7-2014

Constructive Mathematics in St. Petersburg, Russia: A (Somewhat Subjective) View from Within

Vladik Kreinovich
The University of Texas at El Paso, vladik@utep.edu

Comments:

Technical Report: UTEP-CS-14-44a

To appear in: Francine F. Abeles and Mark E. Fuller, *Modern Logic 1850-1950. East and West,* Birkhauser, Basel.

Constructive Mathematics in St. Petersburg, Russia: A (Somewhat Subjective) View from Within

Vladik Kreinovich

Abstract. In the 1970 and 1980s, logic and constructive mathematics were an important part of my life; it's what I defended in my Master's thesis, it was an important part of my PhD dissertation. I was privileged to work with the giants. I visited them in their homes. They were who I went to for advice. And this is my story.

Mathematics Subject Classification (2000). Primary 01A60; Secondary 01A72, 03F60.

Keywords. Constructive mathematics, history of constructive mathematics, Russian mathematics.

1. Why Constructive Mathematics Is One of the Most Important Activities in the World – As Well as Physics and Game Theory

What do we humans want?

Why science and physics are important. We want to *understand the world*, we want to predict what will happen – including what will happen if we do nothing and what will happen if we perform certain actions. This is what *physics* – and science in general – is about. Physicists come up with equations describing how the state of the world changes with time, and we would like to use these equations to come up with the actual predictions.

This work was supported in part by the National Science Foundation grants HRD-0734825 and HRD-1242122 (Cyber-ShARE Center of Excellence) and DUE-0926721.

Why is constructive mathematics important? How do we go from equations to predictions? At first glance, this is what mathematicians (especially specialists in numerical methods) are doing – and sometimes they are doing it – but in general, mathematics is about proving theorems, not generating numbers.

In Russia, many of us heard a story (possibly a legend) that once a famous mathematician, a colleague of the Nobelist physicist Lev Landau, asked Landau what he was working on. Landau wrote down a complex system of partial differential equations describing the physical phenomena that interested him at that time. After a few months, a happy mathematician came back to Landau with a thick manuscript: "I have solved your problem! It was not easy, but I have proven that your system of equations has a solution!" :-)

This may be an exaggeration, definitely Kolmogorov and other prominent applied mathematicians helped efficiently solve many complex practical problems – but this story shows that there is a need to formally distinguish between proving theorems and actually producing solutions.

This distinction is what *constructive mathematics* is about: crudely speaking, constructive mathematics is about algorithms – in constructive mathematics, existence means that we can already produce the corresponding description – and not simply that we have proven its existence.

We also want to change the word: another reason why constructive mathematics is important. In addition to understanding, we also want to *change* the world, we want to find the appropriate actions and designs that will lead to the best possible outcomes. This is what engineering is about:

- We want to design a bridge that would withstand the prevailing winds and possible hurricanes and earthquakes;
- We want to design an efficient and safe airplane;
- We want to come up with a control strategy for a vehicle which would, for example,
  - lead an emergency vehicle to its destination in the shortest possible time or
  - make a bus spend as little fuel as possible while following the prescribed route.

In all these problems, we want to actually produce a solution. Here, it is even more important to actually produce the corresponding design or control algorithm.

Yes, numerical methods aim to do just that, they even use the word "algorithm", but often, what they call an algorithm is not exactly what computer scientists would call an algorithm. Rather it is a blueprint for an algorithm. For example, Newton's method for finding a root is a potentially infinite iterative process.

- We are not given any specific recommendation on when to stop;[1] and

[1] To be more precise, we are shown several possible recommendations, and told that none of them is perfect.

- We are not sure that this method will always work – usually, we know that in many cases, it does not work.

We need a way to clearly distinguish between such heuristic "algorithms" and algorithms in the computer science sense: when the sequence of steps is pre-determined and always leads to a correct solution. Constructive mathematics provides such distinction.

Why game theory is important. Finally, when selecting an appropriate solution, we need to take into account the preferences and opinions of different people who are (or may be) potentially affected by the solution. The discipline that takes these preferences into account is well established, it goes under the somewhat misleading name of *game theory*.

People in political science and humanities in general, political leaders, spiritual leaders, business leaders, may think that they should solve these problems – and at present, in most cases, they are solving these problems now. But the goal of game theory has always been to resolve many of these problems by applying appropriate mathematical methods – and in solving such problems, specialists in game theory and decision making have succeeded a lot.

