence of formulas each one of which is either an axiom of PA_E or follows from two earlier formulas in the sequence by R_1, or follows from an earlier formula in the sequence by R_2.*

Definition 37 (provable in PA$_E$) *A formula F of $\mathcal{L}_{AE}$ is provable in PA_E, symbolized as $PA_E \vdash F$, if there exists a proof in PA_E of which F is a member.*

4.3 An arithmetized proof predicate for $\mathbf{PA}_E$

Each numbered paragraph in this section is both a definition of a property or relation of numbers, and a proposition that this property or relation is expressible in $\mathcal{L}_{AE}$, which is established by the formula that follows. Because it will be needed for later results, we prove, in all but cases (4) and (6), a stronger result than is needed for the weak form of Gödel's first incompleteness theorem, namely that the expressing formulas in $\mathcal{L}_{AE}$ contain only bounded quantifiers. The quantifications in (4) and (6) can also be bounded, but these cases are considerably more complicated. The property of being the Gödel number of a provable formula requires one unbounded existential quantifier.

Both for ease of reading and of typesetting, I will usually abbreviate the abbreviation $*_{13}$ for base 13 concatenation by using concatenation itself, e.g. for $v_i *_{13} v_j$ I will write $v_i v_j$. Note that on our arithmetization of $\mathcal{L}_{AE}$, we have, for example, $\ulcorner f, \urcorner = \ulcorner f \urcorner *_{13} \ulcorner , \urcorner = 4 *_{13} 5 = 45_{13} = 57_{10}$, which is denoted in $\mathcal{L}_{AE}$ by the term $0\overbrace{'\cdots'}^{57}$.

(1) $Var(v_1)$: v_1 is the Gödel number of a variable.

Recall that a variable is an expression of the form $v_{,,\ldots,}$, i.e. the variable symbol followed by a one or more subscript symbols. We have stipulated that the Gödel number of the variable symbol is 6 (base 13) and of the subscript symbol is 5 (base 13), and that the Gödel number of an expression is the base 13 number that results from concatenating the base 13 digits assigned to the symbols in the expression in their corresponding order. Hence a number is the Gödel number of a variable if and only if its base 13 expression begins with a 6 followed by one or more 5s.

$(\exists v_2 \leq v_1)((\forall v_3 \leq v_1)(v_3 P_{13} v_2 \supset 0''''' P_{13} v_3) \wedge v_1 = 0'''''' *_{13} v_2)$

In this formula I write out the formal numerals $0'''''$ and $0''''''$ rather than abbreviating them as $\overline{5}$ and $\overline{6}$, respectively, to bring out the fact that the relation $0''''' P_{13} v_3)$ is between numbers, in this case between the number 5 (as we write it in base 10, and also, as it happens, in base 13 notation) and some other number and not a relationship between numerals, though the relation between numbers is determined to hold or not by going from the number to its, in this case, base 13 representation. The number required to exist by the quantification over the variable v_2 is the Gödel number of a string of subscripts, by the condition that the subscript symbol is a part of every part of that expression. All of which is to say that when the formula $Var(v_1)$ is written in the primitive notation of $\mathcal{L}_{AE}$, the numbers in it will be expressed by numerals, i.e. $0'''''$ for 5 and $0''''''$ for 6, and similarly in the rest of the formulas expressing arithmetized syntax.

(2) $Num(v_1)$: v_1 is the Gödel number of a *numeral*, i.e. an expression of the form $0'^{\dots\prime}$

$Pow_{13}(v_1)$

(3) $Seqt(v_1)$: v_1 is the Gödel number of a *formation sequence for a term*, i.e. a sequence of expressions each one of which is either a variable or a numeral or the result of applying one of the four functions of successor, addition, multiplication, or exponentiation to an expression or expressions occurring earlier in the sequence, i.e. of the form t' or $(t_1 f_{,} t_2)$ or $(t_1 f_{,,} t_2)$ or $(t_1 f_{,,,} t_2)$.

$(Seq(v_1) \wedge (\forall v_2 \leq v_1)(v_2 \in v_1 \supset (Var(v_2) \vee Num(v_2) \vee (\exists v_3 \leq v_1)(v_3 \underset{v_1}{\prec} v_2 \wedge v_2 = v_3 0) \vee (\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(v_3 \underset{v_1}{\prec} v_2 \wedge v_4 \underset{v_1}{\prec} v_2 \wedge (v_2 = 2v_3 45 v_4 3 \vee v_2 = 2v_3 455 v_4 3 \vee v_2 = 2v_3 4555 v_4 3)))))$

(4) $Tm(v_1)$: v_1 is the Gödel number of a term.

$\exists v_2 (Seqt(v_2) \wedge v_1 \in v_2)$

Note: The formula $Seqt(v_2)$ in (4) is obtained from the formula $Seqt(v_1)$ in (3) by changing the free variable from v_1 to v_2. In changing the free variable in this way corresponding changes of bound variables in $Seqt(v_1)$ must be made so that v_1 is free for v_2 in a logically equivalent transform of $Seqt(v_1)$, e.g.

$(Seq(v_1) \wedge (\forall v_5 \leq v_1)(v_5 \in v_1 \supset (Var(v_5) \vee Num(v_5) \vee (\exists v_3 \leq v_1)(v_3 \underset{v_1}{\prec} v_5 \wedge v_5 = v_3 0) \vee (\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(v_3 \underset{v_1}{\prec} v_5 \wedge v_4 \underset{v_1}{\prec} v_5 \wedge (v_5 = 2v_3 45 v_4 3 \vee v_5 = 2v_3 455 v_4 3 \vee v_5 = 2v_3 4555 v_4 3)))))$

If we had given the formula in (4) as the logically equivalent formula $\exists v_5 (Seqt(v_5) \wedge v_1 \in v_5)$, the only change needed to obtain $Seqt(v_5)$ from $Seqt(v_1)$ is to replace all occurrences of v_1 by v_5.

Note: The formula in (4) above contains an initial unbounded existential quantifier. This quantifier can be bounded by the correlate in arithmetized syntax of Problem 2 on Problem sheet 1, i.e. decidability of whether or not an expression is a term, but it's a delicate question in which languages, i.e. with what primitives, that bound can be expressed by a term. We can live with it as an unbounded existential quantifier since the proof predicate necessarily contains an unbounded existential quantifier, and the unbounded existential quantifier of this formula will occur inside the scope of that quantifier.

(5) $AF(v_1)$: v_1 is the Gödel number of an atomic formula, i.e. of the form $t_1 = t_2$ or $t_1 \leq t_2$ for t_1, t_2 terms.

$(\exists v_2 \leq v_1)(\exists v_3 \leq v_1)(Tm(v_2) \wedge Tm(v_3) \wedge (v_1 = v_2 \eta v_3 \vee v_1 = v_2 \epsilon v_3))$

(6) $Seqf(v_1)$: v_1 is the Gödel number of a *formation sequence for a formula*, i.e. a finite sequence of expressions each one of which is either an atomic formula or of the form $\sim E$ for E occurring earlier in the sequence or of the form $(E_i \supset E_j)$ for E_i and E_j occurring earlier in the sequence or of the form $\forall v_i E$ for v_i any variable and E occurring earlier in the sequence.

$(Seq(v_1) \wedge (\forall v_2 \leq v_1)(v_2 \in v_1 \supset (AF(v_2) \vee (\exists v_3 \leq v_1)(v_3 \underset{v_1}{\prec} v_2 \wedge v_2 = 7v_3) \vee (\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(v_3 \underset{v_1}{\prec} v_2 \wedge v_4 \underset{v_1}{\prec} v_2 \wedge v_2 = 2v_3 8 v_4 3) \vee (\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(v_3 \underset{v_1}{\prec} v_2 \wedge Var(v_4) \wedge v_2 = 9v_4v_3))))$

(7) $Fm(v_1)$: v_1 is the Gödel number of a formula.

$\exists v_2(Seqf(v_2) \wedge v_1 \in v_2))$

Note: The remark as at (4) above applies here also. We know by Problem 2 on Problem sheet 1, that we can determine by a finite search whether an expression is a formula, but it's a delicate matter to determine in exactly what language of arithmetic, i.e. with what primitives, this numerical quantifier can be bounded by a term of the language.

(8) $Ax(v_1)$: v_1 is the Gödel number of an axiom of PA_E. There are seven schemata of logical axioms $L_1 - L_7$ and eleven axioms of arithmetic $N_1 - N_{11}$ plus one axiom schema of arithmetic N_{12} (Induction). We need formulas $L_i(v_1)$ such that $L_i(v_1)$ iff v_1 is the Gödel number of an axiom of form L_i, and $N_i(v_1)$ such that $N_i(v_1)$ iff v_1 is the Gödel number of an axiom of form N_i. The property that v_1 is the Gödel number of an axiom of PA_E is expressed by $(L_1(v_1) \vee \ldots \vee L_7(v_1) \vee N_1(v_1) \vee \ldots \vee N_{12}(v_1))$. Finding $N_{12}(v_1)$ is given as a problem on Problem sheet 2. I will treat a couple of cases from each of the other groups of axioms.

Logical axioms:

Group I

$L_1(v_1)$: $(\exists v_2 \leq v_1)(\exists v_3 \leq v_1)(Fm(v_2) \wedge Fm(v_3) \wedge v_1 = 2v_2 8 2 v_3 8 v_2 33)$

$L_3(v_1)$: $(\exists v_2 \leq v_1)(\exists v_3 \leq v_1)(Fm(v_2) \wedge Fm(v_3) \wedge x = 227v_2 87 v_3 382 v_3 8 v_2 33)$

Group II

$L_4(v_1)$: $(\exists v_2 \leq v_1)(\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(Fm(v_2) \wedge Fm(v_3) \wedge Var(v_4) \wedge v_1 = 29v_4 2 v_2 8 v_3 3829 v_4 v_2 89 v_4 v_3 33)$

$L_7(v_1)$: $(\exists v_2 \leq v_1)(\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(\exists v_5 \leq v_1)(\exists v_6 \leq v_1)(\exists v_7 \leq v_1)(Var(v_2) \wedge Tm(v_3) \wedge v_6 = v_4 v_2 v_5 \wedge AF(v_6) \wedge v_7 = v_4 v_3 v_5 \wedge v_1 = 2v_2 \eta v_3 82 v_6 8 v_7 33)$

Group III

$N_1(v_1)$: v_1 is the Gödel number of the axiom N_1, which in primitive notation is $(v_\prime{}' = v_{\prime\prime}{}' \supset v_\prime = v_{\prime\prime})$.

$v_1 = 2650\eta6550865\eta6553$.

$N_7(v_1)$: v_1 is the Gödel number of the axiom N_7. To compute the Gödel number of N_7 we must write it in primitive notation. This requires expressing $\equiv$ in terms of $\sim$ and $\supset$, by the truth-functional definitions: $((A \equiv B) =_{df} (A \supset B) \wedge (B \supset A))$ and $((C \wedge D) =_{df}$
$\sim(C \supset \sim D))$, which yields $((A \equiv B) =_{df} \sim((A \supset B) \supset \sim(B \supset A)))$.

So $N_7 = \sim((v_\prime \leq 0 \supset v_\prime = 0) \supset \sim(v_\prime = 0 \supset v_\prime \leq 0))$

$v_1 = 72265\epsilon1865\eta138765\eta18265\epsilon133$

(9) $Prf_{PA_E}(v_1)$: v_1 is the Gödel number of a proof in PA$_E$, i.e. a sequence of formulas each one of which is either an axiom of PA$_E$, or is the result of applying R_1 [Modus Ponens] to two formulas occurring earlier in the sequence, or is the result of applying R_2 [Generalization] to a formula occurring earlier in the sequence.

$(Seq(v_1) \wedge (\forall v_2 \leq v_1)(v_2 \in v_1 \supset (Ax(v_2) \vee (\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(v_3 \underset{v_1}{\prec} v_2 \wedge v_4 \underset{v_1}{\prec} v_2 \wedge v_4 = 2v_3 8 v_2 3) \vee (\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(Var(v_3) \wedge v_4 \underset{v_1}{\prec} v_2 \wedge v_2 = 9v_3v_4)))))$

(10) $Prov_{PA_E}(v_1, v_2)$: v_2 is the Gödel number of a proof of the formula with Gödel number v_1.

$Prov_{PA_E}(v_1, v_2) \equiv (Prf_{PA_E}(v_2) \wedge v_1 \in v_2))$

(11) $Pr_{PA_E}(v_1)$: v_1 is the Gödel number of a formula in the language $\mathcal{L}_{AE}$ that is provable in PA$_E$.

$\exists v_2 Prov_{PA_E}(v_1, v_2)$.

Theorem 21 (Arithmetical proof relation) *The two place relation between numbers m and n given by the condition that n is the Gödel number of a proof in PA_E of the formula whose Gödel number is m is expressible in $\mathcal{L}_{AE}$.*

Proof. The formula $Prov_{PA_E}(v_1, v_2)$ in (10) expresses "E_{v_2} is a proof of E_{v_1}". This is evident from this formula and (1) - (9). ▲

Corollary 22 (Arithmetical proof predicate) *The property of a number that it is the Gödel number of a formula provable in PA_E is expressible in $\mathcal{L}_{AE}$,*

Proof. PA$_E \vdash E_n$ if and only if $\exists v_2 Prov_{PA_E}(\overline{n}, v_2)$ is true. ▲

Remark: As the construction of the proof predicate for PA_E shows, arithmetization by assignment of digits to symbols and of concatenation of corresponding sequences of digits (numbers in the given base notation) to concatenation of sequences of symbols (expressions) makes the correspondence between formal expressions and numbers, which Hilbert recognized (see quotation in the first lecture) completely direct. What Gödel achieved, going beyond Hilbert's insight, was to show that the formal syntax of strings of symbols by which a formal system of proof is established corresponds exactly with arithmetically definable properties of the corresponding numbers.

4.4 The weak form of Gödel's First Incompleteness Theorem from Lecture 1 has now been established for $\mathbf{PA}_E$

We have now established hypotheses (i) and (ii) of Theorem 1 for PA_E in the language $\mathcal{L}_{AE}$, and so are able to prove

Theorem 23 (weak form of Gödel's First Incompleteness Theorem for $\mathbf{PA}_E$) *There is a sentence G in $\mathcal{L}_{AE}$ such that, if every sentence provable in PA_E is true, $PA_E \nvdash G$, G is true, and $PA_E \nvdash \sim G$.*

Proof. By the immediately previous result, Corollary 22, there is a formula in $\mathcal{L}_{AE}$, $Pr_{PA_E}(v_1)$, that expresses the property of being the Gödel number of a formula derivable in PA_E, which establishes hypothesis (i) of Theorem 1. By the Diagonal Lemma (Theorem 15) applied to $\sim Pr_{PA_E}(v_1)$, there is a sentence G such that the sentence $(G \equiv \sim Pr_{PA_E}(\overline{\ulcorner G \urcorner}))$ in $\mathcal{L}_{AE}$ is true. This establishes hypothesis (ii) of Theorem 1. Hence on the assumption that every sentence provable in PA_E is true, i.e. hypothesis (iii) of Theorem 1, the proof of Theorem 1 establishes that $\mathrm{PA}_E \nvdash G$, G is true, and $\mathrm{PA}_E \nvdash \sim G$. ▲

4.5 A weaker form of incompleteness of $\mathbf{PA}_E$ from the undefinability of truth for $\mathcal{L}_{AE}$ in $\mathcal{L}_{AE}$

Theorem 24 (Tarski's Theorem for $\mathcal{L}_{AE}$) *The set of Gödel numbers of the sentences of $\mathcal{L}_{AE}$ that are true is not expressible in $\mathcal{L}_{AE}$.*

Proof. Exercise (Problem sheet 1 problem 4) ▲

Tarski's Theorem has as an immediate corollary an extremely weak form of Gödel's First Incompleteness Theorem, extremely weak because it establishes, on the assumption that every provable sentence is true, the existence of a true sentence which is unprovable in PA_E and whose negation is unprovable, without providing an instance of such a sentence.

Theorem 25 (a very weak form of incompleteness theorem) *If every sentence provable in PA_E is true, then there must be a sentence which is true but unprovable in PA_E and whose negation is unprovable.*

Proof. If every provable sentence is, by hypothesis, true, and every true sentence is, by assumption (as the basis for an argument by reductio ad absurdum), provable, then the set of Gödel numbers of true sentence in $\mathcal{L}_{AE}$ coincides with the set of Gödel numbers of sentences provable in $\mathcal{L}_{AE}$. But this would mean that the set of Gödel numbers of true sentence in $\mathcal{L}_{AE}$ is expressible in $\mathcal{L}_{AE}$ by the formula $Pr_{PA_E}(v_1)$, which would contradict Tarski's Theorem. So the assumption that every true sentence is provable is refuted, i.e. there is a true unprovable sentence. Since the negation of a true is false, the hypothesis that every every provable sentence is true tells us that the negation of this true unprovable sentence is not provable. ▲

Note that this argument for incompleteness via Tarski's theorem is highly inefficient since it does not generate a specific true unprovable sentence, but at the same time requires all the work by which to generate a particular true unprovable sentence, namely the Diagonal Lemma, needed in proving Tarski's theorem, and the arithmetical proof predicate in order to establish that truth and proof do not coincide.

Lecture 5

The system PA, with symbols for zero, successor, addition, mutiplication, and less than or equals as primitive; Σ_0 and Σ_1-formulas and relations; a Σ_0-coding of finite sets of ordered pairs; exponentiation, and all primitive recursive functions are Δ_1-expressible in the language of PA

.

Monday 28 January 2019

5.1 The system PA, with zero, successor, addition, multiplication, and less than or equals as primitive

Definition 38 (the system PA) *The language $\mathcal{L}_A$ for PA is obtained from the language $\mathcal{L}_{AE}$ for PA_E by dropping the condition in the definition of terms for $\mathcal{L}_{AE}$, Definition 12, that if t_1 and t_2 are terms, then $(t_1 f_{'''} t_2)$ is a term, which correspondingly removes from formulas of the language $\mathcal{L}_E$ any expressions that contain the expression $f_{'''}$ (without having to make any change to Definition 15). The axioms of PA are obtained from those of PA_E by dropping axioms N_{10} and N_{11} (which are not formulas in the language $\mathcal{L}_A$).*

By very simple modification of the construction of an arithmetized proof predicate for PA_E in the language $\mathcal{L}_{AE}$, given in Section 4.3, we obtain an arithmetized proof predicate for PA in $\mathcal{L}_{AE}$. By showing that exponentiation is expressible by $\mathcal{L}_A$, we establish the existence of an arithmetized proof predicate for PA in $\mathcal{L}_A$.

Theorem 26 (proof predicate for PA in $\mathcal{L}_{AE}$) *There are formulas $Prf^E_{PA}(v_1)$, $Prov^E_{PA}(v_1, v_2)$, and $Pr^E_{PA}(v_1)$ in the language $\mathcal{L}_{AE}$ that express the property of being the Gödel number of a proof sequence for PA, the relation of being the Gödel number of a formula that occurs in the proof sequence coded by a given number, and the property of being the Gödel number of a theorem of PA.*

Proof. We show this by modifying the constructions for PA_E in Theorem 21 and Corollary 22. We drop the disjunct corresponding to term formation by the function expression $f_{'''}$, i.e. $v_2 = 2v_3 4555 v_4 3$, so that $Seqt(v_1)$ is $(Seq(v_1) \wedge (\forall v_2 \leq v_1)(v_2 \in v_1 \supset (Var(v_2) \vee Num(v_2) \vee (\exists v_3 \leq v_1)(v_3 \underset{v_1}{\prec} v_2 \wedge v_2 = v_3 0) \vee (\exists v_3 \leq v_1)(\exists v_4 \leq v_1)(v_3 \underset{v_1}{\prec} v_2 \wedge v_4 \underset{v_1}{\prec} v_2 \wedge (v_2 = 2v_3 45 v_4 3 \vee v_2 = 2v_3 455 v_4 3)))))$. In the formula $Ax(v_1)$ that expresses "E_{v_1} is an axiom of PA_E" (p. 41) the disjuncts $N_{10}(v_1)$ and $N_{11}(v_1)$ are dropped. With these modifications, the formulas for $Prf_{PA_E}(v_1)$, $Prov_{PA_E}(v_1, v_2)$, and $Pr_{PR_E}(v_1)$ on p. 42 are transformed to corresponding formulas $Prf^E_{PA}(v_1)$, $Prov^E_{PA}(v_1, v_2)$, and $Pr^E_{PA}(v_1)$. ▲

5.2 Σ_0-formulas

Definition 39 (bounded quantifiers) *For v_i any variable and t a term which is either a numeral $\overline{n}$ or a variable v_j $j \neq i$, an expression in either of the forms $\forall v_i(v_i \leq t \supset F)$ or $\exists v_i(v_i \leq t \wedge F)$ is called bounded quantification, abbreviated as*

$(\forall v_i \leq t)F$ and $(\exists v_i \leq t)F$, respectively, and the expressions $(\forall v_i \leq t)$, and $(\exists v_i \leq t)$ are called bounded quantifiers.

Remark. The restriction that the variable v_j be distinct from the variable v_i when the bound on the quantification is a variable is essential, since $\forall v_i(v_i \leq v_i \supset F)$ is logically equivalent to $\forall v_i F$, which is unbounded quantification. Also, note that bounded existential quantifications are, in primitive notation, formulas of the form $\sim \forall v_i \sim (v_i \leq \overline{n} \wedge F)$ and $\sim \forall v_i \sim (v_i \leq v_j \wedge F)$.

Definition 40 (Σ_0-formulas) *(a) Every atomic formula of the language $\mathcal{L}_{\mathcal{A}}$ of PA is a Σ_0-formula, i.e. for any terms t_1 and t_2, $t_1 = t_2$ and $t_1 \leq t_2$ are Σ_0-formulas.*

(b) If F is a Σ_0-formula, then $\sim F$ is a Σ_0-formula.

If F and G are Σ_0-formulas, then $(F \supset G)$ is a Σ_0-formula.

(c) If F is a Σ_0-formula, v_i any variable, and t either a variable distinct from v_i or a numeral, then $(\forall v_i \leq t)F$, i.e. $\forall v_i(v_i \leq t \supset F)$, is a Σ_0-formula.

Definition 41 (a formula is Σ_0) *If a formula F is provably equivalent in PA to a Σ_0-formula, we say that F is Σ_0.*

Corollary 27 (of Definition 40) *For F any Σ_0-formula, v_i any variable, and t either a variable different from v_i or a numeral, $(\exists v_i \leq t)F$, i.e. $\exists v_i(v_i \leq t \wedge F)$, is Σ_0.*

Proof. By (b), $\sim F$ is a Σ_0-formula. Then by (c), $\forall v_i(v_i \leq t \supset\sim F)$ is a Σ_0-formula. Hence by (b) again, $\sim \forall v_i(v_i \leq t \supset\sim F)$ is a Σ_0-formula, and this formula is logically equivalent, and hence provably equivalent in PA, to $\sim \forall v_i{\sim\sim}(v_i \leq t \supset\sim F)$, which is abbreviated as $\sim \forall v_i \sim (v_i \leq t \wedge F)$, which is abbreviated as $\exists v_i(v_i \leq t \wedge F)$, which is abbreviated as $(\exists v_i \leq t)F$ is Σ_0. ▲

Proposition 28 (Decidability of Σ_0-sentences) *We can effectively decide (compute) the truth or falsity of each Σ_0-sentence, i.e. closed Σ_0-formula.*

Proof. By induction over the recursive definition of Σ_0-formulas, corresponding to the clauses (a), (b), and (c) of the definition:

(a) A closed term is computable to a numeral (by Proposition 6), and sentences of the form $\overline{m} = \overline{n}$ and $\overline{m} \leq \overline{n}$ are immediately decidable.

(b) Given the truth value of F and G, we can compute the truth value of $\sim F$ and of $(F \supset G)$.

(c) Suppose v_i occurs free in F. Then $\forall v_i(v_i \leq \overline{n} \supset F(v_i))$ is equivalent to $(F(0) \wedge \ldots \wedge F(\overline{n}))$. By induction hypothesis each conjunct is decidable, so the conjunction is decidable.

Suppose v_i is not free in F. Suppose F is true. Then $(H \supset F)$ is true for any H, so $\forall v_i(v_i \leq \overline{n} \supset F)$ is true. Suppose F is false. Since $0 \leq \overline{n}$ is true, $(0 \leq \overline{n} \supset F)$ is false, so $\forall v_i(v_i \leq \overline{n} \supset F)$ is false. ▲

For reasons that will be apparent shortly, we also label Σ_0-formulas as Π_0 and as Δ_0.

5.3 Σ_1 and Π_1-formulas; Σ_1, Π_1, and Δ_1-relations

Definition 42 (Σ_1-formula) *A Σ_1 formula is any formula of the form $\exists v_i F$ where F is a Σ_0-formula.*

Definition 43 (Σ_1-relation) *A relation $R \subseteq \boldsymbol{N}^k$ is Σ_1 iff there is a Σ_1-formula $G(v_1, \ldots, v_k)$ that expresses R, i.e. such that for each k-tuple $(n_1, \ldots, n_k)$, $G(\overline{n}_1, \ldots \overline{n}_k)$ is true iff $(n_1, \ldots, n_k) \in R$.*

Note that a Σ_1-formula *begins* with *one* unbounded existential quantifier. (It may contain other quantifiers so long as they are bounded.)

Proposition 29 *Every Σ_0-formula is logically, and hence provably, equivalent to a Σ_1-formula.*

Proof. For v_i not free in F, $(F \equiv \exists v_i F)$ is logically valid (on our assumption that all domains of interpretation are non-empty), and hence provable in every system complete with respect to first-order logical validity. ▲

Definition 44 (Π_1-formula) *A Π_1 formula is any formula of the form $\forall v_i F$ where F is a Σ_0-formula.*

Definition 45 (Π_1-relation) *A relation $R \subseteq \boldsymbol{N}^k$ is Π_1 iff there is a Π_1-formula $G(v_1, \ldots, v_k)$ that expresses R, i.e. such that for each k-tuple $(n_1, \ldots, n_k)$, $G(\overline{n}_1, \ldots \overline{n}_k)$ is true iff $(n_1, \ldots, n_k) \in R$.*

Lemma 30 (Σ_1 and Π_1 are dual to each other) *The negation of a Σ_1-formula is Π_1, and the negation of Π_1-formula is Σ_1.*

Proof. By logic and the fact that the negation of a Σ_0-formula is Σ_0. ▲

Definition 46 (Δ_1 relations) *A relation is Δ_1 if and only if it is both Σ_1 and Π_1.*

Remark. There is no such thing as a Δ_1-formula, i.e. Δ_1 is not a syntactic form, but we shall sometimes say of a formula with is equivalent both to a Σ_1-formula and to a Π_1-formula that it is Δ_1.

Corollary 31 *A relation is Δ_1 iff it is Σ_1 and its complement is Σ_1.*

Proof. Immediate from Definition 46 and Lemma 30. ▲

Remark. The definitions of Σ_1 and Π_1-formulas, and by the Remark above also Δ_1-formulas, include the case of formulas with no free variables, i.e. sentences. The definitions of Σ_1, Π_1, and Δ_1-relations include the case of 0-ary relations, i.e. propositions, and 1-ary relations, i.e. sets.

Example of a Δ_1-proposition: The proposition expressed by the equivalent sentences $\exists v_1(v_1 = \overline{n} \wedge F(v_1))$ and $\forall v_i(v_i = \overline{n} \supset F(v_i))$, where v_1 is the only free variable in a Σ_0-formula $F(v_1)$. However, as we have seen, these sentences are logically equivalent to $F(\overline{n})$, which is Σ_0 for $F(v_1)$ Σ_0. It is an important fact that there are Δ_1 relations, including 0-ary relations, that are Δ_1 and not Σ_0, but this is not a fact we can prove on the basis of results so far obtained.

Definition 47 (a function is Σ_1, Π_1, Δ_1) *For f an n-ary function from $\mathbf{N}^n$ to $\mathbf{N}$, f is Σ_1 iff the $n+1$-ary relation $f(v_1, \ldots, v_n) = v_{n+1}$ is Σ_1. Similarly for Π_1 and Δ_1.*

Lemma 32 *If a total function $f : \mathbf{N}^n \to \mathbf{N}$ is Σ_1, then it is Δ_1.*

Proof. We show that the relation $f(v_1, \ldots, v_n) \neq v_{n+1}$ is also Σ_1, from which it follows that the relation $f(v_1, \ldots, v_n) = v_{n+1}$ is Δ_1. By the hypothesis that f is Σ_1, there is a Σ_0-formula $F(v_1, \ldots, v_n, v_{n+1}, v_{n+2})$ such that $\exists v_{n+2} F(v_1, \ldots, v_n, v_{n+1}, v_{n+2})$ expresses the relation $f(v_1, \ldots, v_n) = v_{n+1}$. Then, since f is total so that, for every $v_1, \ldots, v_n$, there is v_{n+1} such that $f(v_1, \ldots, v_n) = v_{n+1}$, the relation $f(v_1, \ldots, v_n) \neq v_{n+1}$ is expressed by the condition $\exists v_{n+3} \exists v_{n+2}(F(v_1, \ldots, v_n, v_{n+3}, v_{n+2}) \wedge \sim v_{n+3} = v_{n+1})$. This condition is equivalently expressed by

$$\exists v_{n+4}(\exists v_{n+3} \leq v_{n+4})(\exists v_{n+2} \leq v_{n+4})(F(v_1, \ldots, v_n, v_{n+3}, v_{n+2}) \wedge \sim v_{n+3} = v_{n+1})$$

By Definition 40,

$$(\exists v_{n+3} \leq v_{n+4})(\exists v_{n+2} \leq v_{n+4})(F(v_1, \ldots, v_n, v_{n+3}, v_{n+2}) \wedge \sim v_{n+3} = v_{n+1})$$

is a Σ_0-formula. Hence the preceding formula is Σ_1. ▲

5.4 Arithmetization of the syntax of PA in the language of PA

We have already seen that for any base $b \geq 2$, concatenation to base b, $x *_b y = z$, is expressible in $\mathcal{L}_{AE}$. We now show that for base p for p a prime number, concatenation to base p is expressible in $\mathcal{L}_A$, and indeed that it is Σ_0-expressible in this language. This result is based on an observation by John Myhill (see Smullyan, p. 43) that the property of a number x that it is a power of a given prime p can be expressed without using the exponentiation function, since x is a power of the given prime p if and only if every proper divisor of x is divisible by p.

Lemma 33 *For every prime number p the following conditions are* Σ_0.

1. $x \mid y$ —x *divides* y.

2. $Pow_p(x)$ —x *is a power of* p.

3. $p^{\ell_p(x)} = y$ —y *is the smallest positive power of* p *greater than* x.

Proof.

1. $x \mid y$ if and only if $(\exists z\, x \cdot z = y \wedge y \neq 0)$. We write this in the primitive notation of $\mathcal{L}_A$ by $\sim(\sim\forall v_{,,,} \sim (v, f_{,,} v_{,,,}) = v_{,,} \supset v_{,,} = 0)$ (see problem 3(a) on Problem sheet 1).

2. $(\forall z \leq x)((z \mid x \wedge z \neq 1) \supset p \mid z)$.

3. $(Pow_p(y) \wedge y > x \wedge y > 1 \wedge (\forall z < y) \sim (Pow_p(z) \wedge z > x \wedge z > 1)$. ▲

Lemma 34 (base p concatenation is Σ_0) *For any prime p, the relation* $x *_p y = z$ *is* Σ_0.

Proof.

$x *_p y = z$ iff $p^{\ell_p(y)} + y = z$ iff $(\exists v_1 \leq z)(v_1 = p^{\ell_p(y)} \wedge ((x \cdot v_1) + y) = z)$.

The result follows by part 3 of Lemma 33. ▲

Lemma 35 *The relation* xP_by *('x is part of y') for base b a prime number is* Σ_0.

Proof. By Theorem 16, Lemma 33 and Lemma 34. ▲

5.5 A Σ_0-coding of finite sets of ordered pairs of numbers

We now establish that the relation $x^y = z$ can be expressed in $\mathcal{L}_A$, i.e. without exponentiation. The key to this result is a Σ_0-coding of finite sets of ordered pairs of numbers. This was proved by Gödel in his 1931 paper, for which he used the Chinese Remainder Theorem, on solving simultaneous congruences. Given our different Gödel numbering, we obtain this result without using the Chinese Remainder Theorem.

Remark: This coding of finite sets of ordered pairs of numbers requires a more subtle idea than the coding of sequences of Gödel numbers of formulas. For sequence numbers we could simply add a new symbol, for which we chose $\sharp$, to mark the boundary between expressions in the sequence, and then use our Gödel numbering to code the resulting sequence of symbols. For coding finite sets of pairs of numbers, we can't add a new symbol to mark the boundaries between them since any such symbol would have to be coded by a number, and that number could be among those in the ordered pairs. Instead, we have to look at the particular finite set of ordered pairs of numbers to find a number which for those numbers can function as a boundary, in such a way that from the number coding the set of ordered pairs, we can determine what the boundary number is, and thereby decode what numbers are in the ordered pairs.

Theorem 36 (Σ_0-coding of finite sets of ordered pairs) *There is a Σ_0-formula $K(v_1, v_2, v_3)$ such that*

1. For any finite set of ordered pairs of natural numbers $(a_1, b_1), (a_2, b_2), \ldots, (a_r, b_r)$, there is a number k such that for any numbers m and n, $K(m, n, k)$ holds if and only if (m, n) is one of the pairs $(a_1, b_1), (a_2, b_2), \ldots, (a_r, b_r)$.

2. For any numbers v_1, v_2, v_3, if $K(v_1, v_2, v_3)$ holds, then $v_1 \leq v_3$ and $v_2 \leq v_3$.

Proof. We need to describe two things which are dependently related to each other: a process whereby a set of ordered pairs of numbers $(a_1, b_1), (a_2, b_2), \ldots, (a_r, b_r)$ is coded by a number k, and a formula $K(v_1, v_2, v_3)$ whereby, given a code number k, the set of ordered pairs is decoded.

(1) *Coding*. By a *frame number* we shall mean a number whose numeral in a specified base, in our case 13, is of the form $2t2$ where t is a string of 1s (this idea goes back to W.V. Quine, "Concatenation as a basis for arithmetic", *Journal of Symbolic Logic* 11 (1946), pp. 105-114; see Smullyan p. 45). The condition that the base b numeral for x is a string of 1s, which we shall abbreviate as $1_b(x)$, is expressible by the condition, $(\forall y \leq x)(yP_bx \supset 1P_by)$. Examples: $1_{10}(111_{10})$, but $\sim 1_{10}(111_{13})$ since $111_{13} = 183_{10}$, which also means that $1_{13}(183_{10})$.

Let θ be a finite sequence of ordered pairs, $(a_1, b_1), (a_2, b_2), \ldots, (a_r, b_r)$, and let f be *any* frame number that has a longer string of 1s in it than the longest string of 1s that occurs in any of the numbers $a_1, b_1, \ldots, a_r, b_r$. The code k of θ with respect to f is the number $ffa_1fb_1ffa_2fb_2ff \ldots ffa_rfb_rff$.

Note that for given θ, there are infinitely many frame numbers that have a longer string of 1s in them than the longest strings of 1s that occur in any of the numbers in θ, so there are infinitely many k that code θ.

(2) *Decoding* We call x a *maximal frame* in y if x is a frame number, x is part of

y, and x is as long as any frame that is part of y. This relation is expressed by the following formula, which we will label $MF(x,y)$:

$(xPy \wedge (\exists z \leq y)(1(z) \wedge x = 2z2 \wedge \sim (\exists w \leq y)(1(w) \wedge 2zw2Py)))$.

By Lemmas 34 and 35, $MF(x,y)$ is Σ_0.

Note that if y has a frame number in it, it has a maximal frame number, since the length of frame numbers in y is bounded by the length of y. Since frame numbers whose numerals in the given base are of the same length are equal, any number which contains a frame number contains a unique maximal frame number.

Having expressed the notion of a maximal frame, we are then able to define a formula $K(v_1, v_2, v_3)$ with which to decode the ordered pairs coded by the process specified in (1).

$K(v_1, v_2, v_3) =_{df}$
$(\exists v_4 \leq v_3)(MF(v_4, v_3) \wedge v_4v_4v_1v_4v_2v_4v_4Pv_3 \wedge \sim v_4Pv_1 \wedge \sim v_4Pv_2)$.

By Lemmas 34 and 35, the formula $K(v_1, v_2, v_3)$ is Σ_0.

Let us suppose that the sequence of ordered pairs $(a_1, b_1), (a_2, b_2), \ldots, (a_r, b_r)$ has been coded by the number k using the above procedure. The frame number f chosen for the coding occurs in k, and will be a maximal frame number for k since the string of 1s in f is longer than any strings of 1s from the a_i and b_i. As noted above, it is unique. Having recovered f from k, we can then decode each of the pairs of numbers coded in k.

The second clause of the theorem, that for any numbers v_1, v_2, v_3, if $K(v_1, v_2, v_3)$, then $v_1 \leq v_3$ and $v_2 \leq v_3$, holds since the condition $v_4v_4v_1v_4v_2v_4v_4Pv_3$ immediately implies that $v_1 < v_3$ and $v_2 < v_3$, so *a fortiori* $v_1 \leq v_3$ and $v_2 \leq v_3$. ▲

Note that there is an error in Smullyan's proof (p. 45), at the point when he says, let f be any frame number that is longer than any frame which is part of any of the numbers $a_1, b_1, \ldots, a_r, b_r$. The problem is if one or more of the a_i or b_i *is* a string of 1s that is longer than any string of 1s occurring in a frame number in one of the numbers being coded. Let c be the longest such number and let f be the frame number specified by Smullyan. In this case the maximal frame in $ffa_1fb_1ffa_2fb_2ff \ldots ffa_rfb_rff$ is $2c2$ and not f, and the decoding fails.

5.6 The relation $x^y = z$ is Δ_1-expressible in the language of PA

Theorem 37 (exponentiation is Σ_1) *The relation $x^y = z$ is Σ_1.*

Proof. The relation $x^y = z$ holds if and only if there is a set of ordered pairs $\{(0,1),(1,x),(2,x^2),\ldots,(y,x^y)\}$ and (y,z) is a member of that set. Given the coding and Σ_0-decoding of finite sets of ordered pairs of numbers by Theorem 36, we can express this by the formula $\exists w(K(y,z,w) \wedge (\forall u \leq w)(\forall v \leq w)(K(u,v,w) \supset ((u = 0 \wedge v = 1) \vee (\exists r \leq w)(\exists s \leq w)(K(r,s,w) \wedge u = r' \wedge v = s \cdot x)))$. The bounds in the quantifiers in this formula are justified by part 2. of Theorem 36. Since $K(v_1,v_2,v_3)$ is Σ_0, this whole formula is Σ_1. ▲

Corollary 38 (exponentiation is Δ_1) *The relation $x^y = z$ is Δ_1.*

Proof. Immediate from Theorem 37 by Lemma 32. ▲

A stronger result than Corollary 38 is true, namely:

Theorem 39 (exponentiation is Σ_0) *The relation $x^y = z$ is Σ_0.*

Proof. The proof is too complicated to give here. See Petr Hájek and Pavel Pudlák, *Metamathematics of First-Order Arithmetic*, Springer, Berlin, 1993, Chapter V section 3 part (c) (pp. 299-303), which has as its aim to construct a Σ_0-formula $Exp(x,y,z)$ and to prove in $I\Sigma_0$, i.e. PA with Induction restricted to Σ_0-formulas, the following formulae:

(c.1) $Exp(x,0,z) \equiv z = \overline{1}$

(c.2) $Exp(x,y+\overline{1},z) \equiv \exists v(Exp(x,y,v) \wedge z = v \cdot x)$,

from which it follows (Lemma 3.8) that $Exp(\overline{m},\overline{n},\overline{k}) \equiv \overline{m}^{\overline{n}} = \overline{k}$. ▲

5.7 Primitive recursive functions

The primitive recursive functions, which include addition, multiplication, and exponentiation, are a natural class of effectively computable total functions. The proof of Theorem 39 for exponentiation does not generalize to other primitive recursive functions, but the proof of Theorem 37 with Corollary 38 does generalize.

