Hilbert's Tenth Problem

Tove Verner-Carlsson

March 10, 2017

Abstract

This essay concerns Hilbert's tenth problem. We begin by stating this problem and describe its origins, moving on to disecting the components of said statement, the main one being Diophantine equations. After extensive discussion about Diophantine equations and sets, logic of different forms, and computability we move on to the chronology of the solution of Hilbert's tenth problem. Covering the main events that led to the unsolvability proof of the problem, the essay culminates in the fact that Diophantine sets coincide with semidecidable sets. This completes the unsolvability of Hilbert's tenth problem.

Contents

1 Introduction

Who was Hilbert? What is his tenth problem, and why is it called the tenth? Let us begin by giving the topic of this essay a historical context, accompanied by some philosophical discussion.

1.1 Hilbert's 23 problems

At the International Congress of Mathematicians in Paris in 1900, *David Hilbert* gave a memorable talk entitled "The Problems of Mathematics". During this talk he presented a compilation of problems that he predicted would be of great importance for the upcoming century. In the actual talk he presented ten problems (though it is a common misconception that all 23 occurred), but the list was soon to be extended to the legendary 23 problems of Hilbert.

The list became very influential, and is generally reckoned the most successful and deeply considered list of open problems ever to be produced by an individual mathematician. Some of these were solved within a short time, some were discussed throughout all of the 20th century with varying results. Some still remain a challenge for mathematicians to resolve.

One of these 23 problems, namely the tenth, is about solutions to *Diophantine equations*. This is the one with which we shall concern ourselves in this essay.

1.2 Formulation of Hilbert's tenth problem

Hilbert's original publication of the 23 problems was in German, but the following translation of the tenth is one of the more widely spread.

Problem 10. Hilbert's Tenth Problem

> Given a Diophantine equation with any number of unknown quantities and with rational integral numerical coefficients: to devise a process according to which it can be determined in a finite number of operations whether the equation is solvable in rational integers.

There are some things with this formulation that needs to be clarified. First we note that common practise is to consider Hilbert's reference to "rational integers" to mean the same numbers we would usually just call integers. So did those who eventually solved the problem during the 50's and 60's, and hence it is also what we will do here. There has been a lot of research about the analogue of Problem 10 for other domains, for example the rational numbers $\mathbb{Q}$, but we will restrict ourselves to the integers.

Nowadays we would say that Hilbert was asking for an algorithm, but that word wasn't in broad use until decades later with the importance of computers growing rapidly. What he is asking for is, in modern terms, a computer program that will take any Diophantine equation as input, and output YES if it is solvable and NO if it is not. We will investigate this through the notion of *Turing machines*, i.e. very primitive and theoretical, but yet complete, computers in section 3.5. In fact Hilbert couldn't really have used the word algorithm, not merely because the word wasn't in use but also because there was no rigorous definition of this word. Instead he proposed the notion of a process with a finite number of operations, which is a bit vague. It is troublesome to prove something about a vague statement, which contributed to Problem 10 being open for so long. The development of Turing machines allowed for a definition of this type of processes, and enabled mathematics to make universal statements about such. This was a key factor to be able so solve Hilbert's tenth problem at all.

1.3 The Hilbert Program

Hilbert is said to have been very optimistic about the future of mathematics. His ambitious Hilbert Program indicates this, as well as quotations such as the famous "We must know. We will know." which is also engraved on his tombstone. The main purpose of the program was to formalise and axiomatise all of mathematics — indeed an ambitious goal. This was partially a response to certain problems arising in mathematics at the time. Unpleasant paradoxes like the famous one of Russell had arisen and needed to be dealt with.

Example 1. Russell's Paradox
Mathematician Bertrand Russell made reference to "the set of all sets not containing themselves". In symbols this it the set $S = \{A : A \notin A\}$. But then what happens with the set S itself? Suppose that $S \notin S$. Then it doesn't contain itself and should be in S — but we just stated that it didn't! Suppose instead $S \in S$. By definition S contains only sets which does not contain themselves, so this also leads to a contradiction. We have encountered a paradox. △

Certainly this was a burden for contemporary mathematicians, and probably caused many sleepless nights. One had to think twice about how to treat sets and how not to. In the case of Russell's paradox, set theory was modified so as not to include such sets at all to avoid this inconsistency.

With the positive attitude Hilbert held, he certainly expected the process he was asking for in his tenth problem to exist — though we will see that in fact it

doesn't. However he is in spite of this optimism said to have uttered the words "Occasionally it happens that we seek the solution under insufficient hypotheses or in an incorrect sense, and for this reason do not succeed. The problem then arises: to show the impossibility of the solution under the given hypotheses or in the sense contemplated."This impossibility is exactly what will be established in what follows.

1.4 Decision problems

Hilbert's tenth problem is a so called *decision problem*. Such problems consist of countably many individual problems, which can be called subproblems, each one with a YES- or NO-answer. The essence of a decision problem lies in the search for a general method that provides this YES- or NO-answer for any given subproblem. Each subproblem is specified by a finite amount of information; in the case of Hilbert's tenth problem, this is a given Diophantine equation and the domain in which to search for solutions (here integers).

Obviously we know how to solve some of these subproblems. For example, linear Diophantine equations (see section 2.2) are very easy to solve, and for such we have a sufficient algorithm based on the elementary *Euclidean algorithm*. But then again, Hilbert asks for a universal method that would work for *any* subproblem, and so Euclid's algorithm certainly will not do.

If there exists a general method that solves every single subproblem, one way to show this would be to specify that very method. The method might be crazy or extremely complex, and might not even be possible to find — but if we could find it, and specify it, we would simply be done. That would be what we call a *direct* proof. The other way would be to reduce the decision problem to another one that has already been solved, which would be more of an *indirect* proof. If this would have been the case — that the process Hilbert was asking for would exist — it is possible that Problem 10 might have been solved much earlier.

However if there doesn't exist such a general method, or process, or algorithm, or whatever word you prefer, we have to show that none can ever exist. It will not do to simply show that, say, Euclid's algorithm doesn't suffice. We have to be able to say that about every possible algorithm, every combination of instructions — but these are infinitely many! Now this is a universal attempt at a completely different level: we need to prove that for all algorithms, none will solve each subproblem. It would, obviously, be futile to try to eliminate them one at a time since we would never be done.