This is another reason why we need constructive mathematics. And again, in game theory and decision making, we do not just need existence proofs, we need algorithms, we need explicit solutions.

Summarizing: three things are most important:

- Physics – understood in the general sense, as a description of the physical world – which enables us to describe how the world changes;
- Constructive mathematics, which enables us to describe how to best affect the world;
- Game theory, which enables us to take into account preferences of different people.

2. From The Mathematical Viewpoint, These Three Research Areas Have Much In Common: They Are All About Important Partial Pre-Orders

Physics: causality. In physics, some things change by themselves, other things change because some objects affect other objects. Before we start studying *how* objects affect each other, it is very important to first understand *which* pairs of objects, which pairs of events can causally affect each other.

In other words, we need to understand the notion of *causality*, which, according to many physicists, is one of the most important notions of physics; see, e.g., [21].

The study of the causality relation is more important than it may seem at first glance. For example, in special relativity, even the linear structure on space-time can be determined based only on the causality relation; this result was first proven by the Russian geometer A. D. Alexandrov in 1949 [3, 5] and became widely known after a somewhat stronger result was proven by E. C. Zeeman (later of catastrophe theory fame) in 1964 [123].

From the mathematical viewpoint, causality is a *partial order*. To be more precise, it is a partial order only in relativistic physics. In Newtonian physics, with the possibility of instantaneous effect, simultaneous events can affect each other, i.e., we have $a \leq b$ and $b \leq a$, but $a \neq b$. So, causality is a *pre-order*.

Constructive mathematics: derivability relation. In constructive mathematics, there is also a natural ordering relation.

Namely, in some cases, we derive the corresponding algorithmic result "from scratch"' – similarly to the fact that in mathematics, we sometimes prove results directly from the axioms. However, in most cases, both in traditional and in constructive mathematics, we use previous results.

Ideally, we should know how exactly we use the previous results – i.e., we need to know the actual proofs. However, in many cases, it is sufficient to know which results can be derived from other results.

The study of such a "derivability" relation is known as *logic*; a derivability relation corresponding to constructive mathematics is known as *constructive logic*. Logic is indeed often helpful in proving results in both traditional and constructive mathematics. From the mathematical viewpoint, derivability is also a partial order – to be more precise, it is a *pre-order*, since for two different statements $a \neq b$, we can have a implying b and b implying a.[2]

Game theory: preference relation. Finally, in game theory, there is also a natural pre-order.

Indeed, to make a decision that takes into account individual human preferences, we need to know these human preferences. Again, ideally, we should know *why* a person prefers one alternative to another and how strong the corresponding preference is. But first we need to know which alternatives are preferable and which are not – i.e., first we need to know each person's preference relation – yet another partial pre-order.

Moreover, in decision making theory, we can restore the numerical characteristics of human behavior – so-called *utility values* – based on the corresponding preference order; see, e.g., [23, 84, 97, 105].

3. My Personal Story: How I Came to Constructive Mathematics

I was interested in mathematics and physics. I have always been fascinated by mathematics and physics. I participated in Olympiads in math and physics, I went to a math circle led by university students. When the time came for me to

[2]Moreover, many important mathematical theorems establish exactly such equivalences: when we know necessary and sufficient conditions for some property, this brings a sense of completion and satisfaction.

enter high school, I went to a special high school with an emphasis on math and physics.

Enter game theory. When I was in high school, Igor Frenkel, then a student at a similar math high school (and a winner of city math Olympiads; he is now a professor at Yale) gave me, for my birthday, the best birthday present I ever got – an exciting book on game theory. I was awed by the fact that many real-life problems can potentially be solved by reasonably convincing mathematics.

I also saw that while this theoretically is possible, the available algorithms would require an unrealistic computation time to solve complex real-life conflict situations. This was one of the first cases when I realized that many open problems are not about answering purely mathematical questions (although there are many such questions in game theory as well), but rather about coming up with efficient algorithms which would implement the known ideas and techniques.

Game theory: there is room for optimism. While the overall optimization may not be achievable, it is clear that algorithms have helped practical decision-makers. It is also clear that there is a strong need for new algorithms, algorithms which can produce optimal decisions, decisions which are better than heuristic suboptimal decisions people that use now based on their intuition and expertise.

Mathematics and physics beyond game theory: my high school experience. A game theory book further increased my interest in mathematics and physics. I wanted to read more. However, new books on mathematics and physics were difficult to buy. So to find a good book, one had to regularly go to one of the academic old books stores, where we would sometimes find monographs, edited books, journal issues. (This is, by the way, why I so much appreciated Igor Frenkel's gift.)