Definition 48 (primitive recursive functions) *This definition is given in a language $\mathcal{L}_{APR}$ (the language of arithmetic for primitive recursive functions) which* *contains the symbols 0 and $'$ and variables and numerals as for $\mathcal{L}_A$, and infinitely many function symbols for each arity, i.e. for each number k and r, there is a k-ary function symbol f_r^k such that $f_r^k(v_1, \ldots, v_k) = v_{k+1}$ is a well-formed formula. To reduce clutter we may write different function symbols f_{r_1} and f_{r_2} with other letters, for example g and h. The definition is by recursion:*

Basis: The following explicitly defined functions are primitive recursive:

$S(v_1) = v_1'$

$C(v_1) = 0$

For each $i \geq 1$ and each j such that $1 \leq j \leq i$, $P_j^i(v_1, \ldots, v_j, \ldots v_i) = v_j$ (Projection Functions)

Recursion: Primitive recursive functions are generated from given primitive recursive functions by the following two operations:

Composition: Given a primitive recursive function $f_r^k(v_1, \ldots, v_k)$ and k-many t-ary primitive recursive functions $g_{r_1}^t(v_1, \ldots v_t), \ldots, g_{r_k}^t(v_1, \ldots v_t)$, then

$h(v_1, \ldots, v_t) = f_r^k(g_{r_1}^t(v_1, \ldots v_t), \ldots, g_{r_k}^t(v_1, \ldots v_t))$ *is a primitive recursive function.*

Primitive recursion: Let $f^k(v_1, \ldots, v_k)$ and $g^{k+2}(v_{k+1}, v_{k+2}, v_1, \ldots, v_k)$ be primitive recursive functions, then the function $h^{k+1}(v_{k+1}, v_1, \ldots, v_k)$ such that

$h^{k+1}(0, v_1, \ldots, v_k) = f^k(v_1, \ldots v_k)$, *and*

$h^{k+1}(S(v_{k+1}), v_1, \ldots, v_k) = g^{k+2}(v_{k+1}, h(v_{k+1}, v_1, \ldots, v_k), v_1, \ldots, v_k)$

is a primitive recursive function.

Theorem 40 (primitive recursive functions are Δ_1) *Every primitive recursive function $f_r^k(v_1, \ldots v_k) = v_{k+1}$ is Δ_1.*

Proof (sketch) By recursion over the recursive definition of primitive recursive functions, arithmetizing the conditions for the definition of a primitive recursive function by one of the clauses in the definition of primitive recursive functions in terms of the Σ_0-formula $K(v_1, v_2, v_3)$ from Theorem 36. ▲

5.8 General recursive functions

The most general notion of computable function, which includes the primitive recursive functions but goes beyond them, is that of general recursive function. We add to primitive recursion the use of the μ-operator, where for a Σ_0-formula $F(v_1, \ldots, v_k, v_{k+}))$,

$\mu v_{k+1} F(v_1, \ldots, v_k, v_{k+1})$ is the minimum v_{k+1} such that $F(v_1, \ldots, v_k, v_{k+1})$. The μ-operator can be used to compute a general recursive function $f(v_1, \ldots, v_k) = \mu v_{k+1} F(v_1, \ldots, v_k, v_{k+1})$. Whether or not such an f is defined at $(v_1, \ldots, v_k)$ depends on whether or not $\exists v_{k+1} F(v_1, \ldots, v_k, v_{k+1})$, which is not decidable. Even if we know that $\forall v_1 \ldots \forall v_k \exists v_{k+1} F(v_1, \ldots, v_k, v_{k+1})$, i.e. that f is total, we don't in advance how many steps are required for the computation of F for given arguments, unlike for primitive recursive functions, for which we know from the construction of the function how long each computation will take. All of this shows how much beyond primitive recursion general recursion goes. Even so, general recursion is expressible in $\mathcal{L}_A$, the following theorem, which we won't prove in this course, as it's not needed for our results.

Theorem 41 *A function is general recursive if and only if it is Δ_1.*

Lecture 6

Every Σ-formula is provably equivalent to a Σ_1-formula; the arithmetized proof predicate for PA is Σ_1; the arithmetical hierarchy; the notions of Σ_0-completeness and Σ_1-completeness

Wednesday 30 January 2019

6.1 Σ-formulas

Terminology: We define a class of formulas Σ which extends the class of formulas that are Σ_1 according to Definition 42 to all formulas provably equivalent in PA to a Σ_1-formula. We will often say of a formula in Σ that it is Σ_1, rather than the more ponderously precise statement that it is provably equivalent in PA to a Σ_1-formula. We justify this use of terminology by sketching here a proof that every formula in Σ is provably in PA equivalent to a Σ_1-formula.

Definition 49 (Σ-formula) *Base: Every Σ_1-formula is a Σ-formula.*

Recursion: 1. If F is a Σ-formula, then for any variable v_i, the formula $\exists v_i F$ is a Σ-formula.

2. For any Σ-formulas F and G, $(F \vee G)$ and $(F \wedge G)$ are Σ-formulas.

3. For any Σ_0-formula F and Σ-formula G, $(F \supset G)$ is a Σ-formula.

4. If F is a Σ-formula, then for any distinct variables v_i and v_j, $(\forall v_i \leq v_j)F$ is a Σ-formula, and for any number n, $(\forall v_i \leq \overline{n})F$ is a Σ-formula.

5. If F is a Σ-formula, then for any distinct variables v_i and v_j, $(\exists v_i \leq v_j)F$ is a Σ-formula, and for any number n, $(\exists v_i \leq \overline{n})F$ is a Σ-formula.

To show that every Σ-formula is provably equivalent in PA to a Σ_1-formula we prove the following five lemmas, corresponding to the five recursion clauses in the definition of a Σ-formula.

Lemma 42 *$PA \vdash (\exists v_j \exists v_i F(v_i, v_j) \equiv \exists v_k(\exists v_j \leq v_k)(\exists v_i \leq v_k)F(v_i, v_j))$.*

Proof. Right to left is logically valid, so provable in PA. Left to right is provable in PA using N_9 $(v_1 \leq v_2 \vee v_2 \vee v_1)$. ▲

Lemma 43 *(a) $PA \vdash ((\exists v_i F(v_i) \vee \exists v_j G(v_j)) \equiv \exists v_k(F(v_k) \vee G(v_k)))$, for v_k any variable substitutable into F and G.*

(b) $PA \vdash ((\exists v_i F(v_i) \wedge \exists v_j G(v_j)) \equiv \exists v_i \exists v_j(F(v_i) \wedge G(v_j)))$, on the assumption, without loss of generality, that v_i does not occur free in G and v_j does not occur free in F.

Proof. (a) This equivalence is logically valid (Problem sheet 0 problem 2(a)), hence provable in PA.

(b) This equivalence is logically valid, and hence provable in PA. ▲

Lemma 44 *For F a formula in which v_i does not occur free,*
$PA \vdash ((F \supset \exists v_i H(v_i)) \equiv \exists v_i(F \supset H(v_i)))$.

Proof. This equivalence is logically valid, and hence provable in PA. ▲

Lemma 45 *$PA \vdash ((\forall v_j \leq v_k)\exists v_i R(v_i, v_j) \equiv \exists v_r(\forall v_j \leq v_k)(\exists v_i \leq v_r)R(v_i, v_j))$.* ▲

Proof. Right to left is logically valid, and hence provable in PA. The following is an informal argument that the left to right implication is true in the natural numbers: There are finitely many $v_j \leq v_k$. If $(\forall v_j \leq v_k)\exists v_i R(v_i, v_j)$, then there are finitely many minimum v_i such that $R(v_i, v_j)$ for $v_j \leq v_k$. Among finitely many natural numbers, there is a maximum, which is the required v_r. Exercise: Show that this informal argument can be formalized in PA. This requires an argument by induction on v_k from a PA induction axiom, using all three non-logical axioms of PA governing $\leq$. ▲

Lemma 46 $PA \vdash ((\exists v_j \leq t)\exists v_i R(v_i, v_j) \equiv \exists v_i(\exists v_j \leq t)R(v_i, v_j))$

Proof. This equivalence, with the bounded quantifier expressed in primitive notation, is logically valid. $(\exists v_j \leq t)\exists v_i R(v_i, v_j)$ in primitive notation is $\exists v_j(v_j \leq t \wedge \exists v_i R(v_i, v_j))$, which by prenexing is logically equivalent to $\exists v_j \exists v_i(v_j \leq t \wedge R(v_i, v_j))$, which is logically equivalent to $\exists v_i \exists v_j(v_j \leq t \wedge R(v_i, v_j))$, which is expressed in the defined notation for bounded quantifiers as $\exists v_i(\exists v_j \leq t)R(v_i, v_j)$. ▲

Remark comparing Lemmas 42 and 46: The formula in Lemma 46 is logically equivalent to a formula with a quantifier prefix of two existential quantifiers. However, Lemma 42 is irrelevant since a logical equivalence and definitional equivalence results in a quantifier prefix with a single existential quantifier, without appeal to any non-logical axioms.

Theorem 47 (Σ equivalent to Σ_1) *Every Σ-formula is provably equivalent in PA to a Σ_1-formula.*

Proof. By induction on the recursive definition of Σ-formulas, with each induction step established by one of Lemmas 42–46.

Base case: From $PA \vdash (F \equiv F)$, for every formula F.

Induction steps:

1. If $\exists v_i F(v_i, v_j)$ is Σ_1, then it must be that $F(v_i, v_j)$ is Σ_0, in which case, $(\exists v_j \leq v_k)(\exists v_i \leq v_k)F(v_i, v_j))$ is Σ_0. Hence by Lemma 42, if H is provably equivalent to a Σ_1-formula, $\exists v_i H$ is provably equivalent to a Σ_1-formula.

2. (a) If $F(v_i)$ and $G(v_j)$ are Σ_0, then $(F(v_i) \vee G(v_j))$ is Σ_0. Hence by Lemma 43(a), if H and K are provably equivalent to Σ_1-formulas, $(H \vee K)$ is provably equivalent to a Σ_1 formula.

(b) If $F(v_i)$ and $G(v_j)$ are Σ_0, then $(F(v_i) \wedge G(v_j))$ is Σ_0. Hence by Lemma 43(b) and Lemma 42, if H and K are provably equivalent to Σ_1-formulas, then $(H \wedge K)$ is provability equivalent to a Σ_1-formula.

3. If $\exists v_i H(v_i)$ is Σ_1, then $H(v_1)$ is Σ_0. Then if F is Σ_0, $(F \supset H(v_i))$ is Σ_0. Hence by Lemma 44, if G is provably equivalent to a Σ_1-formula, then $(F \supset G)$ is provably equivalent to a Σ_1-formula.

4. If $\exists v_i R(v_i, v_j)$ is Σ_1, then $R(v_i, v_j)$ is Σ_0, and $(\forall v_j \leq v_k)(\exists v_i \leq v_r)R(v_i, v_j)$ is Σ_0. Hence by Lemma 45, if H is provably equivalent to a Σ_1-formula, than $(\forall v_i \leq v_j)H$ is provably equivalent to a Σ_1-formula. Similarly if the bound is a numeral.

5. For $R(v_i, v_j)$ Σ_0, is $\exists v_i(\exists v_j \leq t)R(v_i, v_j)$ is Σ_1. By Corollary 27, for $R(v_i, v_j)$ Σ_0, $(\exists v_j \leq t)R(v_i, v_j)$ is Σ_0, so $\exists v_i(\exists v_j \leq t)R(v_i, v_j)$ is Σ_1, so $(\exists v_j \leq t)\exists v_i R(v_i, v_j)$ is logically equivalent to a Σ_1-formula for $R(v_i, v_j)$ Σ_0. Hence by Lemma 46, if H is provably equivalent to a Σ_1-formula, the $(\exists v_i \leq v_j)H$ is provably equivalent to a Σ_1-formula. Similarly if the bound is a numeral. ▲

6.2 The arithmetized proof predicate for PA is Σ_1

Theorem 48 *The set of Gödel numbers of formulas provable in PA is expressible in $\mathcal{L}_A$.*

Proof. By Theorem 26 and Lemmas 34 and 35. ▲

A much stronger result than Theorem 48 is true and is needed for what is to come, namely that the arithmetized proof predicate for PA is Σ_1. The first step is to show that the two-place formula $(Prf_{PA}(v_2) \wedge v_1 \in v_2)$ is Σ_1. With considerable effort we could actually establish that it's Σ_0. We know from Problem 2 on Problem sheet 1 that whether a string of symbols is a term or a formula is decidable, so we know that there is a bound on those quantifiers, but giving an explicit formulation of that bound in the language of PA is hard work which we can avoid, since for the result that the proof predicate for PA is Σ_1 it's sufficient, by Lemma 42 to show that $(Prf_{PA}(v_2) \wedge v_1 \in v_2)$ is Σ_1.

Lemma 49 *The formula* $(Prf_{PA}(v_2) \wedge v_1 \in v_2)$, *i.e.* $(Seq(v_2) \wedge (\forall v_3 \leq v_2)(v_3 \in v_2 \supset (Ax_{PA}(v_3) \vee (\exists v_4 \leq v_2)(\exists v_5 \leq v_2)(v_4 \underset{v_2}{\prec} v_3 \wedge v_5 \underset{v_2}{\prec} v_3 \wedge v_5 = 2v_4 8 v_3 3) \vee (\exists v_4 \leq v_2)(\exists v_5 \leq v_2)(Var(v_4) \wedge v_5 \underset{v_2}{\prec} v_3 \wedge v_3 = 9 v_4 v_5))) \wedge v_1 \in v_2)$, *is logically equivalent and hence provably equivalent to a Σ_1-formula.*

Proof. The key point is that the only place in the construction of $Prf_{PA}(v_2)$ in which we used an unbounded quantifier was in $Tm(v_1)$ (v_1 is the Gödel number of a term) as $\exists v_2(Seqt(v_2) \wedge v_1 \in v_2)$ and $Fm(v_1)$ (v_1 is the Gödel number of a formula) as $\exists v_2(Seqf(v_2 \wedge v_1 \in v_2))$. These occur in the formula $Ax_{PA}(v_1)$, i.e. $(L_1(v_1) \vee L_2(v_1) \vee L_3(v_1) \vee L_4(v_1) \vee L_5(v_1) \vee L_6(v_1) \vee L_7(v_1) \vee N_1(v_1) \vee N_2(v_1) \vee N_3(v_1) \vee N_4(v_1) \vee N_5(v_1) \vee N_6(v_1) \vee N_7(v_1) \vee N_8(v_1) \vee N_9(v_1) \vee N_{12}(v_1))$, in the disjuncts for the axiom schemata $L_1 - L_7$ and N_{12}. None of these occurrences of unbounded existential quantifiers is in the antecedent of a conditional, and the occurrence of $Ax(v_1)$ in $(Prf_{PA}(v_2) \wedge v_1 \in v_2)$ is also not in the antecedent of a conditional. Hence in a prenex normal form of $(Prf_{PA}(v_2) \wedge v_1 \in v_2)$ these several existential quantifiers come out as prenex existential quantifiers. By repeated application of Lemma 43, this formula is provably equivalent to a Σ_1 formula. ▲

Theorem 50 (proof predicate for PA is Σ_1) *The formula $Pr_{PA}(v_1)$, i.e. $\exists v_2(Prf_{PA}(v_2) \wedge v_1 \in v_2)$, is equivalent to a Σ_1-formula.*

Proof. By Lemma 49 and repeated applications of Lemma 42. ▲

Corollary 51 $\{n : PA \vdash E_n\}$ *is* Σ_1.

Proof. Immediate by Theorem 50. ▲

The following result requires the result, which we will prove in the next lecture, that PA is complete for true Σ_1 sentences.

Theorem 52 *If every sentence provable in PA is true, then $\{n : PA \vdash E_n\}$, which by Corollary 51 is Σ_1, is not Δ_1.*

Proof. Suppose the complement of $\{n : PA \vdash E_n\}$ is expressed by a Σ_1-formula, call it $NPr_{PA}(v_1)$. Then the Gödel sentence G, such that $(G \equiv\sim Pr(\overline{\ulcorner G \urcorner}))$ is true, is equivalent to the Σ_1-sentence $NPR_{PA}(\overline{\ulcorner G \urcorner})$. By a simple modification of Theorem 23 so that it applies to PA rather than PA_E, G is true and, if every sentence provable in PA is true, not provable in PA. But by the Σ_1-completeness of PA (to be proved in the next lecture), if G is true and equivalent to a Σ_1-sentence, then PA$\vdash G$, which contradicts the unprovability of G on the hypothesis that every sentence provable in PA is true. (In Lecture 8 we shall prove this result on a much weaker hypothesis.) ▲

Proposition 53 $\{n : \text{PA} \vdash E_n[\overline{n}]\}$ *is* Σ_1.

Proof. Exercise. ▲

6.3 The arithmetical hierarchy

The kind of classifications we introduced in Lecture 5 with the notions of Σ_0, Π_0, Δ_0 and Σ_1, Π_1, Δ_1 can be extended, by Prenex Normal Form Theorem and a generalization of Theorem 47, to a hierarchy of all formulas in the language of PA. This correspondingly defines a hierarchy of relations on natural numbers (the arithmetical hierarchy).

Remark. A simple cardinality argument tells us that most relations on the natural numbers, in particular the 1-ary relations, i.e. the sets of natural numbers, are not in this hierarchy (there are uncountably many sets of natural numbers and there are countably many formulas in the language of arithmetic).

Definition 50 (arithmetical hierarchy of formulas) *(a) Σ_0-formulas ($=_{df}$ Π_0-formulas) are as given by Definition 40.*

(b) If F is Σ_n, then $\forall v_i F$ is Π_{n+1}.

(c) If F is Π_n, then $\exists v_i F$ is Σ_{n+1}.

There is a corresponding arithmetical hierarchy of sets and relations.

Definition 51 (arithmetical hierarchy of sets and relations) *A relation on natural numbers is Σ_n (or Π_n) if and only if it is expressible by a Σ_n-formula (respectively a Π_n-formula) in $\mathcal{L}$.*

Definition 52 *If a relation is both Σ_n and Π_n, it is said to be Δ_n.*

In order to generalize Theorem 47 in the arithmetical hierarchy of formulas, and for many other purposes, we require a Σ_0-pairing function.

Lemma 54 (Σ_0 pairing function) *The function $p(m,n) = \frac{1}{2}(m+n+1)(m+n)+m$ is a bijection between the natural numbers and pairs of natural numbers which is strictly increasing in both arguments, and it is Σ_0.*

Proof. Exercise. ▲

Theorem 55 *(1) For $n > 0$, formulas provably equivalent in PA to Σ_n-formulas are closed under existential quantification and formulas provably equivalent to Π_n-formulas are closed under universal quantification, and (2) Formulas provably equivalent to Σ_n-formulas, and formulas provably equivalent to Π_n-formulas, are both closed under conjunction and disjunction.*

Proof. (1) and (2) are proved by a single induction on n in the conjunction of (1) and (2). (Exercise). ▲

Corollary 56 *Every formula in $\mathcal{L}_A$ is equivalent to a Σ_n or Π_n formula for some n.*

Proof. For a given formula, find a prenex normal form for it. By Theorem 55 (1), adjacent like quantifiers can be collapsed to a single quantifier. ▲

Corollary 57 *Formulas provably Δ_n are closed under conjunction, disjunction, and negation.*

Proof. Exercise. ▲

Lecture 7

The notions of Σ_0-completeness and Σ_1-completeness; Σ_0-completeness of a very weak system of arithmetic R

Monday 4 February 2019

7.1 Notions of Σ_0-completeness and Σ_1-completeness

We now introduce the notions of Σ_0-completeness and Σ_1-completeness of formal systems, which will play a crucial role throughout the rest of these lectures, either explicitly, or as a background assumption. Σ_0-completeness is a very weak condition, but it is sufficient for the arithmetization of syntax, as we shall see.

Definition 53 (Σ_0-completeness) *A system S is Σ_0-complete iff for each true Σ_0-sentence X, $S \vdash X$.*

Definition 54 (Σ_1-completeness) *A system S is Σ_1 complete iff for each true Σ_1-sentence X, $S \vdash X$.*

On the face of it, Σ_1-completeness may appear to be a stronger condition than Σ_0-completeness, since Σ_1-sentences contain an unbounded quantifier. However, the two conditions are equivalent, as shown by the following simple argument.

Proposition 58 *A system S is Σ_1-complete iff S is Σ_0-complete.*

Proof. Assume S is Σ_0-complete and let X be a true Σ_1-sentence, i.e. a sentence of the form $\exists v_i F(v_i)$ where $F(v_i)$ is a Σ_0-formula. Since X is true, for some number k, $F(\overline{k})$ is a true Σ_0-sentence. By Σ_0-completeness of S, $S \vdash F(\overline{k})$. Then by $\exists$-Introduction in S, $S \vdash \exists v_i F(v_i)$.

For the converse, assume S is Σ_1-complete and let X be a true Σ_0-sentence. By Proposition 29, X is logically equivalent to a Σ_1-formula Y, so since X is true, Y is true. Since Y is a true Σ_1-sentence, $S \vdash Y$. Since $(X \equiv Y)$ is logically valid, $S \vdash (X \equiv Y)$, so $S \vdash X$. ▲

In this lecture we shall see that PA, and even very weak subsystems of PA, are Σ_0-complete, and hence by Proposition 58 Σ_1-complete.

7.2 Sufficient conditions for Σ_0-completeness

Definition 55 (system S correctly decides sentence X) *System S correctly decides sentence X iff either X is true and $S \vdash X$, or X is false and $S \vdash \sim X$.*

Lemma 59 (Σ_0-complete iff correctly decides Σ_0-sentences) *A system S is Σ_0-complete iff S correctly decides every Σ_0-sentence.*

Proof. Half of the condition that S correctly decides every Σ_0-sentence is that if X is a true Σ_0-sentence then $S \vdash X$, i.e. Σ_0-completeness of S.

Conversely, suppose S is Σ_0-complete, and X is any false Σ_0-sentence. Then $\sim X$ is a true Σ_0-sentence, so by Σ_0-completeness, $S \vdash \sim X$, as required. ▲

Lemma 60 (two conditions that imply S correctly decides Σ_0-sentences) *The following two conditions on a system S together imply that S correctly decides every Σ_0-sentence.*

C_1. S correctly decides every atomic sentence.

C_2. For any Σ_0-formula $F(v_i)$ with v_i the only free variable and for every number n, if $S \vdash F(0), \ldots, S \vdash F(\overline{n})$, then $S \vdash (\forall v_i \leq \overline{n})F(v_i)$.

Proof. By induction over the recursive definition of Σ_0-formulas.

1. By C_1 S correctly decides all atomic Σ_0-sentences.

2. By propositional logic in S, if S correctly decides X, then S correctly decides $\sim X$, and if S correctly decides X and Y, then S correctly decides $(X \supset Y)$. [Exercise]

3. Any Σ_0-sentence Z that is neither atomic nor of the form $\sim X$ or $(X \supset Y)$ must be of the form $(\forall v_i \leq \overline{n})F(v_i)$ where $F(v_i)$ is a Σ_0-formula of lower degree in the

inductive generation of Σ_0-formulas than Z and contains v_i as its only free variable.

(i) Suppose $(\forall v_i \leq \overline{n})F(v_i)$ is true. That means that each of the sentences $F(0), \ldots, F(\overline{n})$ is true. Then by induction hypothesis, $S \vdash F(0), \ldots, S \vdash F(\overline{n})$. Then by condition C_2, $S \vdash (\forall v_i \leq \overline{n})F(v_i)$.

(ii) Suppose $(\forall v_i \leq \overline{n})F(v_i)$ is false. Then for some $m \leq n$, $F(\overline{m})$ is false. Then by induction hypothesis we have that $S \vdash\sim F(\overline{m})$. Since $\overline{m} \leq \overline{n}$ is a true atomic Σ_0-formula, by C_1, $S \vdash \overline{m} \leq \overline{n}$. Then since $\overline{m} \leq \overline{n}$ is true and $F(\overline{m})$ is false, $(\overline{m} \leq \overline{n} \supset F(\overline{m}))$ is false, and since by the preceding S correctly decides $\overline{m} \leq \overline{n}$ and $F(\overline{m})$, then by 2. $S \vdash\sim (\overline{m} \leq \overline{n} \supset F(\overline{m}))$.
$(\forall v_i(v_i \leq \overline{n} \supset F(v_i)) \supset (\overline{m} \leq \overline{n} \supset F(\overline{m})))$ is logically valid, and hence provable in S, so by propositional logic in S, $S \vdash\sim \forall v_i(v_i \leq \overline{n} \supset F(v_i))$, which is to say $S \vdash\sim (\forall v_i \leq \overline{n})F(v_i)$. ▲

Lemma 61 (three conditions that imply Σ_0-completeness) *The following three conditions on a system S jointly imply that S is Σ_0-complete.*

D_1. All true atomic sentences are provable in S.

D_2. For any distinct numbers m and n, $S \vdash\sim \overline{m} = \overline{n}$.

D_3. For any number n, $S \vdash (v_1 \leq \overline{n} \supset (v_1 = 0 \vee \ldots \vee v_1 = \overline{n}))$.

Proof. We show that conditions D_1, D_2, D_3 imply conditions C_1, C_2 of Lemma 60, which establishes that S correctly decides every Σ_0-sentence, and so by Lemma 59 is Σ_0-complete.

Specifically, D_1, D_2, D_3 together imply C_1, and D_3 implies C_2.

1. To establish C_1, i.e. that S correctly decides all atomic sentences, we are given by D_1 that all true atomic sentences are provable in S, so it remains to show that all false atomic sentences are refutable in S.

(i) Suppose the false atomic sentence is of the form $t_1 = t_2$. By Lemma 6, there are numbers m and n such that the atomic sentences $t_1 = \overline{m}$ and $t_2 = \overline{n}$ are true, so by D_1, $S \vdash t_1 = \overline{m}$ and $S \vdash t_2 = \overline{n}$. Since $t_1 = t_2$ is false, $\overline{m} \neq on$, so by D_2 $S \vdash\sim \overline{m} = \overline{n}$. Then by propositional logic in S, $S \vdash\sim t_1 = t_2$.

(ii) Suppose the false atomic sentence is of the form $t_1 \leq t_2$. By Lemma 6, there are numbers m and n such that the atomic sentences $t_1 = \overline{m}$ and $t_2 = \overline{n}$ are true, so by D_1, $S \vdash t_1 = \overline{m}$ and $S \vdash t_2 = \overline{n}$. Since $t_1 \leq t_2$ is false, $\sim m \leq n$ is true, which means that all the sentences $\overline{m} = 0, \ldots, \overline{m} = \overline{n}$ are false. Hence by D_2, $S \vdash\sim \overline{m} = 0, \ldots, S \vdash\sim \overline{m} = \overline{n}$. Then by repeated application of $\wedge$-Introduction, $S \vdash (\sim \overline{m} = 0 \wedge \ldots \wedge \sim \overline{m} = \overline{n})$, and so by propositional logic in S, $S \vdash\sim (\overline{m} = 0 \vee \ldots \vee \overline{m} = \overline{n})$. From D_3, by $\forall$-Introduction and $\forall$-Elimination,

$S \vdash (\overline{m} \leq \overline{n} \supset (\overline{m} = 0 \vee \ldots \vee \overline{m} = \overline{n}))$. Hence by propositional logic in S, $S \vdash\sim \overline{m} \leq \overline{n}$. Then by substitutivity of identity, $\vdash\sim t_1 \leq t_2$.

2. To show that D_3 implies C_2.

Suppose $F(v_i)$ is a Σ_0 formula with v_i its only free variable, and that n is a number such that $S \vdash F(0), \ldots, S \vdash F(\overline{n})$. Then by pure logic (with identity) in S, $S \vdash (v_i = 0 \supset F(v_i)), \ldots, S \vdash (v_i = \overline{n} \supset F(v_i))$. Then by $\vee$-Elimination, $\supset$-Elimination, and $\supset$-Introduction in S, $S \vdash ((v_i = 0 \vee \ldots \vee v_i = \overline{n}) \supset F(v_i))$. From D_3, by $\forall$-Introduction and $\forall$-Elimination, $S \vdash (v_i \leq \overline{n} \supset (v_i = 0 \vee \ldots \vee v_i = \overline{n}))$. By transitivity of $\supset$ in S, $S \vdash (v_i \leq \overline{n} \supset F(v_i))$. Then by $\forall$-Introduction in S, $S \vdash \forall v_i(v_i \leq \overline{n} \supset F(v_i))$, i.e. $S \vdash (\forall v_i \leq \overline{n})F(v_i)$. ▲

7.3 Weak systems of arithmetic Q and R (without induction)

Definition 56 (system Q) *The system Q is obtained from PA by dropping the axiom schema for induction, N_{12}. Thus Q has only finitely many (nine) nonlogical axioms:*

N_1 $(v_1' = v_2' \supset v_1 = v_2)$

N_2 $\sim 0 = v_1'$

N_3 $v_1 + 0 = v_1$

N_4 $v_1 + v_2' = (v_1 + v_2)'$.

N_5 $v_1 \cdot 0 = 0$

N_6 $v_1 \cdot v_2' = (v_1 \cdot v_2) + v_1$

N_7 $(v_1 \leq 0 \equiv v_1 = 0)$

N_8 $(v_1 \leq v_2' \equiv (v_1 \leq v_2 \vee v_1 = v_2'))$.

N_9 $(v_1 \leq v_2 \vee v_2 \leq v_1)$.

The system Q is a variant of one due to Raphael Robinson. We will show that all true Σ_0-sentences are provable Q and so provable in PA, since all the axioms of Q are axioms of PA. We prove this by proving a yet stronger result, namely that an even weaker system R, also due to Raphael Robinson, is Σ_0-complete. Instead of the (finitely) many recursion axioms of PA and Q, it has as axioms infinitely many instances of computations of addition, multiplication, and inequality, in three axiom schemata, and two axiom schemata expressing properties of $\leq$.

Definition 57 (system R**)** *The axioms of R are all sentences and formulas of $\mathcal{L}_A$ generated from natural numbers m and n by the following axiom schemata:*

Ω_1 $\overline{m} + \overline{n} = \overline{m+n}$.

Ω_2 $\overline{m} \cdot \overline{n} = \overline{m \cdot n}$.

Ω_3 $\sim \overline{m} = \overline{n}$ *where* $m \neq n$.

Ω_4 $(v_1 \leq \overline{n} \supset (v_1 = 0 \vee \ldots \vee v_1 = \overline{n}))$.

Ω_5 $(v_1 \leq \overline{n} \vee \overline{n} \leq v_1)$.

Note that the occurrence of the symbol $+$ on the left side of the the formulation of Ω_1 above abbreviates the expression $f_{\prime}$ in $\mathcal{L}_A$, and on the right does not abbreviate the occurrence of an expression, but rather describes the expression that occurs on the right, i.e. the instances of Ω_1 are generated from pairs of natural numbers, m and n, by writing an equation between the term $(\overline{m} f_{\prime} \overline{n})$ on the left and the term 0 with $m+n$ many occurrences of the symbol $'$ suffixed to it on the right. A corresponding remark holds concerning occurrences of the dot symbol for multiplication in the formulation of Ω_2, e.g. an instance of Ω_2 is $(0'' f_{\prime\prime} 0''') = 0''''''$.

The converse of Ω_4 is provable in R. For this result we need

Lemma 62 *For each natural number n, $R \vdash \overline{n} \leq \overline{n}$.*

Proof. By $\forall$-Intro and $\forall$-Elim on the instance of Ω_5 for n, $\Omega_5 \vdash (\overline{n} \leq \overline{n} \vee \overline{n} \leq \overline{n})$, so by propositional logic, $\Omega_5 \vdash \overline{n} \leq \overline{n}$. ▲

Theorem 63 (converse of Ω_4**)** *For each number n,*
$R \vdash ((v_1 = 0 \vee \ldots \vee v_1 = \overline{n}) \supset v_1 \leq \overline{n})$.

Proof. Exercise ▲

7.4 Σ_0-completeness of systems R, Q, and PA

We will now show the R is Σ_0-complete. Proposition 6 in Lecture 2 showed that for each closed closed term t in $\mathcal{L}_A$, there is a number n such that the sentence $t = \overline{n}$ is true. We now need to show that these sentences are provable in R.

Lemma 64 (evaluation of closed terms by R**)** *For each closed term t in $\mathcal{L}_A$, there is a unique number n such that $R \vdash t = \overline{n}$.*

Proof. If t is a closed term it is either a numeral, or there is a closed term t_1 such that t is the term t_1', or there are closed terms t_1 and t_2 such that t is $(t_1 f' t_2)$, i.e. $t_1 + t_2$, or t is $(t_1 f'' t_2)$, i.e. $t_1 \cdot t_2$. If t is a numeral $\overline{n}$ then $R \vdash t = \overline{n}$ by reflexivity of identity. If t is of the form t_1' for some term t_1, then by induction hypothesis there is a numeral $\overline{n}$ such that $R \vdash t_1 = \overline{n}$. Then by the logic of identity, $R \vdash t_1' = \overline{n}'$. But $\overline{n}'$ is $\overline{n+1}$, which is to say that $R \vdash t = \overline{n+1}$. If t is of the form $t_1 + t_2$, then by induction hypothesis there are numbers m_1 and m_2 such that $R \vdash t_1 = \overline{m}_1$ and $R \vdash t_2 = \overline{m}_2$. Let $m_1 + m_2 = k$. Then by Ω_1, $R \vdash \overline{m}_1 + \overline{m}_2 = \overline{k}$. Then by substitutivity of identity, $R \vdash t_1 + t_2 = \overline{k}$. Similarly, if $m_1 \cdot m_2 = k$, then by Ω_2, $R \vdash \overline{m}_1 \cdot \overline{m}_2 = \overline{k}$, and by substitutivity of identity, $R \vdash t_1 \cdot t_2 = \overline{k}$.

The uniqueness of $\overline{n}$ for a given term t follows by transitivity of identity. ▲

Proposition 65 *The system R is Σ_0-complete.*

Proof. We establish this result by showing that R satisfies the conditions D_1 D_2 D_3 of Lemma 61.

D_1. (i) Suppose t_1 and t_2 are terms such that $t_1 = t_2$ is a true sentence. Since $t_1 = t_2$ is a sentence, the terms t_1 and t_2 contain no variables. Hence by Lemma 64 there are numbers m_1 and m_2 such that $R \vdash t_1 = \overline{m}_1$ and $R \vdash t_2 = \overline{m}_2$. Since $t_1 = t_2$ is true, $m_1 = m_2$, so $\overline{m}_1$ and $\overline{m}_2$ are the same numeral, so by reflexivity of identity in R, $R \vdash \overline{m}_1 = \overline{m}_2$. Hence by transitivity of identity in R, $R \vdash t_1 = t_2$.

(ii) Suppose t_1 and t_2 are terms such that $t_1 \leq t_2$ is a true sentence. Then t_1 and t_2 have no free variables, so by Lemma 64 there are natural numbers m_1 and m_2 such that $t_1 = \overline{m}_1$ and $t_2 = \overline{m}_2$ are true sentences, and so by (i) provable in S. Since $t_1 \leq t_2$ is true, $m_1 \leq m_2$. By the logic of identity, $R \vdash \overline{m}_1 = \overline{m}_1$ and so by propositional logic, $R \vdash (\overline{m}_1 = 0 \vee \ldots \vee \overline{m}_1 = \overline{m}_1 \vee \ldots \vee \overline{m}_1 = \overline{m}_2)$. By Theorem 63, with $\forall$-introduction and $\forall$-elimination, and Modus ponens, $R \vdash \overline{m}_1 \leq \overline{m}_2$. Then by substitutivity of identity, $R \vdash t_1 \leq t_2$.

D_2. This is Ω_3.

D_3. This is Ω_4. ▲

Lecture 8

Σ_0-completeness of the intermediate system Q and Σ_0-completeness of PA; Σ_0-soundness and Σ_1-soundness; the notions of consistency, ω-consistency and n-consistency; Gödel's First Incompleteness Theorem on the assumption of 1-consistency; truth of the Gödel sentence; ω-incompleteness.

Wednesday 6 February 2019

8.1 The intermediate system Q and the system PA are Σ_0-complete

We prove this result by showing that

Proposition 66 *R is a subsystem of Q.*

Proof. We must show that every axiom of R is provable in Q, i.e. for every natural number m and n, the instances of Ω_1, Ω_2, and Ω_3 for m and n are provable in Q, and for ever natural number n, the instances of Ω_4 and Ω_5 for n are provable in Q. For each Ω_i we use induction on n to show that each instance of Ω_i with respect to n is provable in Q. **Note** that Q does not contain induction, and the arguments we give belong to informal mathematics—or as it is sometimes called in this kind of application, metamathematics—and are not **in** Q, but **about** Q.

These arguments make essential use of Corollary 5 to the definition of numerals in $\mathcal{L}_A$ in Lecture 2, that $\overline{n+1}$ is $\overline{n}'$.

Ω_1: We argue by induction on n that for all n, $Q \vdash \overline{m} + \overline{n} = \overline{m+n}$.

$n = 0$. By N_3, $\forall$-I and $\forall$-E, $Q \vdash \overline{m} + 0 = \overline{m}$. Since $m = m + 0$, $\overline{m}$ and $\overline{m+0}$ are the same term (formal numeral), so $Q \vdash \overline{m} + 0 = \overline{m+0}$.

Induction step.

(1) $Q \vdash \overline{m} + \overline{n} = \overline{m+n}$	Induction hypothesis
(2) $Q \vdash \overline{m} + \overline{n}' = (\overline{m} + \overline{n})'$	N_4, $\forall$-I, $\forall$-E
(3) $Q \vdash \overline{m} + \overline{n}' = (\overline{m+n})'$	(1), (2), substitutivity of = in Q
(4) $\overline{n}'$ is $\overline{n+1}$	Corollary 5
(5) $(\overline{m+n})'$ is $\overline{(m+n)+1}$	Corollary 5
(6) $Q \vdash \overline{m} + \overline{n+1} = (\overline{(m+n)+1})$	(3)(4)(5)
(7) $((m+n)+1) = (m+(n+1))$	truth of arithmetic
(8) $\overline{(m+n)+1)}$ and $\overline{(m+(n+1))}$ are the same term	(7)
(9) $Q \vdash \overline{m} + \overline{n+1} = \overline{m+(n+1)}$	(6), (8)

Hence by induction on n, for all n, $Q \vdash \overline{m} + \overline{n} = \overline{m+n}$

Ω_2: We show by induction on n that $Q \vdash \overline{m} \cdot \overline{n} = \overline{m \cdot n}$.

$n = 0$ By N_5, $\forall$-Intro, $\forall$-Elim, $Q \vdash \overline{m} \cdot 0 = 0$. Since $0 = m \cdot 0$, 0 and $\overline{m \cdot 0}$ are the same term. So $Q \vdash \overline{m} \cdot 0 = \overline{m \cdot 0}$.

Induction step.

(1) $Q \vdash \overline{m} \cdot \overline{n} = \overline{m \cdot n}$	Induction hypothesis
(2) $Q \vdash \overline{m} \cdot \overline{n}' = \overline{m} \cdot \overline{n} + \overline{m}$	N_6, $\forall$-I, $\forall$-E
(3) $Q \vdash \overline{m} \cdot \overline{n}' = \overline{m \cdot n} + \overline{m}$	(1), (2), substitutivity of $=$ in Q
(4) $Q \vdash \overline{m} \cdot \overline{n+1} = \overline{m \cdot n} + \overline{m}$	(3), Corollary 5
(5) $Q \vdash \overline{m \cdot n} + \overline{m} = \overline{(m \cdot n) + m}$	by Ω_1 (already proved)
(6) $Q \vdash \overline{m} \cdot \overline{n+1} = \overline{(m \cdot n) + m}$	(4)(5) trans of $=$ in Q
(7) $(m \cdot n) + m = m \cdot (n+1)$	truth of arithmetic
(8) $\overline{(m \cdot n) + m}$ and $\overline{m \cdot (n+1)}$ are the same term	(7)
(9) $Q \vdash \overline{m} \cdot \overline{n+1} = \overline{m \cdot (n+1)}$	(6), (8)

Hence by induction on n, for all n, $Q \vdash \overline{m} \cdot \overline{n} = \overline{m \cdot n}$

Ω_3: To show that for every m and n such that $m \neq n$, $Q \vdash \sim \overline{m} = \overline{n}$. Suppose that $m \neq n$. Without loss of generality, we may suppose that $m > n$, since by logic of identity in Q, if $Q \vdash \sim \overline{m} = \overline{n}$, then $Q \vdash \sim \overline{n} = \overline{m}$. We argue by cases.

