Undecidable problems about polynomials

Around Hilbert's 10th Problem

Anton Sadovnikov

Saint Petersburg State University

April 21, 2007

Abstract

In his Tenth Problem Hilbert asked for an algorithm capable to determine if an arbitrary Diophantine equation is solvable. In this paper we discuss the statement of the Hilbert's Tenth Problem, the history and the proof of its (negative) solution. Also, we discuss some other undecidable problems concerning some sorts of equations in real numbers and Diophantine games.

This review is based mainly on [Mat00] and also on [Mat93].

Contents

1 Hilbert's 10th Problem

In this section we consider the statement of Hilbert's Tenth Problem and the history of its solution.

1.1 The Statement of Hilbert's 10th Problem

Diophantine Equations A *Diophantine equation* is an equation of the form

$$D(x_1, \ldots, x_m) = 0,$$

where D is a polynomial with integer coefficients.

Hilbert's Tenth Problem David Hilbert stated the Tenth Problem as follows:

10. Entscheidung der Lösbarkeit einer diophantischen Gleichung. Eine diophantische Gleichung mit irgendwelchen Unbekannten und mit ganzen rationalen Zahlkoefficienten sei vorgelegt: *man soll ein Verfahren angeben, nach welchen sich mittels einer endlichen Anzahl von Operationen entscheiden lässt, ob die Gleichung in ganzen rationalen Zahlen lösbar ist.*

10. Determination of the Solvability of a Diophantine Equation. 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 by a finite number of operations whether the equation is solvable in rational integers.*

Decision Problems Using modern notions, we consider this problem as a *decision problem.*

A decision problem P has a positive solution provided there is an algorithm M_P which, for an arbitrary problem instance (or one can say, individual problem) P_i, will give a (correct) answer YES or NO.

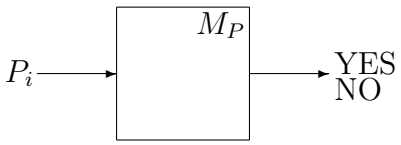

Figure 1. Decision problem P

And it has a negative solution provided there is no such algorithm.

Negative Solution of Hilbert's Tenth Problem Today, we know that Hilbert's Tenth Problem has a negative solution:

Theorem 1 (The undecidability of Hilbert's tenth problem) *There is no algorithm which, for a given arbitrary Diophantine equation, would tell whether the equation has a solution or not.*

Moreover, a stronger statement is true:

Theorem 2 (A stronger form) *There is an algorithm which, for a given algorithm A, produces a counterexample to the assumption that A solves Hilbert's tenth problem.*

Hilbert's Tenth Problem in Natural Numbers Besides the original statement of the problem, which concerns equations in integers, we can consider the similar problem in natural numbers:

$H10_{\mathbb{Z}}$: Given a diophantine equation with any number of unknown quantities and with integer numerical coefficients: *To devise a process according to which it can be determined by a finite number of operations whether the equation is solvable* **in integers**.	$H10_{\mathbb{N}}$: Given a diophantine equation with any number of unknown quantities and with integer numerical coefficients: *To devise a process according to which it can be determined by a finite number of operations whether the equation is solvable* **in natural numbers**.

For technical reasons, we consider 0 as a natural number: $\mathbb{N} = \{0, 1, 2, \ldots\}$.

Two these problems, $H10_{\mathbb{Z}}$ and $H10_{\mathbb{N}}$, turned out to be equivalent as decision problems. Two decision problems are equivalent provided they are reducible to each other.

Reducibility of Decision Problems Decision problem P is reducible to decision problem Q provided we can decide P with the use of Q as an oracle:

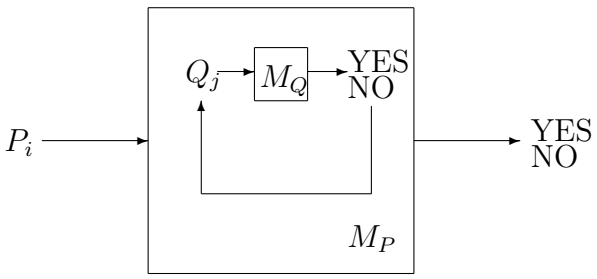

Figure 2. Reducing decision problem P to a decision problem Q

First, we prove that $\mathrm{H10}_{\mathbb{Z}}$ is reducible to $\mathrm{H10}_{\mathbb{N}}$. We have a problem instance in $\mathbb{Z}$:

$$D(x_1, \ldots, x_m) = 0, \quad x_k \in \mathbb{Z} \tag{1}$$

We build a problem instance in $\mathbb{N}$ for it:

$$D(p_1 - q_1, \ldots, p_m - q_m) = 0, \quad p_k, q_k \in \mathbb{N} \tag{2}$$

We see that if (1) has a solution then (2) has a solution because any integer is the difference of two natural numbers. If (2) has a solution then (1) has a solution because we can take

$$x_i = p_i - q_i$$

for any $i = 1, \ldots, m$.

Now, we prove the reduction in opposite direction: $\mathrm{H10}_{\mathbb{N}} \longrightarrow \mathrm{H10}_{\mathbb{Z}}$. We have a problem instance in $\mathbb{N}$:

$$D(x_1, \ldots, x_m) = 0, \quad x_k \in \mathbb{N} \tag{1}$$

We build a problem instance in $\mathbb{Z}$ for it:

$$D(p_1^2 + q_1^2 + r_1^2 + s_1^2, \ldots, p_m^2 + q_m^2 + r_m^2 + s_m^2) = 0, \tag{2}$$

$$p_k, q_k, r_k, s_k \in \mathbb{Z}$$

If (1) has a solution then (2) has a solution because any integer is the sum of four squares of integer numbers (Langrange's theorem). If (2) has a solution then (1) has a solution because we can take

$$x_i = p_i^2 + q_i^2 + r_i^2 + s_i^2$$

for any $i = 1, \ldots, m$.

Since two these problems are equivalent, for us it will be more convenient to consider the problem in natural numbers. So, through the rest of the paper all the variables assume natural values if the opposite isn't stated.

1.2 The History of the Problem's Solution

Parametric Equations A *parametric Diophantine equation* is an equation of the form

$$D(a_1, \ldots, a_n; x_1, \ldots, x_m) = 0,$$

where

- $a_1, \ldots, a_n$ are *parameters*,
- $x_1, \ldots, x_m$ are *unknowns*.

Parameters and unknowns can assume natural values only.

Diophantine Sets Consider the set S of all n-tuples $\langle a_1, \ldots, a_n \rangle$ for which the equation

$$D(a_1, \ldots, a_n; x_1, \ldots, x_m) = 0$$

has a solution in $x_1, \ldots, x_m$:

$$\langle a_1, \ldots, a_n \rangle \in S \iff \exists x_1, \ldots, x_m : D(a_1, \ldots, a_n; x_1, \ldots, x_m) = 0.$$

Sets having such representations are called *Diophantine sets*. An equivalence of the form above is called a *Diophantine representation* of the set S.

Let us consider some examples of Diophantine sets:

- The set of all squares: $a - x^2 = 0$.
- The set of all composite numbers: $a - (x_1 + 2)(x_2 + 2) = 0$.
- The set of all positive integers which are not powers of 2:
  $a - (2x_1 + 3)(x_2 + 1) = 0$.

It would be natural to ask if the complements of the sets listed above are also Diophantine. We can easily build a Diophantine representation for the complement of the first set while the answer for two other complements is not so evident:

- The set of all numbers which are not squares:
  $(a - z^2 - x - 1)^2 + ((z + 1)^2 - a - y - 1)^2 = 0$.
- Is the set of all prime numbers Diophantine?
- Is the set of all powers of 2 Diophantine?