This is where something called algorithmic unsolvability comes into play.

The development of the Turing machine suddenly axiomatised algorithms, and we were enabled to talk about such things as possible and impossible algorithms. This new notion of computability theory, or recursion theory as it is also called, was of indispensable importance to the solution of Hilbert's tenth problem. Before, there was no definition of algorithm, or of a process with a finite number of operation to put it the way Hilbert himself did, and therefore impossible to state anything about algorithms in general. But suddenly we could do such a thing. It might be worth mentioning that this assertion requires the acceptance of the *Church-Turing Thesis*, that we will elaborate in section 3.6.

1.5 Making reductions

When solving a mathematical problem, or constructing or searching for a proof, it is almost inevitable to make use of *reductions*. This means that if we want to solve say problem A, we might first show that if we can solve problem B we will also be able to solve problem A — and then we move on to trying to solve problem B instead. This method of reducing one problem to another-technique was so frequently used during the solving of Hilbert's tenth problem, that one suffers a severe risk of losing oneself amongst the different angles and aspects, and what happened when, and what actually implies what. All we can say for sure at this point is that everything taken together, apparently provides a negative solution to Hilbert's tenth problem!

1.6 Ready, set, solved! Or not?

We conclude this section by asking whether Hilbert's tenth problem was really solved after all. This might be considered semantics and disregarded with a reference to it being of mere "academic interest" — but as the reader probably will acknowledge, that is not an argument to dismiss it, but rather the opposite! What would Hilbert himself think? Considering the quotation in section 1.3, he would probably accept the negative solution as an actual solution. However in hindsight, he might wish he would have stated it differently. A positive solution would have yielded a process for determining solutions in rational numbers as well, and in his optimism he might have thought that his formulation would suffice, but the negative solution provides no further information about such solutions. It is reasonable to think that Hilbert would have included a question about solutions in rational numbers as well if he would have foreseen said outcome. Problem 10 modified as to ask for rational solutions remains to date an open problem.

When we say that Hilbert's tenth problem given its original statement was solved, we can't mean that the decision problem Hilbert's asked for has been solved. It hasn't. It has been *proven unsolvable*. An unsolvability proof is not the same as a solution. I would suggest that we distinguish between the tenth problem Hilbert posed, and "Hilbert's tenth problem". Then we could say that the former has been given an unsolvability proof and the latter has been solved, in the sense that it is no longer a problem and we know the outcome.

2 Diophantine equations

To be able to understand the work that led to the unsolvability proof of which we spoke above we naturally need some theory and some place to start off. We begin with some simple and familiar examples, before moving on to stating what Diophantine equations formally are and some properties they possess.

2.1 Diophantine equations in one variable

It might be of pedagogical interest to start by considering some very basic special cases of Diophantine equations to familiarise with the concept. We can state a Diophantine equation with only one variable as

$$a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x^n + a_0 = 0, \text{ where } a_i \in \mathbb{Z}. \tag{1}$$

As one can see, this is essentially a polynomial equation in one variable, with n complex roots. It becomes Diophantine when we limit the solutions to integers. No doubt, for $n \leq 4$ we could find all solutions via well-known algebraic formulas, however irrational or complex they would be. Then we could take these solutions and check one at a time if they are integers, to find all integer solutions. However as shown by Niels Abel in 1824, when the degree is 5 or greater there exists no such general formula. Here one is referred to so called numerical methods to approximately find all solutions.

There is a difference when we are looking only for integer solutions. Consider a polynomial

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0. \tag{2}$$

If $a_0 = 0$, we can factor an x from the polynomial, and 0 will be a root. We can do this until the constant term is not 0, and hence we without loss of generality may consider only polynomials with $a_0 \neq 0$.

Every integer solution to $p(x)$ will divide a_0. Because of this if suffices to first find the divisors of a_n, and then input these in p to see if it evaluates to 0. Surely, factoring integers isn't always a pleasurable method, but it is algorithmically manageable.

Example 2. A Diophantine polynomial equation of order 5
Consider the polynomial

$$p(x) = x^5 - 36x^3 - 84x^2 - 37x - 84. \tag{3}$$

We find the prime factorisation as $84 = 2^2 \cdot 3 \cdot 7$ without much trouble. This gives us 24 (we can have both positive and negative values) possible solutions for

$x : \{\pm 1, \pm 2, \pm 3, \pm 4, \pm 6, \pm 7, \pm 12, \pm 14, \pm 21, \pm 28, \pm 42, \pm 84\}$. Inputting these values in p one after another and noting when it is equal to 0, we see that the integer roots of polynomial p are $x = -3$, $x = -4$ and $x = 7$. Hence these are the solutions to the Diophantine equation $p(x) = 0$. $\triangle$

If one wants to avoid factoring the constant term (which is reasonable to want for large values of a_0), there are also ways to give an upper bound for the solutions. We will illustrate one way to do this. Take the polynomial from (2) and factor out signs from the coefficients so that we may write it in the form

$$p(x) = a_n x^n - b_{n-1} x^{n-1} - \cdots - b_1 x - b_0 \tag{4}$$

(where $b_i = -a_i$ for $0 \le i \le n-1$). When the absolute value of x is greater than or equal to 1, we have that $|x^n| = |x|^n \ge |x|^i = |x^i|$, for $0 \le i \le n-1$. Rewriting the polynomial (4) and applying this inequality as well as the triangle inequality we get

$$\begin{aligned} p(x) = 0 \iff \quad & a_n x^n = a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \\ \implies \quad & |a_n| \cdot |x|^n = |a_{n-1} x^{n-1} + \cdots + a_1 x + a_0| \\ & \le |a_{n-1} x^{n-1}| + \cdots + |a_1 x| + |a_0| \\ & \le |a_{n-1} x^{n-1}| + \cdots + |a_1 x^{n-1}| + |a_0 x^{n-1}| \\ & = |x|^{n-1} (|a_{n-1}| + \cdots + |a_1| + |a_0|) \end{aligned} \tag{5}$$

Dividing by $|a_n| \cdot |x|^{n-1}$ we obtain

$$|x| \le \frac{1}{|a_n|} \, (|a_{n-1}| + \cdots + |a_1| + |a_0|). \tag{6}$$

Now it is possible to test all integers below this upper bound. Surely if this number is high, there will be a lot of testing, but evaluating a polynomial for a specific value is not the hardest thing to do computationally.

