SCIENTIFIC REVIEW (1996), Number 16, pp. 75-82

GÖDEL'S THEOREMS

Žarko MIJAJLOVIĆ

Faculty of Mathematics, University of Belgrade
Studentski trg 16, 11000 Belgrade, Yugoslavia

Abstract. This paper gives an overview of Gödel's incompleteness theorems.

Keywords: mathematical logic, Gödel's theorems, incompleteness, Hilbert's program, formal arithmetic.

1. Hilbert's program

With the progress of mathematics some mathematicians have been interested in the development of a formal system which would yield all true mathematical statements and only them. Special places in these efforts belong to Gottfried Leibniz (1646-1716) and David Hilbert (1862-1943).

One of Leibniz' favorite ideas was to build "an alphabet of human thought that makes it possible to deductively derive new ideas by means of definite rules for combining symbols"[1]. This idea is contained in his conception of *lingua characteristica*. Leibniz had in mind two aspects of this notion. The first one is associated with the linguistic idea of creating a *universal language*, suggested to Leibniz by the British philologist George Dalgarno who had tried to realize it[2]. The other aspect consists in the requirement of developing a universal language and symbolism which should not be used only for communicating thoughts between two men, but also to facilitate the very process of thought. Thus, Leibniz distinguished the linguistic and logical aspect of his universal characteristic. If taken as a logical project, the universal characteristic is a system of rigorously defined symbols that can be used in logic and other deductive sciences to denote simple elements of the object under the investigation. Some properties of the symbols are presupposed. They would have to be brief and not overlapping, and they would have to include the maximum information. Further, there would be an isomorphic

[1] *Historia et commendatio lingue characteristicae universalis, quae simul sit ars inveniendiet judicandi* (*Oewres*, Raspe, 1765).

[2] *Ars signorum vulgo character universalis et lingua philosophica*, London, 1661.

correspondence between symbols and the objects they denote: simple elements in logical derivations by letters; complex logical considerations by formulas; and sentences by equations. This would make it possible to obtain all logical consequences that necessarily follow from given hypothesis. Leibniz adopted Descartes thesis on mathematics as a universal science and stated it in terms of his logical studies. So he held that there exists a universal mathematics from which all of the mathematical sciences pull their principles and most general theorems. This merges mathematics with logic, and there is no mere formal analogy, nor parallelism between mathematics and logic, but an identity, or at least a partial identity. Leibniz not only made logic mathematical but mathematics logical. For example, he attempted[3] to give the concept of number in purely logical definition. However, his endeavor to include all of mathematics in formal logic failed. Even if we leave aside technical difficulties and the extreme complexity of the task, this failure in the light of Gödel results on inherent incompleteness of formalized arithmetic today is very comprehensible. So not all of Leibniz' logical program resisted the test of time. In particular the development of the contemporary mathematics and other sciences showed that the concept of universal characteristic is unrealizable. But his attempt at reducing all of meaningful human thought to a finite number of formal mathematical calculuses, and consequences of this notion such as Leibniz' try to confine all of meaningful mathematics to the narrow frame of formal logic attracted many mathematicians that came after Leibniz. Probably the most notable formulation and attempt of this type is *Hilbert's Entscheidungsproblem*.

Hilbert had long[4] been interested in the famous problems of mathematics. He held that the problems of mathematics can all ultimately be solved, as justifies his famed sentence "Wir müssen wissen, wir werden wissen". Also in 1928, in a small book with W. Ackerman, he formulated an explicit version of his *Entscheidungsproblem* for a certain specific formal system: the first order predicate calculus. In 1931, just as Hilbert stated that we must know, Gödel showed that Entscheidungsproblem is unsolvable. More explicitly, it could not be solved in a formal system as for example "Principia Mathematica" of Russel and Whitehead. A Church's paper from 1936 did the same for the specific formal system (predicate calculus) used by Hilbert. However, there are opinions that the problem for a given system can be solved by means of some other more powerful system. A result of this type have in fact been obtained by Gerhard Gentzen, an early leader in the branch of logic called *proof theory*.