Listable Sets Next basic notion we need is the notion of a listable set. A set S of n-tuples of natural numbers is called *listable* (or *effectively enumerable*, or *recursively enumerable*) if there is an algorithm, possibly working for an unlimited amount of time, which would print in some order, possibly with repetitions, all the elements of the set S.

It is evident that the following sets are listable:

- The set of all prime numbers.
- The set of all powers of 2.

Can we find any relation between two these classes of Diophantine sets and of listable sets? It is easy to see that one of these classes lies in the other one:

Fact 1 *Any Diophantine set is a listable set.*

Proof: Assume S is Diophantine. Then it has a Diophantine representation:

$$\langle a_1, \ldots, a_n \rangle \in S \iff \exists x_1, \ldots, x_m : D(a_1, \ldots, a_n; x_1, \ldots, x_m) = 0$$

We have to build an algorithm which lists all elements of S. First, we enumerate all the $(n+m)$-tuples of natural numbers. The algorithm must go through all these $(n+m)$-tuples, put them in the equation and print first n numbers from them if and only if the equality holds. □

It was an American mathematician Martin Davis who first stated the following conjecture:

Davis's Conjecture *Any listable set is a Diophantine set.*

At that time, this conjecture looked utterly unbelievable because of its rather striking corollaries (one of them states that there exists such a polynomial that the set of its positive values coincides with the set of all prime numbers. These corollaries are discussed in detail in [Mat00] and in [Mat93]). It was hard to believe that two notions from different fields, the theory of numbers and the theory of computability, coincide.

Negative Solution of Hilbert's 10th Problem This conjecture was very important because if it were true then the negative solution of H10 problem would immediately follow from it. It can be easily shown. First, we recollect the notion of a decidable set. A set is called *decidable* provided there is an algorithm, which determines in a finite number of steps whether an arbitrary object is its element or not. Denoting the complement of any set S by $\overline{S}$, we claim:

Fact 2 *S is decidable $\iff$ S and $\overline{S}$ are listable.*

Proof: We prove that any decidable set S (its complement $\overline{S}$) is listable. The algorithm must go through all the inputs, give them to the algorithm which "decides" S and print them if and only if they lie in S (do not lie in S).

Now, suppose that we have two algorithms M and $\overline{M}$ which list S and $\overline{S}$, respectively. To show that S is decidable we build the following algorithm. For any given input a it runs both algorithms M and $\overline{M}$ (it requires more detailed description of how one algorithm can simulate the concurrent work of two other algorithms; this description is omitted here). If a appears as the output of M then a lies in S. Else a appears as the output of $\overline{M}$ and that means that a does not lie in S. In any case our algorithm works for a limited amount of time. □

Now, let S be an undecidable listable set (the example of such set is found, for instance, in [Mat93]). It has a Diophantine representation:

$$a \in S \iff \exists x_1, \ldots, x_m : W(a, x_1, \ldots, x_m) = 0.$$

Suppose we have an algorithm which would tell whether the equation

$$W(a, x_1, \ldots, x_m) = 0$$

is solvable for a given a. Then we have an algorithm for deciding whether an arbitrary a lies in S. This leads to a contradiction with the undecidability of S.

Thus, H10 problem would have a negative solution if the Davis's conjecture were true.

Davis's Normal Form Martin Davis tried to prove his own conjecture and he obtained that every listable set has an *almost* Diophantine representation (which was called Davis's normal form).

Theorem 3 (Martin Davis, 1953) *Every listable set S has a representation of the form*

$$\langle a_1, \ldots, a_n \rangle \in S \iff$$
$$\exists z \; \forall y \leq z \; \exists x_1, \ldots, x_m : D(a_1, \ldots, a_n; x_1, \ldots, x_m, y, z) = 0$$

All that was left to do was to remove the universal quantifier.

Exponential Diophantine Equations Finally, a group of American mathematicians (Martin Davis, Hillary Putnam and Julia Robinson) succeded to remove the universal quantifier though at the cost of considering a wider class of equations. It was the class of *exponential* Diophantine equations.

Exponential Diophantine equation is an equation of the form

$$E_L(x_1, \ldots, x_m) = E_R(x_1, \ldots, x_m),$$

where E_L and E_R are *exponential polynomials*. The exponential polynomials are functions in several variables constructed with the use of integers and traditional laws of addition, multiplication and *exponentiation*. The example of such exponential Diophantine equation is as follows:

$$(x+1)^{y+2} + x^3 = y^{(x+1)^x} + y^4.$$

Similarly to parametric Diophantine equations, one can define parametric exponential Diophantine equations.

So, it was obtained that every listable set has an *exponential* Diophantine representation:

Theorem 4 (Davis, Putnam, Robinson, 1961) *For every listable set S of n-tuples of non-negative integers there is a representation of the form*

$$\langle a_1, \ldots, a_n \rangle \in S \iff \exists x_1, \ldots, x_m :$$
$$E_L(a_1, \ldots, a_n; x_1, \ldots, x_m) = E_R(a_1, \ldots, a_n; x_1, \ldots, x_m)$$

where E_L and E_R are exponential polynomials.

We could easily obtain a genuine Diophantine representation from an exponential Diophantine representation if only the set $S = \{\langle a, b, a^b\rangle\}$ were Diophantine (or one can say, if only the exponentiation function were Diophantine).

Assume S is Diophantine. Then it has a Diophantine representation:

$$\langle a, b, c\rangle \in S \iff \exists z_1, \ldots, z_m : A(a, b, c, z_1, \ldots, z_m) = 0.$$

Now, suppose that we have an exponential Diophantine representation of some listable set. Let it look like our sample exponential Diophantine equation:

$$(x+1)^{y+2} + x^3 = y^{(x+1)^x} + y^4.$$

The genuine Diophantine representation obtained from the exponential one (with the use of $A(a, b, c, z_1, \ldots, z_m)$) looks like:

$$A^2(x+1, y+2, s', z_1', \ldots, z_m') + A^2(x+1, x, s'', z_1'', \ldots, z_m'') +$$
$$A^2(y, s'', s''', z_1''', \ldots, z_m''') + (s' + x^3 - s''' - y^4)^2 = 0$$

One can easily see that two these representations are indeed equivalent:

$$A^2(x+1, y+2, s', z_1', \ldots, z_m') + A^2(x+1, x, s'', z_1'', \ldots, z_m'') +$$
$$A^2(y, s'', s''', z_1''', \ldots, z_m''') + (s' + x^3 - s''' - y^4)^2 = 0$$
$$\iff \begin{cases} A(x+1, y+2, s', z_1', \ldots, z_m') = 0 \\ A(x+1, x, s'', z_1'', \ldots, z_m'') = 0 \\ A(y, s'', s''', z_1''', \ldots, z_m''') = 0 \\ s' + x^3 - s''' - y^4 = 0 \end{cases}$$
$$\iff \begin{cases} (x+1)^{y+2} = s' \\ (x+1)^x = s'' \\ y^{s''} = s''' \\ s' + x^3 - s''' - y^4 = 0 \end{cases}$$
$$\iff (x+1)^{y+2} + x^3 = y^{(x+1)^x} + y^4$$

Thus, the last step in proving Davis's conjecture was to show that exponentiation is Diophantine.