I often went to an academic old book store on Liteiny Prospect with my classmate Nikolay "Kolya" Vavilov. Kolya's father was a professor, so he knew in person – or heard about – many of the city's mathematicians, and the corresponding interesting personal stories added to my fascination.

Space-time geometry and physics. For example, when we came across a book on space-time geometry and space-time physics by R. I. Pimenov [102], Kolya explained to me that Pimenov spent some time in jail for his political activities.

This was not that surprising: in Stalin's times, many families had someone arrested – including my own grandfather. Many scientists and engineers were jailed, including:

- Tupolev (of the airplane fame),
- Korolev (later the leader of the successful Soviet space program),
- Lev Landau,

and many others (and there were lucky ones, who got out alive).

There was a known story that after Tupolev was arrested, the KGB told him that he could atone for his political "sins" and get released by forming a jail-based team and designing a good plane for the Motherland. They asked him to make a list of possible helpers. Tupolev was understandably afraid that the KGB would be tempted to arrest innocent people – just to make his jail team stronger – so he made a list of all the numerous specialists he knew – thinking that the KGB would not arrest everyone. It turned out that most people on his list had already been arrested.

But this was during Stalin's time, and, as Kolya explained, the unusual thing about Pimenov is that he was in jail not in Stalin's time, but under Khruschev, the Communist leader who denounced Stalin's crimes and freed people from jails and concentration camps. Kolya also mentioned that Pimenov was a student of A. D. Alexandrov – a geometer who used to be President ("Rector") of St. Petersburg University in the 1950s and 1960s (until he moved to Siberia to promote science there).

According to Kolya, Pimenov was probably the most beloved of Alexandrov's students – for his great scientific ideas and results – and probably the most hated – since Pimenov publicly accused Alexandrov of complicity with Stalin's crimes and of praising Stalin's outrageous behavior in his official speeches and articles (I think this was an unfair accusation: millions had to do that, those who refused were usually jailed themselves.)

Logic and constructive mathematics. Kolya attracted my attention to many articles in Zapiski Seminarov LOMI, a local mathematical journal, written by Yuri Matiyasevich and Vladimir Lifschitz, two young talented mathematicians who, according to Kolya, were driven not only by their love of science, but also by their competition with each other.

I later knew both, I think the competition part was, to put it mildly, exaggerated, but the papers were interesting, and their talents clear.

I joined the Mathematics department. I was fascinated by game theory, by algorithms, by physics. I was especially fascinated by the foundations of physics – so I wanted to major either in physics or in the philosophy of physics. Fate – in the avatar of our Communist dictators – decided otherwise.

It is was well known that Jews were not allowed to become students of philosophy or physics at St. Petersburg University. So, I joined the Mathematics department.

Seminars. Talk about a kid in a toy store. I immediately found three seminars which satisfied all three of my needs, and I started actively attending all three of them.

First, I attended a seminar on space-time geometry and physics led by Revolt Pimenov himself. A few years before that, Pimenov started a deep analysis of space-time and physics in general based on the the causality relation.[3]

I also started going to a seminar on game theory led by Nikolay N. Vorobiev, the leader of Russian game theory researchers [116, 117].

---

[3]It looks like this ideas was up in the intellectual air, since at that same time, in addition to Pimenov, similar ideas were proposed by the famous geometer Busemann [12] and by physicists Kronheimer and Penrose [77].

And finally, I started going to seminars on logic and constructive mathematics. In contrast to space-time physics and game theory, there were actually three different seminars:

- A city-wide official seminar, where completed results would be presented to a very general audience, including people from different schools;
- A working seminar, in which preliminary results and open problems were presented, as well as interesting papers published by others (the seminar leaders regularly assigned to seminar participants to review and present);
- An informal seminar "on systems", led by Sergey Maslov, where raw ideas were welcome, and where, in addition to logicians, interested (and interesting) people from humanities would often give presentations.

My purpose is to describe what happened at the seminars on constructive mathematics. To get a better understanding of this, let us first briefly recall what happened earlier, before the Fall 1969 when I started attending their seminars.

4. A Brief History of Constructive Mathematics up to the 1960s

Brouwer's ideas: intuitionism. The need to have efficiency in mathematics started with Brouwer's *intuitionism* [11].

