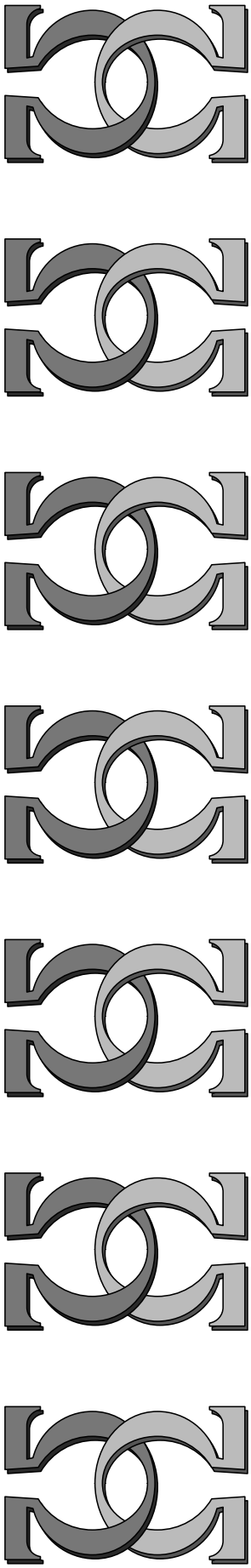

CDMTCS
Research
Report
Series

Proving as a Computable Procedure

C.S. Calude and S. Rudeanu
University of Auckland, NZ
University of Bucharest, Romania

CDMTCS-251
October 2004

Centre for Discrete Mathematics and
Theoretical Computer Science

Fundamenta Informaticae xx (2005) 1–10
IOS Press

Proving as a Computable Procedure

Cristian S. Calude
Department of Computer Science
The University of Auckland
Private Bag 92019, Auckland, New Zealand
cristian@cs.auckland.ac.nz

Sergiu Rudeanu
Faculty of Mathematics and Computer Science
The University of Bucharest
Str. Academiei 14, Bucharest, Romania
rud@funinf.cs.unibuc.ro

Abstract. Gödel's incompleteness theorem states that every finitely-presented, consistent, sound theory which is strong enough to include arithmetic is incomplete. In this paper we present elementary proofs for three axiomatic variants of Gödel's incompleteness theorem and we use them (a) to illustrate the idea that there is more than "complete vs. incomplete", there are degrees of incompleteness, and (b) to discuss the implications of incompleteness and computer-assisted proofs for Hilbert's Programme. We argue that the impossibility of carrying out Hilbert's Programme is a *thesis* and has a similar status to the Church-Turing thesis.

1. Introduction

By 1930 research in the foundations of mathematics was in full swing. In 1929 Presburger [23] proved that the arithmetic without multiplication is decidable, in 1930 Gödel proved the completeness of the first-order logic giving a justification for the logical axioms and inference rules, in 1931 Skolem [24] showed that the arithmetic without addition and successor is decidable and Herbrand [17] found finitary consistency proofs for interesting fragments of arithmetic. All these results seemed to fit well with Hilbert's Programme—designed to safeguard mathematical axiomatic theories by proving their consistency—except for a new theorem proved by Gödel in 1930.

Gödel's incompleteness theorem, announced on 7 October 1930 in Königsberg at the First International Conference on the Philosophy of Mathematics (part of the German Mathematical Congress) was referred to as a landmark of the twentieth century mathematics. It says that *in a consistent, sound, finitely-specified theory strong enough to formalize arithmetic, there are true, but unprovable statements* (such a statement is called *independent*); so, such a theory is *incomplete*.

According to Hintikka ([18], p. 4), with the exception of von Neumann, who immediately grasped Gödel's line of thought and its importance, in Königsberg incompleteness passed un-noticed. In spite of being praised, discussed, used (or abused) by many authors, the incompleteness theorem seems, even after so many years since its discovery, stranger than most mathematical theorems.

In this paper we prove in an elementary way three axiomatic variants of Gödel's incompleteness theorem. We first use these results to illustrate the idea that there is more than "complete vs. incomplete", there are degrees of incompleteness. We emphasize that ultimately a mathematical theory or, more generally, a system of logic, is simply a *method of mechanically listing the theorems (logically true sentences)*. Gödel's theorem *does not prove that there are any true but* absolutely *unprovable sentences*: it shows that not all true arithmetical sentences can be proved *mechanically*. The method of mechanical listing (proving) is essential: it can be a method obtained by a group of mathematicians, a standard PC or a PC augmented with a source generating truly random numbers. Secondly, we address the following questions: Is incompleteness "invariant" in some way with respect to the listing method? Has incompleteness (especially when powerful computer-assisted methods of listing are available) any bearing on Hilbert's Programme? We argue that the impossibility of carrying out Hilbert's Programme is a *thesis* and has a similar status to the Church-Turing thesis.