If we want to understand better how Gödel's theorems originated we have to explain the so called Second Hilbert problem: The Compatibility of Arithmetical Axioms. In the beginning of the statement of the problem Hilbert says[5] "When

---

[3] *Specimen calculi universalis.*

[4] at least since 1900 when Hilbert delivered the famous lecture on twenty three mathematical problems before the Second International Congress of mathematicians at Paris in 1990.

[5] *Mathematical problems*, D. Hilbert, Göttinger Nachrichten, 1900, pp.253-297, and in the Archiv der Mathematik und Physik, 3d ser., 1, (1901), and English translation in Bulletin of AMS, 8, (1902), 437–479.

we are engaged in the investigating the foundation of a science, we must set up a system of axioms which contains an exact and complete description of the relations subsisting between elementary ideas of that science." In the same statement Hilbert designate the following problem as the most important among the numerous questions which can be asked with the regard to the axioms:[6]

To prove that they (arithmetical axioms) are not contradictory, that is, a finite number of logical steps based upon them can never lead to contradictory results.

The notion of the contradiction in the above statement had some deeper meaning for Hilbert then a mere impossibility of syntactical derivation of a contradiction from the system: "If contradictory attributes be assigned to a concept, I say, *that mathematically the concept does not exist.* So, for example, a real number whose square is -1 does not exist mathematically. But if it can be proved that the attributes assigned to the concept can never lead to a contradiction by the application of a finite number of logical processes, I say that the mathematical existence of the concept (for example, of a number or a function which satisfies certain conditions) is thereby proved." The consistency problem for arithmetic is hard since it is not quite clear how to approach it. Namely, Hilbert did not expect to prove the consistency of arithmetic by interpreting it into some other theory or by some infinite structure, but he wished a *direct* or a finitist proof. This is quite understandable, as arithmetic is the starting point of whole mathematics, so the question of the consistency of whole mathematics in this sense is reducible to the consistency of arithmetic. An acceptable way to do it would be, as Hilbert indicated by himself, to prove that the formulas which represents the axioms have a certain syntactical property, and that this property is inherited by the formal rules used for obtaining the theorems of the system. This presupposes a kind of a *formalization* of arithmetic. So we need to have a language, effectively[7] given set of formulas, set of axioms, inference rules, proofs and other syntactical notions of the system. The most important formalization of arithmetic is *formal Peano aritmetic*, or simply formal arithmetic, formulated as a certain theory in the first order logic. These ideas guided Hilbert's further analysis of the problem, during the first decades of the century, and led to *Hilbert's program.* So the main goal of his program is to eliminate the infinity from mathematics, or at least to show that it is possible to establish mathematics without referring to infinity. One could try to find here the analogy to Weierstrass $\varepsilon - \delta$ analysis which eliminated actual infinitesimals from the Leibniz analysis. However, we should mention that Hilbert's intention was not to prohibit the use of infinity in mathematics, but to justify it. The realization of the program should have been shown by finitary methods forming a proper part of

[6]It should be said that the original presentation of the second problem had serious ambiguities in contrast to the most of other problems. This can be explained by the fact that it dealt with notions that had not been previously analyzed. So the axioms that were described in the statement of the problem are the axioms of the arithmetic of the real continuum, not the positive integers. Later in the twenties they are referred to as the axiom of formal arithmetic.

[7]in modern terms *recursive.*

formal arithmetic. Namely, unbounded quantifiers are used in arithmetic, so the sentences containing them refer to potentially infinite domains. Therefore in the modern reinterpretation of Hilbert's program the finitist part would be the formal arithmetic with only bounded quantifiers, or primitive recursive arithmetic. So the problem of the possibility of reduction of (all) "infinitary" statements to finitary statements is equivalent to the conservativity[8] of formal arithmetic with respect to the primitive recursive arithmetic. The consistency problem is treated in the same way as the conservativity problem, so the finitary proof of the consistency of formal arithmetic should be found. As a response to the program, Gödel obtained his incompleteness theorems and showed that neither task of the program is achievable.