$n = 0$. Then since $m \neq 0$, there is a number k such that $k + 1 = m$. By Corollary 5, $\overline{m}$ is $\overline{k}'$. By N_2, $\forall$-Intro, $\forall$-Elim, $Q \vdash \sim \overline{k}' = 0$, i.e. $Q \vdash \sim \overline{m} = 0$.

$n \neq 0$. Since $m > n$, there is a non-zero number d such that $m = d + n$. By taking d in place of m in the argument for the previous case, $Q \vdash \sim \overline{d} = 0$. By contraposition of N_1, $\forall$-Intro, $\forall$-Elim, $Q \vdash (\sim \overline{d} = 0 \supset \sim \overline{d}' = 0')$, so by Modus ponens, $Q \vdash \sim \overline{d}' = 0'$. By n many applications of this argument,

$Q \vdash \sim \overline{d}^{\overbrace{\prime \cdots \prime}^{n}} = 0^{\overbrace{\prime \cdots \prime}^{n}}$, i.e. $Q \vdash \sim \overline{m} = \overline{n}$.

Ω_4: We show by induction on n that for each n, $Q \vdash (v_1 \leq \overline{n} \supset (v_1 = 0 \vee \ldots \vee v_1 = \overline{n}))$.

$n = 0$. By $\wedge$-Elim from N_7, $Q \vdash (v_1 \leq 0 \supset v_1 = 0)$.

Assume, as induction hyposthesis, that $Q \vdash (v_1 \leq \overline{n} \supset (v_1 = 0 \vee \ldots \vee v_1 = \overline{n}))$.

By $\wedge$-Elim, $\forall$-Intro, $\forall$-Elim from N_8, $Q \vdash (v_1 \leq \overline{n}' \supset (v_1 \leq \overline{n} \vee v_1 = \overline{n}'))$. Then by $\vee$-Elim, $Q \vdash (v_1 \leq \overline{n}' \supset ((v_1 = 0 \vee \ldots \vee v_1 = \overline{n}) \vee v_1 = \overline{n}'))$, which is to say, by Corollary 5, $Q \vdash (v_1 \leq \overline{n+1} \supset (v_1 = 0 \vee \ldots \vee v_1 = \overline{n} \vee v_1 = \overline{n+1}))$.

Ω_5: $N_9 \vdash \Omega_5$ by $\forall$-Intro and $\forall$-Elim. ▲

Proposition 67 *Q is Σ_0-complete.*

Proof. By Propositions 65 and 66. ▲

Theorem 68 PA *is Σ_0-complete.*

Proof. By Proposition 67 and the fact that PA is an extension of Q. ▲

8.2 Σ_0-soundness and Σ_1-soundness

Definition 58 (Σ_0-soundness) *A system S is Σ_0-sound if and only if for every Σ_0-sentence X, if $S \vdash X$, then X is true (in the structure of the natural numbers).*

Definition 59 (Σ_1-soundness) *A system S is Σ_1-sound if and only if for every Σ_1-sentence X, if $S \vdash X$, then X is true (in the structure of the natural numbers).*

Proposition 69 *If a consistent system is Σ_0-complete, it is Σ_0-sound.*

Proof. Let S be a Σ_0-complete system and let X be a false Σ_0-sentence such that $S \vdash X$. Since X is Σ_0 and false, $\sim X$ is Σ_0 and true. Hence by Σ_0-completeness of S, $S \vdash\sim X$. But this means that S is inconsistent, contrary to hypothesis. ▲

Remark. Though a system is Σ_0-complete if and only if it is Σ_1-complete (Proposition 58), the proof of Proposition 69 does not extend to the case of Σ_1-completeness since in general the negation of a Σ_1-sentence is not Σ_1, and indeed such a result does not hold in general for systems that are consistent and Σ_1-complete. Gödel's First Incompleteness Theorem, in the strong form in which we will establish it in Lecture 8, i.e. that for G the Gödel sentence of PA, if PA is consistent, $PA \nvdash G$, gives a counterexample: From $PA \nvdash G$, it follows that $PA \cup \{\sim G\}$ is consistent, and since if PA is consistent, G is true, and by its construction, G is Π_1, so $\sim G$ is Σ_1, and if PA is consistent, false. Thus on the assumption that PA is consistent, $PA \cup \{\sim G\}$ is a consistent Σ_1-complete system which is not sound.

8.3 The notions of consistency, ω-consistency and 1-consistency.

Definition 60 (consistency) *A system S is consistent if there is no formula X in the language of S such that $S \vdash X$ and $S \vdash\sim X$.*

Proposition 70 *A system S (containing propositional logic) is consistent if and only if there is a formula Y in the language of S such that $S \nvdash Y$.*

Proof. (i) Left to right: We prove the contrapositive. Suppose for every X, $S \vdash X$. Then in particular for any formula Y, $S \vdash Y$ and $S \vdash\sim Y$.

(ii) Right to left: We prove the contrapositive. Suppose S is inconsistent, i.e. there is a formula Y such that $S \vdash Y$ and $X \vdash \sim Y$. For any formulas Y and Z, $((Y \supset (\sim Y \supset Z))$ is valid. So by two applications of $\supset$-Elimination, $S \vdash Z$. (This result is sometimes called *ex falso quodlibet*—from a falsity everything follows.)▲

Remark. Proposition 70 shows that we could equivalently have defined consistency by:

Definition 61 (alternative definition of consistency) *S is consistent if and only if there is a formula X such that $S \nvdash X$.*

This property of consistency shows that consistency is a necessary condition for any unprovability result. We shall see that it is also sufficient for the unprovability of the Gödel sentence. However, as we shall also see, consistency is not sufficient for the unprovability of the negation of the Gödel sentence. Gödel introduced the notion of ω-consistency in order to prove that the negation of the Gödel sentence is not provable in the system. ω-consistency is a much weaker hypothesis than soundness, i.e. that all theorems are true (which we saw already in Lecture 1 is sufficient for the result), established by Proposition 75, which shows that ω-consistency only implies a very limited amount of truth. Even so, ω-consistency is a considerably stronger hypothesis than is necessary to established formal incompleteness. From the fact that a proof predicate for a formal deductive system with arithmetized syntax is Σ_1, the First Incompleteness Theorem can be proved, as we shall see, with just the assumption that there is no ω-inconsistency with a Σ_1-formula. Kreisel in 1957 [11] noted that the minimum case of ω-consistency, which he labeled 1-consistency, is sufficient for the second half of Gödel First Incompleteness Theorem. This special case of ω-consistency perhaps strictly should be labeled something like Σ_1 ω-consistency, but the label 1-consistency introduced by Kreisel has the virtue of brevity and is standard in the literature.

While 1-consistency is both weaker and more natural than ω-consistency as a hypothesis for proof of the second half of the First Incompleteness Theorem, we shall see later that 1-consistency is also stronger than necessary for this result, but this can only be shown after we have proved the Second Incompleteness Theorem.

Definition 62 (ω-consistency) *A system S in a language $\mathcal{L}$ which for each natural number n contains a closed term $\overline{n}$ that denotes n is said to be ω-*consistent *if and only if there is no formula $F(v_i)$ with one free variable in $\mathcal{L}$ such that $S \vdash \exists v_i F(v_i)$ and for each natural number n, $S \vdash \sim F(\overline{n})$.*

Proposition 71 (ω-consistency implies consistency) *If a system is ω-consistent, it is consistent.*

Proof. The contrapositive is immediate: if S is inconsistent, S proves every formula in the language of S, so in particular for any formula $F(w)$ with one free variable, $S \vdash \exists w F(w)$, and for each n $S \vdash \sim F(\overline{n})$, i.e. S is ω-inconsistent. ▲

The converse of Proposition 71 does not hold, i.e.

Proposition 72 *There are consistent systems that are ω-inconsistent.*

Proof. We shall establish for PA that its Gödel sentence G is a true Π_1 sentence, i.e. of the form $\forall v_1 F(v_1)$ where $F(v_1)$ is Σ_0 such that if PA is consistent, then $PA \nvdash G$. Hence if PA is consistent, $PA \cup \{\sim G\}$ is consistent. The sentence $\sim G$ is equivalent to $\exists v_1 \sim F(v_1)$. Since G is true, for each n $F(\overline{n})$ is a true Σ_0-sentence. Since PA is Σ_0-complete, as we proved in Lecture 7 (Theorem 68), for each n $PA \vdash F(\overline{n})$, so also $PA \vdash \sim\sim F(\overline{n})$. This shows that $PA \cup \{\sim G\}$ is ω-inconsistent.

A simpler example of a consistent ω-inconsistent system is $Q \cup \{\exists v_1(1+v_1) = v_1\}$ ▲

Proposition 73 *If a system is sound with respect to truth in arithmetic, then it is ω-consistent.*

Proof. Let S be a system whose language contains numerals for the natural numbers and which is sound with respect to truth in arithmetic. Suppose $S \vdash \exists w F(w)$. Then $\exists w F(w)$ is true, i.e. there is a natural number n such that $F(\overline{n})$ is true, which is to say that $\sim F(\overline{n})$ is false. So $S \nvdash \sim F(\overline{n})$, which is to say that S is ω-consistent. ▲

The converse holds only to a strictly limited extent, as detailed by the following two propositions, and in general ω-consistency does not imply truth.

Proposition 74 *If a system S is Σ_0-complete and ω-consistent, it is Σ_2-sound, i.e. if sentence X is Σ_2 and $S \vdash X$, then X is true.*

Proof. Exercise. ▲

Remark The following proposition shows that Proposition 74 is best possible.

Proposition 75 *There is an ω-consistent system that proves a false Σ_3-sentence.*

Proof. Exercise. ▲

Definition 63 (1-consistency) *A system S in a language that contains a closed term $\overline{n}$ for each natural number is said to be* 1-consistent *if and only if there is no Σ_1-formula $\exists v_i F(v_i)$, with no free variables, in the language of S such that $S \vdash \exists v_i F(v_i)$ and for each natural number n, $S \vdash \sim F(\overline{n})$.*

Proposition 76 (1-consistency implies consistency) *If a system S is 1-consistent, then S is consistent.*

Proof. The contrapositive is immediate. If S is inconsistent, then S proves everything, so in particular it proves 1-inconsistencies. ▲

Lemma 77 *If S a Σ_0-complete and 1-consistent, then S is Σ_1-sound.*

Proof. Suppose S is Σ_0-complete and 1-consistent. Suppose $S \vdash \exists v_i F(v_i)$ and $\exists v_i F(v_i)$ is false. By the supposed falsity of $\exists v_i F(v_i)$, for each number n, $\sim F(\overline{n})$ is true. Since $\exists v_i F(v_i)$ is a Σ_1-sentence, $F(v_i)$ is Σ_0, and since Σ_0-formulas are closed under negation, $\sim F(v_i)$ is a Σ_0-formula. Hence by Σ_0-completeness of S, for each natural number n, $S \vdash\sim F(\overline{n})$. This violates the hypothesized 1-consistency of S, so if $S \vdash \exists v_i F(v_i)$, then $\exists v_i F(v_i)$ is true, i.e. S is Σ_1-sound. ▲

Lemma 78 *If a system S is Σ_1-sound, then S is 1-consistent.*

Proof. Suppose S is Σ_1-sound, and suppose $S \vdash \exists v_i F(v_i)$. Then by Σ_1-soundness of S, $\exists v_i F(v_i)$ is true, which means that there is a natural number k such that $F(\overline{k})$ is a true Σ_0-sentence, so $\sim F(\overline{k})$ is a false Σ_0-sentence which, by Lemma 29, is logically equivalent to a false Σ_1-sentence. Then by Σ_1-soundness of S, $S \nvdash\sim F(\overline{k})$. Hence S cannot prove a 1-inconsistency, i.e. S is 1-consistent. ▲

Note that Lemma 78 does not require the S be Σ_0-complete.

Theorem 79 (equivalence of Σ_1-soundness and 1-consistency) *If a system S is Σ_0-complete, then S is Σ_1-sound if and only if S is 1-consistent.*

Proof. By Lemma 77 and Lemma 78.

Definition 64 (2-consistency) *A system S in a language that contains a closed term $\overline{n}$, i.e. a numeral, for each natural number, is said to be* 2-consistent *if and only if there is no Σ_2-formula $\exists v_i F(v_i)$ with one free variable in the language such that $S \vdash \exists v_i F(v_i)$ and for each natural number n, $S \vdash\sim F(\overline{n})$.*

Lemma 80 *If a system is Σ_2-sound, then it is Σ_1-sound*

Proof. The proof is by vacuous quantification. Let $S \vdash \exists v_1 F(v_1)$ for $F(v_1)$ a Σ_0-sentence (i.e. no free variables). Then $S \vdash \exists v_1 \forall v_2 F(v_1)$, and so by Σ_2-soundness, $\exists v_1 \forall v_2 F(v_1)$ is true. Then $\exists v_1 F(v_1)$ is true. ▲

Theorem 81 *For a Σ_0-complete system, 2-consistency is equivalent to Σ_2-soundness.*

Proof. (i) To show that if S is 2-consistent, then S is Σ_2-sound: Suppose S is 2-consistent and suppose $S \vdash \exists v_1 \forall v_2 F(v_1, v_2)$, where $F(v_1, v_2)$ is a Σ_0-formula, and $\exists v_1 \forall v_2 F(v_1, v_2)$ is false, which is to say that for each natural number n, $\exists v_2 {\sim} F(\overline{n}, v_2)$ is a true Σ_1-sentence. Then by the Σ_1-completeness of every Σ_0-complete theory and predicate logic, for each natural number n, $S \vdash\sim \forall v_2 F(\overline{n}, v_2)$. But then S is 2-inconsistent. So by RAA, $\exists v_1 \forall v_2 F(v_1, v_2)$ is true.

(ii) To show that if S is Σ_2-sound, then S is 2-consistent: Suppose S is Σ_2-sound and suppose $S \vdash \exists v_1 \forall v_2 F(v_1, v_2)$. Then $\exists v_1 \forall v_2 F(v_1, v_2)$ is true, so for some number k, $\forall v_2 F(\overline{k}, v_2)$ is true. Suppose $S \vdash\sim \forall v_2 F(\overline{k}, v_2)$. Then $S \vdash \exists v_2 \sim F(\overline{k}, v_2)$. Since S is Σ_2-sound, by Lemma 80 it is Σ_1-sound, so $\exists v_2 \sim F(\overline{k}, v_2)$ is true. But this contradicts the truth of $\forall v_2 F(\overline{k}, v_2)$, so by RAA, $S \nvdash\sim \forall v_2 F(\overline{k}, v_2)$. This means that S is 2-consistent. ▲

Theorem 82 (1-consistency is strictly weaker than ω-consistency) *There are 1-consistent systems that are ω-inconsistent.*

Proof. By Theorem 83, 1-consistency does not imply 2-consistency, but by Proposition 74, ω-consistency does imply 2-consistency, so 1-consistency doesn't imply ω-consistency. ▲

Theorem 83 (1-consistency is strictly weaker than 2-consistency) *There are 1-consistent systems that are not 2-consistent.*

Proof. Exercise. ▲

Remark. We can define 3-consistency and n-consistency for larger n exactly as for 1-consistency and 2-consistency, but these notions are not natural in the way 1-consistency and 2-consistency are, since there is no equivalence with a corresponding degree of soundness as in Corollary 79 Theorem 81, i.e.

Corollary 84 (of Proposition 75) *There is a 3-consistent system that is not Σ_3-sound.*

Proof. Since ω-consistency implies n-consistency for each n and so in particular 3-consistency, Proposition 75 means that there is a 3-consistent system that is not Σ_3-sound. ▲

8.4 Incompleteness of PA from the assumption of 1-consistency

We are now in a position to prove Gödel's First Incompleteness Theorem in its most natural and thereby strongest formulation, in which the unprovability of $\sim G$ is proved from the assumption of 1-consistency. (Later we shall see, after proving the Second Incompleteness Theorem, that $S \nvdash \sim G$ can be proved from the weaker assumption that $S \cup \{ConS\}$ is consistent.) This theorem is much stronger than the version of incompleteness we have already proved, Theorem 23, in that it proves this result from the hypothesis that PA is 1-consistent, which is much weaker than the hypothesis that PA is sound, i.e. that everything PA proves is true (in the structure of the natural numbers). This hypothesis is a purely syntactic (finitary) property, as opposed to a semantic (infinitary) property. It is also strictly stronger than Gödel's original First Incompleteness Theorem, since the hypothesis for the second half of the theorem, that PA is 1-consistent, is strictly weaker than Gödel's hypothesis of ω-consistency, as we have seen, but the argument in the proof is exactly the same as for Gödel's result, i.e. Gödel's original proof made no use of the extra strength of ω-consistency over 1-consistency, so the proof from 1-consistency is really an improvement in clarity rather than in strength.

Since we are no longer working with the notion of truth (in the structure of the natural numbers), we cannot use the Diagonal Lemma, Theorem 15, which establishes the *truth* of the diagonal equivalence. However, we use the construction of the diagonal sentence for the one-place formula $\sim Pr_{PA}(v_1)$ from the proof of the Diagonal Lemma to obtain the same Gödel sentence for this theorem as for the weaker one.

Theorem 85 (Gödel's First Incompleteness Theorem for PA) *There is a* Π_1*-sentence G constructed by arithmetized syntax of PA such that*

1. *If PA is consistent, PA $\nvdash G$, and*
2. *If PA is 1-consistent, PA $\nvdash \sim G$.*

Proof. Let $\exists v_2 A(v_1, v_2)$ be the Σ_1-formula constructed by arithmetization of the syntax of PA carried out in previous lectures that expresses $\{n : \text{PA} \vdash E_n[\overline{n}]\}$ (Proposition 53). Let $a =_{df} \ulcorner \forall v_2 {\sim} A(v_1, v_2) \urcorner$, and let $G =_{df} \forall v_2 {\sim} A(\overline{a}, v_2)$.

1. Assume PA is consistent, and suppose PA $\vdash G$. By Lemma 10 and the completeness of first-order logic of PA, PA $\vdash (\forall v_2 {\sim} A(\overline{a}, v_2) \equiv \forall v_2 {\sim} A([\overline{a}], v_2))$, so then PA $\vdash \forall v_2 {\sim} A([\overline{a}], v_2)$. Then $a \in \{n : \text{PA} \vdash E_n[\overline{n}]\}$. Since $\exists v_2 A(v_1, v_2)$ expresses $\{n : \text{PA} \vdash E_n[\overline{n}]\}$, $\exists v_2 A(\overline{a}, v_2)$ is true. Since PA is Σ_0-complete, by Theorem 68, and hence, by Proposition 58, Σ_1-complete, PA $\vdash \exists v_2 A(\overline{a}, v_2)$, i.e. $PA \vdash \sim G$. This contradicts the assumption that PA is consistent. So if PA is consistent, PA $\nvdash G$.

2. Assume PA is 1-consistent, and suppose PA $\vdash \sim G$, i.e. PA $\vdash \exists v_2 A(\overline{a}, v_2)$. From the assumed 1-consistency of PA, and the fact that PA is Σ_0-complete, PA is Σ_1-sound, by Lemma 77, so $\exists v_2 A(\overline{a}, v_2)$ is true. Since $\exists v_2 A(v_1, v_2)$ expresses $\{n : \text{PA} \vdash E_n[\overline{n}]\}$, PA $\vdash E_a[\overline{a}]$, i.e. $PA \vdash \forall v_2 {\sim} A([\overline{a}], v_2))$. Then since by Lemma 10 and the completeness of first-order logic of PA, PA $\vdash (\forall v_2 {\sim} A([\overline{a}], v_2) \equiv \forall v_2 {\sim} A(\overline{a}, v_2))$, PA $\vdash \forall v_2 {\sim} A(\overline{a}, v_2)$, i.e. PA $\vdash G$, which from the assumption that PA $\vdash \sim G$ means that PA is inconsistent. But by the condition that PA is 1-consistent, PA is consistent, by Proposition 76, so if PA is 1-consistent, PA $\nvdash \sim G$. ▲

8.5 Truth of the Gödel sentence

By the bivalence of truth, one or other of G and $\sim G$ is true in the structure of the natural numbers. Though PA cannot decide G, i.e. does not prove G and does not prove $\sim G$, we have good reason to hold that G is true (in the structure of the natural numbers) by the following considerations.

Theorem 86 *G is true if and only if $PA \nvdash G$.*

Proof. By Proposition 53, there is a Σ_1-formula, $\exists v_2 A(v_1, v_2)$, that expresses $\{n : \text{PA} \vdash E_n[\overline{n}]\}$, i.e. PA $\vdash E_n[\overline{n}]$ if and only if $\exists v_2 A(\overline{n}, v_2)$ is true. Hence for $a =_{df} \ulcorner \forall v_2 {\sim} A(v_1, v_2) \urcorner$, and $G =_{df} \forall v_2 {\sim} A(\overline{a}, v_2)$, PA $\vdash G$ if and only if $\exists v_2 A(\overline{a}, v_2)$ is true, which is to say that PA $\vdash G$ if and only if $\sim G$ is true. So by contraposition, PA $\nvdash G$ if and only if G is true. ▲

Theorem 87 *G is true if and only if PA is consistent.*

Proof.

(i) If PA is consistent, then by the first part of Theorem 85, *$PA \nvdash G$*. Then by Theorem 86, right to left, G is true.

(ii) If G is true, then by Theorem 86 left to right, *$PA \nvdash G$*. But if there is any sentence that a system doesn't prove then the system is consistent, which is to say that PA is consistent. ▲

Remarks (1) By Theorem 87(i), we are justified in holding that the Gödel sentence for PA is true insofar as we are justified in our conviction (universal among all mathematicians except a few with very quirky views) that PA is consistent. Identifying the basis of our conviction that PA is consistent lies outside the scope of these lectures.

(2) Formalizing the proof of the first half of Theorem 85 in PA shows that if PA is consistent it cannot prove its own consistency, since otherwise it would prove G, which we have just shown, by Theorem 85 it cannot if it is consistent. This is Gödel's Second Incompleteness Theorem. Formalizing the proof of the first half of Theorem 85 in PA requires some hard work, the hardest of which is formalizing the proof that PA is Σ_0-complete (provable Σ_0-completeness), which I will do in Lecture 11.

(3) Having shown that G for PA can be seen to be true on the basis of the accepted consistency of PA, it is important to realize that there is no weaker basis on which to hold that G is true than that PA is consistent, i.e. G for PA cannot be established as true on the basis of any considerations that do not also establish the consistency of PA, by Theorem 87(ii).

8.5.1 Any Σ_1-sentence unprovable in a Σ_0-complete theory is false

There is another argument to show that G is true which is weaker than the argument for Theorem 87 because it requires a stronger hypothesis than just consistency, namely whatever it takes to prove the second half of the First Incompleteness Theorem, for which 1-consistency suffices but consistency does not, but the argument itself is of independent interest. It depends on the following general theorem.

Theorem 88 *If a system S is Σ_0-complete and does not prove $\exists v_i F(v_i)$ for $F(v_i)$ a Σ_0-formula, then $\exists v_i F(v_i)$ is false.*

Proof. This theorem is the contrapositive of the implication that every Σ_0-complete system is Σ_1-complete (Proposition 58), i.e. if $\exists v_i F(v_i)$ is true, then for some number k, $F(\overline{k})$ is true and hence provable in any Σ_0-complete system, which then by predicate logic also proves $\exists v_i F(v_i)$. By contraposition, if $S \nvdash \exists v_i F(v_i)$, then $\exists v_i F(v_i)$ is false. ▲

Corollary 89 (truth of the Gödel sentence from 1-consistency)

Proof. By the second half of Theorem 85, if Theorem 88, if PA is 1-consistent, then $PA \nvdash \sim G$. By the construction of G, $\sim G$ is logically and hence provably equivalent to a sentence of the form $\exists v_i F(v_i)$. Hence by Theorem 88, $\sim G$ is false, which is to say that G is true. ▲

8.6 Generalisation of the First Incompleteness Theorem

The arguments by which we have established the First Incompleteness Theorem for PA readily generalize to any Σ_0-complete theory with arithmetized syntax which yields a Σ_1-predicate that expresses $\{n : S \vdash E_n[\overline{n}]\}$.

Proposition 90 *For S any Σ_0-complete theory with arithmetized syntax that has a Σ_1-predicate that expresses $\{n : S \vdash E_n[\overline{n}]\}$, there is a Π_1-sentence G such that if S is consistent, $S \nvdash G$, and if S is 1-consistent, $S \nvdash \sim G$, and G is true if and only if S is consistent.*

Proof. All of the results in this lecture, and in the buildup to this lecture in earlier lectures, can be carried through more or less 'word for word' for such a system S. This claim needs to be verified for any particular such S, e.g. Q and R. ▲

8.7 PA is ω-incomplete

Definition 65 (ω-completeness) *A system S in a language containing a numeral $\overline{n}$ for each natural number n is said to be ω-complete if for every formula $F(v_1)$ in the language of S such that for each natural number n $S \vdash F(\overline{n})$, $S \vdash \forall v_i F(v_i)$, otherwise ω-incomplete.*

An important aspect of the first half of Gödel's First Incompleteness Theorem is that it establishes ω-incompleteness of the systems to which it applies, i.e. that for such a system S, there is a formula with one free variable $F(v_i)$ such that S proves every numerical instance of $F(v_i)$, but cannot prove that $F(v_i)$ holds of every number. In particular, the first half of the First Incompleness Theorem as established for PA shows that:

Theorem 91 *If PA is consistent, PA is ω-incomplete.*

Proof. We have shown that if PA is consistent, PA $\nvdash G$, i.e. PA $\nvdash \forall v_2 {\sim} A(\overline{a}, v_2)$. We have also seen that if PA is consistent then G is true, i.e. for each natural number n, ${\sim}A(\overline{a}, \overline{n})$ is true. The sentences ${\sim}A(\overline{a}, \overline{n})$ are Σ_0. Hence by the Σ_0-completeness of PA, for each n, PA $\vdash {\sim}A(\overline{a}, \overline{n})$. Hence PA is ω-incomplete. ▲

Lecture 9

Enumerability and the Separation Lemma; incompleteness of PA from the assumption of consistency (Rosser's Theorem); weak and strong definability of a function in a system

Monday 11 February 2019

We have noted that since an inconsistent system proves everything, consistency of a system S is a necessary condition for $S \nvdash X$, for any sentence X and in particular for G the Gödel sentence for S. Generalizing from our account of the First Incompleteness Theorem for PA, we can say that for S is any system that can arithmetize its own syntax, $S \nvdash G$, if S is consistent. The condition of ω-consistency is, as we have seen, a stronger condition than consistency, and a weaker condition than soundness. We saw that 1-consistency arises in a natural way as a condition sufficient to establish that $S \nvdash\sim G$ for G the Gödel sentence for S. There turns out to be a form of incompleteness, discovered by J. Barkley Rosser (1936), that is symmetric with respect to negation. Rosser gave the construction of a sentence R for a system S with arithmetized syntax such that if S is consistent, $S \nvdash R$ and $S \nvdash\sim R$. This result is of great interest, but it is not a strengthening of Gödel's First Incompleteness Theorem, as it is sometimes said to be, i.e. it does not show that the Gödel sentence for S is undecidable in S just on the assumption that S is simply consistent, and it is a fact, which we shall establish, that $S \nvdash \sim G$ requires a stronger hypothesis than just that S is consistent. Like the Gödel sentence, if S is consistent, the Rosser sentence for S is true, so if S is consistent, $(G \equiv R)$ is true, but this equivalence cannot be proved in S.

We will prove the Rosser Incompleteness Theorem from a separation property, itself of independent interest and which we use also in proving that the diagonal equivalence in the diagonal lemma is not only true, as we have seen, but also formally provable, which we need in proving Gödel's Second Incompleteness Theorem.

9.1 Enumerability and the Separation Lemma

Definition 66 (enumeration of a relation by a formula in a theory) *A k-ary relation $R \subseteq \mathbf{N}^k$ is enumerated by a $k+1$-place formula $F(v_1, \ldots, v_k, v_{k+1})$ in a system S if and only if:*

1. If $\langle n_1, \ldots, n_k \rangle \in R$, then there exists a number m such that $S \vdash F(\overline{n}_1, \ldots, \overline{n}_k, \overline{m})$. (We say that m is a witness *to the fact that $\langle n_1, \ldots, n_k \rangle \in R$.)*

2. If $\langle n_1, \ldots, n_k \rangle \notin R$, then for every *number m, $S \vdash \sim F(\overline{n}_1, \ldots, \overline{n}_k, \overline{m})$.*

Theorem 92 (enumerability of Σ_1-expressible sets and relations) *If a relation $R \subseteq \mathbf{N}^k$ is expressed by a Σ_1-formula $\exists v_{k+1} G(v_1, \ldots, v_k, v_{k+1})$ in $\mathcal{L}_A$, then if S is Σ_0-complete, R is enumerated in S by $G(v_1, \ldots, v_k, v_{k+1})$, and conversely, if a relation $R \subseteq \mathbf{N}^k$ is enumerated by a Σ_0-formula $G(v_1, \ldots, v_k, v_{k+1})$ in a consistent Σ_0-complete system, then R is expressed by $\exists v_{k+1} G(v_1, \ldots, v_k, v_{k+1})$.*

Proof. Exercise. ▲

Definition 67 *A formula $F(v_1, \ldots, v_k)$ separates a non-empty k-ary relation A from a non-empty k-ary relation B in a system S if and only if for all $(n_1, \ldots n_k) \in A$, $S \vdash F(\overline{n}_1, \ldots, \overline{n}_k)$, and for all $(n_1, \ldots n_k) \in B$, $S \vdash \sim F(\overline{n}_1, \ldots, \overline{n}_k)$.*

Lemma 93 *(1) If $F(v_1, \ldots, v_k)$ separates A from B in S, then $\sim F(v_1, \ldots, v_k)$ separates B from A in S. (2) If $F(v_1, \ldots, v_k)$ separates A from B in S and S is consistent, then $F(v_1, \ldots, v_k)$ does not separate B from A in S. (3) If $F(v_1, \ldots, v_k)$ separates A from B in S and S is consistent, then A and B are disjoint. (4) If S is inconsistent, then for any formula $F(v_1, \ldots, v_k)$ and any k-ary relations A and B, $F(v_1, \ldots, v_k)$ separates A from B in S.*

Proof. Exercise. ▲

Theorem 94 (Separation Lemma) *Let S be a system in which Ω_4 and Ω_5 hold,* *and let A and B be disjoint k-ary relations such that A is enumerated in S by $F(v_1, \ldots, v_k, x)$ and B is enumerated in S by $G(v_1, \ldots, v_k, x)$. Then the formula*

$$\exists x(F(v_1, \ldots, v_k, x) \wedge (\forall y \leq x) \sim G(v_1, \ldots, v_k, y))$$

separates A from B in S.

Proof. In order to shorten formulas on the page, I shall take A and B to be unary relations, i.e. sets. The proof for A and B as k-ary relations is just a notational variant of this proof.

i) To show: if $n \in A$, then $S \vdash \exists x(F(\overline{n}, x) \wedge (\forall y \leq x) \sim G(\overline{n}, y))$.

(1) $n \in A$	Assumption
(2) there exists k such that $S \vdash F(\overline{n}, \overline{k})$	(1) and enumeration of A by $F(v_1, v_2)$ in S
(3) $n \notin B$	(1) and hypothesis that A and B as disjoint
(4) for every m, $S \vdash \sim G(\overline{n}, \overline{m})$	(3) and enumeration of B by $G(v_1, v_2)$ in S
(5) for every m, $S \vdash (y = \overline{m} \supset \sim G(\overline{n}, y))$	(4) substitutivity of $=$
(6) $S \vdash ((y = 0 \vee \ldots \vee y = \overline{k}) \supset \sim G(\overline{n}, y))$	(5) $\supset$-elim, $\vee$-elim, $\supset$-intro
(7) $S \vdash (y \leq \overline{k} \supset \sim G(\overline{n}, y))$	(6), instance of Ω_4 and prop. logic
(8) $S \vdash \forall y(y \leq \overline{k} \supset \sim G(\overline{n}, y))$	(7) by $\forall$-Intro
(9) $S \vdash (\forall y \leq \overline{k}) \sim G(\overline{n}, y)$	(8) definition of $(\forall y \leq \overline{k})$)
(10) $S \vdash (F(\overline{n}, \overline{k}) \wedge (\forall y \leq \overline{k}) \sim G(\overline{n}, y))$	(2) (9) $\wedge$-Intro
(11) $S \vdash \exists x(F(\overline{n}, x) \wedge (\forall y \leq x) \sim G(\overline{n}, y))$	(10) $\exists$-intro

(ii) To show: if $n \in B$ then $S \vdash \sim \exists x(F(\overline{n}, x) \wedge (\forall y \leq x) \sim G(\overline{n}, y))$, which is logically equivalent to $S \vdash \forall x(F(\overline{n}, x) \supset (\exists y \leq x)G(\overline{n}, y))$

(1) $n \in B$	Assumption
(2) there exists k such that $S \vdash G(\overline{n}, \overline{k})$	(1) and enumeration of B by $G(v_1, v_2)$ in S
(3) $n \notin A$	(1) and the hypothesis that A and B as disjoint
(4) for every m, $S \vdash \sim F(\overline{n}, \overline{m})$	(3) and enumeration of A by $F(v_1, v_2)$ in S
(5) for every m, $S \vdash (y = \overline{m} \supset \sim F(\overline{n}, \overline{m}))$	(4) substitutivity of $=$
(6) $S \vdash ((y = 0 \vee \ldots \vee y = \overline{k}) \supset \sim F(\overline{n}, y))$	(5) $\supset$-elim, $\vee$-elim, $\supset$-intro
(7) $S \vdash (y \leq \overline{k} \supset \sim F(\overline{n}, y))$	(6), instance of Ω_4 and prop. logic
(8) $S \vdash (F(\overline{n}, y) \supset \sim y \leq \overline{k})$	(7) by prop logic (contraposition)
(9) $S \vdash (F(\overline{n}, y) \supset \overline{k} \leq y)$	(8) by Ω_5 and prop logic
(10) $S \vdash (F(\overline{n}, y) \supset (\overline{k} \leq y \wedge G(\overline{n}, \overline{k})))$	(2) (9) propositional logic
(11) $S \vdash \exists y(F(\overline{n}, x) \supset (y \leq x \wedge G(\overline{n}, y)))$	(10) $\exists$-Intro
(12) $S \vdash (F(\overline{n}, x) \supset \exists y(y \leq x \wedge G(\overline{n}, y)))$	(11) predicate logic (anti-prenexing)
(13) $S \vdash (F(\overline{n}, x) \supset (\exists y \leq x)G(\overline{n}, y))$	(12) definition of $(\exists y \leq x)$
(14) $S \vdash \forall x(F(\overline{n}, x) \supset (\exists y \leq x)G(\overline{n}, y))$	(13) $\forall$-Intro ▲

Note that in the proof of the Separation Lemma, the argument for (ii) uses both Ω_4 and Ω_5 while the argument for (i) uses just Ω_4.

9.2 Incompleteness of PA from the assumption of consistency (Rosser's Theorem)

Theorem 95 *If a formula $H(v_1)$ separates $\{n : S \vdash\sim E_n[\overline{n}]\}$ from $\{n : S \vdash E_n[\overline{n}]\}$ in a consistent arithmetizable system S, then for $h = \ulcorner H(v_1) \urcorner$, $S \nvdash H(\overline{h})$ and $S \nvdash\sim H(\overline{h})$.*

Proof. (i) Suppose $S \vdash H(\overline{h})$. Then since $S \vdash (H(\overline{h}) \equiv H[\overline{h}])$, $S \vdash H[\overline{h}]$, so $h \in \{n : S \vdash E_n[\overline{n}]\}$. Then by the separation property of $H(v_1)$, $S \vdash\sim H(\overline{h})$. This contradicts the assumption that S is consistent. So $S \nvdash H(\overline{h})$.

(ii) Suppose $S \vdash\sim H(\overline{h})$. Then $h \in \{n : S \vdash\sim E_n[\overline{n}]\}$. Then by the separation property of $H(v_1)$, $S \vdash H(\overline{h})$. This contradicts the assumption that S is consistent. So $S \nvdash\sim H(\overline{h})$. ▲

Lemma 96 *If S be a consistent axiomatizable extension of R in which the formula $Pd(v_1, v_2)$ enumerates $\{n : S \vdash E_n[\overline{n}]\}$ and the formula $Rd(v_1, v_2)$ enumerates $\{n : S \vdash\sim E_n[\overline{n}]\}$, then the formula*

$$\forall v_1(Pd(v_3, v_1) \supset (\exists v_2 \leq v_1)Rd(v_3, v_2))$$

separates $\{n : S \vdash\sim E_n[\overline{n}]\}$ from $\{n : S \vdash E_n[\overline{n}]\}$.

Proof. By Theorem 94 and Lemma 93 (1). ▲

Lemma 97 (expressibility of negated diagonal quasi-substitution) *There is a Σ_1-formula $\exists v_2 Rd(v_1, v_2)$ constructed by arithmetization of the syntax of PA that expresses $\{n : \mathrm{PA} \vdash \sim E_n[\overline{n}]\}$.*

Proof. Problem 1 on Problem sheet 3. ▲

Theorem 98 (Rosser's Theorem) *There is an explicit sentence R such that if PA is consistent, $PA \nvdash R$ and $PA \nvdash \sim R$.*

Proof. By Proposition 53, there is a Σ_1-formula $\exists v_2 Pd(v_1, v_2)$ constructed by arithmetization of the syntax of PA that expresses $\{n : \mathrm{PA} \vdash E_n[\overline{n}]\}$. Then by Theorem 92, $Pd(v_1, v_2)$ enumerates $\{n : \mathrm{PA} \vdash E_n[\overline{n}]\}$ in PA. By Lemma 97, there is a Σ_1-formula $\exists v_2 Rd(v_1, v_2)$ constructed by arithmetization of the syntax of PA that expresses $\{n : \mathrm{PA} \vdash\sim E_n[\overline{n}]\}$. Then by Theorem 92, $Rd(v_1, v_2)$ enumerates $\{n : PA \vdash\sim E_n[\overline{n}]\}$ in PA. Then by Theorem 95 and Lemma 96, there is an explicit sentence R such that if PA is consistent, $\mathrm{PA} \nvdash R$ and $\mathrm{PA} \nvdash\sim R$, ▲

9.3 Weak and strong definability of a function in a system

Definition 68 (weak definability of a function in a system) *A function $f : \mathbf{N}^n \to \mathbf{N}$ is weakly definable in a system S iff there is a formula $F(v_1, \ldots, v_n, v_{n+1})$ such that.*

(1) If $f(a_1, \ldots, a_n) = b$, then $S \vdash F(\overline{a}_1, \ldots, \overline{a}_n, \overline{b})$.

(2) If $f(a_1, \ldots, a_n) \neq b$, then $S \vdash \sim F(\overline{a_1}, \ldots, \overline{a_n}, \overline{b})$.

This notion is also sometimes called expressibility (e.g. Elliott Mendelson, *Introduction to Mathematical Logic*, 4th edn, Chapman and Hall, 1997, p. 170), or numeralwise expressibility (e.g. Stephen Cole Kleene, *Introduction to Metamathematics*, D. Van Nostrand, 1950, p. 195).

Definition 69 (strong definability of a function in a system) *A function $f : \mathbf{N}^n \to \mathbf{N}$ is strongly definable in a system S iff there is a formula $G(v_1, \ldots, v_n, v_{n+1})$ such that if $f(a_1, \ldots, a_n) = b$, then*

$$S \vdash (G(\overline{a}_1, \ldots, \overline{a}_n, \overline{b}) \wedge \forall v_{n+1}(G(\overline{a}_1, \ldots, \overline{a}_n, v_{n+1}) \supset v_{n+1} = \overline{b})).$$

It's easy to show that strong definability implies weak definability (Proposition 99). The converse also holds but is considerably more complicated to prove (Theorem 100).