Julia Robinson Predicates It could be done with the use of result obtained by Julia Robinson much earlier:

Theorem 5 (Julia Robinson, 1952) *There is a polynomial $A(a, b, c, z_1, \ldots, z_m)$ such that*

$$a^b = c \iff \exists z_1, \ldots, z_m : A(a, b, c, z_1, \ldots, z_m) = 0$$

provided that there is a two-parametric Diophantine equation

$$J(u, v, y_1, \ldots, y_w) = 0$$

such that

- *in every solution of the equation we have* $u < v^v$*;*
- *for every* k *there is a solution such that* $u > v^k$*.*

The sequence satisfying two these relations is also said to satisfy *the relations of exponential growth*. One can easily construct many examples of such sequences, but it was not known any example of such sequence which satisfied also some two-parametric Diophantine equation.

Finally, in 1970 Russian mathematician Yury Matiyasevich managed to find such an equation with an exponentially growing solution. He considered the sequence $v = \Phi_{2u}$ (where Φ_i is the sequence of Fibonacci numbers: $\underline{0}, 1, \underline{1}, 2, \underline{3}, 5, \underline{8}, 13, \underline{21}, \ldots$). It is a sequence of exponential growth. The corresponding two-parametric Diophantine equation looks as follows:

$$v = \Phi_{2u} \iff \exists a, b, c, d, e, g, h, k, l, p, q, r, x, y : \begin{cases} u + (a-1) = v \\ v + b = l \\ l^2 - lk - k^2 = 1 \\ g^2 - gh - h^2 = 1 \\ l^2 c = g \\ ld = r - 2 \\ (2h+g)e = r - 3 \\ x^2 - rxy + y^2 = 1 \\ lp = x - u \\ (2h+g)q = x - v \end{cases}$$

Formally speaking, it is not a Diophantine representation. But we can immediately obtain a Diophatine representation from it using the equivalence

$$D_1^2 + D_2^2 = 0 \iff \begin{cases} D_1 = 0 \\ D_2 = 0 \end{cases}$$

It is left to the reader to show that we can also use logical *or* in Diophantine representations.

DPRM-theorem It was the last step in proving the Davis's conjecture turning the conjecture into a theorem. This theorem is known as DPRM-theorem (after Davis, Putnam, Robinson, Matiyasevich):

Theorem 6 (DPRM-theorem, 1970) *Every listable set* S *of* n*-tuples of non-negative integers has a Diophantine representation, that is*

$$\langle a_1, \ldots, a_n \rangle \in S \iff$$

$$\exists x_1, \ldots, x_m : D(a_1, \ldots, a_n; x_1, \ldots, x_m) = 0$$

for some polynomial D *with integer coefficients.*

2 Proof of DPRM-theorem

Now we shall prove the DPRM-theorem. The proof consists of two parts:

1. First, we prove that *exponentiation is Diophantine*.
2. Then, we prove that *any listable set is Diophantine*. This is done by showing that *any listable set has an exponential Diophantine representation*. It was shown in previous section how one can obtain a genuine Diophantine representation from an exponential one with the use of Diophantine representation of exponentiation function.

2.1 Exponentiation Is Diophantine

The proof of this part is omitted due to its rather technical nature (those interested in complete proof may find it in [Mat00] and in [Mat93]). We only show the view of Diophantine representation of exponentiation function.

As it was shown in the previous section of this paper, the crucial role in proving Davis's conjecture was played by the sequence of even-numbered Fibonacci numbers:

$$\phi(n) = \Phi(2n).$$

One can define this sequence in a recursive way:

$$\phi(0) = 0;\ \phi(1) = 1;$$

$$\phi(n+2) = 3\phi(n+1) - \phi(n).$$

We can consider a sequence of a generalized form:

$$\alpha_b(0) = 0;\ \alpha_b(1) = 1;$$

$$\alpha_b(n+2) = b\alpha_b(n+1) - \alpha_b(n),$$

where

$$b \geq 2.$$

The sequence $\phi(n)$ coincides with the sequnce $\alpha_b(n)$ if $b = 3$.

One can find a Diophantine representation of the sequence $\alpha_b(n)$:

$$3 < b \;\&\; a = \alpha_b(c) \iff \exists r, s, t, u, v, w, x, y : \begin{cases} 3 < b \\ u^2 - but + t^2 = 1 \\ s^2 - bsr + r^2 = 1 \\ r < s \\ u^2 \mid s \\ v = bs - 2r \\ w \equiv b \pmod{v} \\ w \equiv 2 \pmod{u} \\ 2 < w \\ x^2 - wxy + y^2 = 1 \\ 2a < u \\ 2a < v \\ a \equiv x \pmod{v} \\ 2c < u \\ c \equiv x \pmod{u} \end{cases}$$

To show that this representation is indeed Diophantine we have to write relations "less then" ($<$), "divides" ($|$) and "is equivalent modulo" ($\equiv$) in a Diophantine form. For example, $3 < b$ can be rewritten as $b = 4 + z$ where z is a new variable assuming natural values. The other cases are also evident and hence are left to the reader.

A Diophantine representation of exponentiation is built with the use of the representation of the sequence $\alpha_b(n)$:

$$p = q^r \iff \left[\begin{array}{l} q = 0 \;\&\; r = 0 \;\&\; p = 1 \\ q = 0 \;\&\; 0 < r \;\&\; p = 0 \\ \exists b, m : \begin{cases} b = \alpha_{q+4}(r+1) + q^2 + 2 \\ m = bq - q^2 - 1 \\ p < m \\ p \equiv q\alpha_b(r) - (b\alpha_b(r) - \alpha_b(r+1)) \pmod{m} \end{cases} \end{array} \right.$$

2.2 Listable Sets Are Diophantine

Listable Sets The definition of a decidable set is as follows: *Set S is decidable if and only if there exists an algorithm M which, for an arbitrary input a, would tell (in a finite number of steps) if this element lies in S or not.* The following picture illustrates this definition:

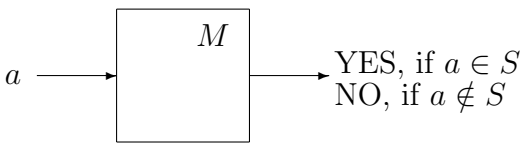

Figure 3. Decidable set S

We can slightly modify this definition to obtain a new one: *Set S is listable if and only if there exists an algorithm M_1 which, for an arbitrary input a, would tell YES in a finite number of steps if a lies in S and would never stop in opposite case.* This definition is illustrated as follows:

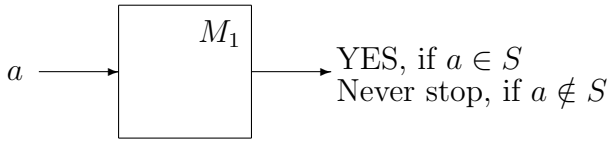

Figure 4. Listable set S (new definition)

One can show that this new definition of a listable set is equivalent to the old definition given in the previous section of this paper: *Set S is listable if and only if there exists an algorithm M_2 which would print (possibly in an infinite number of steps) all elements of S.* This definition can be illustrated by the following picture:

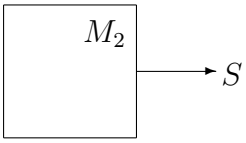

Figure 5. Listable set S (old definition)

Let us show that two these definitions of a listable set are indeed equivalent.

Proof: First, we have an arbitrary set S which is listable in 2nd sense. We prove that it is listable in 1st sense, too.

We have that there exists algorithm M_2 which prints all elements of S. Let us build M_1 from the first definition. For an input a it runs M_2 and waits until it prints a. If this happens, then M_1 stops saying YES, else M_1 works without stop.