2.2 Linear Diophantine equations

Another simple special case is when a Diophantine equation is *linear*. Then there exists a simple algorithm to not just decide if the equation has solutions, but even determine these solutions.

Definition 1. Linear Diophantine equation
A linear Diophantine equation is an equation of the form

$$ax + by = c, \tag{7}$$

with given $a, b, c \in \mathbb{Z}$ and two unknowns x and y. $\triangle$

Again, we are looking only for integer solutions. A necessary condition for the equation $ax + by = c$ to have a solution is that the greatest common divisor of a and b divides c (in symbols $gcd(a,b)|c$). Therefore we can easily see that, for example, $6x + 9y = 2$ has no (integer) solutions; $gcd(6,9) = 3$, and $3 \nmid 2$. Moving on to an example where solutions exist.

Example 3. Consider the equation

$$24x + 9y = 15. \tag{8}$$

First we compute $gcd(24,9)$ with the Euclidean algorithm;

$$\begin{aligned} 24 &= 2 \cdot 9 + 6 \\ 9 &= 1 \cdot 6 + 3 \\ 6 &= 2 \cdot 3 + 0. \end{aligned} \tag{9}$$

We find that $gcd(24,9) = 3$ (this procedure might surely be a bit overkill in this specific case). If we now divide both sides of (8) by 3, we get

$$8x + 3y = 5. \tag{10}$$

Equations (8) and (10) will have the same set of solutions, so in fact we may always assume that $gcd(a,b) = 1$ while solving linear Diophantine equations. This is the case in (10): $gcd(8,3) = 1$. Once again we apply Euclid's algorithm, but this time not to find the gcd but rather to be able to express 8 and 3 as a combination of the remainders that occur in the process.

$$\begin{aligned} 8 &= 2 \cdot 3 + 2 \\ 3 &= 1 \cdot 2 + 1. \end{aligned} \tag{11}$$

Now we can combine the remainders "backwards", and get

$$\begin{aligned} 2 &= 8 - 2 \cdot 3 \\ 1 &= 3 - 1(8 - 2 \cdot 3) = -1 \cdot 8 + 3 \cdot 3. \end{aligned} \tag{12}$$

We solve the help-equation $8x + 3y = 1$ where the right hand side equals 1 using (12);

$$\begin{cases} 1 = 8x + 3y \\ 1 = -1 \cdot 8 + 3 \cdot 3 \end{cases} \quad \Longrightarrow \quad \begin{cases} x_0 = -1 \\ y_0 = 3 \end{cases} \tag{13}$$

Hence $x_0 = -1$, $y_0 = 3$ is a solution to our help-equation, and multiplying with the right hand side of equation (10) we obtain *one* solution to $8x + 3y = 5$ (and to (8) as well) as $x_0' = -5$, $y_0' = 15$. It remains to find *all* solutions. These will be $x = 3 \cdot (-5) - 3k$, $y = 3 \cdot 15 + 8k$, $k \in \mathbb{Z}$. $\triangle$

2.3 General Diophantine equations

A Diophantine equation is an equation in which only integer (or another range if specified) solutions are allowed. Thus there will be plenty of such equations with no solutions. As a simple example there is no integer x such that $x^2 = 2$, so the Diophantine equation $x^2 - 2 = 0$ lacks solutions.

Definition 2. Diophantine equation
A *Diophantine equation* is an equation of the form

$$D(x_1, ..., x_m) = 0, \tag{14}$$

where D is a polynomial with integer coefficients. △

The questions asked in Diophantine analysis include: Are there any solutions? Are there any solutions beyond some that are easily found by inspection? Are there finitely or infinitely many solutions? Can all solutions be found in theory? Can one in practice compute a complete list of solutions?

Hilbert's tenth problem concerns itself only with the first of these questions. But to answer any of the questions for a given Diophantine equation we must provide a representation of the form in Definition 2 as well as a range of the unknowns. We have already established that with Hilbert's tenth problem, that range is the integers. However *as a decision problem*, it is equivalent to speak of the natural numbers as the domain of solutions. For now we accept this as a fact, and conclude that to establish the unsolvability of Hilbert's tenth problem in its original form, it is sufficient to establish it for natural numbers.

We will want to divide the variables of the Diophantine equation into two different categories: unknowns and *parameters*. Then we obtain the following concept.

Definition 3. Parametric Diophantine equation
A parametric Diophantine equation is an equation of the form

$$D(a_1, \ldots, a_n; x_1, \ldots, x_m) = 0, \tag{15}$$

where D is a polynomial with integer parameters $a_1, \ldots, a_n$ and unknowns $x_1, \ldots, x_m$. △

Example 4. The linear Diophantine equations as parametric equations
The linear Diophantine equation from Definition 1 may be written in this form:

$$\begin{aligned} &D(a, b, c; x, y) = ax + by - c \\ &D(a, b, c; x, y) = 0. \end{aligned} \tag{16}$$

Then x and y will be unknowns and $a,\ b,\ c$ will be parameters that we have to fix to see if the corresponding linear Diophantine equation has a solution. If we recall Example 3, we had $a = 24,\ b = 9,\ c = 15$. It is worth mentioning that the c has switched sides; while it was intuitive to write it on the right hand side above, we have to transfer it to the left hand side to match our formal definition of a general Diophantine equation. There is no real difference to be confused by here. $\triangle$

Let's see an example of a Diophantine equation that is neither linear, nor in only one variable.

Example 5. Lagrange's four-square theorem
Consider the parametric Diophantine equation

$$x^2 + y^2 + z^2 + w^2 = a. \tag{17}$$

To correspond to our formal Definition 3 we should write this

$$\begin{aligned} D(a; x, y, z, w) &= x^2 + y^2 + z^2 + w^2 - a \\ D(a; x, y, z, w) &= 0, \end{aligned} \tag{18}$$

but transposing the parameter a makes it easier to read.