2. Prerequisites

The first-order language of arithmetic includes the following elements: $+$ (addition), $\times$ (product), $\mathbf{0}$ (zero), s (successor), $=$ (equality), $\neg, \wedge, \vee$ (propositional connectives) $\forall, \exists$ (quantifiers), $(,)$ (parentheses), $'$ (prime symbol), and the symbol x. The variables of the language are expressions $x, x', x'', x''', \ldots$ built up from the symbols x and $'$. The symbol s denotes the successor function. Variables are assumed to have values in the set of non-negative integers which can be generated as follows: $\mathbf{0}, \mathbf{1} = s(\mathbf{0}), \mathbf{2} = s(\mathbf{1}) = s(s(\mathbf{0})), \ldots$, so each $n \in \mathbb{N}$ is mirrored by $\mathbf{n}$. Sometimes variables will be abbreviated by single letters: x_1, x_2, y, z, etc. Formulas are built up in the natural way. For example, $\forall x \exists x'(x' = s(x))$ is a true formula, $\forall x \exists x'(x = s(x'))$ is a false formula. All variables appearing in the above formulas are quantified—they are bound. In the formula $\exists x'(s(x) + x' = x'')$, denoted, say by $\varphi(x, x'')$, the variable x' is bound, but the variables x, x'' are not—they are free. The truth value of the formula $\varphi(x, x'')$ depends upon the truth values of free variables: intuitively, $\varphi(\mathbf{0}, \mathbf{1})$ is true, but $\varphi(\mathbf{1}, \mathbf{0})$ is false. A *sentence* is a formula with no free variables; hence, a sentence is either true or false. If $\varphi(x)$ is a formula containing only the free variable x, then for each $n \in \mathbb{N}$, $\varphi(\mathbf{n})$ is an *instance* of $\varphi(x)$, that is a sentence; hence, $\varphi(\mathbf{n})$ is either true or false.

According to Tarski (see [21], p. 364), the *truth* of a sentence φ, in writing, $\mathbb{N} \models \varphi$, is inductively

defined as follows:

$$\begin{aligned}
\mathbb{N} \models \mathbf{n} + \mathbf{m} = \mathbf{p} &\Leftrightarrow m + n = p \\
\mathbb{N} \models \mathbf{n} \times \mathbf{m} = \mathbf{p} &\Leftrightarrow m \cdot n = p \\
\mathbb{N} \models \neg\psi &\Leftrightarrow not(\mathbb{N} \models \psi) \\
\mathbb{N} \models \psi_0 \wedge \psi_1 &\Leftrightarrow \mathbb{N} \models \psi_0 \text{ and } \mathbb{N} \models \psi_1 \\
\mathbb{N} \models \psi_0 \vee \psi_1 &\Leftrightarrow \mathbb{N} \models \psi_0 \text{ or } \mathbb{N} \models \psi_1 \\
\mathbb{N} \models \exists x \psi(x) &\Leftrightarrow \text{for some } n,\ \mathbb{N} \models \psi(\mathbf{n}) \\
\mathbb{N} \models \forall x \psi(x) &\Leftrightarrow \text{for all } n,\ \mathbb{N} \models \psi(\mathbf{n})
\end{aligned}$$

The first-order language of arithmetic is quite powerful; for example, Goldbach's conjecture or Riemann's hypothesis can be formalized in this language.

An *axiomatic theory* consists of a formal language plus a system of axioms and inference rules. *Presburger arithmetic* is the first-order theory of the natural numbers with axioms for successor and addition. Concepts such as divisibility or prime number cannot be formalized in Presburger arithmetic. There is an algorithm which decides, for any given statement in Presburger arithmetic, whether it is true or not; the theory is consistent and complete, cf. [23]. *Robinson arithmetic* is a first-order theory of the natural numbers with addition and multiplication; it contains seven axioms, called the theory Q, relating the successor function, addition and multiplication. Robinson arithmetic cannot prove the formula $\forall x(\neg(x = s(x)))$, cf. [8], p. 183–184. *Peano arithmetic* is a stronger theory: it consists in adding every instance of the first-order schema of induction to the theory Q.