Proposition 99 *Let S be a system in which Ω_3 holds. If a function is strongly definable in S then it is weakly definable in S.*

Proof. If $f : \mathbf{N}^n \to \mathbf{N}$ is strongly defined in S by the formula $F(v_1, \ldots, v_n, v_{n+1})$, then it is weakly defined in S by the same formula. Assume that $f(a_1, \ldots, a_n) = b$. Then by the first conjunct of the condition for strong definability, $S \vdash F(\overline{a}_1, \ldots, \overline{a}_n, \overline{b})$, which is condition (1) of weak definability. Suppose $c \neq b$, so $f(a_1, \ldots, a_n) \neq c$. By $\forall$-elimination from the second conjunction of the condition for strong definability in S, $S \vdash F(\overline{a_1}, \ldots, \overline{a}_n, \overline{c}) \supset \overline{c} = \overline{b}$. If $c \neq b$, then by Ω_3, $S \vdash \sim \overline{c} = \overline{b}$. So by propositional logic in S, $S \vdash \sim F(\overline{a_1}, \ldots, \overline{a_n}, \overline{b})$, i.e. clause (2) of the definition of weak definability of a function. ▲

We will now show, just with the use of Ω_4 and Ω_5, that any function weakly definable in a system S is strongly definable in S. Weak definability of a function $f(x) = y$ in S is the condition that there is a formula $F(v_1, v_2)$ that separates in S the graph of $f(x) = y$ from its complement. The first conjunct of the condition for strong definability is the same as the first clause for weak definability. The second conjunct expresses the functionality condition for a given argument of the function, i.e. that the only number that bears the defining relation to the given argument is the value of the function for that argument. The way we do this is to define a new formula from the formula that weakly defines the function with the additional condition that the relationship between the argument and a number holds just in case it's the least number for which the weak definability relation holds.

Theorem 100 (weak definability implies strong definability) *If S is an extension of $\{\Omega_4, \Omega_5\}$, then any function weakly definable in S is strongly definable in S.*

Proof. To reduce clutter, I give the proof for the case of a unary function, which is also the case of immediate interest since the diagonal function is unary. The argument for the general case is a notational variant.

Let $F(x,y)$ be a formula that weakly defines $f(x)$ in S, i.e.

(1) If $f(a) = b$, then $S \vdash F(\overline{a}, \overline{b})$.

(2) If $f(a) \neq b$, then $S \vdash \sim F(\overline{a}, \overline{b})$.

Let $G(x,y)$ be the formula $(F(x,y) \wedge \forall z(F(x,z) \supset y \leq z))$, i.e. $G(x,y)$ if and only $F(x,y)$ and y is the least number z such that $F(x,z)$. We show that $G(x,y)$ strongly defines $f(x)$ in S, i.e.

If $f(a) = b$ then $S \vdash (G(\overline{a}, \overline{b}) \wedge \forall y(G(\overline{a}, y) \supset y = \overline{b}))$.

which we do by showing that, on the assumption $f(a) = b$, S proves each conjunct of $(G(\overline{a}, \overline{b}) \wedge \forall y(G(\overline{a}, y) \supset y = \overline{b}))$.

(i) First conjunct: to show that if $f(a) = b$, $S \vdash (F(\overline{a}, \overline{b}) \wedge \forall z(F(\overline{a}, z) \supset \overline{b} \leq z))$.

(1) Since $f(x)$ in weakly definable in S by $F(v_1, v_2)$, $S \vdash F(\overline{a}, \overline{b})$.

(2) To prove the second conjunct we establish $(F(\overline{a}, v_1) \supset \overline{b} \leq v_1)$ by $\vee$-elimination from the instance of Ω_5 for b, i.e. $(v_1 \leq \overline{b} \vee \overline{b} \leq v_1)$.

(3) We have $S \vdash (\overline{b} \leq v_1 \supset (F(\overline{a}, v_1) \supset \overline{b} \leq v_1))$, as an instance of L_3. So it remains to show: $S \vdash (v_1 \leq \overline{b} \supset (F(\overline{a}, v_1) \supset \overline{b} \leq v_1)))$.

(4) Let k be any number such that $k < b$. Then $k \neq b$, so by clause (2) of weak definability, $S \vdash \sim F(\overline{a}, \overline{k})$. Then by propositional logic $S \vdash (F(\overline{a}, \overline{k}) \supset \overline{b} \leq \overline{k})$. Then by substitutivity of identity, $S \vdash (v_1 = \overline{k} \supset (F(\overline{a}, v_1) \supset \overline{b} \leq v_1))$, for each $k < b$.

(5) Suppose $k = b$. We know by Lemma 62 that $\Omega_5 \vdash \overline{b} \leq \overline{b}$. So by L_3 and Modus ponens, $S \vdash (F(\overline{a}, \overline{b}) \supset \overline{b} \leq \overline{b})$. Then by substitutivity of identity, $S \vdash (v_1 = \overline{b} \supset (F(\overline{a}, v_1) \supset \overline{b} \leq v_1)$.

(6) By $\vee$-elimination from the cases $v_1 = 0, \ldots, v_1 = \overline{b}$ established in (4) and (5), $S \vdash ((v_1 = 0 \vee \ldots \vee v_1 = \overline{b}) \supset (F(\overline{a}, v_1) \supset \overline{b} \leq v_1))$.

(7) From (6) by Ω_4 and propositional logic, $S \vdash (v_1 \leq \overline{b} \supset (F(\overline{a}, v_1) \supset \overline{b} \leq v_1))$.

(8) By $\vee$-elimination from Ω_5 with (3) and (7), $S \vdash (F(\overline{a}, v_1) \supset \overline{b} \leq v_1)$.

(9) by $\forall$-Intro from (8), $S \vdash \forall y(F(\overline{a}, y) \supset \overline{b} \leq y)$.

(ii) Second conjunct: to show that that if $f(a) = b$, $S \vdash \forall y(G(\overline{a}, y) \supset y = \overline{b})$, i.e. $S \vdash \forall y((F(\overline{a}, y) \wedge \forall z(F(\overline{a}, z) \supset y \leq z)) \supset y = \overline{b})$.

(1) By propositional logic, i.e. $\wedge$-elimination and $\supset$-Introduction, $S \vdash ((F(\overline{a}, y) \wedge \forall z(F(\overline{a}, z) \supset y \leq z)) \supset \forall z(F(\overline{a}, z) \supset y \leq z))$.

(2) Since $(\forall z(F(\overline{a}, z) \supset y \leq z) \supset (F(\overline{a}, \overline{b}) \supset y \leq \overline{b}))$ is logically valid, from (1) it follows that $S \vdash ((F(\overline{a}, y) \wedge \forall z(F(\overline{a}, z) \supset y \leq z)) \supset (F(\overline{a}, \overline{b}) \supset y \leq \overline{b}))$.

(3) From (2), $S \cup \{(F(\overline{a}, y) \wedge \forall z(F(\overline{a}, z) \supset y \leq z)) \vdash (F(\overline{a}, \overline{b}) \supset y \leq \overline{b})$.

(4) By condition (1) of weak definability, $S \vdash F(\overline{a}, \overline{b})$, so from (3) by $\supset$-Elimination, $S \cup \{(F(\overline{a}, y) \wedge \forall z(F(\overline{a}, z) \supset y \leq z)) \vdash y \leq \overline{b}$.

(5) Hence from (4) by $\supset$-Introduction, $S \vdash ((F(\overline{a}, y) \wedge \forall z(F(\overline{a}, z) \supset y \leq z)) \supset y \leq \overline{b})$, i.e. $S \vdash (G(\overline{a}, y) \supset y \leq \overline{b})$.

(6) We aim to show that $S \vdash (y \leq \overline{b} \supset (G(\overline{a}, y) \supset y = \overline{b}))$ [proved at (11)], which with (5), by the transitivity of $\supset$, yields $S \vdash (G(\overline{a}, y) \supset (G(\overline{a}, y) \supset y = \overline{b}))$, and so by propositional logic $S \vdash (G(\overline{a}, y) \supset y = \overline{b})$, and so by $\forall$-Introduction, we have (ii). Thus it remains to show that $S \vdash (y \leq \overline{b} \supset (G(\overline{a}, y) \supset y = \overline{b}))$.

(7) For k any number such that $k < b$, $k \neq b$, so $f(a) \neq k$, so by weak definability of $f(x)$ by $F(v_1, v_2)$, $S \vdash \sim F(\overline{a}, \overline{k})$, so $S \vdash \sim G(\overline{a}, \overline{k})$. So by propositional logic (every sentence follows from a contradiction) $S \vdash (G(\overline{a}, \overline{k}) \supset \overline{k} = \overline{b})$.

(8) From (7) by substitutivity of identity, for $k < b$, $S \vdash (y = \overline{k} \supset (G(\overline{a}, y) \supset y = \overline{b})$

(9) By L_3, $S \vdash (y = \overline{b} \supset (G(\overline{a}, y) \supset y = \overline{b}))$.

(10) By $\vee$-elimination from (8) and (9) and $\supset$-introduction, $S \vdash ((y = 0 \vee \ldots \vee y = \overline{b}) \supset (G(\overline{a}, y) \supset y = \overline{b}))$.

(11) From (10) by Ω_4 and propositional logic, $S \vdash (y \leq \overline{b} \supset (G(\overline{a}, y) \supset y = \overline{b}))$.

(12) From (11) and the argument given in (6), $S \vdash (G(\overline{a}, y) \supset y = \overline{b})$.

(13) From (12) by $\forall$-introduction, $S \vdash \forall y(G(\overline{a}, y) \supset y = \overline{b})$. ▲

Theorem 101 (Σ_1-expressible functions strongly definable) *If a total function f is Σ_1-expressible, then for S any Σ_0-complete theory, f is strongly definable S.*

Proof. By Lemma 32 in Lecture 5, if the graph of a total function is Σ_1, the complement of its graph is also Σ_1. By Theorem 92, that Σ_1-relations are enumerable in Σ_0-complete systems, f is enumerable in any Σ_0-complete system S, and its complement is enumerable S. Then by the Separation Lemma, f is weakly definable in S. Then by Theorem 100, f is strongly definable in S. ▲

Lecture 10

Arithmetization of consistency; provable diagonal equivalences; provability predicates; Gödel's Second Incompleteness Theorem; Löb's Theorem

Wednesday 13 February 2019

10.1 Arithmetization of the statement that a system S is consistent

Gödel's Second Incompleteness Theorem for a system S satisfying certain conditions is the inference that if S is consistent, S cannot prove the consistency of S. Clearly, then, a condition for Gödel's Second Incompleteness Theorem to hold for a system S is that the consistency of the system be expressible by a sentence in the language of S.

The consistency of S is the condition that for every sentence X in the language of S, $S \nvdash (X \wedge \sim X)$, or equivalently (by propositional logic), there is some sentence X such that $S \nvdash X$. Accordingly, if we have constructed a formula $Pr_S(v_1)$ in the langauge of S that expresses $\{n : S \vdash E_n\}$, S is consistent if and only the sentence $\forall v_2 {\sim} Pr_S(\ulcorner(E_{v_2} \wedge {\sim}E_{v_2})\urcorner)$, or equivalently the sentence $\exists v_1 {\sim} Pr_S(v_1)$ is true.

Definition 70 (definition of Con_S) *For $Pr(v_1)$ a formula that expresses $\{n : S \vdash E_n\}$ and X any sentence in the language of S such that $S \vdash \sim X$, we let Con_S, the formal expression in the language of S of the consistency of S be $\sim Pr(\overline{\ulcorner X \urcorner})$.*

The notation Con_S does not notate relativity to the sentence X, nor relativity to the chosen Gödel numbering and arithmetization of syntax. I will say something later about provable invariance of the Gödel sentence and of Con_S with respect to Gödel numbering and arithmetization of syntax. As to relativity of Con_S to the unprovable sentence X, we have the following:

Lemma 102 *If $\sim Pr_S(\overline{\ulcorner X \urcorner})$ is true, for X any sentence in $\mathcal{L}(S)$ and $Pr(v_1)$ any formula in $\mathcal{L}(S)$ which expresses $\{n : S \vdash E_n\}$, then S is consistent.*

Proof. By the two hypotheses, $S \nvdash X$. Since an inconsistent system proves everything, S must be consistent. ▲

Lemma 103 *If S is consistent and $S \vdash \sim X$, then for $Pr(v_1)$ any formula in $\mathcal{L}(S)$ which expresses $\{n : S \vdash E_n\}$, $\sim Pr(\overline{\ulcorner X \urcorner})$ is true.*

Proof. By the first two hypotheses, $S \nvdash X$. then by the third hypothesis, $\sim Pr_S(\overline{\ulcorner X \urcorner})$ is true. ▲

Lemma 104 (justifying the definition of Con_S) *A system S is consistent if and only if $\sim Pr(\overline{\ulcorner X \urcorner})$ is true, for $Pr(v_1)$ any formula in the language of S that expresses $\{n : S \vdash E_n\}$, and X any sentence in the language of S such that $S \vdash \sim X$.*

Proof. Immediate from Lemmas 102 amd 103. ▲

Remark. The separation of the proof of Lemma 104 into Lemmas 102 amd 103 brings out that only one direction of the biconditional requires that $S \vdash \sim X$. The relativity to X is dealt with by fixing on a specific X such that $S \vdash \sim X$. A particularly simple such X is $0 = 0'$, since it is immediate from axiom N_2 for PA and Q, or from Ω_3 for R, that $S \vdash \sim 0 = 0'$, and it is fairly standard to take Con_S to be $\sim Pr(\overline{\ulcorner 0 = 0'})$.

The Second Incompleteness Theorem is established by formalizing in S the proof of the first half of the First Incompleteness Theorem, that if S is consistent, then $S \nvdash G$, which is arithmetized as $(Con_S \supset \sim Pr(\overline{\ulcorner G \urcorner}))$. Hence if $S \vdash Con_S$, then $S \vdash \sim Pr(\overline{\ulcorner G \urcorner})$ If the Diagonal Lemma is provable in S, then $S \vdash (G \equiv \sim Pr(\overline{\ulcorner G \urcorner})$, in that case $S \vdash G$, which, if S is consistent, it doesn't, by the first half of the First Incompleteness Theorem.

This argument requires that the Diagonal Lemma be provable in S, not just that it is true, as previously shown.

10.2 Formal provability of the Diagonal Lemma

As a first step to proving the Diagonal Lemma within a system S, we prove that for f a strongly definable total function, and $G(v_1)$ a formula with one free in the language of a system S, the property of n that $G(v_1)$ holds of $f(n)$ is expressible by a formula in the language of S.

Theorem 105 (provable substitution) *If a total function $f(x)$ is strongly definable in a system S, then for each formula $G(v_1)$ with one free variable, there is a formula $H(v_1)$ such that for each number n, $S \vdash (H(\overline{n}) \equiv G(\overline{f(n)}))$.*

Proof. Let $F(v_1, v_2)$ be a formula that strongly defines $f(x)$ in S. For given formula $G(v_1)$ let $H(v_1) =_{df} \exists v_2(F(v_1, v_2) \wedge G(v_2))$. Let n and m be such that $f(n) = m$. We establish the provability of the two halves of the required biconditional as follows:

(i) To show that $S \vdash (G(\overline{f(n)}) \supset H(\overline{n}))$: Since $F(v_1, v_2)$ strongly defines $f(x)$, by clause (1) $S \vdash F(\overline{n}, \overline{m})$. Hence by propositional logic in S, $S \vdash (G(\overline{m}) \supset (F(\overline{n}, \overline{m}) \wedge G(\overline{m}))$. Then by $\exists$-introduction in S (note that in inferring $\exists v_1 A(v_1)$ from $A(t)$, v_1 is substituted for some but not necessarily all occurrences of t in $A(t)$), $S \vdash (G(\overline{m}) \supset \exists v_2(F(\overline{n}, v_2) \wedge G(v_2)))$.

(ii) To show that $S \vdash (H(\overline{n}) \supset G(\overline{f(n)}))$: By $\forall$-elimination from the second conjunct of the condition for strong definability, $S \vdash (F(\overline{n}, v_2) \supset v_2 = \overline{m})$. Therefore by propositional logic in S, $S \vdash ((F(\overline{n}, v_2) \wedge G(v_2)) \supset (v_2 = \overline{m} \wedge G(v_2)))$. By =-Elimination in S, $S \vdash ((v_2 = \overline{m} \wedge G(v_2)) \supset G(\overline{m}))$, so by transitivity of $\supset$,
$S \vdash ((F(\overline{n}, v_2) \wedge G(v_2)) \supset G(\overline{m}))$. Then by $\forall$-Introduction,
$S \vdash \forall v_2(F(\overline{n}, v_2) \wedge G(v_2) \supset G(\overline{m}))$, and so by anti-prenexing,
$S \vdash (\exists v_2(F(\overline{n}, v_2) \wedge G(v_2)) \supset G(\overline{m}))$ ▲

Theorem 106 (provable diagonal equivalence) *Let S be any extension of R. For any formula with one free variable, $F(v_1)$, in the language of S, there is a sentence C such that $S \vdash (C \equiv F(\ulcorner C \urcorner))$.*

Proof. The diagonal function is total, and we have shown, in the proof of Proposition 53 (given as Problem 1 on Problem sheet 3) that it is Σ_1-expressible. Hence by Theorem 101, $d(v_1) = v_2$ is strongly definable in any Σ_0-complete system. By Theorem 105, given $F(v_1)$, there is a formula $H(v_1)$ such that for each n, $S \vdash (H(\overline{n}) \equiv F(\overline{d(n)}))$, so in particular, for $h = \ulcorner H(v_1) \urcorner$, $S \vdash (H(\overline{h}) \equiv F(\overline{d(h)}))$. $S \vdash (H(\overline{h}) \equiv H[\overline{h}])$, since substitution and quasi-substitution are logical equivalent, so $S \vdash (H[\overline{h}] \equiv F(\overline{d(h)}))$. By the construction of $d(x)$, $d(h) = \ulcorner H[\overline{h}] \urcorner$, so taking $C =_{df} H[\overline{h}]$, we have $S \vdash (C \equiv F(\overline{\ulcorner C \urcorner}))$, as required. ▲

10.3 Provability predicates.

The process of formalizing the argument for the First Incompleteness Theorem for S in S is intricate. By work of Paul Bernays and Martin Löb, the requirements for formalization are reduced to three conditions on the proof predicate for S. We shall establish the Second Incompleteness Theorem from the assumption of these three conditions. We also then have to show that the arithmetized proof predicate we have established for PA satisfies these three conditions.

Definition 71 (provability predicate) *A formula $P(v_1)$ is called a provability predicate for a system S if for all sentences X and Y in $\mathcal{L}(S)$ the following three conditions hold:*

P_1*: If $S \vdash X$, then $S \vdash P(\overline{\ulcorner X \urcorner})$.*

P_2*: $S \vdash (P(\overline{\ulcorner (X \supset Y) \urcorner}) \supset (P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner Y \urcorner})))$*

P_3*: $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner P(\overline{\ulcorner X \urcorner}) \urcorner}))$*

Note that P_1 is a one-way implication and not a biconditional. The converse implication is an instance of Σ_1-soundness. The effect of this is that these conditions on being a provability predicate do *not* require that a provability predicate expresses $\{n : S \vdash E_n\}$. Rather, it can express a superset of that set, in particular, the formula $x = x$ is a provability predicate.

Being a provability predicate is not extensional, i.e. as we shall see, there are pairs of formulas, even in the same class of the arithmetical hierarchy, that have the same extension, yet one of which is a provability predicate and the other is not.

Theorem 107 ($Pr(v_1)$ for PA is a provability predicate for PA) *The arithmetized proof predicate for PA which we have constructed is a provability predicate for PA.*

Proof.

P_1: We established property P_1 (and its converse) for the arithmetized proof predicate for PA by Theorem 26.

P_2: We need to show that for X and Y any sentences in the language of PA, PA$\vdash$ $(Pr(\overline{\ulcorner X \supset Y \urcorner}) \supset (Pr(\overline{\ulcorner X \urcorner}) \supset Pr(\overline{\ulcorner Y \urcorner})))$. This is given as an exercise on Problem sheet 5.

P_3: This is the arithmetization of P_1, and follows from the arithmetization in PA of the proof that PA is Σ_0-complete, referred to as provable Σ_0-completeness, which we will establish in Lecture 11. ▲

The following properties of a provability predicate are immediate consequences of the three conditions that define what it is to be a provability predicate.

Lemma 108 *For $P(v_1)$ a provability predicate for a system S,*

P_4 *If* $S \vdash (X \supset Y)$, *then* $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner Y \urcorner}))$.

P_5 *If* $S \vdash (X \supset (Y \supset Z))$, *then* $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset (P(\overline{\ulcorner Y \urcorner}) \supset P(\overline{\ulcorner Z \urcorner})))$.

P_6 *If* $S \vdash (X \supset (P(\overline{\ulcorner X \urcorner}) \supset Y))$, *then* $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner Y \urcorner}))$.

Proof.

P_4: If $S \vdash (X \supset Y)$, then by P_1, $S \vdash P(\overline{\ulcorner (X \supset Y) \urcorner})$. Then by P_2 and Modus ponens, $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner Y \urcorner}))$.

P_5: By P_4, $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner (Y \supset Z) \urcorner}))$. By P_2, $S \vdash (P(\overline{\ulcorner (Y \supset Z) \urcorner}) \supset (P(\overline{\ulcorner Y \urcorner}) \supset P(\overline{\ulcorner Z \urcorner})))$. Then by propositional logic, $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset (P(\overline{\ulcorner Y \urcorner}) \supset P(\overline{\ulcorner Z \urcorner}))$.

There is no corresponding result for $S \vdash ((X \supset Y) \supset Z)$.

P_6: If $S \vdash (X \supset (P(\overline{\ulcorner X \urcorner}) \supset Y))$, then by P_5
$S \vdash (P(\overline{\ulcorner X \urcorner}) \supset (P(\overline{\ulcorner P(\overline{\ulcorner X \urcorner}) \urcorner}) \supset P(\overline{\ulcorner Y \urcorner})))$. Then by L_2 and Modus ponens,
$S \vdash ((P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner P(\overline{\ulcorner X \urcorner}) \urcorner})) \supset (P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner Y \urcorner})))$ Then by P_3 and Modus ponens, $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset P(\overline{\ulcorner Y \urcorner}))$. ▲

10.4 Gödel's Second Incompleteness Theorem

Lemma 109 *For $P(v_1)$ a provability predicate for a system S which is an extension of R, there is a sentence G in the language of S such that $S \vdash (G \equiv {\sim}P(\overline{\ulcorner G \urcorner}))$.*

Proof. *Immediate from Theorem 106 by taking $F(v_1)$ as $P(v_1)$.* ▲

Lemma 110 (arithmetization of the First Incompleteness Theorem) *Let S be a system which extends R and has a provability predicate $P(v_1)$, and let G be a sentence in the language of S such that $S \vdash (G \equiv {\sim}P(\overline{\ulcorner G \urcorner}))$. Then for X any sentence in the language of S, $S \vdash ({\sim}P(\overline{\ulcorner X \urcorner}) \supset {\sim}P(\overline{\ulcorner G \urcorner}))$.*

Proof.

(1) $S \vdash (G \equiv {\sim}P(\overline{\ulcorner G \urcorner}))$	Lemma 109
(2) $S \vdash (G \supset {\sim}P(\overline{\ulcorner G \urcorner}))$	(1) $\wedge$-elimination
(3) $S \vdash (G \supset (P(\overline{\ulcorner G \urcorner}) \supset X))$	(2) Prop Logic
(4) $S \vdash (P(\overline{\ulcorner G \urcorner}) \supset P(\overline{\ulcorner X \urcorner}))$	(3) P_6

(5) $S \vdash (\sim P(\overline{\ulcorner X \urcorner}) \supset \sim P(\overline{\ulcorner G \urcorner}))$	(4) contraposition ▲

Theorem 111 (Gödel's Second Incompleteness Theorem) *Let S be a system which extends R and has provability predicate $P(v_1)$. Then for X any sentence in the language of S, if S is consistent, $S \nvdash \sim P(\overline{\ulcorner X \urcorner})$.*

Proof.

(1) $S \vdash (G \equiv \sim P(\overline{\ulcorner G \urcorner}))$	Lemma 109
(2) $S \vdash (\sim P(\overline{\ulcorner G \urcorner}) \supset G)$	(1) $\wedge$-Elimination$_R$
(3) $S \vdash (\sim P(\overline{\ulcorner X \urcorner}) \supset G)$	(2) Lemma 110 and prop logic
(4) $S \vdash \sim P(\overline{\ulcorner X \urcorner})$	Assumption
(5) $S \vdash G$	(3) (4) $\supset$-elimination
(6) $S \vdash P(\overline{\ulcorner G \urcorner})$	(5) P_1
(7) $S \vdash (G \supset \sim P(\overline{\ulcorner G \urcorner}))$	(1) $\wedge$-Elimination$_L$
(8) $S \vdash \sim G$	(6)(7) prop Logic
(9) S is inconsistent	(5)(8)
(10) $S \nvdash \sim P(\overline{\ulcorner X \urcorner})$	by RAA (4)(9) and consistency of S ▲

Remarks on this proof:

(1) The proof of Lemma 110 is strictly part of the proof of Gödel's Second Incompleteness Theorem. I have made it into a separate Lemma to highlight that part of the proof Gödel's Second Incompleteness Theorem which is the arithmetization of the proof of the first half of the First Incompleteness Theorem.

(2) Note that the three conditions on a Provability Predicate are used in this proof at the following points: P_1 at line (5) of Theorem 111 and P_2 and P_3 in the use of P_6 at line (4) of Lemma 109.

Remark: Gödel's Second Incompleteness Theorem is a generalization of the first half of the First Incompleteness Theorem, in the following sense. The first half of the First Incompleteness Theorem establishes, for G the Gödel sentence S, that if S is consistent, $S \nvdash G$. By the diagonal equivalence, this is tantamount to $S \nvdash \sim P(\overline{\ulcorner G \urcorner})$. The Second Incompleteness Theorem establishes that for *every* sentence X in the language of S, if S is consistent, $S \nvdash \sim P(\overline{\ulcorner X \urcorner}))$. The heart of the matter, however, is that the Gödel sentence for S is equivalent provably in S to the consistency of S. One direction of this equivalence is established at line (3) of the above proof of the Second Incompleteness Theorem: If S is consistent, i.e. some sentence is unprovable, then G holds. The converse of (3) for arbitrary X cannot be proved because it doesn't hold, if S is 1-inconsistent: If $S \vdash (G \supset \sim P(\overline{\ulcorner X \urcorner}))$ for X such that $S \vdash X$, then since by P_1, $S \vdash P(\ulcorner X \urcorner)$, $S \vdash\sim G$. However, if $S \vdash\sim X$, then indeed the implication holds, i.e.

Proposition 112 *Let $P(v_1)$ be a provability predicate for a system S, let G be a sentence in the language of S such that $S \vdash (G \equiv\sim P(\ulcorner G \urcorner))$, and let X be any sentence in the language of S such that $S \vdash\sim X$. Then $S \vdash (G \supset\sim P(\ulcorner X \urcorner))$.*

Proof. Since $S \vdash \sim X$, by propositional logic in S, $S \vdash (X \supset G)$. Then by P_4, $S \vdash (P(\ulcorner X \urcorner) \supset P(\ulcorner G \urcorner))$. By contraposition, $S \vdash (\sim P(\ulcorner G \urcorner) \supset\sim P(\ulcorner X \urcorner))$, so by $S \vdash (G \equiv\sim P(\ulcorner G \urcorner))$ and transitivity of implication in S, $S \vdash (G \supset\sim P(\ulcorner X \urcorner))$. ▲

Corollary 113 (to Proposition 112 and the proof of Theorem 111) *Let $P(v_1)$ be a provability predicate for a system S, let G be a sentence in the language of S such that $S \vdash (G \equiv\sim P(\ulcorner G \urcorner))$. and let X be any sentence in the language of S such that $S \vdash\sim X$. Then $S \vdash (G \equiv\sim P(\ulcorner X \urcorner))$.*

Proof. Proposition 112 and line (3) of the proof of Theorem 111 are the two halves of the biconditional. ▲

10.5 Löb's Theorem

Löb's Theorem is a deep result which characterizes the abstract properties of provability, i.e. it can be used as the fundamental axiom for a theory of provability, as we shall see in Lectures 13 and 14. It is also a generalization of the Second Incompleteness Theorem, though this was not immediately realized,. It arose in response to an almost jokey question in the 1950s by Leon Henkin: Is the sentence that asserts its own provability (there is such a sentence, by diagonalization, i.e. $S \vdash (Pr(\ulcorner H \urcorner) \equiv H))$ provable (in which case it is true) or unprovable (in which case it is false)? What Martin Löb showed was that from just half of that diagonal equivalence, i.e. $S \vdash (P(\ulcorner H \urcorner) \supset H)$, it follows that $S \vdash H$ (so H is provable and true), because for *any* sentence X and *any* provability predicate $P(v_1)$, if $S \vdash (P(\ulcorner X \urcorner) \supset X)$, then $S \vdash X$. **Note** that the converse holds by propositional logic, from L_1 and Modus ponens.

Theorem 114 (Löb's theorem) *Let S be a system* ***in*** *which the Diagonal Lemma is provable, and let $P(v_1)$ be a provability predicate for S. For X any sentence in the language of S, if $S \vdash (P(\ulcorner X \urcorner) \supset X)$, then $S \vdash X$.*

Proof. (1) Assume that $S \vdash (P(\ulcorner X \urcorner) \supset X)$.

(2) Let L be a provably diagonal sentence for the predicate $(P(v_1) \supset X)$, i.e. $S \vdash (L \equiv (P(\ulcorner L \urcorner) \supset X)$.

(3) $S \vdash (L \supset ((P(\overline{\ulcorner L \urcorner}) \supset X)$ from (2) by $\wedge$-Elimination.

(4) $S \vdash (P(\overline{\ulcorner L \urcorner}) \supset P(\overline{\ulcorner X \urcorner}))$ from (3) by P_6 from P_2 and P_3.

(5) $S \vdash (P(\overline{\ulcorner L \urcorner}) \supset X)$ from (4) and (1) by transitivity of $\supset$.

(6) $S \vdash ((P(\overline{\ulcorner L \urcorner}) \supset X) \supset L)$ from (2) by $\wedge$-Elimination.

(7) $S \vdash L$ from (5) and (6) by $\supset$-Elimination .

(8) $S \vdash P(\overline{\ulcorner L \urcorner})$ from (7) by P_1 .

(9) $S \vdash X$ from (8) and (5) by $\supset$-Elimination. ▲

Definition 72 *A sentence of the form $(P(\overline{\ulcorner X \urcorner}) \supset X)$ is called a Reflection Principle, or more specifically, a Local Reflection Principle.*

Remark: A reflection principle $(P(\overline{\ulcorner X \urcorner}) \supset X)$ expresses the soundness with respect to provability of X of the system for which $P(v_1)$ expresses provability, i.e. it says that if X is provable, then X, i.e. X is true. Löb's Theorem says that the only such statements that can be proved in a system are the ones that hold trivially by propositional logic, i.e. for which $S \vdash X$.

Theorem 115 *Löb's Theorem is a generalization of Gödel's Second Incompleteness Theorem.*

partial Proof. (i) Löb's Theorem implies the Second Incompleteness Theorem, as follows: Suppose $S \vdash \sim P(\overline{\ulcorner X \urcorner})$. Then by propositional logic in S, $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset X)$. Then by Löb's Theorem for S, $S \vdash X$. Then by property P_1 of $P(v_1)$ as a provability predicate for S, $S \vdash P(\overline{\ulcorner X \urcorner})$. Then S is inconsistent. So if S is consistent, $S \nvdash \sim P(\overline{\ulcorner X \urcorner})$.

(ii)The Second Incompleteness Theorem proves Löb's Theorem in those cases where $S \vdash \sim X$, as follows: Suppose $S \vdash ((P(\overline{\ulcorner X \urcorner}) \supset X))$ and $S \vdash \sim X$. Then by propositional logic, $S \vdash \sim P(\overline{\ulcorner X \urcorner})$. Then by the Second Incompleteness Theorem, S is inconsistent, so proves everything, so in particular, $S \vdash X$.

(iii) Löb's Theorem holds by proposition logic for X such that $S \vdash X$ (an implication is true if the consequent is true).

(iv) The Second Incompleteness Theorem for S does not establish Löb's Theorem for sentences X such that $S \nvdash X$ and $S \nvdash \sim X$. In particular, for G a Gödel sentence for S, the Second Incompleteness Theorem does not extablish that if $S \vdash (Pr(\overline{\ulcorner G \urcorner}) \supset G)$, then $S \vdash G$. We shall see a rigorous proof of this result when we come to the formalization of Provability Logic (in Lectures 13 and 14). ▲

Though Löb's Theorem does not follow uniformly from the Second Incompleteness Theorem, Löb's Theorem can be proved from the Second Incompleteness Theorem on a sentence by sentence basis. The situation is the following:

Theorem 116 *For each sentence X in the language of S, the Second Incompleteness Theorem for $S \cup \{\sim X\}$ implies that if $S \vdash (P(\overline{\ulcorner X \urcorner}) \supset X)$, then $S \vdash X$.*

Proof. Problem 4 on Problem sheet 5. ▲

Lecture 11

Provable Σ_1-completeness

Monday 18 February 2019

The aim of these notes is to give a detailed and rigorous proof of condition P_3 for a provability predicate. This is the most complex piece of arithmetization of syntax in this subject, and proof of this result is seldom included in expositions of the Second Incompleteness Theorem. Peter Smith in his book *An Introduction to Gödel's Theorems* gives a "sketch of a proof sketch" in half a page (p. 235). George Boolos, in his brilliant and invaluable book, *The Logic of Provability*, gives a detailed exposition of this result spread over 34 pages (pp. 15-49), but it seems to me that there is a gap in his proof at a crucial point (p. 46), which it is not easy to see how to fill.

Proposition 117 *For $Pr(v_1)$ a formula in the language of PA that expresses $\{n : PA \vdash E_n\}$ and X any Σ_1-sentence in the language of PA, the sentence $(X \supset Pr(\overline{\ulcorner X \urcorner}))$ is true.*

Proof. (i) If X is true, then by Σ_1-completeness of PA, PA $\vdash X$, and since $Pr(v_1)$ expresses $\{n : \text{PA} \vdash E_n\}$, $Pr(\overline{\ulcorner X \urcorner})$ is true. So $(X \supset Pr(\overline{\ulcorner X \urcorner}))$ is true.

(ii) If X is false, then $(X \supset Pr(\overline{\ulcorner X \urcorner}))$ is true. ▲

In this lecture we will establish that all these true sentences are provable in PA (provable Σ_1-completeness). Since for Y any sentence in the language of PA, the sentence $Pr(\overline{\ulcorner Y \urcorner})$ is Σ_1, these sentences include all sentences of the form $(Pr(\overline{\ulcorner Y \urcorner}) \supset Pr(\overline{\ulcorner Pr(\overline{\ulcorner Y \urcorner}) \urcorner}))$, i.e. P_3, the third condition on a provability predicate.

Note that for X a Σ_1-sentence, $(X \supset Pr(\overline{\ulcorner X \urcorner}))$ is Δ_2. As we shall see, PA is not Δ_2-complete. So the provability of these sentences in PA is specific to provability properties of Σ_1-sentences and of arithmetization of provability in PA.

While provable Σ_1-completeness is a deep theorem whose proof is complicated, provable completeness for Σ_0-sentences is very easy to show.

Proposition 118 (provable Σ_0-completeness for sentences) *For $Pr(v_1)$ a formula in the language of PA that expresses $\{n : PA \vdash E_n\}$ and X any Σ_0-sentence in the language of PA, $PA \vdash (X \supset Pr(\overline{\ulcorner X \urcorner}))$.*

Proof. Argument (1): (i) If X is true, then by Σ_0-completeness of PA, PA $\vdash X$, and since $Pr(v_1)$ is Σ_1, and expresses $\{n : \text{PA} \vdash E_n\}$, $Pr(\overline{\ulcorner X \urcorner})$ is a true Σ_1-sentence. Then by Σ_1-completeness of PA, $PA \vdash Pr(\overline{\ulcorner X \urcorner})$, so by propositional logic in PA, $PA \vdash (X \supset Pr(\overline{\ulcorner X \urcorner}))$.

(ii) If X is false, then $\sim X$ is a true Σ_0-sentence, so by Σ_0-completeness, $PA \vdash\sim X$, so by propositional logic in PA, $PA \vdash (X \supset Pr(\overline{\ulcorner X \urcorner}))$.

Argument (2): If X is Σ_0, then $(X \supset Pr(\overline{\ulcorner X \urcorner}))$ is Σ_1, and hence provable in PA by Σ_1-completeness. ▲

Remark about the strategy for proving provable Σ_1-completeness. Neither of the two arguments for Proposition 118 can be extended to the case of X a Σ_1-sentence. For Argument (1), (i) holds for X a Σ_1-sentence, but (ii) fails since the negation of a Σ_1-sentence is not, in general, a Σ_1-sentence. For Argument (2), we noted above that for X a Σ_1-sentence, $(X \supset Pr(\overline{\ulcorner X \urcorner}))$ is Δ_2, and as we shall see later, PA is not Δ_2-complete. The definition of what it is to be a Σ_1-formula, Definition 42, is explicit, rather than recursive, i.e. a Σ_1 formula is any formula of the form $\exists v_i F$ where F is a Σ_0-formula. So as in our proof that R and thereby Q and PA are Σ_1-complete (Propositions 58, 65, and 67, and Theorem 68), the proof of provable Σ_1-completeness has to go via a proof of provable Σ_0-completeness. However, for these purposes, provable Σ_0-completeness cannot be as given by Proposition 118, with its quite trivial proof.

The definition of Σ_0-formula, Definition 40, is recursive, so the proof of provable Σ_0-completeness must proceed by induction over the recursive definition of Σ_0-formulas. The sequence of formulas by which a Σ_0-sentence is generated by this recursion will in general contain free variables. Thus we must prove provable Σ_0-completeness for formulas that may contain free variables.

We cannot formulate this result for Σ_0-formulas with free variables as we did for Proposition 118. Writing $PA \vdash (v_1 + v_2 = v_3 \supset Pr(\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}))$, for example, does not express that PA proves that for any three numbers a, b, and c such that $a + b = c$, $PA \vdash \overline{a} + \overline{b} = \overline{c}$, since $Pr(\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}))$ says that PA proves the one particular formula $v_1 + v_2 = v_3$ (which it doesn't). Instead, to express what's wanted we use a notation (invented by Solomon Feferman) of putting a dot over a variable that occurs within an expression that occurs within $\ulcorner\urcorner$ to signify that the expression whose Gödel number is being generated varies with the value of that variable, rather than that the symbols which constitute that variable (i.e. v followed by a string of subscript symbols) are part of that expression. For example, $Pr(\overline{\ulcorner{\sim}0 = v'_{,}\urcorner})$ says that $Pr(v_i)$ holds of the Gödel number of Axiom N_2, while $\forall v_{,} Pr(\overline{\ulcorner{\sim}0 = \dot{v}'_{,}\urcorner})$ says that $Pr(v_i)$ holds of the Gödel number of each numerical instance of the axiom. What we need to show is that for each Σ_0-formula $F(v_{k_1}, \ldots, v_{k_m})$,

$$PA \vdash (F(v_{k_1}, \ldots, v_{k_m}) \supset Pr(\overline{\ulcorner F(\dot{v}_{k_1}, \ldots, \dot{v}_{k_m})\urcorner}))$$

To formulate this result, we need to find a way to express the condition that a substitution instance of a formula is provable.

To do this we need first to modify the definition we gave in Lecture 2 of quasi-substitution (Definition 26), which was defined just for substitution on the free variable v_1, i.e. $s(x, y) = \ulcorner\forall v_1(v_1 = \overline{y} \supset E_x)\urcorner$, to allow substitution on any specified variable, i.e. $s(x, y, z) = \ulcorner\forall v_z(v_z = \overline{y} \supset E_x)\urcorner$

Proposition 119 *For the function $s(x, y, z) = \ulcorner\forall v_z(v_z = \overline{y} \supset E_x)\urcorner$ there is a Σ_1-formula $S(x, y, z, w)$ in $\mathcal{L}_A$ such that for all natural numbers n_1, n_2, n_3, n_4, $S(\overline{n}_1, \overline{n}_2, \overline{n}_3, \overline{n}_4)$ is true if and only if $n_4 = \ulcorner\forall v_{n_3}(v_{n_3} = \overline{n}_2 \supset E_{n_1})\urcorner$.*

Proof.