After we fix the parameter a we can ask whether (17) has any solutions in unknowns (x, y, z, w). Let us try this with $a = 21$. Clearly $21 = 16 + 4 + 1 = 4^2 + 2^2 + 1^2 + 0^2$, so one solution is $x = 4,\ y = 2,\ z = 1,\ w = 0$; but we can permute these values and also negating them since the squaring eliminates the signs. Hence another solution is $x = 0,\ y = -4,\ z = -1,\ w = 2$.

The fact that equation (17) is solvable in integers (x, y, z, w) for *any* natural number a is called Lagrange's four-square theorem, and was proved by Lagrange in 1770. $\triangle$

2.4 Diophantine sets

Intimately related to the Diophantine equations are the *Diophantine sets*, which will be of great importance in this text. Using parametric Diophantine equations, we can define sets of n-tuples. Suppose we have a parametric Diophantine equation that defines a set S. We take an n-tuple of natural numbers to be the parameters and fix these values. Now we have an "ordinary" (non-parametric) Diophantine equation which may or may not have solutions for its unknowns (we still don't care about what these solutions are). If it does have a solution, the fixed n-tuple is a member of the set S, and if it does not, it is not a member of the set.

Definition 4. Diophantine representation
For a parametric Diophantine equation and a set S, we have a Diophantine representation of S given by

$$(a_1, \ldots, a_n) \in S \iff \exists x_1 \ldots x_m [D(a_1, \ldots, a_n; x_1, \ldots, x_m) = 0]. \tag{19}$$

△

Definition 5. Diophantine set
A set S of n-tuples which has a Diophantine representation will be called a Diophantine set. The number n is called the dimension of S. △

Every Diophantine set has infinitely many Diophantine representations, since Diophantine equations are essentially polynomials, which can be multiplied by a constant without changing the solutions. They also have the properties of being closed under union and intersection if the dimension is the same, though it is not always the case that the *complement* of a Diophantine set is also a Diophantine set.

Example 6. A finite set of integers
In Example 2 we saw that the Diophantine equation $x^5 - 36x^3 - 84x^2 - 37x - 84 = 0$ has integer solutions $x = -3$, $x = -4$ and $x = 7$. Hence this equation can be used as a representation of the set $I = \{-3, -4, 7\}$:

$$x \in I \iff x^5 - 36x^3 - 84x^2 - 37x - 84 = 0. \tag{20}$$

For this example we see that we have no parameters, but merely a set of solutions to a Diophantine equation. If we want it to look more like Definition 4, we could consider x here as a parameter and use a random "dummy variable" as an unknown, e.g.

$$x \in I \iff \exists y [x^5 - 36x^3 - 84x^2 - 37x - 84 = 0]. \tag{21}$$

△

Example 7. The natural numbers
We could take $a \in \mathbb{N} \iff \exists x[x = a]$ and only allow non-negative values of x and we would have a canonical and trivial Diophantine representation of the set $\mathbb{N}$. If we wanted to complicate things just for fun we may use the equation (17) to obtain a representation of the natural numbers. Then we get

$$a \in \mathbb{N} \iff \exists xyzw [x^2 + y^2 + z^2 + w^2 = a]. \tag{22}$$

The left implication is because a sum of squared integers will be always be a natural number and the right implication holds because of Lagrange's four-square theorem, stating that every natural number can be written as the sum of four squares. △

Sometimes it is more convenient to think of Diophantine sets than of Diophantine equations; instead of asking whether a Diophantine equation has a solution, we can ask if a particular set of n-tuples is Diophantine (i.e., if it has a Diophantine representation).

2.5 Exponential Diophantine equations

Roughly we can say that if a Diophantine equation has as an additional variable or variables occurring as exponents, it is an exponential Diophantine equation. These are equations of the form

$$E_1(x_1, \ldots, x_m) = E_2(x_1, \ldots, x_m), \tag{23}$$

where E_1 and E_2 are expressions constructed from variables and particular natural numbers using addition, multiplication, and exponentiation. We don't have 0 on the right hand side because we don't allow subtraction here (so we can't shuffle all terms to one side).

Example 8. Fermat's Last Theorem
A legendary example of an exponential Diophantine equation is the famous conjecture of Fermat, later to be proven true by Andrew Wiles in 1994 — a seldom seen success story. Indeed, this now theorem occurs in literature about Diophantine equations as an example of an open problem that might not be solved for another century or more (this is the case in for example [3] and [4] in the Bibliography). How extraordinary then that only decades or less after these books were published, the solution was fully settled.

> **Theorem 1.** Fermat's Last Theorem
> The equation $a^n + b^n = c^n$ has no non-trivial solution in integers a, b, c for any integer value of $n > 2$.

This equation is Diophantine in the sense that we are looking only for integer solutions, and exponential because the unknown (here n) is an exponent. △

It is reasonable to assume that Hilbert did not include exponential Diophantine equations in his demand for a process to solve any Diophantine equation.

That is, the process or algorithm we are looking for need not be able to solve exponential Diophantine equations. In sections 2.1 and 2.2 we talked about how to find solutions for different types of Diophantine equations, not only about telling whether there are any. Methods for finding solutions to exponential Diophantine equations include examining divisibility properties of integers or numbers over other algebraic fields, numeric approximations, and even trial and error. Since we're really not interested in finding particular solutions for any type of Diophantine equations when discussing Hilbert's tenth problem, we won't develop that further here. In previous sections methods for finding solutions were discussed so as to gain some familiarity with Diophantine equations, but here it would more likely have the opposite effect of drawing attention from the main topic.

Just like with genuine Diophantine equations, exponential Diophantine equations can be used to form representations of sets (in analogue with Definition 4).

Definition 6. Exponential Diophantine set
If a set is such that it has an exponential Diophantine representation, in other words if it can be represented by an exponential Diophantine equation, we will call it an exponential Diophantine set. △

One might think that this would enable us to represent more sets, i.e. there would be sets which have an exponential Diophantine representation but no ordinary Diophantine representation. However this turns out to not be the case, by a far from trivial result. It will become evident in later sections why exponential Diophantine equations played such an important role for the solution of Hilbert's tenth problem.

2.6 Diophantus himself

We conclude this section with a (very) brief historical comment. Why are these equations called "Diophantine", you may wonder. Like so many other mathematical objects of whatever form, the name comes from a person. The near-mythical Diophantus of Alexandria was a Greek mathematician, sometimes called "the father of Algebra", is the name of the person in this case. He wrote the famous book series known as *Arithmetica*, many of which are now lost, dealing with solving algebraic equations. Not much is known of Diophantus' personal life, and it is even somewhat unclear when he lived, but he is believed to have made most of his work around 250 CE.