The essence of all theories is the fact that theorems can be enumerated by some kind of machines. In the classical cases, from the propositional or predicate calculi to Peano arithmetic, theorems are enumerated by Turing machines, they are *computably enumerable*. For more details see [1, 8, 26]. But, there are more powerful machines!

We assume that the reader has some familiarity with computability theory and (self-delimiting) Turing machines processing binary strings (see, for example, [2]). Let us fix $X = \{0, 1\}$; by X^* we denote the set of finite strings (words) on X. The length of the string w is denoted by $|w|$. The computation of a Turing machine T on a string w may or may not halt; if it stops, then it produces a (unique) string denoted by $T(w)$; if it doesn't stop, then it produces no output. The *program set* (*domain*) of the Turing machine T is the set $PROG_T = \{x \in X^* : T \text{ halts on } x\}$. All Turing machines will be *self-delimiting* in the sense that their program sets are prefix-free; a self-delimiting Turing machine will be shortly called *machine*. A partial function φ from strings to strings is called *partial computable* (abbreviated p.c.) if there is a machine T such that: a) $PROG_T = \mathrm{dom}\,(\varphi)$, and b) $T(x) = \varphi(x)$, for each $x \in PROG_T$. The *program-size complexity* of the string $x \in X^*$ (relatively to T) is $H_T(x) = \min\{|y| \ : y \in X^*,\ T(y) = x\}$, where $\min \emptyset = \infty$. The Invariance Theorem states that we can effectively construct a machine U (called *universal*) such that for every machine T there exists a constant $\varepsilon > 0$ such that for all $x \in X^*$, $H_U(x) \leq H_T(x) + \varepsilon$. Clearly, U simulates every machine T. In what follows we will fix U and put $H = H_U$.

A p.c. function φ such that $\mathrm{dom}\,(\varphi) = X^*$ is called *computable*. A set of strings is *computable* if its characteristic function is computable. A set of strings is *computably enumerable* (abbreviated c.e.) if it is the program set of a Turing machine. Every computable set is c.e., but the converse implication is not true: for example, $PROG_U$ is c.e. and not computable, hence its complement is not c.e. The set

$\{x \in X^* : H(x) \leq |x| - t\}$ is c.e. and not computable, for each $t \in \mathbb{N}$; its complement is not only not c.e., it contains no infinite c.e. set (such a set is called *immune*).

Let $Y \subset X^*$. An *oracle* Y–machine is a machine "endowed" with the ability to decide the membership in its "oracle" set of strings Y. In this paper we will be interested in machines working with the oracle set $K = PROG_U$. Deciding the membership in K is equivalent with solving the Halting Problem. Intuitively, a computation executed by a K–oracle machine is a normal computation which has the additional power to use the answers to finitely many questions of the form "Is w in K"? Most concepts and results in the theory of computability relativize to oracle machines, in particular to K–machines. The analogues of computable and c.e. sets are K–computable and K–c.e. sets. Obviously, K–machines are more powerful than machines, as K is not computable. The results on program-size complexity described above relativize as well to K–machines.

There is a very strong relation between c.e. sets and a class of formulas which can be expressed in the language of arithmetic. A formula φ in the language of first-order arithmetic is called Δ_0 if φ contains only bound variables. A formula $\varphi(x_1, \ldots, x_k)$ in the language of arithmetic is called Σ_1 if it is of the form $(\exists x_1 \ldots \exists x_k)\psi$, where ψ is a Δ_0 formula. The negation of a Σ_1 formula is called a Π_1 formula. Similarly, Σ_n and Π_n formulas can be constructed. The Representation Theorem states that a relation $R \subset \mathbb{N}^k$ is c.e. iff there (effectively) exists a Σ_1 formula $\varphi(x_1, \ldots, x_k)$ such that for all $n_1, \ldots, n_k \in \mathbb{N}$: $(n_1, \ldots, n_k) \in R \Leftrightarrow \mathbb{N} \models \varphi(\mathbf{n}_1, \ldots, \mathbf{n}_k)$.