The function $f(z) = \overbrace{5\ldots5}^{z+1}$ is generated by the following primitive recursion:

$f(0) = 5$

$f(n + 1) = f(n) *_{13} 5$, so by generalization of Theorem 37, $f(x) = y$ is Σ_1.

$s(x, y, z) = \ulcorner\forall v_z(v_z = \overline{y} \supset E_x)\urcorner = 96\,\overbrace{5\ldots5}^{z}\,26\,\overbrace{5\ldots5}^{z}\,\eta * 13^y * 8x3 =$
$96 * f(z-1) * 26 * f(z-1) * \eta * 13^y * 8x3$ is expressed by a Σ_1-formula $S(v_1, v_2, v_3, v_4)$ such that $S(\overline{n}_1, \overline{n}_2, \overline{n}_3, \overline{n}_4)$ is true if and only if $\ulcorner\forall v_{n_3}(v_{n_3} = \overline{n}_2 \supset E_{n_1})\urcorner = \overline{n_4}$. ▲

Definition 73 (arithmetized proof predicate with free variables) *For $Pr(v_1)$ a formula in the language of PA that expresses $\{n : PA \vdash E_n\}$ and $F(v_{k_1}, \ldots, v_{k_m})$ any formula in the language of PA with exactly the free variables shown, and $k = \max\{k_1, \ldots, k_m\}$, $Pr[\overline{\ulcorner F(v_{k_1}, \ldots, v_{k_m})\urcorner}](v_{k_1}, \ldots, v_{k_m}) =_{df}$
$\exists v_{k+1} \ldots \exists v_{k+m}(S(\overline{\ulcorner F(v_{k_1}, \ldots, v_{k_m})\urcorner}, v_{k_1}, \overline{k_1}, v_{k+1}) \wedge S(v_{k+1}, v_{k_2}, \overline{k_2}, v_{k+2}) \wedge \ldots$
$\wedge\, S(v_{k+m-1}, v_{k_m}, \overline{k_m}, v_{k+m})) \wedge Pr(v_{k+m}))$.*

Remark 1: $F(v_{k_1}, \ldots, v_{k_m})$ and $Pr[\overline{\ulcorner F(v_{k_1}, \ldots, v_{k_m})\urcorner}](v_{k_1}, \ldots, v_{k_m})$ have the same free variables.

Remark 2: The notation '$Pr[\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner}](v_{k_1},\dots,v_{k_m})$' stresses the important point that it's the Gödel number of the formula $F(v_{k_1},\dots,v_{k_m})$ that occurs in $Pr[\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner}](v_{k_1},\dots,v_{k_m})$, and *not* the formula itself. Occasionally, to avoid clutter, we will abbreviate this formula as $Pr[F(v_{k_1},\dots,v_{k_m})]$, which must be read bearing in mind that $F(v_{k_1},\dots,v_{k_m})$ is *not* a sub-formula of $Pr[F(v_{k_1},\dots,v_{k_m})]$. Note that this abbreviation *cannot* be used if we need to show the result of making a substitution for a free variable of $Pr[\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner}](v_{k_1},\dots,v_{k_m})$.

Choice of provably equivalent definitions: Where a formula $S(v_1,v_2)$ represents a total function $s(v_1)=v_2$ in a theory T, we can express the substitution $F(s(v_1))$ either by $\forall v_2(S(v_1,v_2)\supset F(v_2))$ or by $\exists v_2(S(v_1,v_2)\wedge F(v_2))$, since $\forall v_2(S(v_1,v_2)\supset F(v_2))$ and $\exists v_2(S(v_1,v_2)\wedge F(v_2))$ are logically equivalent. So we could have defined $Pr[\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner}](v_{k_1},\dots,v_{k_m})$ as
$\forall v_{k+1}\dots\forall v_{k+m}(S(\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner},v_{k_1},\overline{k_1},v_{k+1})\supset S(v_{k+1},v_{k_2},\overline{k_2},v_{k+2})\supset\dots\supset S(v_{k+m-1},v_{k_m},\overline{k_m},v_{k+m}))\supset Pr(v_{k+m}))$. We do not use this latter formula as the definition since, given that $Pr(v_1)$ is Σ_1, this formula is Π_2, whereas on the given definition, $Pr[\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner}](v_{k_1},\dots,v_{k_m})$ is, like $Pr(v_1)$, Σ_1.

Proposition 120 *For any natural numbers $a_1,\dots a_m$,*
$Pr[\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner}](\overline{a}_1,\dots,\overline{a}_m)$ *is true if and only if* $Pr(\overline{\ulcorner F(\overline{a}_1,\dots,\overline{a}_m)\urcorner})$ *is true.*

Proof. By the definition of $Pr[\overline{\ulcorner F\urcorner}]$, and provable equivalence of $F(\overline{a})$ and $F[\overline{a}]$. ▲

Corollary 121 (of Proposition 120) *For any natural numbers $a_1,\dots a_m$,*
$Pr[\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner}](\overline{a}_1,\dots,\overline{a}_m)$ *is true if and only if* $PA\vdash F(\overline{a}_1,\dots,\overline{a}_m)$.

Proof. By Proposition 120 and the fact that $\{n : PA\vdash E_n\}$ is expressed by the formula $Pr(v_1)$ (Corollary 51).

Proposition 122 *For a formula F with no free variables,*
$PA\vdash(Pr[\overline{\ulcorner F\urcorner}]\equiv Pr(\overline{\ulcorner F\urcorner}))$.

Proof. For F with no free variables, consider $Pr[\overline{\ulcorner F\urcorner}]$ with one vacuous quantifier, e.g. $\forall v_1(S(\overline{\ulcorner F\urcorner},v_1,0',v_2)\supset Pr(v_2))$. ▲

We need to generalize P_1 and P_2 to allow for occurrence of free variables. The generalization of P_1 expresses that if a formula with free variables is provable in PA, then for each sentence that results from substituting numerals for the free variables of that formula, PA proves the proof predicate for PA applied to the Gödel number of that sentence.

Theorem 123 ($P_1^*=P_1$ generalized to allow free variables) *For any formula $F(v_{k_1},\dots,v_{k_m})$ in the language of PA, if $PA\vdash F(v_{k_1},\dots,v_{k_m})$, then*
$PA\vdash Pr[\overline{\ulcorner F(v_{k_1},\dots,v_{k_m})\urcorner}](v_{k_1},\dots,v_{k_m})$.

Proof. The proof is by induction on the number of free variables in $F(v_{k_1},\dots,v_{k_m})$.

Base case:

From the assumption that $PA \vdash F(v_1)$, we need to show that
$PA \vdash \exists v_2(S(\overline{\ulcorner F(v_1)\urcorner}, v_1, 0', v_2) \wedge Pr(v_2))$

The following argument establishes this result for each numeral $\overline{n}$ in place of the variable v_1. The question then is how to establish this result for the variable v_1, which of course does not follow from establishing it for each $\overline{n}$.

(1) Since $S(v_1, v_2, v_3, v_4)$ is a Σ_1 formula that expresses the function $s(x, y, z) = \ulcorner \forall v_z(v_z = \overline{y} \supset E_x)\urcorner$, for all natural numbers n_1, n_2, n_3, n_4, $S(\overline{n}_1, \overline{n}_2, \overline{n}_3, \overline{n}_4)$ is true if and only if $\ulcorner \forall v_{n_3}(v_{n_3} = \overline{n}_2 \supset E_{n_1})\urcorner = n_4$. Hence
$S(\overline{\ulcorner F(v_1)}, \overline{n}, 0', \overline{\ulcorner \forall v_1(v_1} = \overline{n} \supset F(v_1))\urcorner)$ is a true Σ_1-sentence, and so by Σ_1-completeness of PA, $PA \vdash S(\overline{\ulcorner F(v_1)}, \overline{n}, 0', \ulcorner \forall v_1(v_1 = \overline{n} \supset F(v_1))\urcorner)$.

(2) We are given that $PA \vdash F(v_1)$, so $PA \vdash \forall v_1 F(v_1)$. By pure logic
$PA \vdash (\forall v_1 F(v_1) \supset F(\overline{n}))$, so by $\supset$-Elimination, $PA \vdash F(\overline{n})$, so by P_1,
$PA \vdash Pr(\overline{\ulcorner F(\overline{n})\urcorner})$. By pure logic, $PA \vdash (F(\overline{n}) \supset \forall v_1(v_1 = \overline{n} \supset F(v_1)))$. Then by P_1 and P_2 for PA, $PA \vdash (Pr(\overline{\ulcorner F(\overline{n})\urcorner}) \supset Pr(\overline{\ulcorner \forall v_1(v_1 = \overline{n} \supset F(v_1))\urcorner}))$, so by $\supset$-Elimination, $PA \vdash Pr(\overline{\ulcorner \forall v_1(v_1 = \overline{n} \supset F(v_1))\urcorner})$

By $\wedge$-Introduction from (1) and (2),
$PA \vdash (S(\overline{\ulcorner F(v_1)}, \overline{n}, 0', \ulcorner \forall v_1(v_1 = \overline{n} \supset F(v_1))\urcorner) \wedge Pr(\overline{\ulcorner \forall v_1(v_1 = \overline{n} \supset F(v_1))\urcorner}))$. Then by $\exists$-Introduction, $PA \vdash \exists v_2((S(\overline{\ulcorner F(v_1)}, \overline{n}, 0', v_2) \wedge Pr(v_2))$, i.e. $PA \vdash Pr[\overline{\ulcorner F(v_1)\urcorner}](\overline{n})$, and not $PA \vdash Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1)$, which is what's required. ▲

Theorem 124 ($P_2^* = P_2$ generalized to allow free variables) *For any formulas $F(v_{k_1}, \ldots, v_{k_m})$ and $G(v_{r_1}, \ldots, v_{r_s})$ in the language of PA,*
$PA \vdash (P[(F \supset G)](v_{k_1}, \ldots, v_{k_m}, v_{r_1}, \ldots v_{r_s}) \supset (P[F](v_{k_1}, \ldots, v_{k_m}) \supset P[G](v_{r_1}, \ldots v_{r_s})))$.

Proof. Exercise. ▲

Lemma 125 *For all formulas $F(v_{j_1}, \ldots, v_{j_m})$ and $G(v_{k_1}, \ldots, v_{k_n})$, $PA \vdash (P[\overline{\ulcorner F\urcorner}](v_{j_1}, \ldots, v_{j_m}) \supset (P[\overline{\ulcorner G\urcorner}](v_{k_1}, \ldots, v_{k_n}) \supset P[\overline{\ulcorner (F \wedge G)\urcorner}])(v_{j_1}, \ldots, v_{j_m}, v_{k_1}, \ldots, v_{k_n}))$*

Proof Exercise. ▲

Definition 74 *A term t is free for variable v_i in formula $F(v_i)$ if v_i in $F(v_i)$ does not occur within the scope of a quantifier whose variable of quantification is a free variable in t.*

Definition 75 *For $F(v_i)$ a formula with free variable v_i and t any term, $F(v_i/t)$ is the result of substituting the term t for all occurrences of the variable v_i in the formula $F(v_i)$.*

Lemma 126 *Let $F(v_{k_1}, \ldots, v_{k_m})$ be a formula with free-variables $v_{k_1}, \ldots, v_{k_m}$. Let*

$v_{r_1}, \dots, v_{r_s}$ *be variables free for* $v_{k_1}, \dots, v_{k_m}$*, respectively, in* $F(v_{k_1}, \dots, v_{k_m})$ *and let* $t_1, \dots, t_m$ *be terms free for* $v_{k_1}, \dots, v_{k_m}$ *in* $F(v_{k_1}, \dots, v_{k_m})$*. Then*

$PA \vdash (Pr[\overline{\ulcorner F(v_{k_1}, \dots, v_{k_m})\urcorner}](v_{k_1}/t_1, \dots, v_{k_m}/t_m) \equiv$
$Pr[\overline{\ulcorner F(v_{k_1}, \dots, v_{k_m})\urcorner}](v_{k_1}/v_{r_1}, \dots, v_{k_m}/v_{r_m})(v_{r_1}/t_1, \dots, v_{r_m}/t_m)).$

Proof. From the definition of $Pr[\overline{\ulcorner F(v_{k_1}, \dots, v_{k_m})\urcorner}](v_{k_1}, \dots, v_{k_m})$, substitutivity of $=$, and logical equivalence of $F(v_{k_1}, \dots, v_{k_m})$ and $F[v_{k_1}, \dots, v_{k_m}]$. ▲

Lemma 127 *Let* $F(v_{k_1}, \dots, v_{k_m})$ *be a formula with free-variables* $v_{k_1}, \dots, v_{k_m}$*, and let* $t_1(v_{r_1}), \dots, t_m(v_{r_m})$ *be terms free for* $v_{k_1}, \dots, v_{k_m}$ *in* $F(v_{k_1}, \dots, v_{k_m})$ *with variable* v_{k_i} *distinct from* v_{r_i}*. Then*

$PA \vdash (Pr[\overline{\ulcorner F(v_{k_1}/t_1(v_{r_1}) \dots, v_{k_m}/t_m(v_{r_m}))\urcorner}](v_{r_1}, \dots, v_{r_m}) \equiv$
$Pr[\overline{\ulcorner F(v_{k_1}, \dots, v_{k_m})\urcorner}](v_{k_1}/t_1(v_{r_1}), \dots, v_{k_m}/t_m(v_{r_m})))$

Proof. From the definition of $Pr[\overline{\ulcorner F(v_{k_1}, \dots, v_{k_m})\urcorner}](v_{k_1}, \dots, v_{k_m})$, substitutivity of $=$, and logical equivalence of $F(v_{k_1}, \dots, v_{k_m})$ and $F[v_{k_1}, \dots, v_{k_m}]$. ▲

Our proof in Lecture 7 that PA is Σ_0-complete went by way of proving the very strong result that the extremely weak system R is Σ_o-complete and then showing that R is a subsystem of PA. Proving Σ_0-completeness requires use of mathematical induction, so this proof cannot be formalized in R, which is to say that while we can prove that R is Σ_0-complete, provable Σ_0-completeness does not hold for R. However, it holds for PA, which has induction. We could establish provable Σ_0-completeness of PA by formalizing in PA the proof of the Σ_0-completeness of PA we gave before, but it would be very roundabout to formalize in PA that R is Σ_0-complete and then formalize in PA that R is a subsystem of PA. Instead, I will give a direct proof that PA is Σ_0-complete for the case of atomic formulas $v_1 + v_2 = v_3$, and $v_1 \cdot v_2 = v_3$, then formalize this proof in PA.

Lemma 128 *If* $a + b = c$*, then* $PA \vdash \overline{a} + \overline{b} = \overline{c}$*.*

Proof. We argue by induction on the free variable b in the statement, $\forall c$ (if $a+b = c$, then PA $\vdash \overline{a} + \overline{b} = \overline{c}$). (The universal quantifier on the variable c is to strengthen the induction hypothesis.) The result then follows by $\forall$-Elimination.

$b = 0$:

(1)	(1) $a + 0 = c$	Assumption
	(2) $a + 0 = a$	recursion equation for $+$
(1)	(3) $a = c$	(1)(2) transitivity of $=$
(1)	(4) $\overline{a}$ and $\overline{c}$ are the same expression	(3) defn of $\overline{n}$
(1)	(5) PA $\vdash \overline{a} = \overline{c}$	(4) PA $\vdash x = x$
	(6) PA $\vdash \overline{a} + 0 = \overline{a}$	$N3$ in PA

(1)	(7) PA $\vdash \overline{a} + 0 = \overline{c}$	(5)(6) substitutivity of $=$
	(8) if $a + 0 = c$, then PA $\vdash \overline{a} + 0 = \overline{c}$	(1)(7) if-then Intro
	(9) $\forall c$ (if $a + 0 = c$, then PA $\vdash \overline{a} + 0 = \overline{c}$)	(8) $\forall$-Intro $(*)$

$(*)$ c not free in any assumption on which (8) depends, since (8) depends on no assumptions.

Induction step.

(1)	(1) $\forall c$ (if $a + b = c$, then PA $\vdash \overline{a} + \overline{b} = \overline{c}$)	Induction Hypothesis
(2)	(2) $a + b' = c$	Assumption
	(3) $a + b' = (a + b)'$	Recursion equation for $+$
	(4) $a + b = a + b$	logic of $=$
	(5) $\exists z(a + b = z)$	(4) $\exists$-Intro
(6)	(6) $a + b = d$	Assumption
(6)	(7) $(a + b)' = d'$	(6) logic of $=$
(2)(6)	(8) $d' = c$	(7)(3)(2) transitivity of $=$
(2)(6)	(9) $\overline{d'}$ is the same expression as $\overline{c}$	(8) defn of $\overline{n}$
	(10) $\overline{d'}$ is the same expression as $\overline{d}'$	Corollary 5
(2)(6)	(11) PA $\vdash \overline{d}' = \overline{c}$	(9)(10) PA $\vdash x = x$
(1)	(12) (if $a + b = d$, then PA $\vdash \overline{a} + \overline{b} = \overline{d}$)	(1) $\forall$-Elim
(6)(1)	(13) PA $\vdash \overline{a} + \overline{b} = \overline{d}$	(6)(11) Modus ponens
(6)(1)	(14) PA $\vdash (\overline{a} + \overline{b})' = \overline{d}'$	(13) logic of $=$ in PA
	(15) PA $\vdash \overline{a} + \overline{b}' = (\overline{a} + \overline{b})'$	$N4$ in PA
(6)(1)	(16) PA $\vdash \overline{a} + \overline{b}' = \overline{d}'$	(14)(15) transitivity of $=$
(2)(6)(1)	(17) PA $\vdash \overline{a} + \overline{b}' = \overline{c}$	(11)(16) transitivity of $=$
(2)(1)	(18) PA $\vdash \overline{a} + \overline{b}' = \overline{c}$	(5)(6) $\exists$-Elim $(*)$
(2)(1)	(19) PA $\vdash \overline{a} + \overline{b'} = \overline{c}$	Corollary 5
(1)	(20) (if $a + b' = c$, then PA $\vdash \overline{a} + \overline{b'} = \overline{c}$)	(2)(19) $\supset$-Intro
(1)	(21) $\forall c$ (if $a + b' = c$, then PA $\vdash \overline{a} + \overline{b'} = \overline{c}$)	(20) $\forall$-Intro $(**)$

$(*)$ d not free in (6)(5)(2)(1).

$(**)$ c not free in (1).

Hence by induction on b, $\forall c$ (if $a + b = c$, then PA $\vdash \overline{a} + \overline{b} = \overline{c}$).

Hence by $\forall$-Elimination, if $a + b = c$, then PA $\vdash \overline{a} + \overline{b} = \overline{c}$. ▲

Lemma 129 *If $a \cdot b = c$, then $PA \vdash \overline{a} \cdot \overline{b} = \overline{c}$.*

Proof. We argue by induction on the variable b in the sentence, For all c, if $a \cdot b = c$, then PA $\vdash \overline{a} \cdot \overline{b} = \overline{c}$.

$\underline{b = 0}$ Assume $a \cdot 0 = c$. Since $a \cdot 0 = 0$, $c = 0$.

If $c = 0$, PA $\vdash \overline{c} = 0$.

By N_5, PA $\vdash \overline{a} \cdot 0 = 0$. Hence by logic of identity in PA, PA $\vdash \overline{a} \cdot 0 = \overline{c}$. So we have shown that if $a \cdot 0 = c$, then $PA \vdash \overline{a} \cdot 0 = \overline{c}$.

Then by universal generalization,
for any c, if $a \cdot 0 = c$, then $PA \vdash \overline{a} \cdot 0 = \overline{c}$.

<u>Induction step</u>

Induction hypothesis: for all c, if $a \cdot b = c$, then PA $\vdash \overline{a} \cdot \overline{b} = \overline{c}$.

Assume $a \cdot b' = c$. By the recursion equations for multiplication, $a \cdot b' = a \cdot b + b$. Let d be such that $a \cdot b = d$. By instantiation of the universal quantifier in the IH by d and Modus ponens, $PA \vdash \overline{a} \cdot \overline{b} = \overline{d}$. If $d + b = c$, then $PA \vdash \overline{d} + \overline{b} = \overline{c}$, as proved in lecture. Then by substitutivity of identity, $PA \vdash \overline{a} \cdot \overline{b} + \overline{b} = \overline{c}$. By universal generalization and instantiation from N_6, $PA \vdash \overline{a} \cdot \overline{b}' = \overline{a} \cdot \overline{b} + \overline{b}$. Hence by logic of identity in PA, $PA \vdash \overline{a} \cdot \overline{b}' = \overline{c}$. So if $a \cdot b' = c$, then $PA \vdash \overline{a} \cdot \overline{b}' = \overline{c}$. So for any c, if $a \cdot b' = c$, then $PA \vdash \overline{a} \cdot \overline{b}' = \overline{c}$. ▲

We now turn to proof of the main theorem.

Theorem 130 (provable Σ_0-completeness with <u>free variables</u>) *For each Σ_0-formula $F(v_{k_1}, \ldots, v_{k_m})$, $PA \vdash (F(v_{k_1}, \ldots, v_{k_m}) \supset Pr[\overline{\ulcorner F(v_{k_1}, \ldots, v_{k_m}) \urcorner}](v_{k_1}, \ldots, v_{k_m}))$.*

Proof. By induction over the inductive definition of Σ_0-formulas.

Base case:

F is an atomic formula, i.e. a formula of the form $t_1 = t_2$ or $t_1 \leq t_2$ for t_1, t_2 terms. The proof of this case is by a double induction over the recursive definition of terms. We will prove one of these cases. The others are similar.

PA $\vdash (v_1 + v_2 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3))$, i.e. PA $\vdash (v_1 + v_2 = v_3 \supset \exists v_4 \exists v_5 \exists v_6 (S(\ulcorner v_1 + v_2 = v_3 \urcorner, v_1, 0', v_4) \wedge S(v_4, v_2, 0'', v_5) \wedge S(v_5, v_3, 0''', v_6) \wedge Pr(v_6))$

The following is an informal description of a formal proof *within* PA. The proof is by induction on the variable v_2, but rather than argue by induction on v_2 in the formula $(v_1 + v_2 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3))$, we argue by induction on v_2 in the formula $\forall v_3 (v_1 + v_2 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_3, v_2))$, in order to have a stronger Induction Hypothesis.

Base case: $v_2 = 0$. We need to show that PA $\vdash \forall v_3(v_1{+}0 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, 0, v_3))$.

(1)	(1) $v_1 + 0 = v_3$	Assumption
	(2) $v_1 + 0 = v_1$	N_3
(1)	(3) $v_1 = v_3$	(1) (2) subst =
	(4) $Pr[\overline{\ulcorner v_1 + 0 = v_1 \urcorner}](v_1)$	(2) P_1^*
	(5) $(Pr[\overline{\ulcorner v_1 + 0 = v_1 \urcorner}](v_1) \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/0, v_3/v_1))$	Lemma 127
	(6) $Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/0, v_3/v_1)$	(4)(5) $\supset$-elim
(1)	(7) $Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, 0, v_3/v_1/v_3)$	(6) (3) subst =
	(8) $(v_1 + 0 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, 0, v_3))$	(1)(7) $\supset$-intro
	(9) $\forall v_3(v_1 + 0 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, 0, v_3))$	(8) $\forall$-intro

Induction step:

We have to show, within PA, that there exists a derivation of
$(\forall v_3(v_1 + v_2 = v_3 \supset Pr[\overline{\overline{v}_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3)) \supset$
$\forall v_3(v_1 + v_2' = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3)))$.

(1)	(1) $\forall v_3(v_1 + v_2 = v_3 \supset Pr[\overline{\overline{v}_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3)$	Assumption (induction hypothes
(2)	(2) $v_1 + v_2' = v_3$	Assumption
	(3) $v_1 + v_2' = (v_1 + v_2)'$	N_4
(2)	(4) $(v_1 + v_2)' = v_3$	(2)(3) logic of =
(2)	(5) $\exists v_4(v_4' = v_3)$	(4) $\exists$-Intro
(6)	(6) $v_4' = v_3$	Assumption
(6)(2)	(7) $(v_1 + v_2)' = v_4'$	(4)(6) subst of =
(6)(2)	(8) $v_1 + v_2 = v_4$	(7) N_1, logic
(1)	(9) $(v_1 + v_2 = v_4 \supset Pr[\overline{\overline{v}_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_4))$	(1) $\forall$-elimin
(1)(2)(6)	(10) $Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_4)$	(8)(9) $\supset$-Elim
	(11) $((v_1 + v_2 = v_4 \wedge v_1 + v_2' = (v_1 + v_2)') \supset v_1 + v_2' = v_4')$	substitutivity of =
	(12) $Pr[\overline{\ulcorner ((v_1 + v_2 = v_4 \wedge v_1 + v_2' = (v_1 + v_2)') \supset v_1 + v_2' = v_4') \urcorner}](v_1, v_2, v_4)$	(11) P_1^*
	(13) $((Pr[\overline{\ulcorner v_1 + v_2 = v_4 \urcorner}](v_1, v_2, v_4) \wedge Pr[\overline{\ulcorner v_1 + v_2' = (v_1 + v_2)' \urcorner}](v_1, v_2)) \supset$ $Pr[\overline{\ulcorner v_1 + v_2' = v_4' \urcorner}](v_1, v_2, v_4))$	(12) P_2^*
	(14) $(Pr[\overline{\ulcorner v_1 + v_2 = v_4 \urcorner}](v_1, v_2, v_4) \equiv Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3/v_4))$	Lemma 126
	(15) $(Pr[\overline{\ulcorner v_1 + v_2' = v_4' \urcorner}](v_1, v_2, v_4) \equiv Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3/v_4'))$	Lemma 127
	(16) $((Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_4) \wedge Pr[\overline{\ulcorner v_1 + v_2' = (v_1 + v_2)' \urcorner}](v_1, v_2)) \supset$ $Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3/v_4'))$	(13)(14)(15) prop logic
	(17) $Pr[\overline{\ulcorner v_1 + v_2' = (v_1 + v_2)' \urcorner}](v_1, v_2)$	(3) P_1^*

(1)(2)(6)	(18)	$Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3/v_4')$	(16)(10)(17) $\supset$-Elim
(1)(2)(6)	(19)	$Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3/v_4'/v_3)$	(18)(6) subst $=$
(1)(2)	(20)	$Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3)$	(6)(19) $\exists$-Elim
(1)	(21)	$(v_1 + v_2' = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3))$	(2)(20) $\supset$-intro
(1)	(22)	$\forall v_3(v_1 + v_2' = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3))$	(21) $\forall$-Intro v_3 not free in (1)
	(23)	$(\forall v_3(v_1 + v_2 = v_3 \supset Pr[\overline{\overline{v}_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3)) \supset$ $\forall v_3(v_1 + v_2' = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2/v_2', v_3)))$	(1)(22) $\supset$-Intro

By the instance of N_{12} for induction on v_2 in the formula $\forall v_3(v_1 + v_2 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3))$, we have proved *in* PA, $\forall v_3(v_1 + v_2 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3))$. Then by one step of $\forall$-elimination, we have $(v_1 + v_2 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3 \urcorner}](v_1, v_2, v_3))$, which was to be proved.

We now turn to the induction steps of the proof.

(i) F is $\sim G$ where G is Σ_0.

Exercise.

(ii) F is $(G \wedge H)$, for G and H both Σ_0-formulas.

(1) $(G \supset Pr[G])$	Induction hypothesis
(2) $(H \supset Pr[H])$	Induction hypothesis
(3) $(G \supset (H \supset (G \wedge H)))$	propositional logic
(4) $Pr[(G \supset (H \supset (G \wedge H)))]$	(3) Theorem 123
(5)$(Pr[(G \supset (H \supset (G \wedge H)))] \supset (Pr[G] \supset Pr[(H \supset (G \wedge H))]))$	Theorem 124
(6) $(Pr[(H \supset (G \wedge H))] \supset (Pr[H] \supset Pr[(G \wedge H)]))$	Theorem 124
(7) $(Pr[G] \supset (Pr[H] \supset Pr[(G \wedge H)]))$	(4) (5) (6) propositional logic
(8) $((G \wedge H) \supset Pr[(G \wedge H)])$	(1) (2) (7) propositional logic

(iii) F is $(\forall v_1 \leq v_2)G(v_1)$, for $G(v_1)$ a Σ_0-formula. This means that v_2 is free in F. To simplify notation we shall take it that no other variables are free in F, i.e. v_1 is the only free variable in $G(v_1)$. We need to give a proof *in* PA of $(F \supset Pr[F])$, i.e. $((\forall v_1 \leq v_2)G(v_1) \supset \forall v_3(S(\overline{\ulcorner(\forall v_1 \leq v_2)G(v_1)\urcorner}, v_2, 0'', v_3) \supset P(v_3))$.

The proof is by induction on the variable v_2 occurring free in the formula $(F \supset Pr[F])$.

Base case: We need to prove that $((\forall v_1 \leq 0)G(v_1) \supset \forall v_3(S(\overline{\ulcorner(\forall v_1 \leq 0)G(v_1)\urcorner}, 0, 0'', v_3) \supset P(v_3))$

To simplify notation I shall write v_1 as x and v_2 as y.

(1) $(G(x) \supset Pr[\overline{\ulcorner G(x) \urcorner}](x))$	Induction hypothesis for the main induction
(2) $(G(x) \supset Pr[G(x)](x))(0)$	(1) $\forall$-Intro, $\forall$-Elim
(3) $(G(0) \supset Pr[G(x)(x)](0))$	(2) defn of subst
(4) $(Pr[G(x)](0) \equiv Pr[G(0)])$	Lemma 126
(5) $(G(0) \supset Pr[G(0)])$	(3) (4) propositional logic
(6) $(\forall x \leq 0)G(x) \equiv G(0))$	provable in PA
(7) $((\forall x \leq 0)G(x) \supset Pr[(\forall x \leq 0)G(x)])$	(5) (6) logical equivalences

Induction step: On the assumption $((\forall x \leq y)G(x) \supset Pr[\overline{\ulcorner(\forall x \leq y)G(x)\urcorner}](y))$, which is the induction hypothesis for this sub-induction, we need to establish $((\forall x \leq y')G(x) \supset Pr[\overline{\ulcorner(\forall x \leq y')G(x)\urcorner}](y'))$.

(1)	(1) $((\forall x \leq y)G(x) \supset Pr[\overline{\ulcorner(\forall x \leq y)G(x)\urcorner}](y))$	Assumption IH for sub-induction
(2)	(2) $(\forall x \leq y')G(x)$	Assumption
(2)	(3) $\forall x(x \leq y' \supset G(x))$	(2) defn of $(\forall x \leq y')$
(2)	(4) $\forall x((x \leq y \vee x = y') \supset G(x))$	(3) N_6
(2)	(5) $\forall x((x \leq y \supset G(x)) \wedge (x = y' \supset G(x))$	(4) propositional logic
(2)	(6) $(\forall x(x \leq y \supset G(x)) \wedge \forall x(x = y' \supset G(x)))$	(5) predicate logic
(2)	(7) $((\forall x \leq y)G(x) \wedge G(y'))$	(6) defn of $(\forall x \leq y)$, logic
(8)	(8) $(G(x) \supset Pr[\overline{\ulcorner G(x)\urcorner}](x))$	Assumption IH for main induction
(8)	(9) $(G(y') \supset Pr[\overline{\ulcorner G(x)\urcorner}](y'))$	(8) $\forall$-Intro, $\forall$-Elim
(1)(2)(8)	(10) $(Pr[\overline{\ulcorner(\forall x \leq y)G(x)\urcorner}](y) \wedge Pr[\overline{\ulcorner G(x)\urcorner}](y'))$	(7)(1)(9) propositional logic
(1)(2)(8)	(11) $Pr[\overline{\ulcorner((\forall x \leq y)G(x) \wedge G(y))\urcorner}](x, y')$	(10) Lemma 125
(1)(2)(8)	(12) $Pr[(\forall x \leq y')G(x)]$	(11) N_8, P_1^*, P_2^*
(1)(8)	(13) $((\forall x \leq y')G(x) \supset Pr[(\forall x \leq y')G(x)])$	(2) (12) $\supset$-Intro

Step (9) in the above derivation calls for comment. In deriving (9) from (8) $\forall$-Intro is applied to (8) and the status of (8), as an induction hypothesis, is that of assumption. If the variable in (8) were the variable of induction this $\forall$-Intro would be illegitimate. However, the induction of which (8) is an induction hypothesis is over formulas, so the assumption is about the formula $G(x)$, and not about x, i.e. x is a free variable in (8) to which $\forall$-Intro may be applied. ▲

Corollary 131 (provable Σ_1-completeness with free variables) *For each Σ_1-formula $F(v_{k_1}, \ldots, v_{k_m})$, PA $\vdash (F(v_{k_1}, \ldots, v_{k_m}) \supset Pr[\overline{\ulcorner F(v_{k_1}, \ldots, v_{k_m})\urcorner}](v_{k_1}, \ldots, v_{k_m})$.*

Proof. We abbreviate $v_{k_1}, \ldots, v_{k_m}$ as $\mathbf{v}$, to avoid clutter. That $F(\mathbf{v})$ is a Σ_1-formula means there is a Σ_0-formula $G(\mathbf{v}, v_i)$ such that $F(\mathbf{v}) = \exists v_i G(\mathbf{v}, v_i)$. The following describes a derivation *in* PA.

(1) $(G(\mathbf{v}, v_i) \supset Pr[\overline{\ulcorner G(\mathbf{v}, v_i)\urcorner}](\mathbf{v}, v_i)$	Theorem 130
(2) $(G(\mathbf{v}, v_i) \supset \exists v_i G(\mathbf{v}, v_i))$	predicate logic
(3) $(Pr[\overline{\ulcorner G(\mathbf{v}, v_i)\urcorner}](\mathbf{v}, v_i) \supset Pr[\overline{\ulcorner \exists v_i G(\mathbf{v}, v_i)\urcorner}](\mathbf{v}))$	(2) P_1^*, P_2^*
(4) $(G(\mathbf{v}, v_i) \supset Pr[\overline{\ulcorner \exists v_i G(\mathbf{v}, v_i)\urcorner}](\mathbf{v})$	(1) (3) propositional logic
(5) $\forall v_i (G(\mathbf{v}, v_i) \supset Pr[\overline{\ulcorner \exists v_i G(\mathbf{v}, v_i)\urcorner}](\mathbf{v})$	(4) $\forall$-Intro
(6) $(\exists v_i G(\mathbf{v}, v_i) \supset Pr[\overline{\ulcorner \exists v_i G(\mathbf{v}, v_i)\urcorner}](\mathbf{v})$	(5) anti-prenexing, since v_i not free in the consequent of (5)

▲

Lecture 12

The ω-rule and uniform reflection; PA proves that PA proves every instance of the Gödel sentence; Π_1-uniform reflection and consistency; PA is Π_1-conservative over $\mathbf{PA}_{\Pi_2} \cup \{Con_{PA}\}$

Wednesday 20 February 2019

12.1 The ω-rule

We noted in Section 8.7 that the first half of the Gödel Incompleteness Theorem for a Σ_0-complete system S establishes that if S is consistent, then it's ω-incomplete, i.e. we have, for each n, $S \vdash \sim Prov(\overline{\ulcorner G \urcorner}, \overline{n})$, but $S \nvdash \forall v_2 \sim Prov(\overline{\ulcorner G \urcorner}, v_2)$. Hence, as a means of "overcoming" the Gödel incompleteness of a formal system S, we might think of adding the following an (infinitary) inference rule to S which yields $\forall v_1 F(v_1)$ from each of its numerical instances, i.e.

Definition 76 (adding the ω-rule to a system S) *We extend a system S of the kind we have been considering in this course to a system S^ω by adding the inference* *rule that if for each n, $S^\omega \vdash F(\overline{n})$, then $S^\omega \vdash \forall v_1 F(v_1)$, or in a two-dimensional display.*

$$\frac{S^\omega \vdash F(0), S^\omega \vdash F(\overline{1}), \ldots, S^\omega \vdash F(\overline{n}) \ldots}{S^\omega \vdash \forall y F(y)}$$

It's immediate that

Proposition 132 *The ω-rule is sound, in the sense that if a system S is sound, S^ω is sound.*

Proof. If each numerical instance of a formula $F(v_i)$ with one free variable is true (in the standard model of arithmetic), then $\forall v_i F(v_i)$ is true (in the standard model of arithmetic). ▲

Note that the ω-rule is an infinite version of $\wedge$-introduction, i.e.

$$\frac{S \vdash A \quad S \vdash B}{S \vdash (A \wedge B)}$$

Definition 77 (a proof in a system with the ω-rule) *For S^ω as in Definition 76, a proof of a formula X is a rooted tree of finite height, which has either one, two or ω-many upward branches from each node, at the nodes of which are formulas, with X at the root node of the tree, and at the top nodes are axioms of S. The formula at a non-top node follows by a rule of inference of S from the formulas at the nodes immediately above it. We write $S^\omega \vdash X$ if there is a proof of X in S^ω.*

Corollary 133 (of Definition 77) $PA^\omega \vdash G$

Proof Immediate from Definition 77 and the proof of Theorem 91.

This corollary is merely a relabeling of what we already knew. The ω-rule in this form is so strong (which is not a virtue) that a much stronger result holds by a vastly weaker argument, namely that R plus the ω-rule proves all true sentences.

Proposition 134 (R^ω is complete) *If a sentence X in $\mathcal{L}_A$ is true, $R^\omega \vdash X$.*

Proof. We argue by induction over the arithmetical hierarchy.

Base case: If X is a true Σ_0-sentence, then by Proposition 65, $R \vdash X$, and hence $R^\omega \vdash X$.

Induction step: Assume for Induction Hypothesis that the result holds for Σ_n and Π_n-sentences. (i) Let X be a true Σ_{n+1}-sentence $\exists v_i F(v_i)$, where $F(v_i)$ is a Π_n formula. Then for some natural number m, $F(\overline{m})$ is a true Π_n-sentence. Then by Induction Hypothesis $R^\omega \vdash F(\overline{m})$. Since $R \vdash (F(\overline{m}) \supset \exists v_i F(v_i))$, $R^\omega \vdash \exists v_i F(v_i)$.

(ii) Let X be a true Π_{n+1}-sentence $\forall v_i F(v_i)$. Then for each number n, $F(\overline{n})$ is a true Σ_n-sentence. Then by Induction Hypothesis, for each n, $R^\omega \vdash F(\overline{n})$. Then by one application of the ω-rule, $R^\omega \vdash \forall v_i F(v_i)$. ▲

12.2 The arithmetized ω-rule: Uniform Reflection

The ω-rule has infinitely many premisses. Hence we cannot strictly write down an application of the ω-rule. However, we can finitely express an application of the ω-rule by stating in a single sentence that all numerical instances of a given formula $F(v_1)$ are provable in a given system. By the arithmetization of syntax, such single sentences can be expressed *in* the language of arithmetic, namely as $\forall v_1 Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1)$, where $Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1)$ is defined by Definition 73 in Lecture 11. We can then give finite expression to an ω-rule by the sentence:

$$(\forall v_1 Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1) \supset \forall v_1 F(v_1)).$$

Such sentences are called Uniform Reflection Principles (see Smorynski [15], p. 845):

Definition 78 *For $F(v_1)$ a formula in the language of PA with one free variable, a* Uniform Reflection Principle *is any sentence of the form*
$(\forall v_1 Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1) \supset \forall v_1 F(v_1))$.

PA extended by Uniform Reflection Principles for all one-place formulas is strictly weaker than PA extended by the infinitary ω-rule, since PA + all Uniform Reflection Principles is axiomatic and hence incomplete, while PA + ω-rule = true arithmetic.

12.3 PA proves that PA proves every instance of the Gödel sentence

The proof that PA proves that PA proves every instance of the Gödel sentence makes essential use of provable Σ_0-completeness of PA, as established in the previous lecture.