For the opposite direction implication we give only the idea of the proof. The algorithm M_2 must run an infinite number of copies of M_1 for all possible inputs. If any copy of M_1 stops then M_2 prints the corresponding input. Since it is impossible to run an infinite number of copies of an algorithm, M_2 must start the first copy at a step 1, the second copy at a step 2, and so on. The proof is incomplete since we do not show how M_2 can simulate a concurrent work of any finite number of copies of M_1. □

Register Machines We introduce the notion of a register machine (further denoted by RM). This is a machine supplied with a finite number of *registers*

$$R1, \ldots, Rn$$

capable to store integers. Its work is controlled by a program which looks like a set of *instructions* or *states* (which is the same since there is a bijection between

the states and the instructions which must be done in any state). RM has a finite number of possible states:

$$S1, \ldots, Sm.$$

Instructions can be of three types:

1. $Sk : Rl++;\ Si$
2. $Sk : Rl--;\ Si;\ Sj$
3. $Sk :$ STOP

The instruction of the first type means that RM must increment (by one) the value stored in register Rl and jump to the state Si. The instruction of the second type means that RM must decrement (by one) the value stored in Rl and jump to Si if this value is not equal to zero already, and jump to the state Sj in opposite case. The instruction of the third type makes RM stop.

This machine is equivalent to Turing machine.

Protocol The work of the RM can be recorded in the form of *protocol*:

	q	$\ldots$	$t+1$	t	$\ldots$	0
S_1	$s_{1,q}$	$\ldots$	$s_{1,t+1}$	$s_{1,t}$	$\ldots$	$s_{1,0}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
S_m	$s_{m,q}$	$\ldots$	$s_{m,t+1}$	$s_{m,t}$	$\ldots$	$s_{m,0}$
R_1	$r_{1,q}$	$\ldots$	$r_{1,t+1}$	$r_{1,t}$	$\ldots$	$r_{1,0}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
R_n	$r_{n,q}$	$\ldots$	$r_{n,t+1}$	$r_{n,t}$	$\ldots$	$r_{n,0}$
Z_1	$z_{1,q}$	$\ldots$	$z_{1,t+1}$	$z_{1,t}$	$\ldots$	$z_{1,0}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Z_n	$z_{n,q}$	$\ldots$	$z_{n,t+1}$	$z_{n,t}$	$\ldots$	$z_{n,0}$

The t-th column corresponds to the moment of time t. For some reasons which will be evident later we choose the order of columns from right to the left. Here we set:

$$s_{l,t} = \begin{cases} 1, & \text{if RM is in state } S_l \text{ at a step } t \\ 0, & \text{if RM is in any other state at a step } t \end{cases}$$

and

$$r_{l,t} = \text{value of } R_l \text{ at a step } t$$

Also we introduce auxiliary *zero indicators* $Z1, \ldots, Zn$:

$$z_{l,t} = \begin{cases} 0, \text{ if } r_{l,t} = 0 \\ 1, \text{ otherwise} \end{cases}$$

Initial and Final States We make some arrangements on initial and final states of the RM. Without loss of generality, we count that the following relations concerning the initial state of RM hold:

$$s_{1,0} = 1; \quad s_{2,0} = \ldots = s_{m,0} = 0$$

$$r_{1,0} = a; \quad r_{2,0} = \ldots = r_{n,0} = 0$$

That means that RM begins its work at the state $S1$ and input a is placed in the register $R1$.

At the final state the following relations are supposed to be held:

$$s_{m,q} = 1; \quad s_{1,q} = \ldots = s_{m-1,q} = 0$$

$$r_{1,q} = \ldots = r_{n,q} = 0$$

That means that RM ends its work at the state Sm and it empties all the registers before the end of the work.

From Step to Step It is easy to see that since RM is fully determined, the $(t+1)$-th column of the protocol table can be uniquely recovered from its t-th column.

For example, the following relations hold for values stored in registers:

$$r_{l,t+1} = r_{l,t} + \Sigma^{+} s_{k,t} - \Sigma^{-} z_{l,t} s_{k,t}$$

- where Σ^{+} summation is over all instructions of the form $Sk : Rl + +; Si$,
- and Σ^{-} summation is over all instructions ot the form $Sk : Rl - -; Si; Sj$.

To validate this formula we notice that only one of the $s_{k,t}$ is equal to 1 for any fixed t. And we must increment the value in Rl if we use the instruction of the first form and decrement it if we use the instruction of the second form and if the value of Rl is not already equal to zero.

In similar way one can write the relations for states:

$$s_{d,t+1} = \Sigma^{0} s_{k,t} + \Sigma^{+} z_{l,t} s_{k,t} + \Sigma^{-} (1 - z_{l,t}) s_{k,t}$$

- where Σ^{0} summation is over all instructions of the form $Sk : Rl + +; Sd$,
- Σ^{+} summation is over all instructions of the form $Sk : Rl - -; Sd; Sj$,
- and Σ^{-} summation is over all instructions of the form $Sk : Rl - -; Si; Sd$.

Almost Diophantine Representation Thus, we obtain that RM will stop on input a after a finite number of steps if and only if it starts, it works from step to step and it finishes work. Or, we can say that RM will stop on input a after a finite number of steps if and only if there exist natural numbers $q, s_{k,t}, r_{l,t}, z_{l,t}$ such that the following equations hold:

$$z_{l,t} = \begin{cases} 0, \text{ if } r_{l,t} = 0 \\ 1, \text{ otherwise} \end{cases}$$

$$s_{1,0} = 1; s_{2,0} = \ldots = s_{m,0} = 0$$

$$r_{1,0} = a; r_{2,0} = \ldots = r_{n,0} = 0$$

$$r_{l,t+1} = r_{l,t} + \Sigma^{+} s_{k,t} - \Sigma^{-} z_{l,t} s_{k,t}$$

$$s_{d,t+1} = \Sigma^{0} s_{k,t} + \Sigma^{+} z_{l,t} s_{k,t} + \Sigma^{-}(1 - z_{l,t}) s_{k,t}$$

$$s_{m,q} = 1; s_{1,q} = \ldots = s_{m-1,q} = 0$$

$$r_{1,q} = \ldots = r_{n,q} = 0$$

We see that if we use this RM in the (new) definition of a listable set, we obtain that any listable set has an almost Diophantine representation, expressed via the equations above. Unfortunately, this is not a Diophantine representation since it uses indefinite number of variables.

Positional Coding of the Protocol This difficulty may be overcome in the following way. We simply delete the vertical lines in the protocol table and consider the values in one row as digits of one number (that is why the order of columns is chosen from right to the left):

	q	$\ldots$	$t+1$	t	$\ldots$	0
s_1	$s_{1,q}$	$\ldots$	$s_{1,t+1}$	$s_{1,t}$	$\ldots$	$s_{1,0}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
s_m	$s_{m,q}$	$\ldots$	$s_{m,t+1}$	$s_{m,t}$	$\ldots$	$s_{m,0}$
r_1	$r_{1,q}$	$\ldots$	$r_{1,t+1}$	$r_{1,t}$	$\ldots$	$r_{1,0}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
r_n	$r_{n,q}$	$\ldots$	$r_{n,t+1}$	$r_{n,t}$	$\ldots$	$r_{n,0}$
z_1	$z_{1,q}$	$\ldots$	$z_{1,t+1}$	$z_{1,t}$	$\ldots$	$z_{1,0}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
z_n	$z_{n,q}$	$\ldots$	$z_{n,t+1}$	$z_{n,t}$	$\ldots$	$z_{n,0}$

To speak formally, we select a number

$$b = 2^{c+1}$$

which should be greater then any $2s_{k,t}, 2r_{l,t}$ and $2z_{l,t}$. Then we define:

$$s_k = \sum_{t=0}^{q} s_{k,t} b^t, \quad r_l = \sum_{t=0}^{q} r_{l,t} b^t, \quad z_l = \sum_{t=0}^{q} z_{l,t} b^t$$

Our aim now is to rewrite relations from the almost Diophantine reoresentation of a listable set using the language of new variables s_k, r_l, z_l.