Brouwer was not happy with the fact that in classical logic and in classical mathematics, a statement $A \vee \neg A$ is always true. This seemed to conflict with a reasonable intuitive understanding of "or", according to which knowing $A \vee B$ would means that we either know A or we know B. Indeed, for many open mathematical statements A, we do not know whether these statements are true or false. Brouwer therefore decided to change mathematics in such a way that it would be in better accordance with this reasonable intuition.

To capture this intent, he called this new mathematics *intuitionistic mathematics* – and he called the corresponding logic *intuitionistic logic*.

Can intuitionism ideas be described in formal terms? Brouwer's use of the term "intuitionsim" was even more appropriate since he believed that the problem with the law of excluded middle $A \vee \neg A$ comes from over-emphasizing formalisms – which are inevitably imperfect and thus, lead us astray. He believed that we should always use our intuition as an ultimate test – and he doubted that a formalism would be able to capture, for example, his ideas about the law of the excluded middle $A \vee \neg A$.

These doubts were dispelled by A. Heyting [34], who showed, in 1930, that a large portion of then intuitionistic mathematics can actually be formalized; see also [35].

Intuitionistic mathematics and logic promote effectiveness. In intuitionstic logic:

- The knowledge of $A \vee B$ means that we know either A or B,
- The knowledge of $\exists x\, Ax$ means that we can effectively produce x for which $A(x)$ is true,

- The knowledge of $\forall x\,\exists y\,A(x,y)$ means that, given x, we can effectively produce y for which $A(x,y)$ holds.

How can we describe effectiveness? Effectiveness could not be formally described at that time since in the early 1930s, there was no formal notion of an effective procedure (what we now call an *algorithm*). This formal notion came later, with the pioneering papers by Turing [114] and Church [16].

Enter constructive mathematics. By the late 1940s, the notion of an algorithm was universally accepted:

- Different versions of this definition were proven to be equivalent,
- Most procedures recognized as algorithms were shown to be covered by these definitions.

This enabled researchers to formulate the main ideas of constructive mathematics in precise terms: that $\forall x\,\exists y\,A(x,y)$ means that there exists an algorithm that, given x, returns y for which $A(x,y)$ is true.

The first idea of constructive mathematics came from Andrei A. Markov – and, as usual in the history of mathematics (and in history in general), his path to constructive mathematics was not as straightforward as it may seem now.

Andrei Andreevich Markov Jr.[4] at first chose topology as his area of mathematical interests, and he got interested in the problem of checking whether two given compact manifolds are homeomorphic. The traditional definition of a manifold is not very constructive, but it is known to be equivalent to a very constructive definition: like an assembly-required toy, each compact manifold can be represented as a finite collection of polyhedra, with faces marked so that faces marked with the same mark are glued together. (From the topological viewpoint, we can always assume that all the vertices of all the polyhedra have rational coordinates.)

In the 2-D case, there is a known algorithm for checking when two such manifolds are equivalent. Markov decided to analyze how to extend this algorithm to a 3-D case. If he succeeded in producing an algorithm, then he would just have described it as an efficient procedure, and there would have been no need for him to go into any details into what an algorithm means in the general case – all he would have needed was to show that his particular algorithm is efficient. Luckily for foundations of mathematics, Markov was proving a negative result – that no such algorithm is possible.

However, there was no well-established notion of an algorithm operating on manifolds – and without a precise mathematical notion, it is impossible to prove that no algorithm can check homeomorphism.

So, to transform his intuition into a precise proof, Markov started looking into how to formalize the notion of an algorithm operating on manifolds. To do this, he started by describing algorithms operating on real numbers.

Constructive mathematics: a general idea. Intuitively, a constructible object has a description in terms of a finite sequence of symbols. As we all know, inside a

[4] the son of A. A. Markov Sr., of the Markov processes and the Markov chains fame.

computer, every symbol is represented as a sequence of 0s and 1s, so every sequence of symbols is also represented by a binary sequence. Therefore, every constructible object can be represented as a sequence of 0s and 1s.

The simplest mathematical objects are natural numbers. So, from the mathematical viewpoint, it is natural to interpret each code of a constructible object as a natural number. A seemingly natural is to identify each binary sequence with the corresponding number. For example, a binary sequence 11 corresponds to a natural number 3, since 11_2 is a binary code for the decimal number 3_{10}.