Theorem 135 (*PA* proves that *PA* proves every instance of the Gödel sentence)
$PA \vdash \forall v_1 Pr[\overline{\ulcorner {\sim} Prov(\overline{\ulcorner G\urcorner}, v_1)\urcorner}](v_1)$

Proof. The following natural deduction shows the existence of a formal proof *in* PA of $\forall v_1 Pr[\overline{\ulcorner {\sim} Prov(\overline{\ulcorner G\urcorner}, v_1)\urcorner}](v_1)$.

(1)	(1) $\sim Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}](v_1)$	Assumption
	(2) $(0 = 0' \supset \sim Prov(\overline{\ulcorner G \urcorner}, v_1))$	PA $\vdash \sim 0 = 0'$
	(3) $(Pr[0 = 0'] \supset Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}])(v_1)$	(2) P_1^*, P_2^*, $\supset$-Elimination
	(4) $(\sim Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}](v_1) \supset \sim Pr[0 = 0'])$	(3) contraposition
(1)	(5) $\sim Pr[\overline{\ulcorner 0 = 0' \urcorner}]$	(1) (4) $\supset$-Elimination
(1)	(6) $\sim Pr(\overline{\ulcorner 0 = 0' \urcorner})$	(5) Proposition 122 since $0 = 0'$ has no free variables
	(7) $(\sim Pr(\overline{\ulcorner 0 = 0' \urcorner}) \supset \sim \exists v_1 Prov(\overline{\ulcorner G \urcorner}, v_1))$	Lemma 110
(1)	(8) $\sim \exists v_1 Prov(\overline{\ulcorner G \urcorner}, v_1))$	(7) (6) $\supset$-Elimination
(1)	(9) $\forall v_1 \sim Prov(\overline{\ulcorner G \urcorner}, v_1))$	(8) pred logic
(1)	(10) $\sim Prov(\overline{\ulcorner G \urcorner}, v_1))$	(9) $\forall$-Elimination
	(11) $(\sim Prov(\overline{\ulcorner G \urcorner}, v_1) \supset Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}])(v_1)$	Theorem 130
(1)	(12) $Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}](v_1)$	(10) (11) $\supset$-Elimination
	(13) $\sim\sim Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}](v_1)$	(1)(12) $\sim$-Introduction
	(14) $Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}](v_1)$	(13) propositional logic
	(15) $\forall v_1 Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}](v_1)$	(14) $\forall$-Introduction ▲

Corollary 136 *$PA \cup \{(\forall v_1 Pr[\overline{\ulcorner \sim Prov(\overline{\ulcorner G \urcorner}, v_1)\urcorner}](v_1) \supset \forall v_1 \sim Prov(\overline{\ulcorner G \urcorner}, v_1))\}$ $\vdash G$*

Proof. By Theorem 135, $\supset$-Elimination, and the fact that PA $\vdash (G \equiv \forall v_1 \sim Prov(\overline{\ulcorner G \urcorner}, v_1))$. ▲

12.4 Equivalence of Π_1-Uniform Reflection and consistency

Π_1-Uniform Reflection for PA, i.e. the Uniform Reflection Principle restricted to Π_1-formulas, is provably equivalent in PA to Con_{PA}, the formal consistency statement for PA, $\sim Pr_{PA}(\overline{\ulcorner 0 = 0' \urcorner})$.

Theorem 137 *PA + Π_1-uniform reflection $\vdash \sim Pr_{PA}(\overline{\ulcorner 0 = 0' \urcorner})$.*

Proof. The following argument is in PA. Take the instance of Π_1-uniform reflection for the Π_1-formula $\forall v_1(0 = 0' \wedge v_1 = v_1)$, i.e. $(\forall v_1 Pr[\overline{\ulcorner \forall v_1(0 = 0' \wedge v_1 = v_1)\urcorner}](v_1) \supset \forall v_1(0 = 0' \wedge v_1 = v_1))$. This is provably equivalent to $(Pr(\overline{\ulcorner 0 = 0' \urcorner}) \supset 0 = 0')$, which implies $\sim Pr(\overline{\ulcorner 0 = 0' \urcorner})$. ▲

Theorem 138 (Π_1-uniform reflection provable from consistency) *For a Π_1-formula $\forall v_2 R(v_1, v_2)$, i.e. $R(v_1, v_2)$ a Σ_0-formula, and v_1 the only free variable in $F(v_1)$, PA $\cup\{Con_{PA}\} \vdash (\forall v_1 Pr[\overline{\ulcorner \forall v_2 R(v_1, v_2) \urcorner}](v_1) \supset \forall v_1 \forall v_2 R(v_1, v_2))$.*

Proof. The following deduction shows the existence of a formal proof in PA.

(1)	(1) $\forall v_1 Pr[\overline{\ulcorner \forall v_2 R(v_1, v_2) \urcorner}](v_1)$	Assumption
(2)	(2) $\sim\forall v_2 R(v_1, v_2)$	Assumption
	(3) $(\sim\forall v_2 R(v_1, v_2) \equiv \exists v_2 \sim R(v_1, v_2))$	logic
(2)	(4) $\exists v_2 \sim R(v_1, v_2)$	(2)(3) prop logic
	(5) $(\exists v_2 \sim R(v_1, v_2) \supset Pr[\overline{\ulcorner \exists v_2 \sim R(v_1, v_2) \urcorner}](v_1))$	Corollary 131 $(*)$
(2)	(6) $Pr[\overline{\ulcorner \exists v_2 \sim R(v_1, v_2) \urcorner}](v_1)$	(4)(5) $\supset$-Elimination
	(7) $(Pr[\overline{\sim\forall v_2 R(v_1, v_2)}](v_1) \equiv Pr[\overline{\exists v_2 \sim R(v_1, v_2)}](v_1))$	(3) P_1^* P_2^*
(2)	(8) $Pr[\overline{\ulcorner \sim\forall v_2 R(v_1, v_2) \urcorner}](v_1)$	(6)(7) $\supset$-Elim
(1)	(9) $Pr[\overline{\ulcorner \forall v_2 R(v_1, v_2) \urcorner}](v_1)$	(1) $\forall$-elim
(1)(2)	(10) $Pr[\overline{\ulcorner (\forall v_2 R(v_1, v_2) \wedge \sim \forall v_2 R(v_1, v_2)) \urcorner}](v_1)$	(8)(9) Lemma 125
	(11) $((\forall v_2 R(v_1, v_2) \wedge \sim \forall v_2 R(v_1, v_2)) \supset 0 = 0')$	propositional logic
	(12) $(Pr[\overline{\ulcorner (\forall v_2 R(v_1, v_2) \wedge \sim \forall v_2 R(v_1, v_2)) \urcorner}](v_1) \supset Pr[\overline{\ulcorner 0 = 0' \urcorner}])$	(11) P_1^* P_2^*
	(13) $(\sim Pr[\overline{\ulcorner 0 = 0' \urcorner}] \supset \sim Pr[\overline{(\ulcorner (\forall v_2 R(v_1, v_2) \wedge \sim \forall v_2 R(v_1, v_2)) \urcorner}](v_1)))$	(12) contraposition
(14)	(14) $\sim Pr(\overline{\ulcorner 0 = 0' \urcorner})$	Assumption
(14)	(15) $\sim Pr[\overline{\ulcorner 0 = 0' \urcorner}]$	(14) Proposition 122
(14)	(16) $\sim Pr[\overline{(\ulcorner (\forall v_2 R(v_1, v_2) \wedge \sim \forall v_2 R(v_1, v_2))) \urcorner}](v_1)$	(15) (13) $\supset$-Elimination
(14)(1)(2)	(17) $(Pr[\overline{\ulcorner (\forall v_2 R(v_1, v_2) \wedge \sim \forall v_2 R(v_1, v_2)) \urcorner}](v_1) \wedge \sim Pr[\overline{\ulcorner (\forall v_2 R(v_1, v_2) \wedge \sim \forall v_2 R(v_1, v_2)) \urcorner}](v_1))$	(10) (16) $\wedge$-Introduction
(14)(1)	(18) $\sim\sim\forall v_2 R(v_1, v_2)$	(2) (17) $\sim$-Introduction
(14)(1)	(19) $\forall v_2 R(v_1, v_2)$	(18) propositional logic
(14)(1)	(20) $\forall v_1 \forall v_2 R(v_1, v_2)$	(19) $\forall$-Introduction $(**)$
(14)	(21) $(\forall v_1 Pr[\overline{\ulcorner \forall v_2 R(v_1, v_2) \urcorner}](v_1) \supset \forall v_1 \forall v_2 R(v_1, v_2))$	(1) (20) $\supset$-intro ▲

$(*)$ This is the one point in this proof at which the hypothesis that $\forall v_2 R(v_1, v_2)$ is Π_1 is needed.

$(**)$ v_1 not free in (13)(1).

Theorem 139 (Σ_1-uniform reflection provable from consistency) *For a Σ_1-formula $\exists v_2 R(v_1, v_2)$, i.e. $R(v_1, v_2)$ a Σ_0-formula, and v_1 the only free variable in $\exists v_2 R(v_1, v_2)$, PA $\cup\{Con_{PA}\} \vdash (\forall v_1 Pr[\overline{\ulcorner \exists v_2 R(v_1, v_2) \urcorner}](v_1) \supset \forall v_1 \exists v_2 R(v_1, v_2))$*

Proof. The following deduction shows the existence of a formal proof in PA.

(1)	(1) $\forall v_1 Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1)$	Assumption
(2)	(2) $\sim F(v_1)$	Assumption
	(3) $(\sim F(v_1) \equiv \exists v_2 \sim R(v_1, v_2))$	$F(v_1) = \forall v_2 R(v_1, v_2)$
	(4) $(\exists v_2 \sim R(v_1, v_2) \supset Pr[\overline{\ulcorner \exists v_2 \sim R(v_1, v_2)\urcorner}](v_1)$	Corollary 131
(2)	(5) $Pr[\overline{\ulcorner \exists v_2 \sim R(v_1, v_2)\urcorner}](v_1)$	(2)(3)(4) $\supset$-Elimination
	(6) $(Pr[\sim F(v_1)](v_1) \equiv Pr[\overline{\ulcorner \exists v_2 \sim R(v_1, v_2)\urcorner}](v_1))$	(3) P_1^* P_2^*
(2)	(7) $Pr[\sim F(v_1)](v_1)$	(5) (6)
(1)	(8) $Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1)$	(1) $\forall$-elim
(1)(2)	(9) $Pr[\overline{\ulcorner (F(v_1) \wedge \sim F(v_1))\urcorner}](v_1)$	(7)(8) Lemma 125
	(10) $((F(v_1) \wedge \sim F(v_1)) \supset 0 = 0')$	propositional logic
	(11) $Pr[\overline{\ulcorner (F(v_1) \wedge \sim F(v_1))\urcorner}](v_1) \supset Pr[\overline{\ulcorner 0 = 0'\urcorner}]$	(10) P_1^* P_2^*
	(12) $\sim Pr[\overline{\ulcorner 0 = 0'\urcorner}] \supset \sim Pr[\overline{(\ulcorner (F(v_1) \wedge \sim F(v_1))\urcorner}](v_1)$	(11) contraposition
(13)	(13) $\sim Pr(\overline{\ulcorner 0 = 0'\urcorner})$	Assumption
(13)	(14) $\sim Pr[\overline{\ulcorner 0 = 0'\urcorner}]$	(14) Proposition 122
(13)	(15) $\sim Pr[\overline{(\ulcorner (F(v_1) \wedge \sim F(v_1))\urcorner}](v_1)$	(14) (12) $\supset$-Elimination
(13)(1)(2)	(16) $((9) \wedge (15))$	(9) (15) $\wedge$-Introduction
(13)(1)	(17) $\sim\sim F(v_1)$	(2) (16) $\sim$-Introduction
(13)(1)	(18) $F(v_1)$	(17) propositional logic
(13)(1)	(19) $\forall v_1 F(v_1)$	(18) $\forall$-Introduction
		v_1 not free in (13)(1)
(13)	(20) $(\forall v_1 Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1) \supset \forall v_i F(v_1))$	(1) (19) $\supset$-intro ▲

12.5 PA is Π_1-conservative over $\mathbf{PA}_{\Pi_2} \cup \{Con_{PA}\}$

In the 1920s David Hilbert adumbrated a research programme which had at its heart the project of giving proofs of the consistency of formal systems of infinitary mathematics in finitary mathematics. The motivation for this programme was foundational and philosophical but Hilbert came to see that it also promised mathematical application in terms of establishing "conservative extension" results.

Definition 79 (S_2 an extension of S_1) *For theories S_1 and S_2 formulated in the same language, or such that the language of S_2 is an extension of the language of S_1, S_2 is an extension of S_1 if for each formula X in the language of S_1, if $S_1 \vdash X$, then $S_2 \vdash X$.*

Remark. The notion of extension of one theory by another can be generalized to the situation where the language of the first theory is interpreted in the language of the second theory, rather than being the same or part of the language of the second theory, but we have no need here for this more general notion.

Definition 80 (**S_2 a conservative extension of S_1**) *An extension S_2 of S_1 is conservative over S_1 if whenever $S_2 \vdash X$, already $S_1 \vdash X$.*

We also define conservativeness of one system over another with respect to a restricted class of formulas, e.g. Π_1 or Σ_1, or some other class of formulas in the arithmetical hierarchy.

Definition 81 (**S_2 conservative over S_1 with respect to formulas in Γ**) *An extension S_2 of S_1 is conservative over S_1 with respect to a class of formulas Γ if for each formula X in Γ, if $S_2 \vdash X$, then $S_1 \vdash X$.*

A fundamental insight of Hilbert's that lies at the heart of his programme of proof theory is that if finitary mathematics can prove the consistency of infinitary mathematics, then infinitary mathematics is a conservative extension of finitary mathematics with respect to finitary mathematics. Hilbert sketches an argument for this claim in a lecture in 1927, "Die Grundlagen der Mathematik", published in 1928, English translation as "The Foundations of Mathematics" in the van Heijenoort Source Book [9], p. 474.

Gödel's Second Incompleteness Theorem shows that, insofar as infinitary mathematics is an extension of finitary mathematics, the consistency of infinitary mathematics cannot be proved within finitary mathematics. Nonetheless Hilbert's argument adumbrates a correct mathematical theorem the main content of which is the proof of Theorem 138 that consistency implies the uniform reflection principle for Π_1-sentences. Hilbert formulates his argument in terms of a particular Π_1-sentence, Fermat's last theorem: "Let us suppose, for example, that we had found, for Fermat's great theorem, a proof in which the [infinitary] logical function ϵ was used. We could then make a finitary proof out of it in the following way."

Leaving aside the question in what minimal system can the consistency of PA be proved (which is beyond the scope of this course—the answer is, very roughly, constructive principles of abstract mathematics, rather than finitary principles of concrete mathematics), a precise working out of the argument Hilbert sketched requires that the proof of Theorem 138 be carried out in finitary mathematics. Hilbert never formulated clearly what he meant by finitary mathematics, i.e. he never gave a formal system of finitary mathematics, and I won't enter here the debate over what formal system should be taken to capture the intended notion of finitary arithmetic. Rather, I will address the question, in how weak a subsystem of PA can the argument for Theorem 138 be carried out?

The key point is that such a system must be strong enough to prove provable Σ_0-completeness of PA. Hilbert did not explicitly formulate provable Σ_0-completeness, but it is implicit in his argument, and implicitly he takes it to be a fact of finitary mathematics. "Let us assume that numerals p, a, b, c ($p > 2$) satisfying Fermat's equation $a^p + b^p = c^p$ are given; then we could also obtain this equation as a provable formula by giving the form of a proof to the procedure by which we ascertain that the numbers $a^p + b^p$ and c^p coincide."

Proving provable Σ_0-completeness requires mathematical induction, so we may take the question to be, how much induction, measured by complexity in the arithmetical hierarchy of the induction formula, is needed for this proof?

Definition 82 (subsystems of PA with restricted induction) *For Γ a class of formulas in $\mathcal{L}_A$ (e.g. Σ_1 or Π_1), PA_Γ is the subsystem of PA determined by restricting the instances of the scheme of mathematical induction N_{12} to formulas in Γ.*

The minimum is Σ_1-induction, i.e. axioms N_{12} for Σ_1-formulas, as in the step of the proof in Lecture 11 in which we proved that $((\forall v_1 \leq v_2)G \supset \forall v_3(S(\overline{\ulcorner(\forall v_1 \leq v_2)G\urcorner}, v_2, 0'', v_3) \supset P(v_3))$ by induction on the free variable v_2. However, we also used Π_2-induction in our proof in PA showing that e.g. $(v_1 + v_2 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3\urcorner}](v_1, v_2, v_3))$, since we proved this by induction on the formula $\forall v_3(v_1 + v_2 = v_3 \supset Pr[\overline{\ulcorner v_1 + v_2 = v_3\urcorner}](v_1, v_2, v_3))$, which is Π_2 since $Pr[\overline{\ulcorner F(v_1)\urcorner}](v_1)$ is Σ_1. It might be that there is some clever way to reconstruct that proof so that the universal quantification of the induction formula is not needed. In any case, from what has been established, we have the following theorem corresponding to Hilbert's argument claiming that infinitary mathematics is conservative over finitary mathematics with respect to Π_1-theorems.

Theorem 140 *For X any Π_1-sentence in the language of PA, if $PA \vdash X$, then $PA_{\Pi_2} \cup \{Con_{PA}\} \vdash X$.*

Proof. By Theorem 138 and analysis of the proof of Theorem 130. ▲

Lecture 13

Provability logic: the system GL

Monday 25 February 2019

13.1 The system GL for the logic of provability

A proof predicate $Pr(v_1)$ for a system S can be thought of as an operator on sentences in the language of S, i.e. it generates a sentence from a sentence. To signify this viewpoint, we write $\Box A$ for $Pr(\overline{\ulcorner A \urcorner})$. In this notation, the arithmetization of Löb's theorem (exercise) is expressed as $(\Box(\Box A \supset A) \supset \Box A)$. As we shall see, this formula axiomatizes the logic of provability.

When provability logic first began to be developed, in the 1970s, there existed already, for more than fifty years, systems of logic for a sentence operator $\Box A$ with the intended meaning, "A is necessarily true". Such systems are called modal logic since necessity concerns not only the truth of sentences but also the kind, or mode of their truth. In a brief note published in 1933, Gödel obtained results about intuitionistic logic by interpreting the $\Box$ operator of modal logic as provability, which was the beginning of provability logic. Modal logic provided a framework for setting up systems for provability logic, and also a semantics of possible worlds with an accessibility relation between worlds by which to study properties of such systems, which has been exploited in the study of provability logic, but we will establish the results that concern us here purely syntactically, i.e. not using these semantic techniques. It is important to realize that provability logic is not an extension of the logic of necessity, for which $(\Box X \supset X)$ is valid, in contrast to which Löb's theorem shows that if $S \nvdash X$, then $S \nvdash (Pr(\overline{\ulcorner X \urcorner}) \supset X)$.

The system of provability logic was named GL by George Boolos, after Gödel and Löb. It consists of propositional logic plus the provability operator.

13.1.1 The language of GL

The primitive symbols of GL:

A sentence $\perp$ (This symbol stands for a generic false sentence, and when we consider interpretations of GL in PA, we will interpret $\perp$ as some particular sentence refutable in PA, e.g. $0 = 0'$.)

Infinitely many sentence letters generated from the symbol 'p' and iteration of the subscript symbol '$_{\prime}$', i.e. $p_{\prime}$, $p_{\prime\prime}$, $p_{\prime\prime\prime}$, $\ldots$, which we abbreviate as $p_1, p_2, p_3, \ldots$.

The sentential connective $\supset$.

The sentential operator $\Box$.

Definition 83 (sentences of GL) *By recursion:*

base: $\perp$ *and all* p_i *are sentences.*

recursion: If X *and* Y *are sentences,* $(X \supset Y)$ *is a sentence.*

If X *is a sentence,* $\Box X$ *is a sentence.*

We shall write sentences in the language of GL using the following abbreviations which, on the intended meaning for $\supset$ and $\perp$, express negation, conjunction, disjunction, and equivalence:

Definition 84 $\sim X =_{df} (X \supset \perp)$,

$(X \wedge Y) =_{df} ((X \supset (Y \supset \perp)) \supset \perp)$,

$(X \vee Y) =_{df} ((X \supset \perp) \supset Y)$

$(X \equiv Y) =_{df} (((X \supset Y) \supset ((Y \supset X) \supset \perp)) \supset \perp)$

13.1.2 The axioms and inference rules of GL

Definition 85 (axioms of GL) *A1. (Tautologies) Every sentence in the language of GL that is a truth functional tautology when* $\perp$ *is assigned the truth value F (falsity) and* $\supset$ *is interpreted as the truth function 'if ... then ... ' is an axiom.*

A2. (Distribution) For X *and* Y *any sentences in the language of GL,* $(\Box(X \supset Y) \supset (\Box X \supset \Box Y))$ *is an axiom of GL. (Corresponds to* P_2 *for a provability predicate.)*

A3. (Arithmetized Löb's Theorem) For each sentence X in the language of GL, $(\Box(\Box X \supset X) \supset \Box X)$ is an axiom of GL.

Definition 86 (rules of inference of GL) *R1. From sentences X and $(X \supset Y)$, infer Y. (Modus ponens)*

R2. From sentence X infer $\Box X$. (Corresponds to property P_1 of provability predicates. In modal logic, this rule is known as Necessitation.)

There is no axiom schema corresponding to property P_3 for provability predicates since, as we shall see by Theorem 152, $(\Box X \supset \Box\Box X)$ is derivable from the axioms and rules of inference specified for GL.

13.2 Soundness and completeness of GL

The axioms and inference rules of GL arise by abstraction from the arithmetized proof predicate for PA. Conversely, the axioms and inference rules, and hence all theorems of GL, translate into theorems of PA. This result means that GL is sound with respect to interpretation in PA.

Definition 87 (interpretations of GL in PA) *Let $Pr(v_1)$ be a provability predicate in the language of PA and let i be a mapping from $\{\bot, p_,, \ldots, p_{,\ldots,}, \ldots\}$ to sentences in the language of PA such that for $X = i(\bot)$, $PA \vdash \sim X$. An interpretation $\mathfrak{I}_i$ of GL in PA with respect to i and $Pr(v_1)$ is specified by the following inductive definition:*

Base:

(i) $\mathfrak{I}_i(\bot) = i(\bot)$;

(ii) $\mathfrak{I}_i(p_n) = i(p_n)$, *for n any string of subscript symbols.*

Induction:

(iii) $\mathfrak{I}_i((X \supset Y)) = (\mathfrak{I}_i(X) \supset \mathfrak{I}_i(Y))$;

(iv) $\mathfrak{I}_i(\Box X) = Pr(\overline{\ulcorner \mathfrak{I}_i(X) \urcorner})$.

Note that in clause (iii) of this definition the occurrence of $\supset$ on the left is as a symbol in the language of GL and the occurrence of $\supset$ on the right is as a symbol in the language of PA.

Theorem 141 (soundness of GL with respect to interpretation in PA) *For every assignment i of sentences in the language of PA to the atomic sentences in the language of GL, if $GL \vdash X$, then $PA \vdash \mathfrak{I}_i(X)$, for $\mathfrak{I}_i$ as in Definition 87.*

Proof. By induction on the length of proofs in GL.

Basis step, for X an axiom of GL:

(i) If X is an $A1$ axiom, i.e. a truth functional tautology when $\bot$ is assigned the truth value F, then if $i(\bot) = Z$, $({\sim}Z \supset \mathfrak{I}_i(X))$ is a truth functional tautology in the language of PA, as follows: For Z assigned the value T, $({\sim}Z \supset \mathfrak{I}_i(X))$ takes the value T, and for Z assigned the value F, since X is a tautology when $\bot$ is assigned the value F, $\mathfrak{I}_i(X))$ takes the value T when Z takes the value F, so $({\sim}Z \supset \mathfrak{I}_i(X))$ takes the value T, so $({\sim}Z \supset \mathfrak{I}_i(X))$ is a tautology, and hence $PA \vdash ({\sim}Z \supset \mathfrak{I}_i(X))$. Then since $PA \vdash {\sim}Z$, by modus ponens, $PA \vdash \mathfrak{I}_i(X))$.

(ii) If X is an $A2$ distribution axioms, then $\mathfrak{I}_i(X)$ is an instance of P_2 and hence provable in PA.

(iii) If X is an $A3$ arithmetized Löb's Theorem axiom, then $\mathfrak{I}_i(X)$ is an instance of arithmetized Löb's Theorem, which is provable in PA (exercise).

Induction step:

R1: From $GL \vdash (X \supset Y)$ and $GL \vdash X$, $GL \vdash Y$. Then by Induction Hypothesis, $PA \vdash (\mathfrak{I}_i(X) \supset \mathfrak{I}_i(Y))$ and $PA \vdash \mathfrak{I}_i(X)$. Then by Modus ponens in PA, $PA \vdash \mathfrak{I}_i(Y)$.

R2: From $GL \vdash X$, $GL \vdash \Box X$. By Induction Hypothesis, $PA \vdash \mathfrak{I}_i(X)$. Then by P_1 for PA, $PA \vdash Pr(\overline{\ulcorner \mathfrak{I}_i(X) \urcorner})$. ▲

There is also a completeness theorem, due to Robert Solovay, for GL with respect to interpretation in PA.

Theorem 142 (Solovay completeness theorem for GL) *For X any sentence in the language of GL, if for every interpretation $\mathfrak{I}_i(X)$ (as in Definition 87), $PA \vdash \mathfrak{I}_i(X)$, then $GL \vdash X$; or equivalently, for X any sentence in the language of GL, if $GL \nvdash X$, then there is an interpretation $\mathfrak{I}_i(X)$ such that $PA \nvdash \mathfrak{I}_i(X)$.*

Proof. Beyond the scope of this course.

13.3 Some derivations in GL

Unarithmetized Löb's Theorem holds for GL, i.e.

Lemma 143 *If $GL \vdash (\Box X \supset X)$, then $GL \vdash X$.*

Proof. Suppose $GL \vdash (\Box X \supset X)$. Then by $R2$, $GL \vdash \Box(\Box X \supset X)$. By $A3$, $GL \vdash (\Box(\Box X \supset X) \supset \Box X)$, so by $R1$, $GL \vdash \Box X$. Then by the initial supposition and $R1$, $GL \vdash X$. ▲

Lemma 144 *If $GL \vdash (A \supset B)$, then $GL \vdash (\Box A \supset \Box B)$.*

Proof. This is a notational variant of P_4 and the proof of P_4 in Lemma 108 proves this result. ▲

Next we note that GL is closed under truth functional consequence, i.e.

Theorem 145 *If Y is a truth functional consequence of finitely many formulas which are each provable in GL, then $GL \vdash Y$.*

Proof. If Y follows truth functionally from the finitely many formulas $X_1, \ldots, X_r$, then $(X_1 \supset (X_2 \supset (\ldots \supset (X_r \supset Y) \ldots)))$ is a tautology and hence an axiom of GL, and GL $\vdash Y$ by r-many applications of Modus ponens starting with this axiom. ▲

Remarks. In proofs that depend on this theorem I will say "by propositional logic in GL" or just "by propositional logic", rather than citing Theorem 145. By the compactness theorem for truth functional logic, the above result holds for Y a truth functional consequence of any set of provable formulas, not just finite sets of formulas, but we have no need for this generalization.

Theorem 146 $GL \vdash (\Box(X \wedge Y) \equiv (\Box X \wedge \Box Y))$

Proof. (i) Both $((X \wedge Y) \supset X)$ and $((X \wedge Y) \supset Y)$ are tautologies and hence axioms of GL. By Lemma 144, GL $\vdash (\Box(X \wedge Y) \supset \Box X)$ and GL $\vdash (\Box(X \wedge Y) \supset \Box Y)$. Then by propositional logic in GL, GL $\vdash (\Box(X \wedge Y) \supset (\Box X \wedge \Box Y))$

(ii) Since $(X \supset (Y \supset (X \wedge Y)))$ is a tautology, $GL \vdash (X \supset (Y \supset (X \wedge Y)))$. Hence by Lemma 144, $GL \vdash (\Box X \supset \Box(Y \supset (X \wedge Y)))$. By $A2$, $GL \vdash (\Box(Y \supset (X \wedge Y) \supset (\Box Y \supset \Box(X \wedge Y))))$. Then by propositional logic in GL, $GL \vdash (\Box X \supset (\Box Y \supset \Box(X \wedge Y)))$. The formulas $((\Box X \wedge \Box Y) \supset \Box X)$ and $((\Box X \wedge \Box Y) \supset \Box Y)$ are tautologies and hence axioms of GL, so by propositional logic in GL, $GL \vdash ((\Box X \wedge \Box Y) \supset \Box(X \wedge Y))$. ▲

Proposition 147 $GL \vdash (\Box(X \equiv Y) \supset (\Box X \equiv \Box Y))$

Proof. By Definition 84, $(X \equiv Y) =_{df} ((X \supset Y) \wedge (Y \supset X))$, so by Theorem 146, $GL \vdash (\Box(X \equiv Y) \equiv (\Box(X \supset Y) \wedge \Box(Y \supset X)))$. By A_2 and Theorem 145, $GL \vdash ((\Box(X \supset Y) \wedge \Box(Y \supset X)) \supset ((\Box X \supset \Box Y) \wedge (\Box Y \supset \Box X)))$. Then by propositional logic in GL, $GL \vdash (\Box(X \equiv Y) \supset ((\Box X \supset \Box Y) \wedge (\Box Y \supset \Box X)))$, which by Definition 84 is $GL \vdash (\Box(X \equiv Y) \supset (\Box X \equiv \Box Y))$. ▲

The converse of Proposition 147 does not hold:

Proposition 148 *If PA is 1-consistent,* $GL \nvdash ((\Box X \equiv \Box Y) \supset \Box(X \equiv Y))$.

Proof. $PA \nvdash ((Pr(\overline{\ulcorner G \urcorner}) \equiv Pr(\overline{\ulcorner 0 = 0' \urcorner})) \supset (Pr(\overline{\ulcorner G \equiv 0 = 0' \urcorner})))$, since $PA \vdash (Pr(\overline{\ulcorner G \urcorner}) \equiv Pr(\overline{\ulcorner 0 = 0' \urcorner}))$, by propositional logic from Corollary 113, and $PA \nvdash (Pr(\overline{\ulcorner G \equiv 0 = 0' \urcorner}))$, if PA is 1-consistent, which we see as follows. Suppose $PA \vdash Pr(\overline{\ulcorner G \equiv 0 = 0' \urcorner}))$. Then by P_2, $PA \vdash Pr(\overline{\ulcorner \sim G \urcorner})$. Since we are given that PA is 1-consistent, PA is Σ_1-sound, and so since $Pr(v_1)$ is Σ_1 and expresses $\{n : PA \vdash E_n\}$, $PA \vdash \sim G$. But if PA is 1-consistent, $PA \nvdash \sim G$. Hence by Theorem 141, if PA is 1-consistent, $GL \nvdash ((\Box X \equiv \Box Y) \supset \Box(X \equiv Y))$. ▲

13.4 Closure of GL under substitution by provably equivalent formulas

The result of substituting a sentence A for a sentence letter p in formula F, $F_p(A)$, is defined by the following recursion.

Definition 88 ($F_p(A)$) *1. If* $F = p$, *then* $F_p(A) = A$.

2. If $F = q$ *where* $q \neq p$, *then* $F_p(A) = q$.

3. If $F = \perp$, *then* $F_p(A) = \perp$.

4. If $F = (G \supset H)$, *then* $F_p(A) = (G_p(A) \supset H_p(A))$.

5. If $F = \Box G$, *then* $F_p(A) = \Box(G_p(A))$.

Proposition 149 (closure of GL under substitution) *For X any formula in the language of GL, and $F(p)$ any formula in the language of GL in which the sentence letter p occurs, if $GL \vdash F(p)$, then $GL \vdash F_p(X)$.*

Proof. By induction on the length of a proof of $F(p)$ in GL.

Base case: The proof is of length 1, i.e. $F(p)$ is an axiom, in which case it is of the form $A1$ (tautology), $A2$ (distribution), or $A3$ (arithmetized Löb's Theorem). Then $F_p(X)$ is an axiom, of the same form as $F(p)$ is, so $GL \vdash F_p(X)$.

Induction steps: (i) The proof of $F(p)$ ends with an application of $R1$, i.e. $GL \vdash Y$ and $GL \vdash (Y \supset F(p))$. By Induction Hypothesis, $GL \vdash Y_p(X)$ and $GL \vdash (Y \supset F(p))_p(X)$. Then by clause 4 of Definition 88, $GL \vdash (Y_p(X) \supset F_p(X))$, so by $R1$, $GL \vdash F_p(X)$.

(ii) The proof of $F(p)$ ends with an application of $R2$, i.e. $F(p)$ is of the form $\Box G(p)$ and $GL \vdash G(p)$. By Induction Hypothesis, $GL \vdash G_p(X)$. By $R2$, $GL \vdash \Box(G_p(X))$. Then by clause 5 of Definition 88, $GL \vdash F_p(X)$. ▲

Theorem 150 (provable equivalence of substitution of provable equivalents) *For all formulas A, B, and F and any sentence letter p, if $GL \vdash (A \equiv B)$, then $GL \vdash (F_p(A) \equiv F_p(B))$.*

Proof. We argue by induction over the inductive definition of formulas F.

If $F = p$, what is to be proved is that if GL $\vdash (A \equiv B)$, then GL $\vdash (A \equiv B)$, which holds trivially.

If $F = q$ for $p \neq q$, what is to be proved is that if GL $\vdash (A \equiv B)$, then GL $\vdash (q \equiv q)$. The consequent is provable outright, so the implication holds.

If $F =\perp$, what is to be proved is GL $\vdash (A \equiv B)$, then GL $\vdash (\perp\equiv\perp)$, for which again the consequent is provable outright.

If $F = (G \supset H)$, then we have as induction hypotheses, if GL $\vdash (A \equiv B)$, then $GL \vdash (G_p(A) \equiv G_p(B))$, and $GL \vdash (H_p(A) \equiv H_p(B))$. By Definition 88(4), what is to be proved is if GL $\vdash (A \equiv B)$, then
$GL \vdash ((G_p(A) \supset H_p(A)) \equiv (G_p(B) \supset H_p(B)))$, which follows by propositional logic from the induction hypotheses.

If $F = \Box G$, then we have by induction hypothesis that if GL $\vdash (A \equiv B)$, then GL $\vdash (G_p(A) \equiv G_p(B))$. Then by Lemma 144 and propositional logic in GL, GL $\vdash (\Box(G_p(A)) \equiv \Box(G_p(B)))$, so by Definition 88(5), GL $\vdash ((\Box G)_p(A) \equiv (\Box G)_p(B))$, which is to say, GL $\vdash (F_p(A) \equiv F_p(B))$. ▲

Theorem 150 immediately establishes that GL is closed under substitution of provable equivalents.

Corollary 151 *If $GL \vdash F_p(A)$ and $GL \vdash (A \equiv B)$, then $GL \vdash F_p(B)$.*

Proof. From $GL \vdash (A \equiv B)$ and Theorem 150, we have by $\wedge$-elimination, $GL \vdash (F_p(A) \supset F_p(B))$, so by Modus ponens from $GL \vdash F_p(A)$, we have $GL \vdash F_p(B)$. ▲

From Corollary 151 and results from the previous section, we are able to establish that GL proves P_3.

Theorem 152 $GL \vdash (\Box X \supset \Box\Box X)$

Proof. The formula $(X \supset ((\Box\Box X \wedge \Box X) \supset (\Box X \wedge X))$ is a tautology and hence $GL \vdash (X \supset ((\Box\Box X \wedge \Box X) \supset (\Box X \wedge X))$. Since, by Theorem 146, GL $\vdash ((\Box\Box X \wedge \Box X) \equiv \Box(\Box X \wedge X)))$, by Corollary 151, taking F_p as
$(X \supset (p \supset (\Box X \wedge X))$, GL $\vdash (X \supset (\Box(\Box X \wedge X) \supset (\Box X \wedge X))$. Then by Lemma 144 ($P_4$), GL $\vdash (\Box X \supset \Box(\Box(\Box X \wedge X) \supset (\Box X \wedge X))$. The formula $(\Box(\Box(\Box X \wedge X) \supset (\Box X \wedge X)) \supset \Box(\Box X \wedge X))$ is an $A3$ axiom of GL. Then by propositional logic in GL, $GL \vdash (\Box X \supset \Box(\Box X \wedge X))$. Then by Theorems 146 and propositional logic in

GL, GL $\vdash (\Box X \supset (\Box\Box X \wedge \Box X)$. Since $((\Box\Box X \wedge \Box X) \supset \Box\Box X)$ is a tautology, by propositional logic in GL, $GL \vdash (\Box X \supset \Box\Box X)$. ▲

The proof of Theorem 150 generalizes to establish provable equivalence of substitution on more than one sentence letter.

Theorem 153 (substitution on more than one sentence letter) *For F any formula with sentence letters $p_{k_1}, \ldots, p_{k_n}$, and for pairs of formulas A_i, B_i, $i = 1, \ldots, n$, such that $GL \vdash (A_i \equiv B_i)$, $GL \vdash (F(A_1, \ldots, A_n) \equiv F(B_1, \ldots, B_n))$, where $F(A_1, \ldots, A_n)$ is the result of substituting A_i for p_{k_i} in F, and $F(B_1, \ldots, B_n)$ is the result of substituting B_i for p_{k_i} in F.*

Proof. Exactly the same proof structure as for Theorem 150, with just reformulation of the induction hypothesis so that it's for multiple substitutions, establishes this result. ▲

13.5 Closure of GL under substitution of provably equivalent formulas is provable in GL

We now show that the closure of GL under substitution of provably equivalent formulas, Theorem 150, can be formalized *in* GL.

Theorem 154 (arithmetized substitution theorem) *For all formulas A, B, and F and sentence letter p, $GL \vdash (\Box(A \equiv B) \supset \Box(F_p(A) \equiv F_p(B)))$.*

Proof. The proof is by induction over the recursion that generates the formula F.

If $F = p$, what is to be proved is $GL \vdash (\Box(A \equiv B) \supset \Box(A \equiv B))$, which is a tautology and hence provable in GL.

If $F = q$ for $p \neq q$, what is to be proved is $GL \vdash (\Box(A \equiv B) \supset \Box(q \equiv q))$. Since $(q \equiv q)$ is a tautology, $GL \vdash (q \equiv q)$, so by R_2, $GL \vdash \Box(q \equiv q)$. The result follows by propositional logic in GL.

If $F = \perp$, what is to be proved is $GL \vdash (\Box(A \equiv B) \supset \Box(\perp \equiv \perp))$. The argument is as for the preceding case.

If $F = (G \supset H)$, we have as induction hypotheses, $GL \vdash (\Box(A \equiv B) \supset \Box(G_p(A) \equiv G_p(B)))$, and $GL \vdash (\Box(A \equiv B) \supset \Box(H_p(A) \equiv H_p(B)))$. By propositional logic in GL, $GL \vdash (\Box(A \equiv B) \supset (\Box(G_p(A) \equiv G_p(B)) \wedge \Box(H_p(A) \equiv H_p(B))))$. Then by Theorem 146 and propositional logic in GL, $GL \vdash (\Box(A \equiv B) \supset \Box((G_p(A) \equiv G_p(B)) \wedge (H_p(A) \equiv H_p(B))))$. The following formula is a tautology and so provable in GL (as an axiom): $(((G_p(A) \equiv G_p(B)) \wedge (H_p(A) \equiv H_p(B))) \supset ((G_p(A) \supset H_p(A)) \equiv$

$(G_p(B) \supset H_p(B))))$. Then by Lemma 144 ($P_4$), $GL \vdash (\Box((G_p(A) \equiv G_p(B)) \wedge (H_p(A) \equiv H_p(B))) \supset \Box((G_p(A) \supset H_p(A)) \equiv (G_p(B) \supset H_p(B))))$. From this result and the two steps earlier we have, by propositional logic in GL, $GL \vdash (\Box(A \equiv B) \supset \Box((G_p(A) \supset H_p(A)) \equiv (G_p(B) \supset H_p(B))))$, which by Definition 88(4), is $GL \vdash (\Box(A \equiv B) \supset \Box(F_p(A) \equiv F_p(B)))$.