The first Gödel incompleteness theorem states that there is no (finitist) formal system which would codify all theorems of intuitive arithmetic (all true statements on natural numbers). Namely, for every formal system[9] which includes the formal system of Peano arithmetic, contains a sentence φ so that neither φ, nor $\neg\varphi$ are provable in the system. So there is no formal system which would yield all the true statements on arithmetics[10]. It should be also observed that this theorem not only denies the possibility of complete formalization of mathematics, but also puts the question mark on the equivalence of the conservativity and the consistency.

The second Gödel's incompleteness theorem states that the consistency of formal system containing formal arithmetic cannot be proved by means of the system alone. Namely the sentence of the formal system which codify the consistency of the system is unprovable in the system. Therefore it is not possible to prove the consistency of formal arithmetic in the frame of formal arithmetic, so neither of the stronger formal systems.

2. Formal arithmetic

The aim of this section is to clarify some points in the previous section from the mathematical point of view, and to supply some technical details. Namely, we have to describe the appropriate formal system – the right setting for Gödel incompleteness results, so we shall outline now a formalization of arithmetic. Formal arithmetic is given by an *effective* set of symbols and formulas, while the set of theorems of this system is characterized by effectively given sets of axioms and rules of inference. The theory of algorithms (the theory of effective computability) gives the precise meaning to the word *effective* above. From now on we shall abbreviate the system of formal arithmetic by PA.

[8] this means: if a sentence φ is deducible from formal arithmetic, then it deducible already in the primitive recursive arithmetic. Obviously, formal arithmetic contains the primitive recursive arithmetic.

[9] which is effectively, i.e. recursively, given. The contemporary meaning of the notion of "formal system" is somewhat broader, so this remark is necessary.

[10] In the classical mathematics, for every arithmetical sentence φ, one of sentences φ, $\neg\varphi$ is true.

2.1 Language of PA

We distinguish the following types of symbols:

Logical symbols: $\Rightarrow, \wedge, \vee, \neg, \forall, \exists, =$.

Variables: $v_0, v_1, v_2, \ldots$

Auxiliary symbols: (,).

Signs of arithmetical operations.: $+, \cdot, '$ *Constant symbol*: 0.

Starting from these basic symbols we can build other syntactical objects: *numerals, terms, formulas and sentences.*

Definitions of terms, formulas and sentences are the usual ones as given in standard logic courses. Numerals are defined in the following way:

$\bar{1} = 0'$, $\bar{2} = (0')'$, $\bar{3} = ((0')')'$, ...

So the numerals represent natural numbers.

2.2 Arithmetical axioms

Theory PA is formulated in the first-order predicate calculus. Besides the logical axioms we have the following arithmetical axioms:

$$x' \neq 0, \quad x' = y' \Rightarrow x = y$$

$$x + 0 = x, \quad x + y' = (x + y)',$$

$$x \cdot 0 = 0, \quad x \cdot y' = (x \cdot y) + x$$

$$(\varphi(0) \wedge \forall x(\varphi(x) \Rightarrow \varphi(x'))) \Rightarrow \forall x \varphi(x) \quad \text{(induction scheme)}$$

φ is an arbitrary formula of PA

Now we can prove various theorems in PA, for example that the numerals have all expected properties, as $\overline{m +_N n} = \bar{m} + \bar{n}$, where m, n are non-negative integers, and $+_N$ is the operation of addition in the structure of natural numbers. As an illustration we prove that $\bar{2} + \bar{2} = \bar{4}$[11]:

$\bar{2} + \bar{2} = (0')' + (0')' = ((0')' + 0')' = (((0')' + 0)')' = (((0')')')' = \bar{4}$

2.3. Coding

A coding of a domain S of certain objects is an explicit and effective 1–1 mapping $k\colon S \to N$. Here N denotes the set of natural numbers. We say that an object $s \in S$ is *coded* by the natural number $k(s)$. From the injectivity of k, it follows that to each code n corresponds exactly one object $s \in S$ such that $n = k(s)$. Also, from this definition of coding, we see at once that the domain S is at most countable. Here are some examples of codings.

Example 2.3.1 Cantor's coding function $\langle m, n \rangle : N \times N \to N$ is defined by

$$\langle m, n \rangle = (m + n + 1)(m + n)/2 + n.$$

[11] This "logical" proof of $2 + 2 = 4$ belongs to Leibniz, see [Sty].