3 Logic

The solution to Hilbert's tenth problem is a fascinating interplay between number theory and logic and computability theory. Working from two different angles, progress in these separated fields morphed together and formed an unexpected result. Having talked about the number theoretical background in Chapter 2, we will now take the other perspective.

Though logic is a broad word spanning from philosophy to mathematics and perhaps via computer science back again, the aspect with which we will concern ourselves here is applying logical symbolism and terminology to our Diophantine sets and representations. We will also talk a bit about Turing machines.

3.1 The logical terminology

Symbolism and logic is intimately related. We have already made use of of the so called *existential quantifier* when stating a Diophantine representation (Definition 4). If the reader did not recognise that symbol, now is the time to exhale in relief. The existential quantifier is written $\exists$ in symbols, and spoken "there exists". This is hopefully self-explanatory or well-known since before. We will also make use of the logical connectives conjunction, $\wedge$, meaning "and", and disjunction, $\vee$, meaning "or". These are self-explanatory as well, but we shall make note of the fact that the "or" is an inclusive or, which might differ from ordinary language. That is, the statement "The author of this text is female or likes logic" is true, even though both of the disjuncts are true (indeed I am both female and like logic).

These are the only logical symbols we allow in our restriction to Diophantine language. Alas, we may mention the universal quantifier $\forall$, that means "for all", and the negation connective $\neg$ that negates a statement and makes true false, and false true. We will get back to the universal quantifier later when we discuss the historical step towards the solution of Hilbert's tenth problem that is known as the *Davis' normal form*.

3.2 Properties

A *property* $P(a)$ in logic is a predicate such that for some input $a \in \mathbb{N}$ it is either true or false. Because of this, each property induces a set of natural numbers, where a is a member of the set if $P(a)$ is true, an otherwise not. This set may or may not be Diophantine, and we say that the property is Diophantine if and only if its corresponding set is.

Example 9. The even numbers
Let's consider a very basic example of a Diophantine property — the property of being an even number. This is by definition the same thing as to say that the set of even numbers is a Diophantine set, call it, say, $\mathcal{E}$. To establish this we need to find a Diophantine representation (4) of said set. Take for example the representation

$$a \in \mathcal{E} \iff \exists x[2x = a], \tag{24}$$

where x ranges over the integers. It should be evident that this representation for the set of even numbers will do. Now instead consider the property $Even(a)$, and take the following to be a Diophantine representation of *property* $Even$.

$$Even(a) \iff \exists x[2x = a]. \tag{25}$$

△

3.3 Relations

Generalising properties, a *relation* $R(x_0, \ldots, x_n)$ is a predicate of arity n, so that we now allow n-tuples instead of just natural number as input. So, for some input $x_0, \ldots, x_n \in \mathbb{N}$, $R(x_0, \ldots, x_n)$ is either true or false. In analogue to properties, each relation induces a set of n-tuples, where tuple $(x_0, \ldots, x_n)$ is a member of the set if and only if $R(x_0, \ldots, x_n)$ holds. We say that the relation is Diophantine if its corresponding set is. In fact we do not need to distinguish between these concepts, but rather consider a relation and the set it induces to be the very same thing. Thus we may write

$$(x_0, \ldots, x_n) \in R \iff R(x_0, \ldots, x_n). \tag{26}$$

This is of course also the case for properties.

Because of this intimate correspondence, instead of identifying a Diophantine set with a Diophantine representation, we can represent its corresponding relation with a Diophantine representation. The following definition is very similar to Definition 4, with the difference that in the former we are talking about sets and now we are talking about relations.

Definition 7. Diophantine representation of a relation
For a parametric Diophantine equation and a relation R, we have a Diophantine representation of R given by

$$R(a_1, \ldots, a_n) \iff \exists x_1 \ldots x_m[D(a_1, \ldots, a_n, x_1, \ldots, x_m) = 0], \tag{27}$$

where x_i is demanded to be integer. △

Example 10. Divisibility
An example of a binary relation between integers a and b would be where b is divisible by a : $a|b$. This relation is so commonly used that it has got its own symbol, the vertical bar. We also denote this with *infix notation*, i.e. $a|b$ rather than $|(a,b)$. This is a common simplification for binary relations (and binary operations, like $+$ and $\cdot$). A representation for this relation would be

$$a|b \iff \exists x[ax = b]. \tag{28}$$

△

Example 11. Inequality
Another example of binary relations are inequalities, "less than" and "greater than". Just as with divisibility, these are common enough to have their own symbols and to be written with infix notation. Their Diophantine representations would be

$$\begin{aligned} a < b &\iff \exists x[a + x + 1 = b], \text{ and} \\ a > b &\iff \exists x[a = b + x + 1] \end{aligned} \tag{29}$$

respectively, where x is a non-negative integer. We also have the non-strict inequalities $\leq$ ("less than or equal to") and $\geq$ ("greater than or equal to") whose representations are given by simply removing the 1 :s in the representations above. △

Example 12. Exponentiation
An interesting relation that is not binary but ternary (3-ary) is exponentiation, let us denote this $Exp(a,b,c)$. A possible representation is

$$Exp(a,b,c) \iff a = b^c. \tag{30}$$

However this is not a genuine Diophantine representation since it uses exponentiation, but rather an exponential Diophantine representation of the kind we mentioned in section 2.5. Whether it is able to transform this into a genuine Diophantine representation was a very important question for the solution of Hilbert's tenth problem. △

Basically this difference between representations for sets contra relations lies in a change of perspective and notation. The use of logical notation will prove to be more suitable when we later talk about Turing machines and their connection to Hilbert's tenth problem.

3.4 Recursively ennumerable sets

We want to say a bit more about sets. Not very surprisingly, a recursively enumerable set is a set where you could enumerate the elements via recursion. An equivalent notion is to call these sets *semidecidable*, and I think it might be more intuitive to think of them as such.