We consider the following bijection between non-negative integers and strings on X: $0 \mapsto \lambda, 1 \mapsto 0, 2 \mapsto 1, 3 \mapsto 00, 4 \mapsto 01, 5 \mapsto 10, 6 \mapsto 11, \ldots$ The image of n, denoted $bin(n)$, is the binary representation of the number $n + 1$ without the leading 1. The length of $bin(n)$ is roughly equal to $\log_2(n)$. So, for computability issues we will not differentiate between strings and non-negative integers: if $v = bin(n)$, then the string v is identified with the natural n. In particular, $X^* \models \varphi$ means that φ is true when interpreted on strings. The Representation Theorem is true for strings: a relation $R \subset (X^*)^k$ is c.e. iff there (effectively) exists a Σ_1 formula $\varphi(x_1, \ldots, x_k)$ such that for all $v_1, \ldots, v_k \in X^*$: $(v_1, \ldots, v_k) \in R \Leftrightarrow X^* \models \varphi(\mathbf{v}_1, \ldots, \mathbf{v}_k)$.

3. Degrees of Incompleteness

Consider an axiomatic theory $\mathcal{A}$. We will not be concerned with any details regarding the axioms or inference rules of the axiomatic theory $\mathcal{A}$; the only *syntactical property* we will use is that *the set of theorems proved in $\mathcal{A}$ is c.e.* Hence, in what follows we will work with a c.e. set $T_{\mathcal{A}} \subset X^*$ and assume that $T_{\mathcal{A}}$ is the set of Gödel numbers of theorems proved in $\mathcal{A}$. The "natural" way to enumerate the theorems of a theory $\mathcal{A}$ is via their proofs, so a theorem may appear infinitely many times in $T_{\mathcal{A}}$. This method, pioneered by Gödel, is called *Gödel numbering*; the elements of $T_{\mathcal{A}}$ are the Gödel numbers of the theorems in $\mathcal{A}$.

To achieve generality, our study will concern *axiomatic theories $\mathcal{A}$ proving* (among other facts) *properties of non-negative integers*; in fact, we will focus our attention exactly on the theorems of the theory which establish properties of non-negative integers. We will say that such a theory *contains the first-order arithmetic*. Robinson arithmetic or Peano arithmetic are examples. We note that if the set of theorems of $\mathcal{A}$ is c.e., then the subset of theorems of $\mathcal{A}$ establishing sentences of the first-order language of arithmetic is also c.e. Hence, we will work with the c.e. set $T_{Arith} \subset T_{\mathcal{A}}$ of Gödel numbers of theorems establishing sentences expressed in the first-order language of arithmetic. An *interpretation* of

T_{Arith} is a computable function $\imath$ mapping T_{Arith} into the set of sentences of arithmetic. If $w \in T_{Arith}$ and $\varphi = \imath(w)$, then w is the Gödel number of φ.

We can ask now: Can we find an axiomatic theory $\mathcal{A}$ capable of proving all true first-order arithmetical sentences? To relate the theorems proved in $\mathcal{A}$ and the sentences of arithmetic we will not work with global properties of the theory $\mathcal{A}$, but only with those properties pertaining to non-negative integers: (a) the pair $(\mathcal{A}, \imath)$ is *(arithmetically) sound* if for every $w \in T_{Arith}$, $\imath(w)$ is true, that is, $\mathcal{A}$ proves only true first-order arithmetical sentences, (b) the pair $(\mathcal{A}, \imath)$ is *(arithmetically) complete* if every true sentence φ is provable in $\mathcal{A}$, i.e., there exists a $w \in T_{Arith}$ such that $\imath(w) = \varphi$, (c) the pair $(\mathcal{A}, \imath)$ is *(arithmetically) consistent* if there is no first-order sentence φ and there are no $v, w \in X^*$ such that $\imath(v) = \varphi, \imath(w) = \neg\varphi$, that is, $\mathcal{A}$ does not prove both a sentence and its negation. Note that (a) and (b) are semantical properties, while (c) is syntactical; for more details see [18], p. 16.

In the above definitions we talked about "true" sentences; the way we define/express them directly influences the degree of rigour of our results. Tarski's method of defining truth makes use of the "intuitive" meaning of arithmetic, hence it is only a step toward formalisation. This fact was discussed in detail by Gödel [14], when he distinguished between true mathematical propositions (objective) and demonstrable mathematical propositions (subjective).

In what follows we will work with an axiomatic theory $\mathcal{A}$ (whose set of theorems is c.e.) and a fixed interpretation $\imath$; for brevity, sometime we will refer to the pair $(\mathcal{A}, \imath)$ as $\mathcal{A}$.