However, this idea needs a modification. For example, two different binary sequences 0011 and 11 would then be described by the same code 3. We can avoid this problem if we first add an extra 1 in front of the original binary sequence and then convert the resulting binary sequence into a decimal code. In this case, the sequence 0011 will be transformed into a sequence 10011 and thus, will be represented by a number $10011_2 = 19_{10}$, while a sequence 11 is transformed into 111 and is thus represented by a different code $111_2 = 7_{10}$.[5]

Real numbers in constructive mathematics. In Markov's constructive mathematics, e.g., a constructive real number is simply an algorithm that transform a natural number k into a rational number r_k in such a way that $|r_k - r_\ell| \le 2^{-k} + 2^{-\ell}$. The meaning of r_k is that r_k is a 2^{-k}-approximation to the desired real number.

Each is a code in some programming language. So, we can also represent this algorithm r as a sequence of 0s and 1s – hence, as an integer code.

Real-valued functions in constructive mathematics. A constructive function f from real numbers to real numbers is a function that inputs the code of a real number x and returns the code of the real number $f(x)$[6].

Logic of constructive mathematics. Logical statements related to constructive mathematics are interpreted in accordance with a general idea.

For example, the implication $\exists x\, P(x) \to \exists y\, Q(y)$ means that there exists a constructive function f from reals to reals that is always applicable and for which $P(x)$ implies $Q(f(x))$. In other words, the above implication is interpreted as $(\exists f \in Con)(\forall x(P(x) \to Q(f(x)))$, where $f \in Con$ means that a natural number f is a code of a constructive function.

Similar interpretations can be made for more complex logical formulas as well; see, e.g., [87, 88]. As a result, we arrive at an algorithm that transforms an

---

[5]The fact that we can represent sequences of symbols by natural numbers was first discovered by Gödel and is therefore called *Gödelization*. This idea was new in the 1930s, but with the computers, it is so trivial that we feel that over-using this term to describe an otherwise clear idea may only confuse readers. Besides, the original Gödelization algorithm involved exponentiation $2^a \cdot 3^b \cdot \ldots$; in the 1930s, this was a reasonable idea but now, with the clear distinction between feasible (polynomial-time) and exponential-time (non-feasible) algorithms, it does not make sense to introduce an unnecessary exponential time into something as trivial as representing strings in a computer.

[6]It should be mentioned that constructive functions can only be applied to mathematically constructible real numbers – moreover, to compute the value $f(x)$, we must know the exact code of the program that generates the original number x.

arbitrary formula into a form $\exists x\, A$, where A is an *almost negative* formula (in the sense that only decidable formulas can occur after $\exists, \vee$.) The corresponding algorithm was first explicitly described by Nikolay Alexandrovich Shanin, one of the first converts from topology to constructive mathematics and the future leader of the St. Peterburg School of Constructive Mathematics, in [107]; see also [89].[7]

Based on this idea, Markov, Shanin, and other researchers analyzed different mathematical results to see which results are constructive and which are not; see, e.g., [13, 78, 87, 88, 90, 108].

The Markov principle. One important tool in their analyses was Markov's *principle of constructive selection* – which now is known as the *Markov Principle.* The intuitive meaning behind this principle is related to the fact that, as it is well known, there is no algorithmic way to check whether a given algorithm will stop on given data. The Markov Principle says, in effect, that if it is *not* true that the algorithm never stops, this means that this algorithm *will* stop. In more precise terms, if we have a decidable property $P(x)$ (i.e., a property for which $\forall x\, (P(x) \vee \neg P(x))$), then $\neg\neg\exists x P(x)$ implies $\exists x\, P(x)$.[8]

5. Negative Reaction to Constructive Mathematics: Why

The first reaction of the mathematical community to constructive mathematics was rather negative. The way we have just described it, the activity of constructive mathematics is reasonable and useful both for understanding mathematics and for applications of mathematics.

However, originally, the first reaction of most mathematicians to constructive mathematics was negative. There were at least five reasons for this negative reaction.

First reason: methodological. In their papers and talks, researchers in constructive mathematics did not just propose new ideas and results, they argued that, in effect, all the previous mathematical results and theories made no sense and should be replaced by their constructive versions. For example, Shanin liked to emphasize that when a property is proven to be true only almost everywhere, this result is practically useless, since we still do not have a single example of a point at which this property holds: "pochti vezde znachit neizvestno gde". I think many mathematicians would agree with this statement – but not with Shanin's conclusion that the result about the property being true almost everywhere makes no sense and should not be published.

[7] It is worth mentioning that the algorithm SH is known to be equivalent (under a suitable coding in Heyting's formalized intuitionistic arithmetic) to recursive realizability introduced by S. C. Kleene [42].