Rewriting Zero Indicator Relations Let us first rewrite the zero indicator relations:

$$z_{l,t} = \begin{cases} 0, \text{ if } r_{l,t} = 0 \\ 1, \text{ otherwise} \end{cases}$$

The binary notation of the number $2^c - 1$ looks like: $\underbrace{111\ldots11}_{c}$. Hence $x \leq 2^c - 1$ is equivalent to $x \preceq 2^c - 1$, where $\preceq$ stands for *masking* (or bitwise comparing).

Consequently, we can write

$$r_l \preceq d \quad l = 1, \ldots, n,$$

where

$$d = \sum_{t=0}^{q} (2^c - 1) b^t.$$

Numbers $z_{l,t}$ are either 0 or 1, so the z_l satisfy relations:

$$z_l \preceq e \quad l = 1, \ldots, n,$$

where

$$e = \sum_{t=0}^{q} b^t.$$

Consider the number $r_{l,t} + 2^c - 1$. If $r_{l,t} = 0$, then the binary notation of this number, padded to the length $c + 1$, is

$$01\ldots1.$$

If $r_{l,t} > 0$, then its binary notation looks like

$$1 * \ldots * .$$

In other words, the leading $(c+1)$-th digit of $r_{l,t} + 2^c - 1$ is always equal to $z_{l,t}$ and hence

$$2^c z_{l,t} = (r_{l,t} + 2^c - 1) \wedge 2^c,$$

where $\wedge$ stands for bitwise *and* (or digit-by-digit multiplication). Consequently, the definition of z_l can be written as

$$2^c z_l = (r_l + d) \wedge f,$$

where

$$f = \sum_{t=0}^{q} 2^c b^t.$$

Rewriting Register and State Relations Now, we rewrite the register relations:

$$r_{l,t+1} = r_{l,t} + \Sigma^+ s_{k,t} - \Sigma^- z_{l,t} s_{k,t}.$$

Multiplying both parts by b^{t+1} and summing up for t from 0 to $q-1$, one obtains:

$$r_l = b r_l + b\Sigma^+ s_k - b\Sigma^-(z_l \wedge s_k), \quad l = 2, \ldots, n$$

and

$$r_1 = a + b r_1 + b\Sigma^+ s_k - b\Sigma^-(z_1 \wedge s_k).$$

Then, we have to rewrite the state relations:

$$s_{d,t+1} = \Sigma^0 s_{k,t} + \Sigma^+ z_{l,t} s_{k,t} + \Sigma^-(1 - z_{l,t}) s_{k,t}.$$

In similar to the register relations way, we obtain:

$$s_p = b\Sigma^0 s_k + b\Sigma^+(z_l \wedge s_k) + b\Sigma^-((e - z_l) \wedge s_k), \quad p = 2, \ldots, m$$

and

$$s_1 = 1 + b\Sigma^0 s_k + b\Sigma^+(z_l \wedge s_k) + b\Sigma^-((e - z_l) \wedge s_k).$$

Rewriting Initial and Final Relations Now, we rewrite the initial relations:

$$s_{1,0} = 1; \quad s_{2,0} = \ldots = s_{m,0} = 0$$

$$r_{1,0} = a; \quad r_{2,0} = \ldots = r_{n,0} = 0$$

It is easy to see that they are implied by rewritten register and state relations:

Relation

$$s_p = b\Sigma^0 s_k + b\Sigma^+(z_l \wedge s_k) + b\Sigma^-((e - z_l) \wedge s_k), \quad p = 2, \ldots, m$$

implies that b divides s_p for any $p = 2, \ldots, m$ and hence the last digit of s_p is zero:

$$s_{2,0} = \ldots = s_{m,0} = 0.$$

Similarly, relation

$$r_l = br_l + b\Sigma^+ s_k - b\Sigma^-(z_l \wedge s_k), \quad l = 2, \ldots, n$$

implies that

$$r_{2,0} = \ldots = r_{n,0} = 0.$$

Relation

$$s_1 = 1 + b\Sigma^0 s_k + b\Sigma^+(z_l \wedge s_k) + b\Sigma^-((e - z_l) \wedge s_k)$$

implies that the last digit of s_l is 1 and hence

$$s_{1,0} = 1.$$

We impose

$$a < 2^c$$

to be sure that

$$r_1 = a + br_1 + b\Sigma^+ s_k - b\Sigma^-(z_1 \wedge s_k)$$

implies

$$r_{1,0} = a.$$

Then, we rewrite the final relations:

$$s_{m,q} = 1; \quad s_{1,q} = \ldots = s_{m-1,q} = 0$$

$$r_{1,q} = \ldots = r_{n,q} = 0.$$

One can see that $s_{m,q} = 1$ can be rewritten as

$$s_m = b^q.$$

It is not necessary to consider all other final relations since they state that we can write any quantity of zeros before these numbers which is always true.

Exponential Diophantine Representation of Listable Set We obtain that if RM will stop on input a after a finite number of steps, then there exist b, c, d, e, f, q, $s_1, \ldots, s_m$, $r_1, \ldots, r_n$, $z_1, \ldots, z_n$ such that the following equations hold:

$$b = 2^{c+1}$$

$$r_l \preceq d, \quad l = 1, \ldots, n$$

$$(b-1)d = (2^c - 1)(b^{q+1} - 1)$$

$$z_l \preceq e, \quad l = 1, \ldots, n$$

$$(b-1)e = b^{q+1} - 1$$

$$2^c z_l = (r_l + d) \wedge f, \quad l = 1, \ldots, n$$

$$(b-1)f = 2^c(b^{q+1} - 1)$$

$$r_l = br_l + b\Sigma^+ s_k - b\Sigma^-(z_l \wedge s_k), \quad l = 2, \ldots, n$$

$$r_1 = a + br_1 + b\Sigma^+ s_k - b\Sigma^-(z_1 \wedge s_k)$$

$$s_p = b\Sigma^0 s_k + b\Sigma^+(z_l \wedge s_k) + b\Sigma^-((e - z_l) \wedge s_k), \quad p = 2, \ldots, m$$

$$s_1 = 1 + b\Sigma^0 s_k + b\Sigma^+(z_l \wedge s_k) + b\Sigma^-((e - z_l) \wedge s_k)$$

$$a < 2^c$$

$$s_m = b^q$$

The converse is also true: if some numbers $b, c, d, e, f, q, s_1, \ldots, s_m, r_1, \ldots, r_n$, $z_1, \ldots, z_n$ satisfy these conditions, then on input a RM stops after q steps. Or, we may say that rewritten relations imply original ones.

To show it, first we obtain:

$$r_{l,t} = Digit(r_l, b, t)$$

$$s_{k,t} = Digit(s_k, b, t)$$

$$z_{l,t} = Digit(z_l, b, t)$$

where $Digit(a, b, t)$ is t-th digit in base-b notation of number a. We have seen already that rewritten initial and final relations and zero indicator relations imply original ones. It is left to the reader as a technical exercise to understand why rewritten register and state relations imply original ones (only one of $s_{k,t}$ equals to 1 while the others equal to 0 for any t. Hence there is no carry from digit to digit in the state relations. Similar arguments work for the register relations).

Thus, we obtain the exponential Diophantine representation of a listable set which can be transformed into a genuine Diophantine representation of this set.