So the first few values of this function are:

$\langle 0,0\rangle = 0, \langle 1,0\rangle = 1, \langle 0,1\rangle = 2, \langle 2,0\rangle = 3, \langle 1,1\rangle = 4, \langle 0,2\rangle = 5, \langle 3,0\rangle = 6, \ldots$

Example 2.3.2 Gödel's coding function of finite sequences is defined in the following way. Let $p_1, p_2, p_3, \ldots$ be the sequence of all primes $2, 3, 5, \ldots$. Let s be a hereditary[12] finite sequence of finite objects $s_1, s_2, \ldots, s_m$. Then the coding function k is defined recursively by

$$k(s) = p_1^{k(s_1)+1} p_2^{k(s_2)+1} \ldots p_m^{k(s_m)+1}$$

All above mentioned syntactical objects are hereditary finite. If we are given a coding of starting symbols, i.e. of symbols of the language of PA, then we are given a coding of terms, formulas and sentences. One possible coding of the symbols of the language of PA is:

0	$'$	$+$	$\cdot$	$\Rightarrow$	$\wedge$	$\vee$	$\neg$	$\forall$	$\exists$	$=$	$($	$)$	v_i
1	2	3	4	5	6	7	8	9	10	11	12	13	$20+i$

Here v_i is a variable, and $i \in \{0, 1, 2, \ldots\}$. Then the code of the formula $\varphi = \forall v_1 \exists v_0((v_0 \cdot v_1') = v_2)$ is:

$k(\varphi) = 2^9 3^{21} 5^{10} 7^{20} 11^{12} 13^{12} 17^{20} 19^4 23^{21} 29^2 31^{13} 37^{11} 41^{22} 43^{13}$.

Not only single syntactical objects can be represented in PA, but sets of syntactical objects too. Namely, we can introduce special formulas of PA which stand for certain syntactical constructions needed in proofs of Gödel's theorems. These formulas are:

$\mathrm{Term}(x)$	iff	x	is a code of an arithmetical term.
$\mathrm{For}(x)$	iff	x	is a code of an arithmetical formula.
$\mathrm{Ded}(x, y)$	iff	x	is a code of a proof in PA of a formula whose code is y.
$\mathrm{Pr}(x)$	iff	x	is a code of an arithmetical theorem, so $(\mathrm{Pr}(x) \Leftrightarrow \exists y \mathrm{Ded}(y, x))$
$\mathrm{Con(PA)}$	iff		there is no number who is the code of the proof of the contradiction. So, $\mathrm{Con(PA)} \Leftrightarrow \neg\mathrm{Pr}(k(0=1))$

If y is a code of a finite sequence $\mathcal{Y}$, then $\bar{y}$ is the length of $\mathcal{Y}$ (i.e. the number of distinct primes in the prime decomposition of y), while $(y)_i$ denotes the code of i-th member of $\mathcal{Y}$ (i.e. $(y)_i$ is the exponent of i-th prime in the prime decomposition of y). Then, for example, formula $\mathrm{Term}(x)$ looks like:

$$\exists y((y)_{\bar{y}} = x \wedge (\forall k \leq \bar{y})((y)_k = 2^1 \vee (\exists z \leq (y)_k)(y)_k = 2^{20+z} \vee$$
$$(\exists i \leq k)(\overline{(y)_k} = \overline{(y)_i} + \bar{1} \wedge ((y)_k)_{\overline{(y)_i}+1} = 2 \wedge (\forall z \leq \overline{(y)_i})(((y)_k)_z = ((y)_i)_z) \vee \ldots).$$

[12] this means not only s is finite but members $s_1, s_2, \ldots, s_m$ are finite too, and members of each s_i are finite also, and etc.