If $F = \Box G$, we have as induction hypothesis that $GL \vdash (\Box(A \equiv B) \supset \Box(G_p(A) \equiv G_p(B)))$. Then by Proposition 147 and propositional logic in GL, $GL \vdash (\Box(A \equiv B) \supset (\Box(G_p(A)) \equiv \Box(G_p(B))))$. Then by Definition 88 (5.), $GL \vdash (\Box(A \equiv B) \supset (F_p(A) \equiv F_p(B)))$. Then by Lemma 144 ($P_4$), $GL \vdash (\Box\,\Box\,(A \equiv B) \supset \Box(F_p(A) \equiv F_p(B)))$. By Theorem 152, $GL \vdash (\Box(A \equiv B) \supset \Box\,\Box\,(A \equiv B))$, so by propositional logic in GL, $GL \vdash (\Box(A \equiv B) \supset \Box(F_p(A) \equiv F_p(B)))$. ▲

13.6 Strengthened proof that the closure of GL under substitution of provably equivalent formulas is provable in GL

This strengthened proof makes use of the following technical definition.

Definition 89 $\boxdot X =_{df} (\Box X \wedge X)$

Lemma 155 $GL \vdash (\boxdot X \supset X)$

Proof. $\vdash (\boxdot X \supset X)$ is a tautology. ▲

Remark. Lemma 155 is a triviality but draws attention to a key property of $\boxdot$ that holds for all formulas and which, by Löb's Theorem for GL (Lemma 143), holds for $\Box$ only for formulas provable in GL. The constraint of Löb's Theorem makes it very difficult to derive an unboxed conclusion from a boxed premiss, i.e. of the form $\Box X$. The situation is much more flexible if we are able to strengthen the premiss to $\boxdot X$.

Lemma 156 *For each formula X in the language of GL, the formulas $\Box X$, $\Box \boxdot X$, and $\boxdot \Box X$ are provably equivalent in GL.*

Proof. The provable equivalence of $\Box \boxdot X$ and $\boxdot \Box X$ is, by Definition 89, an instance of Theorem 146, namely $GL \vdash (\Box(\Box X \wedge X) \equiv (\Box\,\Box\,X \wedge \Box X))$.

By Theorem 152 (P_3) and propositional logic in GL, $GL \vdash (\Box X \equiv (\Box\,\Box\,X \wedge \Box X))$, which by Definition 89 is $GL \vdash (\Box X \equiv \boxdot \Box X)$. ▲

Corollary 157 $GL \vdash (\boxdot X \supset \Box \boxdot X)$

Proof. By $\wedge$-Elimination, $(\boxdot X \supset \Box X)$. By Lemma 156, $(\Box X \equiv \Box \boxdot X)$, so by $\wedge$-elimination, $\Box X \supset \Box \boxdot X)$. ▲

Remark. The converse implication is not provable in GL, since
$GL \nvdash (\Box X \supset X)$ unless $GL \vdash X$ (Lemma 143).

Lemma 158 *If* $GL \vdash (\boxdot X \supset Y)$*, then* $GL \vdash (\Box X \supset \Box Y)$.

Proof. Assume GL $\vdash (\boxdot X \supset Y)$. Then by Lemma 144, $GL \vdash (\Box \boxdot X \supset \Box Y)$. Then by Lemma 156 and propositional logic in GL, GL $\vdash (\Box X \supset \Box Y)$. ▲

Lemma 159 $GL \vdash (\boxdot X \supset Y)$ *if and only if* $GL \vdash (\boxdot X \supset \boxdot Y)$.

Proof. (i) From Definition 89 by propositional logic in GL, $GL \vdash (\boxdot X \supset \Box X)$. Assume GL $\vdash (\boxdot X \supset Y)$. Then by Lemma 158, $GL \vdash (\Box X \supset \Box Y)$. Then by Theorem 145, $GL \vdash (\boxdot X \supset \Box Y)$. From the assumption and this last result by propositional logic in GL, GL $\vdash (\boxdot X \supset (\Box Y \wedge Y))$, i.e. GL $\vdash (\boxdot X \supset \boxdot Y)$.

(ii) Assume GL $\vdash (\boxdot X \supset \boxdot Y)$. By Lemma 155, GL $\vdash (\boxdot Y \supset Y)$ and hence by propositional logic in GL, GL $\vdash (\boxdot X \supset Y)$. ▲

We now prove a version of Theorem 154 with strengthened consequent from a strengthened antecedent. The consequent of Theorem 154, $\Box(F_p(A) \equiv F_p(B)))$ is strengthened to $(F_p(A) \equiv F_p(B)))$, i.e. by dropping the $\Box$. That this is a strengthening is down to Löb's Theorem, which tells us that the only way to use an antecedent $\Box X$ in a chain of reasoning is if X itself is provable. The antecedent of Theorem 154, $\Box(A \equiv B)$ is strenghtened by adding the conjunct $(A \equiv B)$, i.e. to make it $\boxdot(A \equiv B)$.

Theorem 160 (strengthened arithmetized substitution theorem) *For all formulas A, B, and F and propositional variable p,*
$GL \vdash (\boxdot(A \equiv B) \supset (F_p(A) \equiv F_p(B)))$.

Proof. The proof is by induction over the recursion that generates the formula F.

If $F = p$, what is to be proved is $GL \vdash (\boxdot(A \equiv B) \supset (A \equiv B))$, which holds by Definition 89 and propositional logic in GL.

If $F = q$ for $p \neq q$, what is to be proved is $GL \vdash (\boxdot(A \equiv B) \supset (q \equiv q))$. For any formula H, $(H \supset (q \supset q))$ is a tautology and hence an axiom of GL, so in particular $GL \vdash (\boxdot(A \equiv B) \supset (q \equiv q))$.

If $F = \bot$, what is to be proved is $GL \vdash (\boxdot(A \equiv B) \supset (\bot \equiv \bot))$. The argument is as for the preceding case.

If $F = (G \supset H)$, we have as induction hypotheses, $GL \vdash (\boxdot(A \equiv B) \supset (G_p(A) \equiv G_p(B)))$, and $GL \vdash (\boxdot(A \equiv B) \supset (H_p(A) \equiv H_p(B)))$. By Definition 88(4), what is to be proved is $GL \vdash (\boxdot(A \equiv B) \supset ((G_p(A) \supset H_p(A)) \equiv (G_p(B) \supset H_p(B))))$, which follows from the induction hypotheses by propositional logic in GL.

If $F = \Box G$, we have as induction hypothesis that $GL \vdash (\boxdot(A \equiv B) \supset (G_p(A) \equiv G_p(B)))$. Then by Lemma 144, $GL \vdash (\Box\boxdot(A \equiv B) \supset \Box(G_p(A) \equiv G_p(B)))$. By Theorem 146, Axioms $A2$, and propositional logic in GL, $GL \vdash (\Box(G_p(A) \equiv G_p(B)) \supset (\Box(G_p(A)) \equiv \Box(G_p(B))))$. From these two last results, by propositional logic in GL and Defintion 88(5), $GL \vdash (\Box \boxdot (A \equiv B) \supset ((\Box G_p)(A) \equiv (\Box G_p)(B)))$. Then by Corollary 157 and propositional logic in GL, $GL \vdash (\boxdot(A \equiv B) \supset ((\Box G_p)(A) \equiv (\Box G_p)(B)))$. ▲

Corollary 161 (variant proof of Theorem 154) *For all formulas A, B, and F and propositional variable p, $GL \vdash (\Box(A \equiv B) \supset \Box(F_p(A) \equiv F_p(B)))$.*

Proof. By Theorem 160 and Lemma 158.

Theorem 162 (Theorem 160 generalized to multiple substitutions) *Let $F(p_{i_1}, \ldots, p_{i_m})$ be a formula with sentence letters $p_{i_1}, \ldots, p_{i_m}$. Then*
$GL \vdash (\boxdot(A_1 \equiv B_1) \wedge \ldots \wedge \boxdot(A_m \equiv B_m)) \supset (F(A_1, \ldots, A_m) \equiv F(B_1, \ldots, B_m)))$.

Proof. Exercise. ▲

Lecture 14

The fixed-point theorem for GL

Wednesday 27 February 2019

14.1 The notion of a sentence letter modalized in a sentence, and arithmetized substitution for modalized sentences

Definition 90 (Y is a subsentence of X) *Y is a subsentence of X is defined recursively by*

Base case: X is a subsentence of X

Recursion clauses: If the sentence $(Z \supset W)$ is a subsentence of X, then Z is a subsentence of X and W is a subsentence of X.

If the sentence $\Box Z$ is a subsentence of X, then Z is a subsentence of X.

If sentence Y is a subsentence of sentence X, we say that Y occurs in X.

Definition 91 (sentence letter p occurs modalized in sentence X) *A sentence letter p occurs modalized in a sentence X iff p is a subsentence of X and wherever p occurs in X, it is a subsentence of a subsentence of X of the form $\Box Y$.*

Examples of sentences in which the sentence letter p is modalized: (1) $\Box p$, (2) $\sim \Box p$, (3) $\Box \sim p$, (4) $\sim \Box \sim p$, (5) $(\Box p \supset q)$ (6) $\Box(\Box p \supset p)$, (7) $(\Box(\Box p \supset p) \supset \Box p)$, (8) $\Box(p \equiv (\Box p \supset q))$, (9) $(\Box(\Box p \supset q) \wedge \sim \Box p)$.

Examples of sentences in which the occurrence of p is not modalized: (10) p, (11) $(p \supset \perp)$, (12) $(\Box p \supset p)$, (13) $(p \equiv (\Box p \supset q)$.

Definition 92 (decomposition with respect to a modalized sentence letter) *For X a sentence in which the sentence letter p occurs modalized, a* ***decomposition*** *of X with respect to p is a sentence $D(p_{k_1}, \ldots, p_{k_n})$ containing sentence letters $p_{k_1}, \ldots, p_{k_n}$ that do not occur in X with sentences $\Box C_{k_1}(p), \ldots, \Box C_{k_n}(p)$ such that $X = D(\Box C_{k_1}(p), \ldots, \Box C_{k_n}(p))$, i.e. X is the result of substituting the sentence $\Box C_{k_i}(p)$ for each occurrence of the sentence letter p_{k_i} in the sentence $D(p_{k_1}, \ldots, p_{k_i}, \ldots, p_{k_n})$. The sentences $\Box C_{k_1}(p), \ldots, \Box C_{k_n}(p)$ are called* ***components*** *of X with respect to the decomposition $D(p_{k_1}, \ldots, p_{k_n})$ with $\Box C_{k_1}(p), \ldots, \Box C_{k_n}(p)$, and $D(p_{k_1}, \ldots, p_{k_n})$ is called the* ***decomposition sentence*** *for this decomposition. We will say that X is* ***composed*** *of the components $\Box C_{k_1}(p), \ldots, \Box C_{k_n}(p)$ by substitution into the decomposition sentence $D(p_{k_1}, \ldots, p_{k_n})$.*

Note that what are intrinsic for a given decomposition are its components and the logical form of the decomposition sentence. The choice of sentence letters occurring in the given decomposition is arbitrary, but the choice of numbering of the sentence letters and of the components must be covariant, so that each component is substituted in the right place in the logical form of the decomposition sentence to result in X.

Lemma 163 *If a sentence letter p occurs modalized in sentence X, then there is a decomposition of X with respect of p.*

Proof. We give two methods which generate a decomposition of X with respect to a sentence letter p modalized in X. As we shall see, for some sentences these two methods generate the same decomposition, for other sentences they generate two different decompositions, and also there are sentences that have decompositions other than those generated by these two methods.

Method 1 (top down): The sentence X cannot consist just of the sentence letter p, since p does not occur modalized in p. So there are two cases, either X is of the form $\Box Y$ for some sentence Y, or X is of the form $(Y \supset Z)$ for sentences Y and Z.

(i) X is of the form $\Box Y$. Then we are done, with the decomposition sentence for X any sentence letter p_i distinct from p and component $\Box Y$.

(ii) X is of the form $(Y \supset Z)$. Then p occurs in Y or in Z, or both, and wherever it occurs it occurs modalized. If the sentence or sentences in which p occurs modalized is/are of the form $\Box W$ then by (i) we are done. If not then it is of the form $(U \supset V)$ and we repeat the argument. Since X is generated in finitely many steps, this process comes to an end, and since p occurs modalized, it ends in components, i.e. sentences of the form $\Box C_i(p)$ in which p occurs modalized.

To obtain a decomposition of X, replace each of the different components $\Box C_i(p)$ by a distinct sentence letter p_i that does not occur in X, while replacing each occurrence of a given component, if it has more than one occurrence, by the same sentence letter. The resulting formula is a decomposition of X with components $\Box C_i(p)$.

Method 2 (bottom up): For each occurrence of p in X, find the innermost occurrence of $\Box$ in whose scope that occurrence of p occurs. For each such occurrence of $\Box$, let $C_i(p)$ be the sentence to which that occurrence of $\Box$ is prefixed. From the resulting set of sentences $\Box C_i(p)$, discard those sentences that are a subsentence of any of the other boxed sentences $\Box F(p)$ in which p occurs unmodalized in $F(p)$ (e.g. in $\Box(\Box p \supset p)$, we discard $\Box p$ which has the occurrence of $\Box$ with minimum scope for the occurrence of p first from the left, because it occurs within the scope of $\Box(\Box p \supset p)$, which contains the minimum scope for the modalized occurrence of the second occurrence of p). The sentences $\Box C_i(p)$ that remain after discarding sentences with minimum scope that occur within other sentences with minimum scope will be the components of the decomposition of X. In X replace each component sentence of distinct form by a distinct sentence letter not occurring in X, at each occurrence in X of that component. Components of the same form should be replaced by the same sentence letter. This gives the decomposition of sentence X with these components. ▲

Examples of decompositions.

(i) $(\Box(\Box(\Box p \supset p) \supset p) \supset \Box p)$

Method 1: The sentence is an implication between boxed formulas, so the components are those two boxed sentences, i.e. $\Box(\Box(\Box p \supset p) \supset p)$ and $\Box p$, with decomposition formula is $(p_1 \supset p_2)$.

Method 2: When applied to the three occurrences of p in the antecedent sentence of this implication, the procedure results in sentence $\Box p$ for the first occurrence of p (going from the left), $\Box(\Box p \supset p))$ for the second occurrence of p, and $\Box(\Box(\Box p \supset p) \supset p)$ for the third occurrence of p, and for the single occurrence of p in the consequent sentence, it results in $\Box p$. From the sentences resulting from the occurrences of p in the antecedent sentence, we discard $\Box p$ because $\Box p$ with that occurrence of p occurs in $\Box(\Box p \supset p))$ (and also in $\Box(\Box(\Box p \supset p) \supset p))$, and we discard $\Box(\Box p \supset p))$ since it is a subsentence of $\Box(\Box(\Box p \supset p) \supset p)$ and the two occurrences of p in $\Box(\Box p \supset p))$ occur in $\Box(\Box(\Box p \supset p) \supset p)$. So the components are $\Box(\Box(\Box p \supset p) \supset p)$ and $\Box p$, and a decomposition sentence for these components is $(p_1 \supset p_2)$.

In this example the two methods yield the same decomposition. An example of a sentence in which a sentence letter occurs modalized for which the two methods result in different decompositions is the following:

(ii) $(\Box(\Box p \supset q) \wedge \sim \Box p)$

By Method 1:

components $\Box C_1(p) = \Box(\Box p \supset q)$, and $\Box C_2(p) = \Box p$, with decomposition $(p_1 \wedge \sim p_2)$

By Method 2:

component $\Box C_1(p) = \Box p$, with decomposition $(\Box(p_1 \supset q) \wedge \sim p_1)$

Some sentences with a modalized sentence letter have more than two decompositions, e.g.

(iii) $\Box\,\Box\,\Box p$ has three decompositions:

(i) component $\Box C_1(p) = \Box\,\Box\,\Box p$, with decomposition p_1.

(ii) component $\Box C_1(p) = \Box\,\Box\, p$, with decomposition $\Box p_1$.

(iii) component $\Box C_1(p) = \Box p$, with decomposition $\Box\,\Box\, p_1$

The decomposition of $\Box\,\Box\,\Box p$ by Method 1 is (i). The decomposition of $\Box\,\Box\,\Box p$ by Method 2 is (iii).

Example (iii) generalizes to:

Proposition 164 *For each n, there is a sentence, modalized in a sentence letter, that has n-many decompositions.*

Proof. For each n, $\underbrace{\Box \ldots \Box}_{n} p$ has decompositions

$D_i = \underbrace{\Box \ldots \Box}_{n-i} p_1$ with component $\underbrace{\Box \ldots \Box}_{i} p$, for each i such that $1 \leq i \leq n$. ▲

For sentences in which a sentence letter occurs modalized, there is an arithmetized substitution theorem which yields the conclusion of Theorem 160 on the hypothesis of Theorem 154, i.e.

Theorem 165 (arithmetized substitution in modalized sentences) *Let $F(p)$ be a sentence in which sentence letter p occurs modalized. Then $GL \vdash (\Box(A \equiv B) \supset (F_p(A) \equiv F_p(B)))$.*

Proof. By Lemma 163, there is a decomposition $D(p_1, \ldots, p_m)$, with sentence letters $p_1, \ldots, p_m$ not in $F(p)$ and components $\Box C_1(p), \ldots, \Box C_m(p)$. The following is (a recipe for) a proof in GL.

(1)	$(\Box(A \equiv B) \supset \Box(C_{i_p}(A) \equiv C_{i_p}(B)))$	$1 \leq i \leq m$ Theorem 154 $(*)$
(2)	$(\Box(A \equiv B) \supset (\Box C_i(A) \equiv \Box C_i(B)))$	(1) Proposition 147 prop logic
(3)	$(\Box\,\Box\,(A \equiv B) \supset \Box(\Box C_i(A) \equiv \Box C_i(B)))$	(2) Lemma 144
(4)	$(\Box(A \equiv B) \supset \Box(\Box C_i(A) \equiv \Box C_i(B)))$	(3) Theorem 152 prop logic $(**)$
(5)	$(\Box(A \equiv B) \supset \boxdot(\Box C_i(A) \equiv \Box C_i(B)))$	(2)(4) $\wedge$-Introduction
(6)	$(((\boxdot(\Box C_1(A) \equiv \Box C_1(B))) \wedge \ldots \wedge \boxdot(\Box C_m(A) \equiv \Box C_m(B))) \supset$ $(D(\Box C_1(A), \ldots, \Box C_m(A)) \equiv D(\Box C_1(B), \ldots, \Box C_m(B))))$	Theorem 162 $(* * *)$
(7)	$(\Box(A \equiv B) \supset (F_p(A) \equiv F_p(B)))$	(5)(6) prop logic ▲

$(*)$ Theorem 154 depends on Theorem 152, which depends on A_3.

$(**)$ Theorem 152 depends on A_3.

$(* * *)$ Theorem 162 is a generalization of Theorem 160, which depends on Corollary 157 of Theorem 156, which uses Theorem 152, which depends on A_3.

14.2 The fixed point theorem for GL

Definition 93 (fixed point) *For X a sentence in the language of GL that contains the sentence letter p_i, a sentence F that contains only sentence letters contained in X and does not contain p_i such that $GL \vdash (F \equiv X_{p_i}(F))$ is a fixed point for X with respect of p_i.*

We shall show that every sentence F modalized in a sentence letter p_i has a fixed point with respect to p_i. The argument is by induction on the number of components in a decomposition of F. We begin with a lemma which establishes the case where F has a decomposition with a single component, i.e. F is of the form $\Box Y(p_i)$.

Lemma 166 (fixed point for $\Box Y$) *For p_i any sentence letter and $Y(p_i)$ any sentence in the language of GL in which p_i occurs, the result of substituting $(\perp\supset\perp)$ for each occurrence of p_i in $\Box Y(p_i)$ is a fixed point for $\Box Y(p_i)$.*

Proof. In this proof we abbreviate $(\perp\supset\perp)$ as $\top$, and write $\Box Y(\top)$ for the result of substituting $(\perp\supset\perp)$ for each occurrence of p_i in $\Box Y(p_i)$. The following derivation is a proof *in* GL.

(1) $(\Box Y(\top) \supset (\top \equiv \Box Y(\top))$ tautology.

(2) $\Box(\Box Y(\top) \supset (\top \equiv \Box Y(\top))$	(1) *R*2.
(3) $(\Box\,\Box\, Y(\top) \supset \Box(\top \equiv \Box Y(\top)))$	(2) *A*2 *R*1
(4) $(\Box Y(\top) \supset \Box(\top \equiv \Box Y(\top)))$	(3) Theorem 152 prop logic.
(5) $(\Box(\top \equiv \Box Y(\top)) \supset \Box(Y(\top) \equiv Y(\Box Y(\top))))$	Theorem 154.
(6) $(\Box(Y(\top) \equiv Y(\Box Y(\top))) \supset (\Box Y(\top) \equiv \Box Y(\Box Y(\top))))$	Proposition 147.
(7) $(\Box Y(\top) \supset (\Box Y(\top) \equiv \Box Y(\Box Y(\top))))$	(4) (5) (6) transitivity of $\supset$.
(8) $(\Box Y(\top) \supset \Box Y(\Box Y(\top)))$	(7) prop logic.
(9) $(\Box Y(\top) \supset \boxdot(\top \equiv \Box Y(\top)))$	(1) (4) prop logic
(10) $(\Box Y(\top) \supset (Y(\top) \equiv Y(\Box Y(\top))))$	(9), Theorem 160, prop logic
(11) $(Y(\Box Y(\top)) \supset (\Box Y(\top) \supset Y(\top)))$	(10) prop logic.
(12) $(\Box Y(\Box Y(\top)) \supset \Box(\Box Y(\top) \supset Y(\top)))$	(11) *R*2 *A*2 *R*1
(13) $(\Box Y(\Box Y(\top)) \supset \Box Y(\top))$	(12) A_3 prop logic.
(14) $(\Box Y(\top) \equiv \Box Y(\Box Y(\top)))$	(8)(13)$\wedge$-Introduction. ▲

Before proceeding to the induction step in our proof of the Fixed Point Theorem, we need another lemma and for that lemma we need a definition:

Definition 94 (solvable systems of simultaneous equivalences) *For* $1 \leq i \leq m$*, let* $A_i(p_{k_1}, \dots, p_{k_m})$*, be* m*-many sentences all with the same sentence letters, among which are the* m*-many sentence letters* $p_{k_1}, \dots, p_{k_m}$*. A system of simultaneous equivalences of the form*

$$(p_{k_i} \equiv A_i(p_{k_1}, \dots, p_{k_i}, \dots, p_{k_m}))$$

is solvable *in GL iff there are sentences* $F_1, \dots, F_m$ *not containing* $p_{k_1}, \dots, p_{k_m}$*, but containing the remaining sentence letters in each* A_i*, such that for* $1 \leq i \leq m$*,*

$$GL \vdash (F_i \equiv A_i(p_{k_1}/F_1, \dots, p_{k_i}/F_i, \dots, p_{k_m}/F_m)).$$

Lemma 167 (solving systems of simultaneous equivalences) *Every system of* m*-many simultaneous equivalences of the form*

$$(p_{k_i} \equiv \Box C_i(p_{k_1}, \dots, p_{k_i}, \dots p_{k_m})),$$

for $1 \leq i \leq m$*, is solvable in GL.*

Proof. The proof is by induction on m.

$m = 1$. This case is Lemma 166.

Suppose the Lemma holds for m. Let $\Box C_i(p_{k_1}, \ldots, p_{k_m}, p_{k_{m+1}})$ be a set of $(m+1)$-many sentences with $m+1$-many sentence letters $p_{k_1}, \ldots, p_{k_m}, p_{k_{m+1}}$. By the Induction Hypothesis, there are sentences $F_i(p_{k_{m+1}})$, $1 \leq i \leq m$, each of which does not contain the sentence letters $p_{k_1}, \ldots, p_{k_m}$ and does contain the sentence letter $p_{k_{m+1}}$, such that for $1 \leq i \leq m$,

$$(1)\ \ GL \vdash (F_i(p_{k_{m+1}}) \equiv \Box C_i(p_1/F_1(p_{k_{m+1}}), \ldots, p_m/F_m(p_{k_{m+1}}), p_{k_{m+1}})).$$

Substitute $F_i(p_{k_{m+1}})$ for p_{k_i} in $\Box C_{m+1}(p_{k_1}, \ldots, p_{k_m}, p_{k_{m+1}})$, for each i such that $1 \leq i \leq m$. This results in the sentence $\Box C_{m+1}(F_1(p_{k_{m+1}}), \ldots, F_m(p_{k_{m+1}}), p_{k_{m+1}})$. By Lemma 166 applied to $\Box C_{m+1}(F_1(p_{k_{m+1}}), \ldots, F_m(p_{k_{m+1}}), p_{k_{m+1}})$ with respect to the sentence letter p_{km+1}, there is a sentence F_{m+1} such that

$$(2)\ \ GL \vdash F_{m+1} \equiv \Box C_{m+1}(F_1(F_{m+1}), \ldots, F_m(F_{m+1}), F_{m+1})).$$

Substituting F_{m+1} for $p_{k_{m+1}}$ in each of the provable equivalences (1), the sentences

$$F_1(p_{k_{m+1}}/F_{m+1}), \ldots, F_m(p_{k_{m+1}}/F_{m+1}), F_{m+1}$$

are a set of solutions to $m+1$-many simultaneous equivalences. ▲

We are now in a position to prove the Fixed Point Theorem. There are a number of different proofs of this theorem, most of which use Kripke models for the modal operator (three such proofs are expounded by George Boolos in his book *The Logic of Provability*, Cambridge University Press, 1993, pp. 104-123). The purely syntactic proof given here follows Per Lindstrom, "Provability logic—a short introduction", *Theoria* 62 (1996), pp. 31-35; see also Craig Smorynski, *Self-Reference and Modal Logic*, Springer, 1985, pp. 78-82.

Theorem 168 (Fixed Point Theorem) *For every sentence X_p modalized in p, there is a sentence F containing only sentence letters that occur in X_p other than p such that $GL \vdash (F \equiv X_p(F))$.*

Proof. The proof we give establishes this result constructively, as follows. For $D(p_1, \ldots, p_k)$ a decomposition of X_p with respect to modalized sentence letter p and components $\Box C_1(p), \ldots, \Box C_k(p)$, we have $F_1, \ldots, F_k$ in which p does not occur and the sentence letters in each F_i are among those occurring in X, such that GL proves the simultaneous equivalences $(F_1 \equiv \Box C_{1_p}(D(F_1, \ldots, F_i, \ldots, F_k)))$, $\ldots$, $(F_i \equiv \Box C_{i_p}(D(F_1, \ldots, F_i, \ldots, F_k))), \ldots, (F_k \equiv \Box C_{k_p}(D(F_1, \ldots, F_i, \ldots, F_k)))$, from which it follows that $GL \vdash (D(F_1, \ldots, F_k) \equiv$ $D(\Box C_{1_p}(D(F_1, \ldots, F_k), \ldots, \Box C_{k_p}(D(F_1, \ldots, F_k)))$, which is to say that $GL \vdash (D(F_1, \ldots, F_k) \equiv X_p(D(F_1, \ldots, F_k)))$.

The argument is as follows. Let X be any sentence modalized in p. Then by Lemma 163, X_p has a decomposition $D(p_{k_1}, \ldots, p_{k_n})$, with one or more sentence letters $p_{k_1}, \ldots, p_{k_n}$ distinct from the sentence letters occurring in X, and components $\Box C_1(p), \ldots, \Box C_n(p)$, i.e. there is a one-to-one correspondence between the sentence letters p_{k_i} and the components $\Box C_i(p)$ so that $X_p = D(p_{k_1}/\Box C_1(p), \ldots, p_{k_n}/\Box C_n(p))$.

We first consider the case where the decomposition D_q contains just one sentence letter, call it q, and correspondingly there is a single component $\Box C_p$, such that $X_p = D_q(\Box C_p)$. Form the sentence $\Box C_p(D_q)$ by substituting the decomposition sentence D_q for the sentence letter p in the component $\Box C_p$. By Lemma 166, $\Box C_p(D_q)(\top)$ is a fixed point for $\Box C_p(D_q)$ with respect to q, i.e. for $F = \Box C_p(D_q)(\top)$, $GL \vdash (F \equiv \Box C_p(D_q)(F))$. Then by Theorem 150 (provable equivalence of substitution of provable equivalents in GL), $GL \vdash (D_q(F) \equiv D_q(\Box C_p(D_q)(F))$. Since $X_p = D_q(\Box C_p)$, this provable equivalence tells us that $D_q(F)$ is a fixed point for X_p.

This argument is generalizable to the case of a decomposition with more than one component, as follows: Form the sentences $\Box C_i(p/D(p_{k_1}, \ldots, p_{k_n}))$ by substituting $D(p_{k_1}, \ldots, p_{k_n})$ for p in each of the sentences $\Box C_i(p)$. By Lemma 167, the set of simultaneous equivalences $(p_{k_i} \equiv \Box C_i(p/D(p_{k_1}, \ldots, p_{k_i}, \ldots p_{k_n})))$, indexed by $i = 1, \ldots, n$, is solvable. Let $F_1, \ldots, F_n$ be a set of solutions to these simultaneous equivalences, i.e. for each $i = 1, \ldots, n$, $GL \vdash (F_i \equiv \Box C_i(D(F_1, \ldots, F_i, \ldots, F_n)))$.

Substitute the left sentences of these n equivalences into the sentence letters $p_{k_1}, \ldots, p_{k_n}$ in the sentence $D(p_{k_1}, \ldots, p_{k_n})$, and substitute the right sentences of these equivalences into the sentence letters $p_{k_1}, \ldots, p_{k_n}$ in the sentence $D(p_{k_1}, \ldots, p_{k_n})$. By Theorem 153 in Lecture 13, the two sentences that result from these substitutions are provably equivalent, i.e.

$$GL \vdash (D(p_{k_1}/F_1, \ldots, p_{k_n}/F_n) \equiv$$

$$D(p_{k_1}/\Box C_1(D(p_{k_1}/F_1, \ldots, p_{k_n}/F_n), \ldots, p_{k_1}/\Box C_n(D(p_{k_1}/F_1, \ldots, p_{k_n}/F_n))).$$

Since $D(p_{k_1}/\Box C_1(p), \ldots, p_{k_n}/\Box C_n(p)) = X(p)$, this equivalence means that $D(p_{k_1}/F_1, \ldots, p_{k_n}/F_n)$ is a fixed point for $X(p)$ with respect to p. ▲

Remark. The sentence F gives the explicit, i.e. non self-referential, content of a self-referential sentence p such that $(p \equiv X(p))$, for X modalized in p.

Example 1: $F(p) = \sim \Box p$. First note that p occurs modalized in $\sim \Box p$, with decomposition $\sim q$ and component $\Box p$. Following through the proof of the Fixed Point Theorem (Theorem 168), we substitute the decomposition sentence for p in each component, which yields $\Box{\sim} q$. By Lemma 166, $\Box{\sim}\top$ is a fixed point for $\Box{\sim} q$, and $\Box\sim\top$ is provably equivalent in GL to $\Box\perp$, so we have $GL \vdash (\Box\perp \equiv \Box\sim\Box\perp)$. Substituting this provable equivalence into the decomposition for $\sim\Box p$, we obtain

$$GL \vdash (\sim\Box\perp \equiv \sim\Box\sim\Box\perp)$$

This result illustrates the power of the Fixed Point Theorem for GL, which gives the intrinsic meaning of diagonal sentences to do with provability in PA, in this case for the Gödel sentence. As constructed, the Gödel sentence has the meaning, "this sentence is not provable". Though there is no logical circularity in this construction, there is a meaning circularity in this formulation of the meaning of G. What the Fixed Point Theorem tells us is that the intrinsic meaning of G is that the system in which it's constructed is consistent.

Note also that left to right of this biconditional is the arithmetized Second Incompleteness Theorem.

Example 2: $F(p) = \Box(\Box(p \wedge q) \wedge \Box(p \wedge r))$

We calculate the fixed point for $F(p)$ with respect to p by two difference compositions, one obtained by the 'top down' method, the other 'bottom up'.

Decomposition 1 (top down): $\Box(\Box(p \wedge q) \wedge \Box(p \wedge r))$ is the component for decomposition p_1.

Substituting the decomposition p_1 for p in the component $\Box(\Box(p \wedge q) \wedge \Box(p_1 \wedge r))$ gives $\Box(\Box(p_1 \wedge q) \wedge \Box(p_1 \wedge r))$. Lemma 166 gives a solution to the equivalence $(p_1 \equiv \Box(\Box(p_1 \wedge q) \wedge \Box(p_1 \wedge r)))$ as $\Box(\Box(\top \wedge q) \wedge \Box(\top \wedge r))$, which is provably equivalent to $\Box(\Box q \wedge \Box r)$.

Decomposition 2 (bottom up): The components are $\Box(p \wedge q)$ and $\Box(p \wedge r)$ for decomposition $\Box(p_1 \wedge p_2)$. Substituting the decomposition in place of p in the two components yields the following simultaneous equivalences to be solved:

(1) $(p_1 \equiv \Box(\Box(p_1 \wedge p_2) \wedge q))$

(2) $(p_2 \equiv \Box(\Box(p_1 \wedge p_2) \wedge r))$

Lemma 166 solves for p_1 in (1) in terms of p_2 to yield

(3) $p_1 = \Box(\Box(\top \wedge p_2) \wedge q)$, which simplifies by provable equivalences to

(4) $p_1 = \Box(\Box p_2 \wedge q)$

Substituting this solution for p_1 in place of p_1 in equivalence (2) yields

(5) $(p_2 \equiv \Box(\Box(\Box(\Box p_2 \wedge q) \wedge p_2) \wedge r))$

Lemma 166 solves for p_2 in (5) as

(6) $p_2 = \Box(\Box(\Box(\Box\top \wedge q) \wedge \top) \wedge r)$, which simplifies by provable equivalences to

(7) $p_2 = \Box(\Box\Box q \wedge r)$, which is provably equivalent to $(\Box\Box\Box q \wedge \Box r)$

Substituting the solution for p_2 in (7) for the solution of p_1 in terms of p_2 in (4) yields

(8) $p_1 = \Box(\Box\Box(\Box\Box q \wedge r) \wedge q)$, provably equivalent to $(\Box\Box\Box\Box\Box q \wedge \Box\Box\Box r \wedge \Box q)$

Substituting the solutions (7) and (8) for p_1 and p_2 into the decomposition we obtain the fixed point

(9) $\Box((\Box\Box\Box\Box\Box q \wedge \Box\Box\Box r \wedge \Box q) \wedge (\Box\Box\Box q \wedge \Box r))$, provably equivalent to

$\Box(\Box\Box\Box\Box\Box q \wedge \Box\Box\Box q \wedge \Box q \wedge \Box\Box\Box r \wedge \Box r)$

(10) $GL \vdash ((\Box\Box\Box\Box\Box q \wedge \Box\Box\Box q \wedge \Box q \wedge \Box\Box\Box r \wedge \Box r) \equiv (\Box q \wedge \Box r))$. Left to right is by $\wedge$-Elimination, and right to left is by repeated application of Theorem 152 (P_3),

which shows that the fixed points obtained by the two different decompositions are provably equivalent.

Remarks about the Fixed Point Theorem in relation to these two examples. The first example shows that the Fixed Point Theorem subsumes the Second Incompleteness Theorem. Both examples show that the Fixed Point Theorem enables us to obtain the intrinsic meaning of diagonal sentences that concern provability. Both examples show that obtaining that intrinsic meaning is not something that can be done 'by inspection'. Do this in the case of the first example amounts to proving the Second Incompleteness Theorem. The second example doesn't connect up with any results we've looked at before, but at the same time illustrates the the power of this result by it's wide applicability. A diagonal sentence for a sentence of the form $\Box(\Box(p \wedge q) \wedge \Box(p \wedge r))$, i.e. p such that $(p \equiv \Box(\Box(p \wedge q) \wedge \Box(p \wedge r)))$ as 'saying' 'this sentence is such that it's provably provable with sentences q and r. But what is such a sentence (which by the Diagonal Lemma we know exists) actually 'saying', i.e. what is it saying about q and r. The Fixed Point Theorem enables us to answer these questions. In the case of example 2, the diagonal sentence is saying that it provable that q and r are both provable. That this is telling us something not easy to see on its own is evident if one works out a derivation in GL of the fixed point equivalence $(\Box(\Box q \wedge \Box r) \equiv \Box(\Box(\Box(\Box q \wedge \Box r) \wedge q) \wedge \Box(\Box(\Box q \wedge \Box r) \wedge r)))$ (exercise).

The fixed points proved to exist by the Fixed Point Theorem are unique to within provable equivalence, as follows.

Theorem 169 (provable equivalence of fixed points) *Let $X(p)$ be a sentence in the language of GL in which the sentence letter p occurs modalized, and let q*

be a sentence letter that does not occur in $X(p)$. Abbreviating $X_p(q)$ as $X(q)$, $GL \vdash ((\Box(p \equiv X(p)) \wedge \Box(q \equiv X(q))) \supset \Box(p \equiv q))$.

Proof. By Theorem 165 (arithmetized substitution into modalized sentences), taking A as p and B as q,

$(\Box(p \equiv q) \supset (X(p) \equiv X(q)))$ is provable in GL. Then by propositional logic,

$(((p \equiv X(p)) \wedge (q \equiv X(q))) \supset (\Box(p \equiv q) \supset (p \equiv q)))$. By Lemma 144 and Theorem 146,

$((\Box(p \equiv X(p)) \wedge \Box(q \equiv X(q))) \supset \Box(\Box(p \equiv q) \supset (p \equiv q)))$. Then by Axiom A_3 and propositional logic,

$((\Box(p \equiv X(p)) \wedge \Box(q \equiv X(q))) \supset \Box(p \equiv q))$. ▲

Proposition 170 (inequivalent fixed points for p non-modalized)

Proof. Any letterless sentence F is a fixed point for the formula p, since $p_p(F) = F$ and $GL \vdash (F \equiv F)$, which is to say that $F_1 = \bot$ and $F_2 = (\bot \supset \bot)$ are both fixed points for the $X = p$, but $GL \nvdash (\bot \equiv (\bot \supset \bot))$. ▲

The uniqueness of the fixed point to within provable equivalence turns essentially on the sentence letter of the fixed point equivalence occurring modalized in the sentence for which the existence of a fixed point follows. We can have fixed points for non-modalized sentences that are not unique with respect to provable equivalence. For example, take $X(p)$ as $((r \supset r) \supset ((q \supset q) \supset p))$. By Definition 93, a fixed point for $X(p)$ is a sentence X containing only sentence letters that occur in X and not containing p such that $GL \vdash (F \equiv X_p(F))$. By this criterion, since $GL \vdash (q \equiv ((r \supset r) \supset ((q \supset q) \supset q)))$ and $GL \vdash (r \equiv ((r \supset r) \supset ((q \supset q) \supset r)))$, q and r are both fixed points of $((r \supset r) \supset ((q \supset q) \supset p))$. By R_2, $GL \vdash \Box(q \equiv ((r \supset r) \supset ((q \supset q) \supset q)))$ and $GL \vdash \Box(r \equiv ((r \supset r) \supset ((q \supset q) \supset r)))$. Suppose $GL \vdash \Box(q \equiv r)$. Then by Proposition 149, $GL \vdash \Box(\bot \equiv (\bot \supset \bot))$, in which case $GL \vdash \Box\bot$. But assuming the PA is Σ_1-sound, by Theorem 141, $GL \nvdash \Box\bot$. In which case $GL \nvdash \Box(q \equiv r)$.

The previous example can be tweaked to give an example of sentence in which p occurs not modalized that has a fixed point with respect to p which is unique to within provable equivalence, e.g. $((q \supset q) \supset p)$. It is also worth noticing how modalizing our original example to non-uniqueness to within provable equivalence of a fixed point for a non-modalized sentence yield provable equivalence, e.g. $((r \supset r) \supset ((q \supset q) \supset \Box p))$. This sentence is provably equivalent to $\Box p$, so all fixed points for it are provably equivalent to the Henkin sentence.

Theorem 171 (Strengthened Fixed Point Theorem for GL) *For every sentence X in the language of GL modalized in p, there is a sentence F containing only*

sentence letters that occur in X and not containing p such that
$GL \vdash (\boxdot(p \equiv X(p)) \equiv \boxdot(p \equiv F))$.