All that is left to show is that the relation of masking is (exponential) Diophantine (which means that the set of pairs satisfying this relation is (exponential) Diophantine) and digit-by-digit multiplication is (exponential) Diophantine function (which means that its graph is an (exponential) Diophantine set).

Digit **Function** Every natural number a has a unique representation of the following form:

$$a = \sum_{k=0}^{\infty} a_k b^k$$

which is called base-b notation of number a. We define $Digit(a, b, k)$ as a_k (it is a k-th digit of number a written in base-b notation counting from the rightmost digit, which is at position number 0). If the form of a in base-b notation is

$$\underbrace{\ldots a_{k+1}}_{x} d \underbrace{a_{k-1} \ldots a_0}_{y},$$

then

$$d = Digit(a, b, k) \iff \exists x, y : \{a = xb^{k+1} + db^k + y \ \&\ d < b \ \&\ y < b^k\}.$$

It is an exponential Diophantine representation of the *Digit* function.

Binomial Coefficients *Digit* function allows to build an exponential Diophantine representation of a binomial coefficient.

We can define binomial coefficients $\binom{a}{b} = C_{a,b}$ through the identity

$$(u+1)^a = C_{a,a}u^a + C_{a,a-1}u^{a-1} + \ldots + C_{a,0}$$

which must be held for any u. Fortunately, it is sufficient to treat this as an equation having a solution with a large enough value of u. This gives us an exponential Diophantine representation for binomial coefficients:

$$c = \binom{a}{b} \iff \exists u : u = 2^a + 1 \ \&\ c = Digit((u+1)^a, u, b).$$

Kummer's Theorem Consider the factorization of a binomial coefficient:

$$\binom{a+b}{b} = 2^{\alpha_2(a,b)} 3^{\alpha_3(a,b)} 5^{\alpha_5(a,b)} \ldots$$

Kummer's theorem gives a recipe of calculation of $\alpha_p(a, b)$:

Theorem 7 (Kummer) *To calculate $\alpha_p(a, b)$ one can write a and b in base-p notation and add them; the number of carries from digit to digit performed during this addition is equals to $\alpha_p(a, b)$.*

Binary Orthogonality Let us have two natural numbers in base-2 notation:

$$a = \sum_{k=0}^{\infty} a_k 2^k, \quad b = \sum_{k=0}^{\infty} b_k 2^k.$$

We call two numbers a and b *orthogonal* ($a \perp b$) provided

$$a_k b_k = 0$$

for any k.

Kummer's theorem gives us an exponential Diophantine representation of the relation of orthogonality:

$$a \perp b \iff Odd\left(\binom{a+b}{b}\right).$$

To validate this equivalence we notice that the left part ($a \perp b$) is true if and only if there is no carry from digit to digit during the calculation of the sum of a and b. The right part ($Odd(\binom{a+b}{b})$) is true if and only if $\alpha_2(a, b)$ from the factorization of $\binom{a+b}{b}$ is equal to zero, which is equivalent (due to Kummer's theorem) to the same statement: we don't have carries from digit to digit while adding a and b.

Binary Masking Let us have two natural numbers in base-2 notation:

$$b = \sum_{k=0}^{\infty} b_k 2^k, \quad c = \sum_{k=0}^{\infty} c_k 2^k.$$

We say that b is *masked* by c ($b \preceq c$) provided

$$b_k \leq c_k$$

for any k.

We can build an exponential Diophantine representation of the relation of masking:

$$b \preceq c \iff Odd\left(\binom{c}{b}\right).$$

Validation of this equivalence is left to the reader (set $c = a + b$).

Digit-by-digit Multiplication Let us have three natural numbers in base-2 notation:

$$a = \sum_{k=0}^{\infty} a_k 2^k, \quad b = \sum_{k=0}^{\infty} b_k 2^k, \quad c = \sum_{k=0}^{\infty} c_k 2^k.$$

We say that c is the result of *digit-by-digit multiplication* of numbers a and b ($c = a \wedge b$) provided

$$c_k = a_k b_k$$

for any k.

It is possible to obtain an exponential Diophantine representation of digit-by-digit multiplication:

$$c = a \wedge b \iff c \preceq a \;\&\; c \preceq b \;\&\; a - c \perp b - c.$$

Validation of this equivalence is also left to the reader.

3 Some Other Undecidable Problems

The negative solution of H10 problem allows us to derive negative solutions for some other decision problems about polynomials. First, we might be interested in solving Diophantine equations in sets different from $\mathbb{N}$ and $\mathbb{Z}$: for example, in $\mathbb{Q}$ and $\mathbb{R}$. The problem $\mathrm{H10}_{\mathbb{Q}}$ is still unsolved.

Then, we consider Diophatine games.

3.1 Passing to Rational and Real Variables

Solving in $\mathbb{Q}$ Consider the problem $\mathrm{H10}_{\mathbb{Q}}$. One can easily build a reduction of this problem to the problem $\mathrm{H10}_{\mathbb{N}}$. If we have a problem instance in rationals

$$D(\chi_1, \ldots, \chi_m) = 0, \quad \chi_1, \ldots, \chi_m \in \mathbb{Q},$$

then since any rational is an integer divided by a positive integer, we can rewrite it in the following way:

$$D(\frac{x_1 - y_1}{z+1}, \ldots, \frac{x_m - y_m}{z+1}) = 0, \quad x_k, y_k, z \in \mathbb{N}$$

and then multiply both parts by $(z+1)^d$ (assuming d is a degree of D):

$$(z+1)^d D(\frac{x_1 - y_1}{z+1}, \ldots, \frac{x_m - y_m}{z+1}) = 0, \quad x_k, y_k, z \in \mathbb{N},$$

thus obtaining a problem instance in natural numbers, which is solvable if and only if the original problem instance in rationals is solvable.

At this time, it is still an open question if there exists a reduction in opposite direction.

It was shown that the problem of decidability of Diophantine equations in $\mathbb{Q}$ and the problem of decidability of *homogeneous* Diophantine equations in $\mathbb{Z}$ are equivalent as decision problems. But both problems are still not known to be decidable or not.

Solving in $\mathbb{R}$ Now we pass to the problem $\mathrm{H10}_{\mathbb{R}}$: *to determine if an arbitrary equation of the form*

$$D(\chi_1, \ldots, \chi_m) = 0, \quad \chi_1, \ldots, \chi_m \in \mathbb{R},$$

where D is a polynomial with integer coefficients, is solvable. This problem is known to be decidable:

- For $m = 1$: Sturm's method allows to determine the solvability of an arbitrary Diophantine equation in real numbers.
- For $m > 1$: We can apply Tarski's generalization of Sturm's method

One of the ways to obtain undecidable problem in real numbers is to consider a wider class of functions D.