[8] From the classical viewpoint, the constructive logic of Markov's school can be completely described using the three above-described basic principles: recursive realizability, the Markov principle, and classical logic for sentences containing no constructive problems, i.e., $\exists, \vee$-free sentences [94, 113].

Other reasons. There were other reasons, of course, why the initial reception of constructive mathematics was negative.

Some of these reasons were related to the abundance of negative results and counterexamples in constructive mathematics. In the beginning, the idea of looking for a constructive proof sounded reasonable: e.g., we have a theorem that proves the existence of a solution to a differential equation, but we do not know how to actually find this solution, so let us come up with such an algorithm. In these terms, the problem sounds like the need to find an algorithm. Somewhat surprisingly, it turned out that in many cases, such an algorithm does not exist.

- A. Turing proved, in effect, that no algorithm can detect whether two real numbers are equal or not.
- E. Specker was one of the first to move from general algorithmic impossibility to specific examples, by showing, in [111], that the maximum of a computable bounded increasing sequence can be non-computable.

Second reason: communication problem. Counterexamples were the second reason for mathematicians' negative reaction to constructive mathematics. For example, in traditional calculus, there is a theorem according to which a continuous function $f(x)$ on an interval $[a, b]$ always attains its supremum at some point x. In constructive mathematics, there is a counterexample to this classical theorem: there exists a constructive function $f(x)$ from reals to reals that does not attain its supremum value on a given interval in any constructive point. When presented in this form, it is an interesting negative result about algorithms: that we cannot algorithmically produce a point x_0 at which $f(x_0) = \sup\limits_{x \in [a,b]} f(x)$.

However, most mathematicians understood this result – by literally interpreting the constructivists' existential quantifier – as claiming that no such point x_0 exists at all. Since their intuition of real numbers included non-constructive numbers (e.g., numbers coming from physical measurements), this non-existence could not be explained by just considering mathematically constructible real numbers.

Third reason: overemphasis on negative results. The second reason is closely related to the third reason – originally, constructive mathematicians placed too much emphasis on counterexamples and negative results (showing that there is no universal algorithm for solving different general problems), while under-emphasizing the more useful part of constructive mathematics: providing positive algorithmic results.

If a general algorithm is impossible, then usually it is possible to have algorithms that work under certain conditions, and/or algorithms that solve a slightly weaker problem. For example, in the above problem, it is possible, for any given accuracy ε, to algorithmically produce a point x_0 for which $f(x_0) \geq \sup f(x) - \varepsilon$. From the viewpoint of practically solving optimization problems, this is quite enough.

Fourth reason: original papers are difficult to read. The fourth reason was that the original papers were very difficult to read. Constructive mathematics tries to describe algorithms, algorithms that deal with higher-order objects – like $f(x)$ takes an algorithm as an input and returns an algorithm as an output. In the early 1950s, before the first programming languages appeared, there was no easy way to describe complex algorithms in a clear understandable way.

Even now, with multiple user-friendly programming languages, it is difficult to describe higher-order algorithms, with functions as inputs and functions as outputs, in an unambiguous and easily readable way. It is difficult to read these algorithms even now – even for computer scientists. Imagine how a mathematician would have felt about such code in the 1950s.

When I started learning constructive mathematics, we did not read Shanin's fundamental papers such as [108], since they were too difficult. Instead, we relied on an instructor's descriptions and later re-wordings.

Fifth reason: political. There was a special political reason for this negativity. The main ideas of constructive mathematics arose in the late 1940s and early 1950s, when Stalin was still alive. That was a period when he purged the sciences which were considered to be ideologically impure:

- In 1948, genetics was condemned as a capitalist science, with researchers fired, jailed, and shot;
- Then came cybernetics and linguistics.

After these three campaigns, it looked like Stalin decided to go after physicists. A vicious media campaign was launched against "capitalist" relativity theory and quantum physics. Luckily, this campaign stopped – probably because physicists were considered to be useful in designing and improving atomic bombs. A few people who were denounced and arrested – among then, Vladimir Fock, known to physicists for Fock spaces – were soon released. Fock even had – a rarity in those days – all his belongings and manuscripts returned to him intact.

If not physics, then what? Everyone was afraid that their science was to be the next target.

And then, as A. D. Alexandrov described later, a "bomb" exploded on the ideological front: someone in the communist party noticed the philosophical differences between strict constructive mathematics – where only constructive objects exist – and traditional mathematics. He suggested that there be a "philosophical discussion" – similar to the one that preceded the bloody purge in genetics. Disaster was looming. So, A. D. Alexandrov (President of St. Petersburg University) and A. N. Kolmogorov (the most famous Soviet mathematician of that time) came up with a smart plan.