Definition 8. Listable set
A set is listable if and only if there exist an effective method that would list in some order, possibly with repetition, every element of the set sooner or later. △

In other words, we allow this method to go on forever, as long as we know that for any given element of the set, it will appear on the imagined list eventually. Here "effective method" is taken to be an algorithm in the same way the process Hilbert was asking for in his tenth problem was. For a finite set you could just write down the elements one at a time, the method doesn't have to be very effective at all. For an infinite listable set we could imagine that we tell a computer to give us the elements one at a time and put these in a list.

Example 13. The Fibonacci numbers
A recursion relation is a relation where the next element can be defined by operating on previous ones. One example of such a relation defines the Fibonacci numbers, where each number is the sum of the previous two.

$$\begin{aligned} \phi_0 &= 0 \\ \phi_1 &= 1 \\ \phi_n &= \phi_{n-1} + \phi_{n-2} \text{ for } n \geq 2, \end{aligned} \tag{31}$$

where ϕ_n is the n:th Fibonacci number. This recursive definition provides a very obvious way to sort of generate the set of Fibonacci numbers, which is hence an example of a recursively enumerable set, and equally listable since we can list it using the above process. This can also be thought of as a sequence rather than a set

$$0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots \tag{32}$$

This sequence not only infinite, but also is infinitely cool with enormous amounts of fascinating properties. I could have written this essay solely on the Fibonacci number, but since I did not we have to restrain ourselves from saying too much about them (the interested reader is encouraged to find out more

about them as soon as they have finished reading this text). However I can't resist providing a nice illustration of how these numbers relate — see Figure 1.

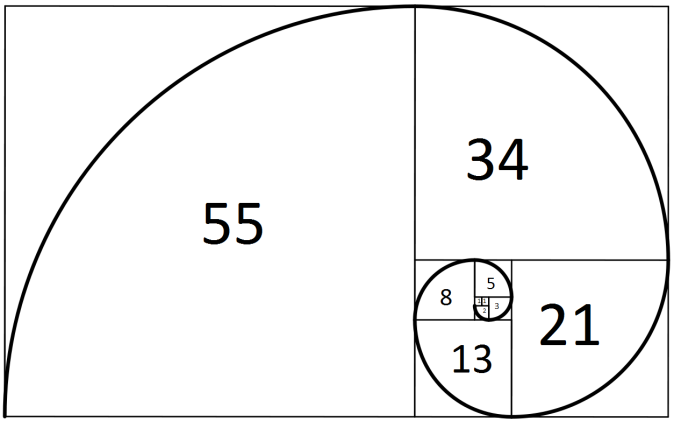

Figure 1: The Fibonacci Spiral

△

Definition 9. Decidable set
A set is decidable if and only if there exists an effective method that for every input yields an output YES if the input is a member of the set, and NO if the input is not a member of the set. △

For example, every finite set is decidable since you can list the elements of the set, and then check for your input one element at a time if they are equal. This process will stop since the list is finite. The set of prime numbers is decidable because for an input n we have a method of telling whether n is prime: we define 2 to be prime and then for every new number check whether it is divisible with a smaller number that is already on the list. We only have to try this for a finite number of numbers (more specific at most up to the integer part of the square root of n), so the process is guaranteed to terminate. However, we have a related notion for the case when the process is not required to terminate.

Definition 10. Semidecidable set
A set is semidecidable if there is a process, or algorithm, that outputs YES if the input is an element of the set, but is allowed to go on forever if it is not an element of the set. △

The difference between a decidable set and a semidecidable set therefore arises when the input is *not* a member of the set. In the former case we require

the output NO, whilst in the latter we expect no output at all. We will now motivate that if a set is Diophantine, it will also be semidecidable.

Theorem 2. Every Diophantine set is listable.

Proof. Consider a parametric Diophantine equation

$$D(a_1, \ldots, a_n; x_1, \ldots, x_m) = 0. \tag{33}$$

One at a time we can input $(n+m)$-tuples of integers into the equation (33). If it satisfies the equation, we add the first n numbers as a tuple to the list, otherwise we move on. If an n-tuple is added to the list more than once, it is not a problem since we allow repetition according to Definition 8. That we can range through these $(n+m)$-tuples to begin with might not be completely obvious either, but it was proved by Cantor that tuples of integers are countable and hence immediately listable. Let us omit that proof and accept that this kind of methodical testing can effectively be done. □

Theorem 3. Every listable set is semidecidable.

Proof. Remember the definition of semidecidable. For any input one of the cases must hold: i) the input is a member of the set, or ii) the input is not a member of the set. For case i) we require an output YES. We can compare the input to the one element of the list at a time, and whenever we find a match (which we sooner or later will, since it is guaranteed to be on there somewhere because the set was assumed to be listable) we stop and give this output. For case ii) we do the same procedure, we keep testing our input to each element of the list but we will never find a match. Hence if the set is finite, we might arrive at the conclusion that our input is not a member of the set (simply because we have gone through all of the list), and otherwise the process will go on forever and never terminate, which is allowed for semidecidable sets. □

The following corollary follows immediately from the two theorems above.

Corollary 1. Every Diophantine set is semidecidable.

3.5 Turing machines

In 1928, the same Hilbert as above posed another problem, one to be called The Entscheidungsproblem. This literally means "decision problem", so it may seem that it is no so far from Hilbert's tenth problem. However the Entscheidungsproblem has nothing to do with Diophantine equations, but rather

with logic. Here we are asked for an algorithm that takes as input a logical statement, and outputs YES if it is (in some sense) "true", and NO if it is (in the same sense) "false". I could have written my essay on this decision problem instead, but since I did not we won't go further into this.

It happens that the Entscheidungsproblem was solved within fewer years, and it was solved by the outstanding and fascinating Alan Turing (and, actually, simultaneously by Alonzo Church). Turing is known for his cracking of Enigma — a cryptographic device that was used during the World War II, his suicide that followed a penalty for being homosexual, and that with which we will concern ourselves here - his famous Turing machines.

Turing developed the notion of automated processes and algorithms in a sense that he thought of an actual mechanic device, that was capable of performing calculations. One might say the he invented a (very) primitive computer, though purely theoretical.

A Turing machine consists of an infinite *tape of squares*, that is the machine's memory. We also have a *head* that can move along the tape and put symbols in the squares. In addition the machine can be put into different *states*, like an instruction of what to do next. And that is all.