Proof. Theorems 168 and 169 establish this result. ▲

Theorem 172 *Theorem 171 implies Theorems 168 and 169.*

Proof. Substituting F for p in Theorem 171 results in GL $\vdash (\boxdot(F \equiv X(F)) \equiv \boxdot(F \equiv F))$ Since $(F \equiv F)$ is a tautology, GL $\vdash (F \equiv F)$, so GL $\vdash \Box(F \equiv F)$, so by $\wedge$-Introduction, GL $\vdash \boxdot(F \equiv F)$. Hence GL $\vdash \boxdot(F \equiv X(F))$, so by $\wedge$-Elimination, GL $\vdash (F \equiv X(F))$. ▲

Lecture 15

The arithmetized completeness theorem for first-order logic; non-standard models of arithmetic; the Kreisel Δ_2-Incompleteness Theorem

Monday 4 March 2019

15.1 Introduction

As we have seen, first-order Peano Arithmetic, and indeed much weaker systems, are Σ_1-complete.

The sentence that Gödel showed how to construct in systems strong enough to arithmetize formal syntax, such that if the system is sound the sentence is undecidable in the given system, is Π_1. Given Σ_1-completeness, Gödel's incompleteness theorem is, in terms of the arithmetical hierarchy, best possible. However, as we have noted before, Σ_1-completeness implies that an irrefutable Π_1-sentence is true:

$$\sim \forall v_1 A(v_1) \equiv \exists v_1 \sim A(v_1)$$

is Σ_1, so if true is provable. So if not provable then false, which is to say that $\forall v_1 A(v_1)$ is true, as shown in Theorem 88 (Lecture 8).

In 1950 Kreisel published his first paper in mathematical logic, 'Note on arithmetic models for consistent formulae of the predicate calculus', which included an incompleteness theorem for a Δ_2-sentence, i.e. one equivalent to formulas of both the forms $\forall v_1 \exists v_2 A(v_1, v_2)$ and $\exists v_1 \forall v_2 B(v_1, v_2)$. The effect of the greater logical complexity over the Gödel sentence is that independence no longer determines truth. The negation of the Δ_2-Kreisel sentence is itself Δ_2, so there is symmetry where for the Gödel sentence there is asymmetry. And indeed for almost all instances of the Kreisel sentence, determining its truth value remains an open problem, as we shall consider in Lecture 16.

15.2 Arithmetized completeness theorem for first-order logic

The completeness of (a given system of) first-order logic is the condition that if a sentence θ is a logical consequence of a set of sentences Γ, in the sense that θ is true in every model of Γ, then there is a derivation of θ from Γ in (the given system of) first-order logic.

By contraposition this condition is equivalent to:

Theorem 173 (Completeness of first-order logic) *If a set of sentences S in a first-order language is consistent (with respect to a given formal system of first-order logic whose completeness is being established), then S has a model.*

Proof sketch. Since Gödel's original proof in [1930] a number of other proofs have been given of which the one by Henkin [1949] is now standard. All proofs establish a stronger result than stated above, namely that every consistent theory has a model of at most the cardinality of the natural numbers.

The Henkin proof proceeds by adding to the language of the set of sentences S a countable infinity of new individual constants $c_0, c_1, \ldots, c_k, \ldots$ and then establishing that S has a model whose domain consists of the constant terms c_i. This is done by first adding to S formulas of the form $(\exists x \phi(x) \supset \phi(c_{[\phi]}))$ for each formula ϕ, where $c_{[\phi]}$ is one of the c_i chosen in such a way that to each formula ϕ there corresponds a unique constant $c_{[\phi]}$. Clearly if S is consistent $S^+ = S \cup \{(\exists x \phi(x) \supset \phi(c_{[\phi]})\}$ for all ϕ is also consistent. One then enumerates all sentences $\phi_0, \phi_1, \ldots, \phi_k, \ldots$ of this augmented language and defines a complete consistent theory starting with S^+ and adding ϕ_n at stage n if ϕ_n is consistent with $S^+ \cup \{\text{earlier choices}\}$, otherwise $\sim \phi_n$. Every stage is consistent and so the union is consistent, and since for every sentence ϕ, ϕ or $\sim \phi$ has been chosen, the union is a complete consistent extension of S

(Lindenbaum's Lemma)[1]. The added instantiation axioms are then used to show that this complete consistent extension of S determines the complete diagram of a model of S whose domain is the constant terms c_i. Since the set of added constant terms c_i is countable, the model shown to exist is countable. ▲

If a model is countable, that means there's a bijection between the elements of its domain and the natural numbers. We can then wonder whether the model can be defined in the language of arithmetic. Paul Bernays, in Hilbert and Bernays, *Grundlagen der Mathematik*, volume II (1939), showed that this is the case. Georg Kreisel, in his first published paper in logic, "Note on arithmetic models for consistent formulae of the predicate calculus", *Fundamenta Mathematicae* 37 (1950), pp. 265-285, showed that Bernays' proof can itself be carried in a system of arithmetic. Bernays and Kreisel were working from Gödel's original proof of the completeness theorem, but the result can also be formulated as an arithmetization of Henkin's proof of the completeness theorem. The proof is highly complicated. Here I just sketch the result, for the particular case where S is the theorems of PA, in order to draw from it a new incompleteness theorem for PA.

Theorem 174 (Bernays-Kreisel Arithemetized Completeness Theorem for PA) *Let $Pr(v_1)$ be a Σ_1 arithmetized proof predicate for PA (as we have constructed), and let Con_{PA} be $\sim Pr(\ulcorner 0 = 0' \urcorner)$. There exists a Δ_2-formula with one free variable $Tr(v_2)$ which arithmetizes the Henkin proof of the completeness theorem for first-order logic applied to $\{\varphi : PA \vdash \varphi\}$, and $PA \cup \{Con_{PA}\} \vdash \forall v_1(Pr(v_1) \supset Tr(v_1))$.*

Proof sketch. Note first that this result is an expression in arithmetized syntax of the completeness theorem applied to $\{\varphi : PA \vdash \varphi\}$, i.e. it the set of theorems of PA is consistent, then it has a complete consistent extension (which determines a model of that set of sentences, i.e. a model of PA).

Each step of the proof of the completeness theorem can be expressed in arithmetized syntax, in particular the condition that if $T_n \cup \{\phi_n\}$ is consistent then $T_{n+1} = T_n \cup \{\phi_n\}$ and otherwise $T_{n+1} = T_n \cup \{\sim \phi_n\}$, where T_n is $S^+ \cup \{\text{earlier choices}\}$ at stage n.

How do we know that the resulting 'truth' predicate is Δ_2? The simplest way to see this is by appeal to Post's Theorem on relative recursiveness.

A number-theoretic predicate $R(v_1)$ is recusive relative to a number-theoretic predicate $H(v_1)$ just in case there is a recursive procedure for deciding if a given number

[1]This complete consistent extension is a *set* of sentences such that each sentence in the language of S or its negation belongs to it, and not both. This extension is not axiomatic, which is to say that the Gödel numbers of sentences in it is not Σ_1, so there is no incompatibility with Gödel's First Incompleteness Theorem.

satisfies $R(v_1)$ that uses in its computation procedure the yes/no answers that an oracle gives as to whether particular numbers satisfy $H(v_1)$. Post showed that a predicate recursive in a Σ_n-predicate is Δ_{n+1} (see S.C. Kleene, *Introduction to Metamathematics* (1952), p. 293, Theorem XI). Inspection of the arithmetization of the construction of a complete consistent extension of the axiomatizable theory S is recursive in the Σ_1-proof predicate of S. Hence by Post's Theorem, the resulting predicate is Δ_2. ▲

15.3 Non-standard models of arithmetic

The Gödel's First Incompleteness Theorem, combined with the truth of the Gödel sentence and Gödel's Completeness Theorem for first-order logic, establishes the existence of non-standard models of arithmetic. However, this result can be established without appeal to the Incompleteness Theorem, just from the Completeness Theorem via the Compactness Theorem, and indeed a stronger result can be established in this way, namely that there exist countable non-standard models of true arithmetic, i.e. all sentences true in the standard model (which we have encountered before, as being the only complete ω-consistent theory of arithmetic).

There are as many countable non-standard models of PA as cardinality considerations allow, namely continuum many. We can prove this by iteration of Rosser's Theorem.

Nonetheless, all countable non-standard models of PA have the same order type, namely $\omega + (\omega^* + \omega) \cdot \eta$ (Henkin).

The truth predicate generated by the Bernays-Kreisel arithmetization of Gödel's completeness theorem must, by Tarski's theorem on the undefinability of truth, be the truth predicate of a non-standard model of PA.

15.4 The Kreisel Incompleteness Theorem

The Kreisel incompleteness theorem brings together the completeness theorem for first-order predicate logic and the incompleteness of formal systems adequate for arithmetization of syntax. It is based upon the Bernays-Kreisel arithmetized completeness theorem.

Theorem 175 (Kreisel Incompleteness Theorem) *Let $Tr(v_1)$ be a Δ_2-formula that arithmetizes the Henkin proof of the completeness theorem for first-order logic.*

Let K be a provably diagonal sentence for the formula $\sim Tr(v_1)$, i.e. $PA \vdash (K \equiv\sim Tr(\overline{\ulcorner K \urcorner}))$. If PA is consistent, $PA \nvdash K$ and $PA \nvdash\sim K$.

Proof. For our purposes, the way to understand the Bernays-Kreisel Arithmetized Completeness Theorem is as showing, on the assumption that PA is consistent, how within a model of PA a model of PA is definable. The defined model cannot be isomorphic to the model within which it's defined, by Tarski's theorem on the undefinability of truth. For $\mathfrak{M}$ any model of PA, let us denote by $\mathfrak{M}^*$ a model of PA defined in $\mathfrak{M}$ by $Tr(v_1)$.

Since $PA \vdash (K \equiv\sim Tr(\overline{\ulcorner K \urcorner}))$ and $\mathfrak{M}$ is a model of PA,

$$\mathfrak{M} \models (K \equiv\sim Tr(\overline{\ulcorner K \urcorner})) \tag{15.1}$$

Since $Tr(v_1)$ defines $\mathfrak{M}^*$ by its interpretation in $\mathfrak{M}$,

$$\mathfrak{M} \models K \text{ iff } \mathfrak{M}^* \models\sim K \tag{15.2}$$

as follows:

Suppose $\mathfrak{M} \vDash K$. Then by $PA \vdash (K \equiv \sim Tr(\overline{\ulcorner K \urcorner}))$, $\mathfrak{M} \vDash \sim Tr(\overline{\ulcorner K \urcorner}$, i.e. K is not true in $\mathfrak{M}^*$, so $\sim K$ is true in $\mathfrak{M}^*$.

Suppose $\mathfrak{M}^* \vDash \sim K$. Then K is not true in $\mathfrak{M}^*$, which is to say that $\sim Tr(\overline{\ulcorner K \urcorner})$ is true in $\mathfrak{M}$, and so by (15.1), K is true in $\mathfrak{M}$.

Suppose that

$$PA \vdash K \tag{15.3}$$

Let $\mathfrak{N}$ be the standard model of PA. Then

$$\mathfrak{N} \models K \tag{15.4}$$

But then by (15.2)

$$\mathfrak{N}^* \models\sim K \tag{15.5}$$

But $\mathfrak{N}^*$ is a model of PA, so by (15.3) we have

$$\mathfrak{N}^* \models K \tag{15.6}$$

But K cannot be both true and false in a model, so by *reductio ad absurdum*

$$PA \nvdash K \tag{15.7}$$

Suppose that

$$PA \vdash \sim K \tag{15.8}$$

Then since $\mathfrak{N}^*$ is a model of PA

$$\mathfrak{N}^* \models \sim K \tag{15.9}$$

But then by (15.2)

$$\mathfrak{N} \models K \tag{15.10}$$

But also from (15.8)

$$\mathfrak{N} \models \sim K \tag{15.11}$$

Again impossible. So

$$PA \nvdash \sim K \tag{15.12}$$

▲

Lecture 16

Determining the truth or falsity of undecidable Kreisel sentences

Wednesday 6 March 2019

Let us denote by $\mathfrak{H}$ (for Henkin) the operator which generates $\mathfrak{M}^*$ from $\mathfrak{M}$. The following results give information relevant to determining the truth value of K and in some cases determine it. But in the general case this is an open problem.

Lemma 176 *For every model $\mathfrak{M}$ of PA, $\mathfrak{M} \in dom(\mathfrak{H})$ iff $\mathfrak{M} \models Con(PA)$.*

Proof. (i) Suppose $\mathfrak{M} \nvDash Con(PA)$, in which case $\mathfrak{M} \vDash {\sim}Con(PA)$, which is to say that $\mathfrak{M} \vDash Pr_{PA}(\overline{\ulcorner 0 = 0' \urcorner})$. But also, since $PA \vdash {\sim}0 = 0'$, $PA \vdash Pr_{PA}(\overline{\ulcorner {\sim}0 = 0' \urcorner})$, so $\mathfrak{M} \vDash Pr_{PA}(\overline{\ulcorner {\sim}0 = 0' \urcorner})$. Then by the proof of the Arithmetized Completeness Theorem, we have, $\mathfrak{M} \vDash Tr(\overline{\ulcorner 0 = 0' \urcorner})$ and $\mathfrak{M} \vDash Tr(\overline{\ulcorner {\sim}0 = 0' \urcorner})$. This tells us that $\mathfrak{H}$ is not defined on $\mathfrak{M}$.

(ii) Suppose $\mathfrak{H}$ is not defined on $\mathfrak{M}$, i.e. there is a sentence X such that $\mathfrak{M} \vDash Tr(\overline{\ulcorner X \urcorner})$ and $\mathfrak{M} \vDash Tr(\overline{\ulcorner {\sim}X \urcorner})$, and let X be the first such sentence of the language of PA for which this happens. The only way in which this can happen at that point is by $\mathfrak{M} \vDash {\sim}Con(PA)$. ▲

Lemma 177 *The process of defining successive models by $\mathfrak{H}$ starting on $\mathfrak{N}$ terminates in a finite number of steps.*

Lemma 178 *For any model $\mathfrak{M}$ of PA on which $\mathfrak{H}$ is undefined $\mathfrak{M} \models\sim K$*

Proof. We have that $PA \vdash (K \equiv {\sim}Tr(\overline{\ulcorner K \urcorner}))$, so $\mathfrak{M} \vDash (K \equiv {\sim}Tr(\overline{\ulcorner K \urcorner}))$. In a model $\mathfrak{M}$ on which $\mathfrak{H}$ is not defined, for every X, $\mathfrak{M} \vDash Tr(\overline{\ulcorner X \urcorner})$, so in particular $\mathfrak{M} \vDash Tr(\overline{\ulcorner K \urcorner})$. Then by the truth of the diagonal equivalence in $\mathfrak{M}$, $\mathfrak{M} \vDash {\sim}K$. ▲

Remark. We might wonder why a version of the argument for this Lemma doesn't also establish that $\mathfrak{M} \vDash K$? From the fact that in a model $\mathfrak{M}$ on which $\mathfrak{H}$ is not defined, for every X, $\mathfrak{M} \vDash Tr(\overline{\ulcorner X \urcorner})$, so for such an $\mathfrak{M}$, $\mathfrak{M} \vDash Tr(\sim K)$. But to conclude from this and the truth of the Diagonal Equivalence in $\mathfrak{M}$ that $\mathfrak{M} \vDash K$ requires that $\mathfrak{M} \vDash (Tr(\sim K) \equiv \sim Tr(\overline{\ulcorner K \urcorner}))$, and this does not hold in an arbitrary model, in particular it does not hold in a model in which a sentence equivalent to $Pr(\overline{\ulcorner 0 = 0' \urcorner})$ is in the extension of $Tr(v_1)$.

Lemma 179 *If $\mathfrak{H}$ terminates in its successive generation of models starting with $\mathfrak{N}$ after an odd number of steps then K is true, and if after an even number of steps then K is false.*

Lemma 180 *Whether the sequence stops after an even or an odd number n of steps depends on the order in which the formulae are enumerated in the process of generating a complete consistent extension. For example, if the sequence begins $Con(PA)$, $\sim Con(PA + Con(PA))$ then $n = 1$ and so K is true. If the sequence begins $Con(PA), Con(PA + Con(PA)), \sim Con(PA + Con(PA) + Con(PA + Con(PA)))$, then $n = 2$ and K is false.*

Let $Tr(x)$ be the Δ^0_2-predicate which determines the model of a consistent theory S. Given a model $\mathfrak{M}$ of S, the model $\mathfrak{M}^*$ determined by $Tr(x)$ cannot be isomorphic to $\mathfrak{M}$ (by Tarski's Theorem on undefinability of truth). Given a model $\mathfrak{M}$ of S, $Tr(x)$ defines another model $\mathfrak{M}^*$ of S in terms of $\mathfrak{M}$. In [?] Kreisel noted that the sequences of models determined by this construction is finite. Manevitz and Stavi [12] (p.146) note that the length of this sequence depends on the order in which the formulae of S are enumerated in the Henkin proof of completeness claiming (without proof, left as an exercise to the reader) that if the enumeration begins with the formula $Con^1(PA), Con^2(PA), \ldots, Con^{n-1}(PA), \sim Con^n(PA)$, where $Con^1(PA) = Con(PA)$ and $Con^{k+1}(PA) = Con(PA + Con^k(PA))$ that the length of the sequence of models starting with $\mathbb{N}$, the standard model of PA, is n.

We are dealing with the Henkin construction starting from the recursive set of axioms of PA. The rule of construction is

$T_{n+1} = T_n \cup \{\phi_n\}$ if $T_n \cup \{\phi_n\}$ is consistent,

$T_{n+1} = T_n \cup \{\sim \phi_n\}$ if $T_n \cup \{\phi_n\}$ is inconsistent,

This construction is from the assumption that the set of formulae S with which the sequence starts is consistent.

This condition is expressed uniformly by a Δ^0_2-predicate. *What set of (Gödel numbers of) formulae it picks out depends on the model of PA in which the formula is interpreted.*

Let us consider first what set of 'true' formulae is picked out if the Δ_2^0 predicate expresses this construction in $\mathfrak{N}$, the standard model. Each of these n formulae is picked out, as shown by the following arguments. $PA \cup \{Con(PA)\}$ is consistent since $PA \nvdash \sim Con(PA)$. This latter fact can be established proof theoretically on the assumption that PA is ω-consistent and $PA \vdash (Con(PA) \supset G)$ (where G is the Gödel sentence for PA).

It can also be established model theoretically from the fact that PA is consistent (has a model, namely $\mathbb{N}$), so that $\sim Con(PA)$ is false in $\mathbb{N}$, that the Δ_2-predicate picks out $Con(PA)$ as the first formula to add to PA, so $T_1 = PA \cup \{Con(PA)\}$. Since $\mathbb{N}$ is a model for T_1, $Con(PA \cup Con(PA))$ is true, so $T_1 \nvdash \sim Con(PA \cup Con(PA))$, i.e. $T_1 \cup \{Con(PA \cup \{Con(PA\}\}$ is consistent, so $T_2 = T_1 \cup \{Con(PA \cup Con(PA)\})\}$. By the same argument we have that $T_{n-1} = T_{n-2} \cup \{Con^{n-1}(PA)\}$. By the Second Incompleteness Theorem applied to T_{n-1} we have $T_{n-1} \nvdash Con(T_{n-1})$ if T_{n-1} is consistent. So if T_{n-1} is consistent $T_{n-1} \cup \{\sim Con(T_{n-1})\}$ is consistent. $T_{n-1} = PA \cup \{Con(PA)\} \cup Con(PA \cup \{Con(PA)\}) \cup \cdots \cup \{Con(PA \cup \{Con^{n-2}(PA)\})\}$ so $\sim Con(T_{n-1}) = \sim Con^n(PA)$. So the n^{th} formula of the sequence is $\sim Con^n(PA)$.

Let $\mathfrak{N}_1$ be the model of PA determined by the Δ_2 truth predicate of the arthmetized Henkin proof of completeness interpreted in $\mathbb{N}$. I now want to consider which of the first n formulae are true in $\mathfrak{N}_2$, the model determined by $Tr(x)$ *interpreted in* $\mathfrak{N}_1$.

Lemma 181 $PA, \sim Con(PA + \phi), Con(PA) \models Pr_{PA}(\sim \phi)$

Proof

$$\sim Con(PA + \phi) \equiv_{df} \exists x Prov_{PA \cup \{\phi\}}(x, \overline{\ulcorner 0 = 1 \urcorner})$$

By arithmetized Deduction Theorem (which we can use since we are working in PA),

$$\exists x Prov_{PA \cup \{\phi\}}(x, \overline{\ulcorner 0 = 1 \urcorner}) \equiv \exists x Prov_{PA}(x, \overline{\ulcorner (\phi \supset 0 = 1) \urcorner})$$

$PA \vdash (\sim \phi \equiv (\phi \supset 0 = 1))$, so by arithmetized syntax

$$PA \vdash \exists x Prov_{PA}(x, \overline{\ulcorner (\phi \supset 0 = 1) \urcorner}) \equiv \exists x Prov_{PA}(x, \overline{\ulcorner \sim \phi \urcorner})$$

■

Remark: Let S be a first-order extension of Robinson's system Q, i.e. a system in which the syntax of the system can be arithmetized, such that $S \vdash \phi$ and $S \nvdash \sim Pr_S(\overline{\ulcorner \sim \phi \urcorner})$. Then $S' = S \cup \{Pr_S(\overline{\ulcorner \sim \phi \urcorner})\}$ is consistent and $S' \vdash \phi$ and $S' \vdash Pr_S(\overline{\ulcorner \sim \phi \urcorner})$.

An example of this situation is $PA' = PA \cup \{\exists x Prov_{PA}(x, \overline{\ulcorner 0 = 1 \urcorner})\}$. Given the Σ_0 predicate $Prov_{PA}(x, y)$ we can immediately modify it to obtain $Prov_{PA'}(x, y)$ (add a disjunction to the condition for being an axiom) and we have

$$PA' \vdash (\exists x Prov_{PA}(x, y) \supset \exists x Prov_{PA'}(x, y))$$

Therefore $PA' \vdash \sim 0 = 1$. Then we have a consistent system PA' such that $PA' \vdash \sim 0 = 1$ and $PA' \vdash \exists x Prov_{PA'}(x, \overline{\ulcorner 0 = 1 \urcorner})$.

We have that $\mathfrak{N}_1 \models \sim Con^n(PA) \equiv_{df} \sim Con(PA + Con^{n-1}(PA))$. Also $\mathfrak{N}_1 \models PA$. Hence by the lemma $\mathfrak{N}_1 \models Pr_PA(\overline{\ulcorner \sim Con^{n-1}(PA) \urcorner})$. Then with $Tr(x)$ interpreted in $\mathfrak{N}_1$, $T_{n-1} \cup \{\phi_{n-1}\}$ is *inconsistent* (as expressed in arithmetized syntax). So in $\mathfrak{N}_2$ the $(n-1)^{th}$ formula is $\sim Con^{n-1}(PA)$. By the absoluteness of Σ_1-formulae the n^{th} formula is still $\sim Con^n(PA)$.

Claim: $\mathfrak{N}_2 \models Con(PA)$, $\mathfrak{N}_2 \models Con(PA + Con(PA))$, $\ldots$, $\mathfrak{N}_2 \models Con^{n-2}(PA)$.

How do we know this? Let us look at the case where $n = 3$.

The initial segment of the enumeration of formulae is

$$\begin{array}{ccc} \phi_0 & \phi_1 & \phi_2 \\ Con(PA) & Con(PA + Con(PA)) & \sim Con(PA + Con(PA + Con(PA))) \end{array}$$

How do we know that $\mathfrak{N}_2 \models Con(PA)$ (while $\mathfrak{N}_2 \models \sim Con(PA+Con(PA))$)? Well, $\mathfrak{N}_1 \models Con(PA + Con(PA))$. Therefore *in* $\mathfrak{N}_1$ the condition holds by which $Tr(x)$ chooses $Con(PA)$ rather than $\sim Con(PA)$.

From $\mathfrak{N}_2 \models \sim Con(PA + Con(PA))$ we have $\mathfrak{N}_3 \models \sim Con(PA)$ by lemma 181.

The process of defining a new model in this an existing model (actually an end-extension) *comes to an end at this point* from the lemma: $\sim Con(PA) \equiv \exists x Prov_PA(x, \overline{\ulcorner 0 = 1 \urcorner})$ So in the application of this process to $\mathfrak{N}_3$, the formula $0 = 1$ is picked. But also $\exists x Prov_PA(x, \overline{\ulcorner \sim 0 = 1 \urcorner})$ (since $PA \vdash \sim 0 = 1$). So $\sim 0 = 1$ is also picked. Therefore there is no such model.

Now we need to look at how the parity of this sequence of models determines the truth value of the diagonal sentence for $Tr(x)$.

The diagonal sentence K is such that $PA \vdash K \equiv \sim Tr(\overline{\ulcorner K \urcorner})$.

Lemma 182 *([12] 1.3, p.145) If ϕ is an alternating sentence for $\mathbb{O}$ and $d_{\mathbb{O}}(M) < \infty$, then $M \models \phi$ iff $d_{\mathbb{O}}$ is an odd number.*

Proof Clearly this result must turn crucially on the 'normalization condition': if $M \notin dom(\mathbb{O})$ then $M \not\models \phi$ (otherwise if ϕ is an alternating sentence then $\sim \phi$ is also an alternating sentence). With this condition the proof is indeed, as Manevitz and Stavi say, 'immediate from the definitions'. The obvious point is that, by definition, the last model in the sequence of models $M, \mathbb{O}(M), \mathbb{O}^2(M), \ldots, \mathbb{O}^n(M)$ is such that $\mathbb{O}(\mathbb{O}^n(M))$ is not defined, so by the normalization condition, in particular $\mathbb{O}^n(M) \not\models \phi$.

By the alternating truth value condition on ϕ, $M \models \phi$ iff n is odd.

■

For this lemma to be relevant to determine the truth value of ϕ we must show that condition (ii) holds for ϕ, i.e. if $M \notin dom(\mathbb{O})$ then $M \not\models \phi$, i.e. we must break the symmetry between ϕ and $\sim \phi$.

Lemma 183 *$M \notin dom(\mathbb{O})$ iff $M \models \sim Con(PA)$.*

Proof (i) Assume $M \models \sim Con(PA)$, $\sim Con(PA) \equiv \exists x Prov_P A(x, \overline{\ulcorner 0 = 1 \urcorner})$.

In this situation, for each n '$T_n \cup \{\phi_n\}$ is inconsistent' holds in the arithmetized syntax of PA, so for each n $T_{n+1} = T_n \cup \{\sim \phi_n\}$. So for every formula ϕ (so in particular for some one such formula) $Tr(\sim \phi)$ and $Tr(\sim\sim \phi)$. So $Tr(x)$ does not determine a model. ▲

How do we know that the operator, as a mapping from models of PA to models of PA, is single-valued, i.e. is a model determined by A's complete diagram? The answer is that it is not, immediately by the Upward Löwenheim-Skolem Theorem. But it does not hold even if we restrict ourselves to countable models of PA, since by compactness the complete diagram of $\mathbb{N}$ does not determine $\mathbb{N}$. For the results of concern to us here, that this operator mapping models of PA to models of PA is not univalent doesn't matter. We are not really dealing with models but with classes of elementarily equivalent models, i.e. with complete consistent extensions of PA. What is crucial here is that this construction can only be iterated finitely many times.

[Perhaps the construction of successive models is single-valued if we take the domain of the model to be exactly the countable infinity of terms added to the given language in the Henkin construction. In this way we cut off the construction of an elementary extension. (This corresponds to Richard Epstein's idea for picking out the standard model of arithmetic by specifying that each element is denoted by a natural number.)]

(ii) The proof of the completeness theorem guarantees that so long as the original set of sentences (e.g. the axioms of PA) is consistent, the construction will always generate a complete consistent extension. The construction itself guarantees completeness. So if the above fails to be a complete consistent extension then it must be that the failure is a failure of consistency. At every stage what guarantees consistency is the consistency of the maximal theorem. So if the construction does not yield a consistent theory it must be that $\sim Con(PA)$ is true.

■

The following simple result plays a key role in cases where the truth value of K can be determined.

Lemma 184 *In any model M for which $\mathbb{O}(M)$ is not defined, $M \models \sim K$.*

We have seen that when $\mathbb{O}(M)$ is not defined, the Δ_2 construction gives $Tr(\sim \phi)$ for all ϕ. So in particular, for $\phi = \sim K$, we get $Tr(\sim\sim K)$.

Lemma 185 *If $PA \vdash (\phi \equiv \psi)$ then $PA \vdash (Tr(\phi) \equiv Tr(\psi)$.*

Proof By induction on the inductive definition of $Tr(x)$.

■

Corollary 186 $PA \vdash (Tr(\sim\sim K) \equiv Tr(K))$

Proof From the lemma and the fact that $PA \vdash (\sim\sim K \equiv K)$.

■

We have constructed K as the diagonal sentence of the predicate $\sim Tr(X)$, i.e. $PA \vdash (K \equiv \sim Tr(\overline{\ulcorner K \urcorner})$.

Then by contraposition and $\sim\sim$-elimination, $PA \vdash (\sim K \equiv Tr(\overline{\ulcorner K \urcorner})$. By the above $M \models Tr(\overline{\ulcorner K \urcorner})$. Since M is a model of PA, $M \models \sim K$.

By the alternation of the truth value of K in going from $\mathfrak{N}_i$ to $\mathbb{O}(\mathfrak{N}_i)$ and the fact that K is false in the final model of the sequence, which in being final is one such that $\mathbb{O}$ is not defined on it, we have that $\mathbb{N} \models K$ iff the process has generated an odd number of models beyond $\mathbb{N}$, and $\mathbb{N} \models \sim K$ iff the process has generated an even number of models beyond $\mathbb{N}$.

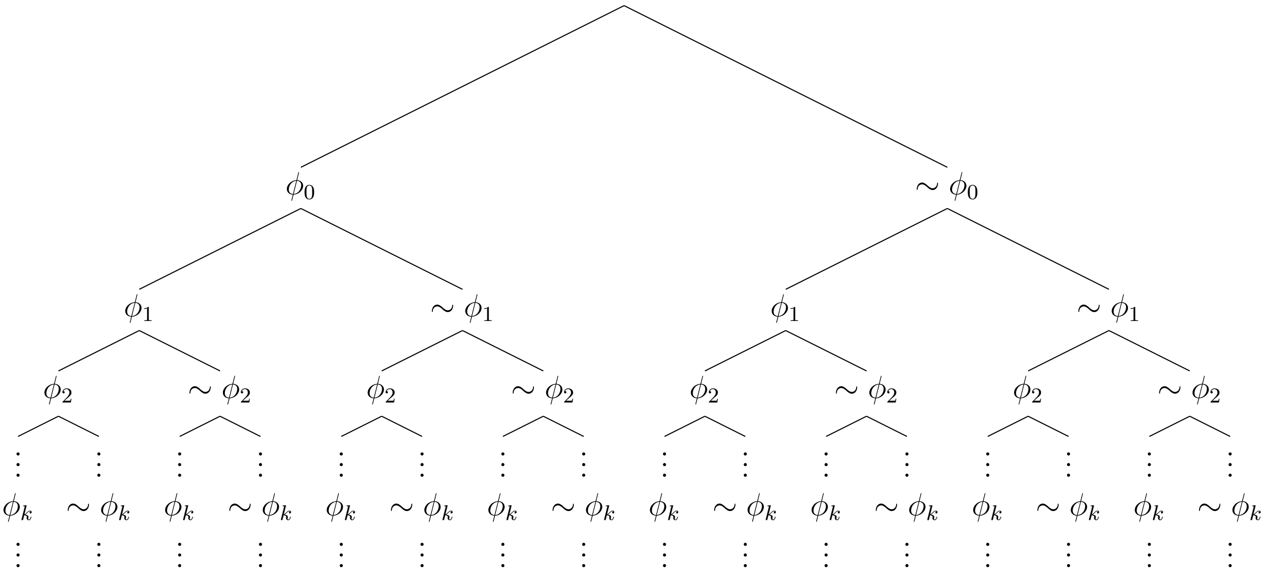
In the above we have seen that if the enumeration of the formulae of PA begins with $Con(PA), \ldots, Con^{n-1}(PA), \sim Con^n(PA)$ that the Henkin completeness theorem operator generates a sequence of (non-standard) models of length n, starting from the standard model $\mathbb{N}$. I want now to go through the proof that the operator generates a finite sequence of (non-standard) models *under every enumeration* of the formulae of PA. This result is stated (without being packaged as a Lemma or Theorem) by Manevitz and Stavi [**?**] p.146 lns 22-24. The result is attributed to Kreisel [**?**] and Smorynski's Handbook article [**?**] is cited for the published proof, 6.2.4(pp.862-3).]

This proof by Smorynski is not quite ideal as an exposition of the proof of this result since, while this is what is in effect being proved, it is not being presented as that, but rather as a model-theoretic proof of the Second Incompleteness Theorem.

Aritmetization of the Completeness Theorem for first order logic – for specificity we take the Henkin proof of Completeness – results in a Δ^0_2 formula which is a truth predicate for a model of the original consistent theory. One way to think of this construction is as follows: given a model of a (consistent) theory, we can, in this the original model, define a (new) model. That the defined model is a *new* model is established by diagonalization – Tarski's undefinability of truth theorems.

Let K be the diagonal sentence for the formula with one free variable $\sim Tr(X)$, i.e. $PA \vdash (K \equiv \sim Tr(\overline{\ulcorner K \urcorner}))$. *Thus the truth value of K in the given model must be opposite to its truth value in the model whose truth definition is given by $Tr(X)$.*

Kreisel's observation in [**?**] (sparked off, by his account there, by some work of Harvey Friedman) is that the iteration of this process of generating a new model of the consistent theory from a given model of the theory must come to an end after a finite number of steps. The basis of this result is to analyze an initial segment of the choice of truth values for formulae of PA made by $Tr(X)$ in the enumeration of formulas. Let $k = \ulcorner K \urcorner$. We consider the sequence of sentences (formulae without free variables) $\phi_0, \phi_1, \ldots, \phi_k = K$. A truth definition for PA (or whatever theory we started from) is an infinite path through the binary truth tree:



We have seen that there are continuum many complete consistent extensions of PA [only one of which is ω-consistent]. Note that there are continuum many paths through this binary fan, in total. I presume that continuum many of these paths are *in*consistent, so that what we have here is an example of the cardinal arithmetic fact that $2^{\aleph_0} + 2^{\aleph_0} = 2^{\aleph_0}$. Thus each of the k-length initial segments of this ω-length binary tree is the initial segment of infinitely (indeed continuum) many complete consistent extensions of PA. However, in the case of the complete consistent extensions defined by $Tr(X)$ Kreisel observed that on the construction by which the defined path is the left-most consistent path (which is the effect of the specification on the construction that $T_{n+1} = T_n \cup \{\phi_n\}$ if $T_n \cup \{\phi_n\}$ is consistent, $T_{n+1} = T_n \cup \{\sim \phi_n\}$ if $T_n \cup \{\phi_n\}$ is inconsistent), each successive k-length initial segment must lie strictly to the right of the previous one. I will give the proof of this fact below but note first that this result shows that this process stops after a finite number of steps.

But before noting that, note that the above remark that there are continuum many complete consistent extensions of PA is *in* the standard model, though perhaps that does not make any difference since the *standard part* of the full binary tree is absolute – i.e. it is the same in all models of PA.

The above fact shows that the sequence of complete consistent theories is finite. The fact that the k-length initial segment of each successive model (complete consistent theory) lies strictly to the right of the previous one (and the fact that lying strictly to the right is a transitive relation) means that the initial k segments of each model is pairwise distinct from all the previous initial k-segments. But there are only 2^{k+1}, i.e. finitely, many possible initial k-sequences. So there are only finitely many models (complete consistent theories) generated in this sequence.

Lemma 187 *Each successive k-length initial segment lies strictly to the right of the preceding k-initial segment.*

Proof (Using Smorynski's notation.) Given $\mathfrak{N}_i$, let $\phi^i =< \phi_0^{e_{0,i}}, \phi_1^{e_{1,i}}, \ldots, \phi_k^{e_{k,i}} >$ denote the portion of the path used in constructing $\mathfrak{N}_{i+1}$, where $e_{j,i} \in \{0,1\}$ and $\phi_j^0 =_{df} \phi_j, \phi_j^1 =_{df} \sim \phi_j$. From the definition of $Tr(x)$ via the proof of the completeness theorem ϕ^i is the left-most consistent path, with $Pr_{PA}(x)$ interpreted in $\mathfrak{N}_i$.

Bibliography

[1] Rasmus Blanck, “On Rosser sentences and proof predicates”, Department of Philosophy, University of Göteborg, 2006.

[2] George Boolos, *The Logic of Provability*, Cambridge University Press, 1993.

[3] William B. Ewald, *From Kant to Hilbert: a Source Book in the Foundations of Mathematics* Volume 2, Oxford University Press, 1996.

[4] Kurt Gödel, “Die Vollständigkeit der Axiome des logischen Funtionenkalküls, *Monatshefte für Mathematik und Physik* 37 (1930), pp. 349-360; English translation by Stefan Bauer-Mengelberg, “The completeness of the axioms of the functional calculus of logic”, Solomon Feferman *et al* eds, *Kurt Gödel Collected Works Volume I: Publications 1929-1936*, Oxford University Press, 1986, pp. 102-123.

[5] Kurt Gödel, “Über formal unentscheidbare Sätze der *Principia mathematica* und verwandter Systeme I”, *Monatshefte für Mathematik und Physik* 38 (1931), pp. 173-198; English translation by Jean van Heijenoort “On formally undecidable propostions of *Principia mathemataica* and related systems I”, Solomon Feferman *et al* eds, *Kurt Gödel Collected Works Volume I: Publications 1929-1936*, Oxford University Press, 1986, pp. 144-195.

[6] David Hilbert, “Mathematische Probleme. Vortrag, gehalten auf dem internationalen Mathematiker-Kongress zu Paris, 1900”, *Nachrichten der Königlichen Gesellschaft der Wissenschaten zu Göttingen*’, pp. 253-97; English translation by Mary Winston Newson, “Mathematical problems: Lecture delivered before the International Congress of Mathematiacian at Paris in 1900”, *Bulletin of the American Mathematical Society* 8 (1902), pp. 437-479; partially reprinted in Ewald.

[7] David Hilbert, “Axiomatische Denken”, *Mathematische Annalen* 78 (1918), pp. 405-415; English translation by William B. Ewald, “Axiomatic thought”,

William B. Ewald (ed,), *From Kant to Hlbert: A Source Book in the Foundations of Mathematics* Volume 2, Oxford University Press, pp. 1107-1115.

[8] David Hilbert, "On the infinite" (1926); English translation in Jean van Heijenoort (ed), (1927), p. 471.

[9] David Hilbert, "Die Grundlagen der Mathematik", *Abhandlungen aus dem mathematischen Seminar der Hamburgeshcen Universität* 6 (1928), English translation by Stephan Bauer-Mengelberg, 'The foundations of mathematics" in Jean van Heijenoort (ed.) *From Frege to Gödel; A Source Book in Mathematical Logic 1879-1931*, Harvard University Press, 1967.

[10] Donald Kalish and Richard Montague, "On Tarski's formalization of predicate logic with identity", *Archiv für mathematische Logik und Grundlagenforschung* 7 (1965), pp. 81-101.

[11] Georg Kreisel, "A refinement of ω-consistency" (Abstract), *The Journal of Symbolic Logic* 22 (1957), pp. 108-109.

[12] Larry Manevitz and Jonathan Stavi, "Δ_2^0 operators and alternating sentences in arithmetic", *The Journal of Symbolic Logic* 45 (1980), pp. 144-154.

[13] Y. Yu. Shavrukov, "On Rosser's provability predicate", *Zeitschrift für Logk und Grundlagen der Mathematik* 37 (1991), pp. 317-330.

[14] Peter Smith, "Rosser's Theorem", *Logic Matters blog*, logicmatters.net, 26 June 2009.

[15] Craig Smorynski, "The incompleteness theorems", Jon Barwise (ed.), *Handbook of Mathematical Logic*, Horth-Holland Publishing Company, 1977, pp. 821-865.

[16] Raymond M. Smullyan, *Gödel's Incompleteness Theorems*, Oxford University Press, 1992.