They convinced the party bosses that mathematics is too complex a science to start a discussion (at least a discussion without proper preparation). Instead, they proposed to first write a definitive book on the methodology and ideology of mathematics.

As A. D. Alexandrov explained, they were motivated by the known story about a legendary Molla Nasreddin. In this story, the Shah liked his pet donkey so much that he believed – as many pet owners do – that his pet donkey was more intelligent than most people. So, he asked Molla to teach his donkey. Molla was afraid to disobey the murderous Shah, so he agreed – but with a warning that he needed at least 15 years to do it. When his horrified wife asked how he was planning to do it, he cheerfully replied: "Do not worry. In 15 years, either I will dead, or the Shah, or the donkey".

Alexandrov and Kolmogorov turned out to be right: while they were working on the book, Stalin died, and the book – a good book actually, re-published by Dover [4] – went out without the need to send anyone to jail.

The ending was happy, but this story left a bad taste in the mouths of many mathematicians. Somewhat understandably, since mathematicians could not do much about the communist dictatorship that nearly killed them, this negative feeling was often directed towards constructive mathematicians who allegedly provoked the government's attack.

6. Constructive Mathematics in the 1970s: A Boom

When I started going to the seminars, all four reasons were slowly being overcome, and constructive mathematics – and logic – were blossoming.

Matiyasevich's solution of the 10th Hilbert problem. The big boost came from the 1970 result by Yuri Matiyasevich who solved [92, 93] the 10th of the Hilbert's 23 problems [36], challenges that 19th century mathematics presented to the 20 century. The 10th problem was about finding an algorithm for solving Diophantine equations and systems of equations – i.e., polynomial equations in which all variables are natural numbers. Matiyasevich proved that no such general algorithm is possible.

Interestingly, what may have seemed, at first, like one of the many negative results turned out to be a very positive result. What Matiyasevich actually proved was that every set which can be eventually generated by some algorithm (such as, e.g., the set of all prime numbers or the set of all prime twin pairs for which both n and $n+2$ are primes) can an be represented as the set of all possible non-negative values of a polynomial of (several) integer-valued variables.

How I learned the details of Matiyasevich's result. I myself learned the details of this result – my apologies to Yuri for the coming English-language metaphor – from the "horse's mouth", i.e., from Yuri himself.

Yuri was giving a talk at the general meeting of the St. Petersburg Mathematical Society, and I was late for his talk and missed the first half. I was very upset about this, since I thought I missed a unique opportunity to learn the details. However, my colleague, Evgeny "Zhenya" Dantsin, suggested that I simply approach Yuri after the lecture, that Yuri would be glad to repeat his descriptions to me.

On my own, as a freshman student, I would not have had the chutzpah to approach a famous mathematician with such a request, but after this advice, I did – and Yuri gladly did explain things to me.[9]

Matiyasevich's result brought attention to the logicians. Matiyasevich's result focused everyone's attention in the logic group, in particular, to their results in constructive mathematics – and the positive character of Matiyasevich's result conveyed that many results of constrictive mathematics have positive algorithmic aspects.

The attitude of constructive mathematics towards non-constructive mathematics became more tolerant. The attitude of constructivists themselves somewhat mellowed. Once in a while, Shanin would repeat – paroding the official line about Marxism – that constructive mathematics is the only scientifically correct approach, but he became much more tolerant of other approaches.

When confronted with the difference between his new views and his more rigid view a few years back, he would always say, half-jokingly, that since all the atoms in the body change every seven years, he is no longer his former physical self and has therefore the right to change his opinions.

Shanin was the only one to have such serious qualms about non-constructive objects. Everyone else in the group agreed that there is some meaning to non-constructive mathematics – moreover, that there is usually even some constructive meaning in seemingly non-constructive proofs and results, and the challenge is how to extract this meaning.

Constructivism papers became more readable. The readability of papers in constructive mathematics also greatly improved. A big push for this readability came with a book by E. Bishop [7], a renowned mathematician who became interested in effective constructions and ended up writing a ground-breaking book on constructive mathematics. Bishop did not use explicit algorithms and did not prove many negative results, his approach was more general, but most of his results could be easily interpreted in Markov-Shanin constructive terms.

Before that, there was a feeling that to learn constructive mathematics, one has to grind his/her teeth and go though barely comprehensible formulas. It turns out that there is a road to constructive mathematics – a road that a working mathematician can rather easily follow.