Theorem 3.1. (Gödel Incompleteness Theorem: First Variant)
For every consistent, c.e. and sound axiomatic theory $(\mathcal{A}, \imath)$ containing the first-order arithmetic there effectively exist a Σ_1 formula $\varphi(x)$ (containing a unique free variable x) and a string $w \in X^*$ such that $\imath(v) \notin \{\varphi(\mathbf{w}), \neg\varphi(\mathbf{w})\}$, for each $v \in X^*$.

Proof:
Use the Representation Theorem for the c.e. set K to effectively construct a Σ_1 formula $\varphi(x)$ in the language of first-order arithmetic such that for every $w \in X^*$ we have: $w \in K \Leftrightarrow X^* \models \varphi(\mathbf{w})$. Assume by absurdity that for every $w \in X^*$ there exists $v \in X^*$ (which depends upon w) such that $\imath(v) = \varphi(\mathbf{w})$ or $\imath(v) = \neg\varphi(\mathbf{w})$. We construct now an enumeration procedure for the complement of K, an impossibility because this would imply the computability of K. The procedure is the following: for each $w \in X^*$ we start enumerating all strings $v \in X^*$ till we obtain $\imath(v) = \varphi(\mathbf{w})$ or $\imath(v) = \neg\varphi(\mathbf{w})$ (a decidable test) and list those w's for which $\imath(v) = \neg\varphi(\mathbf{w})$. In this way we obtain all the elements of the complement of K and only them. Indeed, if w is listed, then $\imath(v) = \neg\varphi(\mathbf{w})$ for some v, so because of soundness, $X^* \models \neg\varphi(\mathbf{w})$, so by the Representation Theorem $w \notin K$. Conversely, if $w \notin K$, then it is impossible to find a v such that $\imath(v) = \neg\varphi(\mathbf{w})$, so w is enumerated in the list. □

Comment The above proof rests on a *broken symmetry*: the class of formulas is closed under the negation, but the class of c.e. sets is not closed under complement. Note that the theorem does not say anything about the truth of the formula $\varphi(\mathbf{w})$: it simply proves that the theory $\mathcal{A}$ cannot prove $\varphi(\mathbf{w})$ and $\neg\varphi(\mathbf{w})$.

Corollary 3.1. For every consistent, c.e. and sound axiomatic theory $(\mathcal{A}, \imath)$ containing the first-order arithmetic there effectively exists a Π_1 formula $\psi(x)$ (containing a unique free variable x) such that for some $w \in X^*$, $X^* \models \psi(\mathbf{w})$, but $\imath(v) \neq \psi(\mathbf{w})$, for each $v \in X^*$. In other words, each consistent, c.e. and sound axiomatic theory containing the first-order arithmetic is incomplete.

Proof:
Take $\psi(x) = \neg\varphi(x)$, where φ is the Σ_1 formula in Theorem 3.1. □

Corollary 3.2. The set of all true sentences expressible in the first-order language of arithmetic is not c.e.

Comment The theory $\mathcal{A}$ cannot prove *all* true sentences $\psi(s)$, but the proof presented does not rule out the possibility that $\mathcal{A}$ proves infinitely many true sentences $\psi(s)$. Reason: the complement of the set K is not c.e., but it contains infinite c.e. subsets. There is, however, another Π_1 formula, containing a unique free variable, such that $\mathcal{A}$ cannot prove more than finitely many true instances of it.

Theorem 3.2. (Gödel Incompleteness Theorem: Second Variant)
For every consistent, c.e. and sound axiomatic theory $(\mathcal{A}, \imath)$ containing the first-order arithmetic there effectively exists a Π_1 formula $\psi(x)$ (containing a unique free variable x) such that the set $\{w \in X^* : X^* \models \psi(\mathbf{w}), \imath(v) = \psi(\mathbf{w}),\ \text{for some } v \in X^*\}$ is finite.

Proof:
instead of a c.e. and non-Use the Representation Theorem for the c.e. set $\{x \in X^* : H(x) \leq |x| - t\}$, for any fixed $t \geq 0$: there effectively exists a Σ_1 formula $\varphi(x)$ such that for every $u \in X^*$: $H(u) \leq |u| - t \Leftrightarrow X^* \models \varphi(\mathbf{u})$. The Π_1 formula $\psi(x) = \neg\varphi(x)$ satisfies the equivalence: $H(u) > |u| - t \Leftrightarrow X^* \models \psi(\mathbf{u})$. Assume that the set $\imath(T_{Arith})$ contains all true arithmetical sentences. Using consistency, the computable function $\imath$ and the formula ψ we can enumerate all theorems w corresponding to true sentences of the form $\psi(\mathbf{w})$ to produce an enumeration of the immune set $\{w \in X^* : H(w) > |w| - t\}$: this is a contradiction. Hence, T_{Arith} cannot contain more than finitely many true instances of ψ. □

Comment Solovay [27] has proved that for every consistent, c.e. and sound axiomatic theory $\mathcal{A}$ containing the first-order arithmetic there effectively exists a Π_2 formula $\varphi(x, y)$ (containing two free variables x, y) such that $\mathcal{A}$ cannot prove any true instance of it.