Adding Sine Function The first idea is to add a sine function into a function D. Consider the following equation (with D being a polynomial with integer coefficients):

$$D^2(\chi_1, \ldots, \chi_m) + \sin^2(\pi\chi_1) + \cdots + \sin^2(\pi\chi_m) = 0, \quad \chi_1, \ldots, \chi_m \in \mathbb{R},$$

which can be rewritten in the form of a system of equations:

$$\begin{cases} D(\chi_1, \ldots, \chi_m) = 0, \quad \chi_1, \ldots, \chi_m \in \mathbb{R} \\ \sin(\pi\chi_1) = 0 \\ \quad \vdots \\ \sin(\pi\chi_m) = 0 \end{cases}$$

This system is equivalent to another system of conditions:

$$\begin{cases} D(\chi_1, \ldots, \chi_m) = 0, \quad \chi_1, \ldots, \chi_m \in \mathbb{R} \\ \chi_1 \in \mathbb{Z} \\ \quad \vdots \\ \chi_m \in \mathbb{Z} \end{cases}$$

or simply to an equation

$$D(\chi_1, \ldots, \chi_m) = 0, \quad \chi_1, \ldots, \chi_m \in \mathbb{Z}.$$

Since we know that there is no universal algorithm for deciding if an arbitrary equation of the last form has a solution, there is no algorithm for deciding if an arbitrary equation of the first form has a solution. Thus, we get the following undecidable problem:

Undecidable Problem 1 *Let $\mathcal{F}_1$ denote the class of functions in several variables that can be defined by expressions constructed from real variables, the integers and the number π, combined through the traditional rules for addition, subtraction, multiplication, and composition with the sine function in arbitrary order. There is no algorithm for deciding for an arbitrary given function $\Phi(\chi_1, \ldots, \chi_m)$ from the class $\mathcal{F}_1$ whether the equation*

$$\Phi(\chi_1, \ldots, \chi_m) = 0$$

has a real solution.

If we have some class of functions for which there is no such algorithm, then any wider class also does not have such an algorithm. But we can not tell anything about any smaller class of functions. Perhaps, if we considered a subclass of $\mathcal{F}_1$ consisting of functions constructed without the use of π, there would be a "decision" algorithm for this subclass.

But the answer is also *no*. Consider the same equation in real numbers:

$$D^2(\chi_1, \ldots, \chi_m) + \sin^2(\pi\chi_1) + \ldots + \sin^2(\pi\chi_m) = 0.$$

We can eliminate the constant π by adding a new real variable:

$$\sin(\psi) = 0, \quad 2 \le \psi \le 4.$$

Rewrite inequality $2 \le \psi \le 4$ (by adding yet another real variable) as:

$$\begin{array}{rcl} 2 \le \psi \le 4 & \iff & -1 \le \psi - 3 \le 1 \\ & \iff & (\psi - 3)^2 \le 1 \\ & \iff & (\psi - 3)^2 = 1 - z^2 \end{array}$$

Hence the original equation is equivalent to the following equation in real numbers:

$$D^2(\chi_1, \ldots, \chi_m) + \sin^2(\psi\chi_1) + \ldots + \sin^2(\psi\chi_m) +$$

$$+ \sin^2(\psi) + (1 - (\psi - 3)^2 - z^2)^2 = 0$$

So, we obtain the following undecidable problem:

Undecidable Problem 2 *Let $\mathcal{F}_2$ denote the class of functions in several variables that can be defined by expressions constructed from real variables and the integers, combined through the traditional rules for addition, subtraction, multiplication, and composition with the sine function in arbitrary order. There is no algorithm for deciding for an arbitrary given function $\Phi(\chi_1, \ldots, \chi_m)$ from the class $\mathcal{F}_2$ whether the equation*

$$\Phi(\chi_1, \ldots, \chi_m) = 0$$

has a real solution.

Actually, we can consider even smaller subclass, consisting of functions only in one variable. The similar problem for this class is undecidable, too. The idea of passing to one variable bases on the following trick. Consider the map from $\mathbb{R}$ to $\mathbb{R}^m$ defined as follows:

$$\chi \mapsto \langle \chi \sin \chi, \chi \sin \chi^3, \ldots, \chi \sin \chi^{2m-1} \rangle.$$

It is easy to check that the range of this map is everywhere dense in $\mathbb{R}^m$. And hence, it could be possible to replace variables $\chi_1, \ldots, \chi_m$ by expressions $\chi \sin \chi$, $\chi \sin \chi^3, \ldots, \chi \sin \chi^{2m-1}$ in one variable.

This idea is discussed in more detail in [Mat00].

Adding Derivatives Another idea is to add not sine functions but derivatives. Let us consider the example of J. Denef and L. Lipshitz.

Consider the differential operator $t_k \frac{\partial}{\partial t_k}$. It acts on a monomial $t_k^{x_k}$ as a multiplication by x_k:

$$t_k \frac{\partial}{\partial t_k} t_k^{x_k} = x_k t_k^{x_k}$$

And

$$\Big(t_k \frac{\partial}{\partial t_k}\Big)^i t_k^{x_k} = \Big(t_k \frac{\partial}{\partial t_k}\Big)\Big(t_k \frac{\partial}{\partial t_k}\Big)^{i-1} t_k^{x_k} =$$

$$= \Big(t_k \frac{\partial}{\partial t_k}\Big)(x_k^{i-1} t_k^{x_k}) = x_k^i t_k^{x_k}$$

This operator acts on a monomial $t_1^{x_1} \dots t_m^{x_m}$ as follows:

$$\Big(t_k \frac{\partial}{\partial t_k}\Big)^i t_1^{x_1} \dots t_m^{x_m} = x_k^i t_1^{x_1} \dots t_m^{x_m}.$$

We can consider $P(t_1 \frac{\partial}{\partial t_1}, \dots, t_m \frac{\partial}{\partial t_m})$, where P is a polynomial with integer coefficients. It acts on a term $t_1^{x_1} \dots t_m^{x_m}$ as follows:

$$P\Big(t_1 \frac{\partial}{\partial t_1}, \dots, t_m \frac{\partial}{\partial t_m}\Big) t_1^{x_1} \dots t_m^{x_m} = P(x_1, \dots, x_m) t_1^{x_1} \dots t_m^{x_m}.$$

Suppose we have a function in the power series form:

$$Y(t_1, \dots, t_m) = \sum_{x_1, \dots, x_m \in \mathbb{N}} c_{x_1, \dots, x_m} t_1^{x_1} \dots t_m^{x_m}.$$

The operator $P(t_1 \frac{\partial}{\partial t_1}, \dots, t_m \frac{\partial}{\partial t_m})$ acts on function $Y(t_1, \dots, t_m)$ as element-wise multiplication by $P(x_1, \dots, x_m)$:

$$P\Big(t_1 \frac{\partial}{\partial t_1}, \dots, t_m \frac{\partial}{\partial t_m}\Big) Y(t_1, \dots, t_m) =$$

$$= \sum_{x_1, \dots, x_m \in \mathbb{N}} c_{x_1, \dots, x_m} P(x_1, \dots, x_m) t_1^{x_1} \dots t_m^{x_m}.$$

Finally, consider the equation

$$(1 - t_1) \dots (1 - t_m) P\Big(t_1 \frac{\partial}{\partial t_1}, \dots, t_m \frac{\partial}{\partial t_m}\Big) Y(t_1, \dots, t_m) = 1.$$

It can be rewritten as

$$\sum_{x_1,\ldots,x_m\in\mathbb{N}} c_{x_1,\ldots,x_m} P(x_1,\ldots,x_m) t_1^{x_1}\ldots t_m^{x_m} = \frac{1}{1-t_1}\cdots\frac{1}{1-t_m} =$$

$$= \sum_{x_1,\ldots,x_m\in\mathbb{N}} t_1^{x_1}\ldots t_m^{x_m}.$$

Hence for all $x_1,\ldots,x_m\in\mathbb{N}$:

$$c_{x_1,\ldots,x_m} P(x_1,\ldots,x_m) = 1.$$

Thus, the original differential equation has a solution in the power series form if and only if the Diophantine equation $P(x_1,\ldots,x_m)=0$ has no solution in natural numbers. So, we obtain the undecidable problem for partial differential equations:

Undecidable Problem 3 *There is no algorithm for deciding for an arbitrary polynomial P with integer coefficients whether the partial differential equation*

$$P\Big(t_1,\ldots,t_m,\frac{\partial}{\partial t_1},\ldots,\frac{\partial}{\partial t_m}\Big)Y(t_1,\ldots,t_m)=1$$

has a formal power series solution.