2.4. Gödelization: coding in PA

The previous coding of metamathematical notions (syntactical objects) is done in the structure of intuitive natural numbers. We use various properties of this structure to prove theorems about the coding function k, for example the Unique Factorization Theorem to prove that k is 1-1. On the other side, notions we have freely used, as the notion of a prime number, and exponential function are not even defined in PA. So, up to now we *did not* represent metamathematics in PA, but in its intuitive counterpart. Gödel proved that this task, coding of all the mentioned syntactical objects, can be done in PA. This process is called *gödelization*, and in most parts mimics coding in N. Main steps in construction are as follows:

- Values of the coding function are *numerals*. The code of a formula φ is denoted by $\lceil\varphi\rceil$.
- The exponential function is representable in PA. So there is a formula $\varphi(x, y, z)$ so that for all $m, n, k \in N$, $\quad m = n^k$ iff PA$\vdash \varphi(\bar{m}, \bar{n}, \bar{k})$.

If instead of $\varphi(x, y, z)$ is written $x = y^z$, then the usual properties of the exponential function y^z are provable in PA: $x^{y+z} = x^y x^z$, $(x^y)^z = x^{yz}$; then Newton binomial formula, etc.

- Finite sequences are representable in PA. There is an arithmetical function $f(y, x)$, which is denoted by $(y)_x$, representable in PA so that for all $k \in N$:

  PA$\vdash (\forall y_1 \ldots y_k)(\exists! y)((y)_1 = y_1 \wedge \ldots (y)_k = y_k)$.

Gödel proved these theorems by use of formalized version of Chinese Reminder Theorem which says that certain finite systems of congruence equations have solution. Now it is easy to define in PA

- The sequence of prime numbers.
- A coding function of finite sequences.
- Formal replicas of metamathematical notions $\mathrm{Term}(x), \mathrm{For}(x), \mathrm{Ded}(x, y), \mathrm{Pr}(x)$, Con(PA).

Now the formal arithmetic "can speak" about itself. For example, the Second Gödel theorem looks like:

not PA$\vdash$ Con(PA), $\qquad N \models$ Con(PA).

Therefore, Con(PA) would be an example of a true sentence (i.e. true in the intuitive structure of natural numbers), but not provable in PA. So *not* PA$\vdash$ $\neg$Con(PA), as all theorems of formal arithmetic are true (or at least if assume that PA is, in fact, consistent). Also, Con(PA) serves as an example of *undecidable* sentence: neither it, nor it's negation are provable in PA. Exactly this is the statement of the *First Gödel's theorem*. Proofs of Gödel's theorems are based on the following lemma:

Diagonalization Lemma Let $\psi(x)$ be a formula of PA which has only x as free. Then there is a sentence φ of PA such that $\vdash \varphi \Leftrightarrow \psi(\lceil\varphi\rceil)$.

If $\mathcal{F}$ is any *consistent* formal system extending PA, then all above arguments can be applied as in the case of PA, so Gödel's theorems hold for $\mathcal{F}$ also. Important examples of this kind are formalized analysis (which is identified with the second order arithmetic), and formalized set theory (as ZFC for example). Some other consequences, or related results to Gödel's theorems are:

1. (Gödel-Rosser) PA is undecidable theory. There is no effective procedure which would decide for any given arithmetical sentence if it is a theorem of PA or not.
2. (T. Skolem) PA is not a categorical theory. There are non-isomorphic structures satisfying all the axioms of PA. In fact, for any infinite cardinal number k, there are 2^k pair-wise non-isomorphic structures of the cardinality k.
3. PA has the continuum many complete extensions.

Bibliography

[Göd] *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter System I*, K. Gödel, Monatschefte Math. Phys., **38**, 173-98.

[Kre] *What we have learnt from Hilbert 's Second problem*, G. Kreisel, in *Mathematical developments arising from Hilbert's problems*, ed. F.E. Browder, AMS, Providence, 1976.

[Mar] *Gödel's results in connection with the completeness and the consistency of formal system* (in Serbian), Z. Marković, **3**, Dijalektika, Beograd, 1972, 25-37.

[MMD] *Hilbert's problems and logic* (in Serbian), Ž. Mijajlović, Z. Marković, K. Došen, Zavod za udžb. i nastavna sredstva, Beograd, 1986.

[Sty] *History of mathematical logic from Leibniz to Peano*, N.I. Styzhkin, (engl. transl.) M.I.T. Press, Cambridge, 1969.

This paper was presented to the Serbian Scientific Society on November 2, 1995.