Assume now that the theorems of the axiomatic theory are generated by a more powerful machine, for example, one which is capable of accessing an oracle solving the Halting Problem. This scenario is motivated by the recent advent of computer-assisted proofs (see, for example, [4]) and the interest in unconventional architectures capable of trespassing the Turing barrier (a Turing machine supplemented with a quantum source of random bits is an example, see [3]). Will such a theory be capable of proving all true first-order properties of non-negative integers?

To answer this question we will work with a K–c.e. axiomatic theory, i.e., a theory whose theorems forms a K–c.e. set. Again, the interpretation will be a computable function $\imath$ from a K–c.e. set into the set of sentences of first-order arithmetic.

Theorem 3.3. (Gödel Incompleteness Theorem: Third Variant)
For every consistent, K–c.e. and sound axiomatic theory $(\mathcal{A}, \imath)$ containing the first-order arithmetic there effectively exists a Π_2 formula $\varphi(x)$ (containing a unique free variable x) such that the set $\{w \in X^* : X^* \models \varphi(\mathbf{w}), \imath(v) = \varphi(\mathbf{w}),\ \text{for some } v \in X^*\}$ is finite.

Proof:
This proof is just a relativized form of the proof of Theorem 3.2, where we use Post Theorem (see [6],

p. 220) for the relativized program-size complexity H^K and we note that the complement of the K–c.e. set $\{w \in X^* : H^K(w) \leq |w| - t\}$ is K–immune, i.e., it is infinite and contains no K–c.e. infinite subset. □

Comment The axiomatic theory in Theorem 3.3 proves all true instances of every Π_1 formula, but fails to prove true instances of some Π_2 formulas. Of course, by relativization, similar results can be proved at every level of the Arithmetical Hierarchy.

4. Consistency and Hilbert's Programme

For Gödel [11] the problem of giving a foundation of mathematics (understood as "the totality of the methods of proof actually used by mathematicians") falls into two parts, the first to reduce the methods "to a minimum number of axioms and primitive rules of inference, which have to be stated as precisely as possible", and then, the second, "a justification in some sense or other has to be sought for these axioms, i.e., a theoretical foundation of the fact that they lead to results agreeing with each other and with empirical facts". The first question was considered to be satisfactorily solved by the "formalization of mathematics in the simple theory of types". However, for the second problem, "... it must be said that the situation is extremely unsatisfactory". The pure syntactical part, in which mathematics is a game with symbols, seems acceptable, but serious problems start to appear as soon as meaning is attached to symbols.

Thirty years after, Gödel's view (see [14], p. 379) was that Hilbert's formalism was a curious hermaphroditic attempt to reconcile the philosophical spirit of (his) time and the true nature of mathematics. On one hand, it is recognised that the truth of axioms cannot be justified, hence the meaning of any consequence deriving from them is only hypothetical, and, on the other hand, "one clung to the belief" that a mathematical proof provides a secure grounding for the statement it proves and "every precisely formulated yes–no question in mathematics must have a clear-cut answer". The aim is to prove that, in the mathematical game, of two sentences ψ and $\neg\psi$, exactly one can always be derived. This is the important property of *consistency*: if it is impossible to derive both ψ and $\neg\psi$, then the mathematical question expressed by ψ can have a clear-cut answer. The question of consistency is purely combinatorial, so one might hope to establish it by *unobjectionable methods*.

Hilbert's 1920 programme proposed that the consistency of more complicated theories, such as real analysis, could be proven in terms of simpler theories, and, ultimately, the consistency of all of mathematics could be reduced to basic arithmetic. In other words, we first build an axiomatic theory $\mathcal{A}$ covering the entire mathematics, and secondly, we use Peano arithmetic to prove the consistency of $\mathcal{A}$. Hilbert requested that the method of proving consistency had to be *finitary*.