3.2 Diophantine Games

Diophantine games were introduced by J. P. Jones in 1974. The definition of a Diophantine game is as follows. Suppose we have one Diophantine equation:

$$D(a_1,\ldots,a_m;x_1,\ldots,x_m)=0$$

with equal numbers of parameters and unknowns, and two players:

- Alexander (who plays for parameters a_i)
- Xerxes (who plays for unknowns x_i)

The game process looks as follows. First, Alexander chooses the value of a_1. Then, (regarding the choice of Alexander) Xerxes chooses the value of x_1. Next, (regarding the choice of Xerxes) Alexander chooses the value of a_2. After that, Xerxes chooses the value of x_2. And so on till the last choice of Xerxes.

The result of a game is determined as follows:

- If *equality* holds, then Xerxes is the winner.
- If *inequality* holds, then Alexander is the winner.

Winning Strategies It is evident that Alexander has a winning strategy if and only if the following statement is true:

$$\exists a_1 \forall x_1 \exists a_2 \forall x_2 \ldots \exists a_m \forall x_m : D(a_1, \ldots, a_m; x_1, \ldots, x_m) \neq 0.$$

Similarly, Xerxes has a winning strategy if and only if

$$\forall a_1 \exists x_1 \forall a_2 \exists x_2 \ldots \forall a_m \exists x_m : D(a_1, \ldots, a_m; x_1, \ldots, x_m) = 0$$

is true. Two these statements are the negations of each other. Hence for any game D either Alexander or Xerxes has a winning strategy.

It looks quite simple but still there can be some difficulties. Consider the following example. We have the Diophantine equation

$$(x_1 + a_2)^2 + 1 = (x_2 + 2)(x_3 + 2).$$

Who has a winning strategy in this game? First, we notice that $(x_2 + 2)(x_3 + 2)$ is always a composite number. Let us intoduce the following statement:

$$A = \text{“There are infinitely many primes of the form } k^2 + 1\text{”}.$$

If A were true then for any Xerxes's choice of x_1 Alexander could choose a_2 in such way that left part would be a prime. So, we obtain that

$$\text{Alexander has a winning strategy } \iff A \text{ is true.}$$

If A were false then Xerxes could choose the value of x_1 equal to the maximal prime of the form $k^2 + 1$. After any choice of Alexander he would be capable to factor the left part of the equation. So,

$$\text{Xerxes has a winning strategy } \iff A \text{ is false.}$$

Since nobody knows at the moment for sure if A is true or not, it is also unknown who of the players has a winning strategy in this game.

Undecidable Problem for Diophantine Games The negative solution of H10 problem leads to the following undecidable problem:

Undecidable Problem 4 *There is no algorithm which would tell for an arbitrary Diophantine game which one of two players has a winning strategy.*

This problem is certainly undecidable. If it were not so, we could obtain an algorithm capable to determine the solvability of an arbitrary Diophantine equation of the form $D(x_1, \ldots, x_m) = 0$ (where Alexander has nothing to choose).

But even if we know for some game who of two players has a winning strategy, there still can be some algorithmic difficulties, which will be considered in the following paragraphs.

Algorithmic Difficulties of Alexander Let S be a Diophantine set with a non-Diophantine complement:

$$a_1 \in S \iff \exists x_1, \ldots, x_m : D(a_1; x_1, \ldots, x_m) = 0.$$

Add $a_2, \ldots, a_m$ as fictitious parameters:

$$a_1 \in S \iff \exists x_1, \ldots, x_m : D(a_1, \ldots, a_m; x_1, \ldots, x_m) = 0.$$

Let Alexander have a winning strategy in this game.

Let A_1 denote the set of all the "right" Alexander's choices of the value of a_1. It is clear that a_1 is "right" if and only if

$$\forall x_1, \ldots, x_m : D(a_1, \ldots, a_m; x_1, \ldots, x_m) \neq 0.$$

Bring the negation out of the quantifier:

$$\neg(\exists x_1, \ldots, x_m : D(a_1, \ldots, a_m; x_1, \ldots, x_m) = 0).$$

We obtained the representation of S under brackets:

$$\neg(a_1 \in S).$$

And so:

$$a_1 \in \overline{S}.$$

Hence $A_1 = \overline{S}$. And hence A_1 is non-Diophantine, that is, not listable and hence undecidable.

So, we obtain that Alexander has no algorithm to determine if an arbitrary a_1 is the "right" choice for him. Of course, this is not that tragic since Alexander can find somehow only one such "right" a_1 and use it for this game unlimited number of times.

Algorithmic Difficulties of Xerxes The algorithmic difficulties of Xerxes can be even greater. Consider the example of a Diophantine game, invented by J. P. Jones in 1982:

$$\Big\{\{a_1 + a_6 + 1 - x_4\}^2 \cdot \{\langle (a_6 + a_7)^2 + 3a_7 + a_6 - 2x_4\rangle^2$$
$$+ \Big\langle [(x_9 - a_7)^2 + (x_{10} - a_9)^2][(x_9 - a_6)^2 + (x_{10} - a_8)^2((x_4 - a_1)^2$$
$$+ (x_{10} - a_9 - x_1)^2)][(x_9 - 3x_4)^2 + (x_{10} - a_8 - a_9)^2][(x_9 - 3x_4 - 1)^2$$
$$+ (x_{10} - a_8 a_9)^2] - a_{12} - 1\Big\rangle^2 + \langle [x_{10} + a_{12} + a_{12}x_9a_4 - a_3]^2$$
$$+ [x_5 + a_{13} - x_9a_4]^2\rangle\Big\} - x_{13} - 1\Big\}\{a_1 + x_5 + 1 - a_5\}\Big\{\langle (x_5 - x_6)^2$$

$$+ 3x_6 + x_5 - 2a_5\rangle^2 + \Big\langle [(a_{10} - x_6)^2 + (a_{11} - x_8)^2][(a_{10} - x_5)^2$$
$$+ (a_{11} - x_7)^2((a_5 - a_1)^2 + (a_{11} - x_8 - a_2)^2)][(a_{10} - 3a_5)^2$$
$$+ (a_{11} - x_7 - x_8)^2][(a_{10} - 3a_5 - 1)^2 + (a_{11} - x_7 x_8)^2] - x_{11} - 1\Big\rangle^2$$
$$+ \langle [a_{11} + x_{11} + x_{11}a_{10}x_3 - x_2]^2 + [a_{11} + x_{12} - a_{10}x_3]^2\rangle\Big\} = 0.$$

It was shown by Jones that in this game Xerxes has a winning strategy but has no effectively computable winning strategy.

References

[Mat00] Yu. Matiyasevich: On Hilbert's Tenth Problem.
Pacific Institute for the Mathematical Sciences Distinguished Lecturer Series, 2000.
See also: http://www.pims.math.ca/science/2000/distchair/matiyasevich/

[Mat93] In Russian:
Yu. Matiyasevich: Desyataya Problema Gilberta.
Nauka, Moscow, 1993.
English translation:
Yu. Matiyasevich: Hilbert's Tenth Problem.
The MIT Press, Cambridge, London, 1993.
Homepage of the book: http://logic.pdmi.ras.ru/~yumat/H10Pbook/index.html

[Mat] Yu. Matiyasevich: Computation Paradigms in the Light of Hilbert's Tenth Problem.

[WWW] http://logic.pdmi.ras.ru/Hilbert10