The tape could be infinite in both directions, but we can also fix one end as a starting point, and let it be infinite only to the right, so let's do that. The head can never move left of the initial square. We will need an alphabet of symbols that the head is allowed to write. This might consist of only $0, 1$ and empty squares, or we might add extra symbols. Let us reserve a symbol $\star$ for the initial square that is not allowed to be changed or written anywhere else on the tape.

We can program the machine with instructions to perform actions in different steps. For each step the machine will start in a state, then read the square the head is placed above, then either i) write something else on that square i.e. change the symbol, ii) move the head to the right, or iii) move the head to the left, and then put itself in another state (though it might of course be the same state).

When we start a Turing machine, the tape will be filled with an input, and the head will be above the star. It will be started in the initial state and since it is not allowed to change the star or go past it to the left, it will move to the right. What happens then depends on the instructions for the states and the initial input on the tape.

A standard question when it comes to Turing machines is whether a certain machine will *halt*, i.e. whether it will eventually stop. We could imagine this

as we input a certain number and a Diophantine set of numbers, we start the machine and we want to know whether the number is an element of the set. If the number is a member of the set the machine will halt in a state that means yes, and if it is not, the machine will either halt in a state that means no, or never halt at all. This is, of course, depending whether the set it decidable or semidecidable. As stated above, Diophantine sets are (at least) semidecidable.

3.6 The Church-Turing thesis

In Hilbert's tenth problem we are asked to provide an effective process that will satisfy some conditions. To prove that no such process exists we need to be able to make universal statements about processes. This was possible first when the algorithms were axiomatised, with the Turing machines discussed above.

To consider Problem 10 solved, we have to accept that what we prove about Turing machines and such algorithms also holds for processes of the kind Hilbert asked for. That these notions coincide is known as Church's thesis, or the Church-Turing thesis.

Thesis 1. The Church-Turing Thesis
Every set that is decidable in the intuitive sense is Turing decidable.

Here we can easily replace decidable with semidecidable. This thesis states that every intuitively computable function is Turing computable and vice versa. This is a bit odd due to the difference in formality: Turing computability is well defined meanwhile "intuitively computable", or "effectively calculable" which is taken to mean the same thing, is rather vague. Hence the label *thesis* — it's not really a theorem or conjecture since it can't be formally proven. Fortunately it is very widely accepted by almost all mathematicians. Although if we reject this thesis, we can't say anymore that Problem 10 has been solved.

4 Steps of the solution

Let us repeat that the solution to complicated mathematical problem is seldom unfolded in chronological order. We will in this chapter try to cover the main achievements, even though some of them aren't needed for the unsolvability proof anymore as the proof has been refined since it was first completed in 1970. One might say that everything began in 1944 with Emil Leon Post declaring that Hilbert's tenth problem "begs for an unsolvability proof". Upon this followed decades of work from several brilliant mathematicians all contributing to the solution.

4.1 Equivalent decision problem

The original statement of Hilbert's tenth problem concerns integers, but in section 2.3 we made a claim that it is equivalent to consider natural numbers (non-negative integers).

A given Diophantine equation may have solutions in integers, but no solutions in natural numbers. Take for example

$$\begin{aligned} D(x) &= x + 4 \\ D(x) &= 0 \end{aligned} \tag{34}$$

This equation has the obvious solution $x = -4$ when the range is integers, but no solution in natural numbers. Hence the claim that while solving Hilbert's tenth problem it will be sufficient to investigate it with the range being the natural numbers needs some motivation.

Take an arbitrary Diophantine equation

$$D(x_1, \ldots, x_m) = 0 \tag{35}$$

where we seek integer solutions, and compare to the following Diophantine equation where we seek only natural number solutions

$$D(a_1 - b_1, \ldots, a_m - b_m) = 0. \tag{36}$$

Here a solution to (36) in natural numbers $a_1, \ldots, a_m, b_1, \ldots, b_m$ would automatically yield a solution in integers $x_1, \ldots, x_m$ to equation (35) according to $x_1 = a_1 - b_1, \ldots, x_m = a_m - b_m$. The other way round, for an integer solution $x_1, \ldots, x_m$ to equation (35), we can find natural numbers $a_1, \ldots, a_m, b_1, \ldots, b_m$ that satisfies (36).

In this way the question of whether (35) is solvable in integers is reducible to the question of whether (36) have solutions in natural numbers, and hence the entire decision problem of solvability of Diophantine equations in integers reduces to the decision problem of solvability of Diophantine equations in natural numbers.

It turns out that the converse is also true: take an arbitrary Diophantine equation

$$D(a_1, \ldots, a_m) = 0 \tag{37}$$

where we seek solutions in natural numbers, and compare to the equation

$$D(w_1^2 + x_1^2 + y_1^2 + z_1^2, \ldots, w_m^2 + x_m^2 + y_m^2 + z_m^2) = 0. \tag{38}$$

Finding an integer solution to (38) gives a natural number solution to (37), and again by Lagrange's four-square theorem (remember Example 5) a natural number solution to (37) can easily be transformed into an integer solution to (38).

Since this reduction applies in both directions, Hilbert's tenth problem as a decision problem for integers and for natural numbers respectively, are equivalent.

4.2 Davis' conjecture and normal form

Eminent mathematician Martin Davis made two conjectures concerning Hilbert's tenth problem, according to himself. The first was that "a clever young Russian" would complete the unsolvability proof of Hilbert's tenth problem. This turned out to be true, but that is more of a fun fact. His more substantial conjecture was this.

Conjecture 1. The Daring Hypothesis
The class of Diophantine sets are identical to the class of semidecidable sets.

We saw in section 3.4 that every Diophantine set is semidecidable, but the opposite was not at all clear. Hence the attachment "daring", which seems to have been something Davis added himself. He had himself proven that Diophantine sets are not closed under complementation and neither are semidecidable sets: this similarity was one of the things that led him to Conjecture 1 which he stated in 1953. Davis' conjecture had plenty of striking corollaries, why many researchers held it unlikely to be true.

Making qualified guesses was not the only thing Davis did. He also obtained what came to be known as Davis' normal form; a form that all semidecidable sets could be defined in.