As noted by Gödel, the first stage was accomplished; Gödel himself was an important contributor. However, according to Gödel's second incompleteness theorem, *Peano arithmetic cannot be used to prove its own consistency*, so it certainly cannot be used to prove the consistency of anything stronger. Does it mean that there is no hope of carrying out Hilbert's programme? Many authors acknowledge that Gödel's second incompleteness theorem, by itself, cannot refute Hilbert's Programme (see, for example, [20, 7, 18]). An important issue is the generality of the notion of formal system used in Gödel's incompleteness theorem. But, even working with a general, axiomatic definition of a formal system (as we did in the previous section) does not give full guarantee because the very generality of such a definition

would be questionable. In this sense, the impossibility of carrying out Hilbert's Programme is a *thesis* and has a similar status to the Church-Turing thesis.

Well before Gödel, Poincaré [22] (volume 2, chapter 4) warned against justifying the induction principle by means of the induction principle, a principle building Peano arithmetic. Gödel himself had subtle variations in interpreting the above argumentation. In [11] he seems to agree. In [12] he explicitely states that his second incompleteness theorem does not contradict Hilbert's formalistic point of view as "it is conceivable that there exist finitary proofs that *cannot* be expressed in the formalism ..."; this point seems not to square well with "Hilbertian combination of materialism and aspects of classical mathematics thus proves to be impossible" (see [15] p. 381). Finally, in [16] we read: "there are ... no conclusive *combinatorial* consistency proofs (such as Hilbert expected to give)" with the important additional notes: "it does not follow from my theorems that there are no *convincing* consistency proofs for the usual mathematical formalisms" and "the concept of a combinatorial proof, although intuitively clear, has not yet been precisely defined".

There are subtle problems related to specific axiomatisations of arithmetic. For example, it is possible to prove the consistency of Robinson arithmetic in Peano arithmetic but not in itself, see [1], Chapter 16. More interestingly, the set theory ZF can prove the sentence asserting the consistency of Peano arithmetic (translated into the language of sets); the original formula cannot be proved in Peano arithmetic. We have an example of a sentence asserting some property of natural numbers which can be proved in ZF, but not in Peano arithmetic. The arithmetic expressed in terms of sets is stronger than first-order arithmetic, which is stronger than Robinson arithmetic. Presburger arithmetic is complete; Robinson and Peano arithmetics are not because they are too powerful. In 1936 Gentzen proved the consistency of Peano arithmetic assuming all principles of induction "at once" (for all well-orderings); see [19], §79. Does this proof qualify as "unobjectionable"?

All consistency and completeness theorems (as for propositional or predicate calculi) are not proved *within* those calculi, but from "outside". A consistency proof "from inside" may not be really credible (like police giving a non-corruption certificate for itself). So, perhaps, the second incompleteness theorem is not a negative result, not a limit, but a *positive* result! Every axiomatic theory is open to extensions. This view seems to match well with Gödel's [15] "the truth lies in the middle" (he accepts the idea of mathematics as a theory of truth, which is complete in the sense that "every precisely formulated yes–no question in mathematics must have a clear-cut answer", but rejects the idea that the basis of mathematics truths is embedded in the axioms) and with Chaitin's [5] observation that all real systems are dynamical, vary in time: "Why mathematics has to be static, fixed once for all?", "How come that in spite of incompleteness, mathematicians are making so much progress?"

Are there any "solutions"? Gödel [13] was optimistic: "But there remains the hope that in the future one may find other and more satisfatory methods of construction beyond the limits of the system A [capturing finitist methods], which may enable us to found classical arithmetic and analysis on them." Later on, Gödel [15], inspired by Husserl's phenomenology, suggested that an alternative to securing the mathematical game of symbols is "cultivating (deepening) knowledge of the abstract concepts themselves which lead to the setting up of these mechanical systems". Reverse mathematics—concerned with the minimal axioms needed to prove a particular theorem—is another solution; see [25]. "Partial completeness" was suggested by Hintikka [18], p. 26: one does not need full completeness to carry out Hilbert's Programme, only completeness as far as the consequences of the specific theory one wants to justify are concerned. This can be achieved, for example, via "practical consistency": prove, for a given very large N, that there is no sentence s of length less than N such that the theory proves both s

and $\neg s$. In this direction, provability with the help of a Turing machine supplemented with a quantum source of random bits may offer at least an empirical confirmation. But there is more. If our "computing device" might solve the Halting Problem, or even the Halting Problem for programs up to a fixed, large length (measured in bits)—an open question for the combination "Turing machine supplemented with a quantum source of random bits" (see [3])—then, we might have a "computational" alternative: simply compute the finite set of provable sentences (which are all halting programs) and check consistency.

References

[1] G.S. Boolos, R.C. Jeffrey. *Computability and Logic*, Cambridge University Press, Cambridge, 1980. Second Edition.

[2] C.S. Calude. *Information and Randomness. An Algorithmic Perspective*, Springer Verlag, Berlin, 1994. 2nd Edition, Revised and Extended, 2002.

[3] C.S. Calude. Algorithmic Randomness, Quantum Physics, and Incompleteness, *CDMTCS Research Report* 248, 2004, 17 pp.

[4] C.S. Calude, S. Marcus. Mathematical proofs at a crossroad? in J. Karhumäki, H. Maurer, G. Păun, G. Rozenberg (eds.). *Theory Is Forever*, Lectures Notes in Comput. Sci. 3113, Springer-Verlag, Berlin, 2004, 15–28.

[5] G.J. Chaitin. Computers, paradoxes and the foundations of mathematics, *American Scientist*, 90 March–April (2002), 164–171.

[6] M. Davis, R. Sigal, E.J. Weyuker. *Computability, Complexity, and Languages*, Academic Press, New York, 1994. Second Edition.

[7] M. Detlefsen. *Hilbert's Program: An Essay on Mathematical Instrumentalism*, Reidel/Kluwer Academic, Dordrecht, 1986.

[8] R.L. Epstein, W.A. Carnielli. *Computability*, Wadsworth & Brooks/Cole, Pacific Grove, California, 1989.

[9] S. Feferman, J. Dawson, Jr., W. Goldfarb, C. Parsons, R.M. Solovay (eds.). *Kurt Gödel Collected Works*, Volume III, Oxford University Press, Oxford, 1995, 45–53.

[10] S. Feferman, J. Dawson, Jr., W. Goldfarb, C. Parsons, W. Sieg (eds.). *Kurt Gödel Collected Works*, Volume V, Clarendon University Press, Oxford, 2003.

[11] K. Gödel. The present situation in the foundations of mathematics (1930), in [9], 45–53.

[12] K. Gödel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, *Monatshefte f. Math. u. Phys.* 38 (1931), 173–198.

[13] K. Gödel. Lecture at Zilsel's (1938), in [9], 87–113.

[14] K. Gödel. Some basic theorems on the foundations of mathematics and their implications (1952), in [9], 304–335.

[15] K. Gödel. The modern development of the foundations of mathematics in the light of philosophy (1961), in [9], 375–387.

[16] K. Gödel. Letter to Leon Rappaport (1962), in [10], 176–177.

[17] J. Herbrand. Sur la non-contradiction de l'Arithmétique, *J. Reine Angew. Math.* 166 (1931), 1–8.

[18] J. Hintikka. *On Gödel*, Wadsworth, Belmont, 2000.

[19] S.C. Kleene. *Introduction to Metamathematics*, North-Holland, Amsterdam, 1952.

[20] G. Kreisel, J. L. Krivine. *Elements of Mathematical Logic*, North-Holland, Amsterdam, 1967.

[21] P. Odiffreddi. *Classical Recursion Theory*, North-Holland, Amsterdam, 1989.

[22] H. Poincaré. *Science et méthode*, Flammarion, Paris, 1908. English translation by F. Maitland, with a preface by B. Russell, Nelson, London, 1914.

[23] M. Presburger. Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt, *Comptes Rendus du I Congrès de Mathématiciens des Pays Slaves*, Warszawa, 1929, 92–101.

[24] T. Skolem. Über einige Satzfunktionen in der Arithmetik, *Skrifter utgitt av Det Norske Videnskaps-Akademi i Oslo, I. Matematisk naturvidenskapelig klasse*, 7 (1931), 1–28.

[25] S. G. Simpson (ed.). *Reverse Mathematics 2001*, `http://www.math.psu.edu/simpson/ revmath/`, to appear.

[26] C. Smoryński. The incompleteness theorems, in J. Barwise (ed.). *Handbook of Mathematical Logic*, North-Holland, Amsterdam, 1977, 821–866.

[27] R.M. Solovay. A version of Ω for which ZFC can not predict a single bit, in C.S. Calude, G. Păun (eds.). *Finite Versus Infinite. Contributions to an Eternal Dilemma*, Springer-Verlag, London, 2000, 323–334.