Theorem 4. Davis' normal form
Every semidecidable S set can be represented in the following form:

$$(a_1, \ldots, a_n) \in S \iff \exists z \forall y \leq z \exists x_1, \ldots, x_m [D(a_1, \ldots, a_n, x_1, \ldots, x_m, y, z) = 0]. \tag{39}$$

Remember that we are looking to prove that every semidecidable set is Diophantine — and a set is Diophantine if it has a Diophantine representation. We mentioned in section 3.1 that the universal quantifier $\forall$ is not allowed within our Diophantine language. Apart from this the representation (39) is purely Diophantine. This means that if we in some way could eliminate the universal quantifier, we would have proven Conjecture 1. This may sound not too difficult, but it would be another 20 years before the problem was solved.

4.3 The JR hypothesis

During the 50's and 60's the fascinating mathematician Julia Robinson pondered the connection between the exponential function and Hilbert's tenth problem. More specifically, she asked whether exponentiation is Diophantine. In terms of sets, this is the question whether the set of triples

$$E = \{(a, b, c) : a = b^c\} \tag{40}$$

is Diophantine.

Remember that this is the question of whether E has a Diophantine representation. She did not succeed to prove this, but instead proposed a hypothesis, later to be called The J.R. Hypothesis.

Conjecture 2. The J.R. hypothesis
There is a Diophantine set D of pairs (a, b) such that $(a, b) \in D \Rightarrow b < a^a$ and for every $k > 0$, there exists $(a, b) \in D$ such that $b > a^k$.

She then proved that if this hypothesis was true, it would imply that exponentiation was indeed Diophantine. Here we have a typical example of a reduction, that we spoke of in section 1.5. Instead of proving something directly, an implication was proved so that it would be sufficient (though not by any means easy) to prove the Conjecture 2.

The J.R. hypothesis suggests the existence of a set satisfying some particular conditions. Following our discussion in Chapter 3, this is equivalent to the existence of a certain relation; more specifically a relation of exponential growth. As always when claiming an existence, one needs only to provide one example to

confirm it — but such a relation proved very hard to find. Even Julia Robinson herself at one point gave up on her own hypothesis, and instead started to search for a positive solution to Hilbert's tenth problem.

4.4 Semidecidable sets are exponential Diophantine

In 1960, Martin Davis, Julia Robinson and Hilary Putnam published a joint paper in which they proved that every semidecidable set is exponential Diophantine. By that we mean that it has an exponential Diophantine representation. In their proof of this they used Davis' normal form together with methods that Robinson had discovered while working with her hypothesis and related topics.

Theorem 5. Every semidecidable set S can be represented in the following form:

$$\begin{gathered}(a_1,\ldots,a_n)\in S \iff \\ \exists x_1,\ldots,x_m[E_L(a_1,\ldots,a_n,x_1,\ldots,x_m)=E_R(a_1,\ldots,a_n,x_1,\ldots,x_m)],\end{gathered} \tag{41}$$

where E_L and E_R are exponential polynomials.

This is a proof of the analogue to Conjecture 1 with exponential Diophantine representations, as opposed to genuine Diophantine representations. Accordingly, so far we have the Davis' normal form which gives us a representation that is almost Diophantine but includes a universal quantifier, and representation 41 which doesn't have the problem with the universal quantifier but on the other hand uses exponential rather than ordinary Diophantine equations.

4.5 Semidecidable sets are Diophantine

The "clever young Russian" that Davis had conjectured would complete the unsolvability proof of Hilbert's tenth problem, appeared in 1970 and turned out to be Yuri Matiyasevich. He had been very interested in Hilbert's tenth problem and particularly in the J.R. hypothesis, patiently trying to find a Diophantine relation of exponential growth.

Matiyasevich studied the Fibonacci numbers, and found that the relation

$$n=\phi_{2m} \tag{42}$$

satisfies the conditions given in Conjecture 2. He then managed to find a Diophantine representation for this relation, i.e. he found a polynomial such that

$$n=\phi_{2m} \iff \exists x_1,\ldots,x_k[P(n,m,x_1,\ldots,x_k)=0]. \tag{43}$$

Now there was a given set that was both Diophantine, and of exponential growth, and through some reductions this proved the J.R. hypothesis, and we have already mentioned that from the J.R. hypothesis follows that every exponential Diophantine set is also genuinely Diophantine. Now together with Theorem 5: that every semidecidable set is exponential Diophantine, we can conclude that every semidecidable set is indeed Diophantine.

This result became known as Matiyasevich' theorem, sometimes called the MRDP-theorem. The latter name honours the four people who made the most important contributions to the solution of Hilbert's tenth problem during more than two decades of mathematics: Yuri Matiyasevich, Julia Robinson, Martin Davis and Hilary Putnam.

Theorem 6. The MRDP-theorem
Every listable set S of n-tuples of natural numbers has a Diophantine representation, that is

$$(a_1, \ldots, a_n) \in S \iff \exists x_1, \ldots, x_m [D(a_1, \ldots, a_n, x_1, \ldots, x_m) = 0] \tag{44}$$

or equivalently *every semidecidable set is Diophantine.*

Together with the Corollary 1, i.e. the long known fact that every Diophantine set is semidecidable, this also proves the above Conjecture 1, The Daring Hypothesis: the classes of Diophantine sets and semidecidable sets coincide.

Now suppose we want to know whether a certain Diophantine equation is solvable in integers. In short this is semidecidable: if it has a solution, we can eventually find it by trial and error. If it has no solution, the search for such would go on forever. But the problem Hilbert proposed was a decision problem, not a "semidecision problem", and hence Hilbert's tenth problem is unsolvable.

Theorem 7. The unsolvability of Hilbert's tenth problem
There is no algorithm which for every Diophantine equation would tell whether that equation has a solution or not.

References

[1] Andreescu, Titu, Andrica, Dorin and Cucurezeanu, Ion. *An Introduction to Diophantine Equations*. Birkhauser, 2010.

[2] Cutland, Nigel. *Computability*. Cambridge University Press, 1980.

[3] Matiyasevich, Yuri. *Hilbert's tenth problem*. The MIT Press, 1993.

[4] Eves, Howard. *Introduction to the history of mathematics*. 6th ed. Brooks/Cole, 1990.

[5] Reid, Constance. *Hilbert*. Springer-Verlag, 1996.