Logicians tried their best to make their papers clear and understandable. Each paper accepted for publication for *Zapiski* was assigned to another author for what we called "eating each other": thorough checking of every single formula and every single phrase. After that, Yuri Matiyasevich and Anatol Slissenko, fearless and tireless editors, would go over every word on their own, making many suggestions (and, to our embarrassment sometimes, corrections) along the way.

[9] While I truly appreciate what Yuri did, I want to add that this was an example of the attitude that was prevalent (and actively cultivated) in our department in general, and among logicians in particular: paraphrasing Rudyard Kipling's Mowgli, we all had a strong feeling that that we are all "of one blood", that we are all brothers and sisters in mathematics and in science.

I remember how Anatol half-jokingly suggested that we erase his pencil marks before coming the next time, so that he would be able to make a different suggestion this time. This was somewhat painful but proudly painful: we all felt like Lev Tolstoy who re-wrote his *War and Peace*, I think, six times. The resulting text was not exactly of Tolstoy caliber, but still clearly improved.

7. Constructive Mathematics in the 1970s: Main Challenges

The main idea of constructive mathematics: a reminder. What were the challenges that motivated our research? To understand these challenges, let us recall the general idea of constructive mathematics:

- We start with a general class of problems,
- We try to analyze whether a general algorithm is possible for solving all the problems from this class.

Challenges naturally emerged from all the aspects of this idea: objects, analysis, and algorithms.

First challenge: the need to extend constructive mathematics to more complex mathematical objects. The first class of challenges came from the fact that most traditional results of constructive mathematics dealt with reasonably simple mathematical *objects*, such as numbers and functions. In modern mathematics and its applications, much more complex objects are used. We need to extend constructive mathematics to these more general objects.

Second challenge: to be useful for data processing, algorithms must be able to handle possibly non-constructive data. Traditional constructive mathematics dealt only with computable objects – e.g., only with computable real numbers, computable functions, etc. In practice, we need to process data coming from measurements, and, according to modern physics, the corresponding data are not necessarily computable: e.g., the results of quantum measurements are inherently random.

We therefore need to extend the algorithms of constructive mathematics to algorithms for handling these not-necessarily-computable objects.

Third challenge: the need for general ways of analyzing problems. The *analysis* of a problem in constructive mathematics was too ad hoc. Crudely speaking, every new result was, in effect, worthy of a Master's thesis or a PhD dissertation.

If we wanted constructive methods to be widely used, we could not afford a situation in which so much effort is needed to analyze the constructiveness of a situation, we needed to develop general results which would make such an analysis easier.

Fourth challenge: when an algorithm is possible, is it feasible? On the algorithm stage, if an *algorithm* has been produced, how efficient is it? An algorithm whose running time exceeds the lifetime of the Universe is clearly not very feasible. If this algorithm is not feasible, is a feasible algorithm possible?

If it is not feasible on existing computers, can computers using some novel physical phenomena make these problems feasibly solvable? And if the problem is not feasibly solvable in general, when is it feasibly solvable?

Fifth challenge: what if no general algorithm is possible? On the other hand, if a general algorithm for solving all the instances of the original problem is proven to be not possible, then the natural questions are:

- How can we relax the problem to make it possible?
- Is it possible to find a reasonable subclass of problems for which the solution is algorithmically possible?
- Is it possible to relax the requirements of the problem and have an algorithm for solving a weaker problem?
- Can computers using some novel physical phenomena make these problems algorithmically solvable?

These were the challenges that we worked on. Let us now briefly enumerate the results of this work.

8. First Challenge: Dealing With More Complex Objects – Which Objects Do We Need?

We need to look into possible application areas. Algorithms are most useful for applications. Thus, to understand which objects should concentrate on, we need to look at possible applications: which mathematical objects are needed to describe the physical world?

Newton's mechanics. Let us start with traditional Newtonian physics (for details of the corresponding physics descriptions, see, e.g., [21]). In Newtonian physics:

- We have a 3-D Euclidean space $\mathbb{R}^3$ and a 1-D time $\mathbb{R}$.
- The world consists of particles.
- The state of the world at any moment of time t can be described by listing the spatial locations $x_i(t)$ of all these particles $i = 1, 2, \ldots$
- Newton's equations – a system of ordinary differential equations – describe how the coordinate $x_i(t)$ of each particle i changes with time.

This model perfectly describes, e.g., celestial mechanics.

This is a description which is well covered by traditional constructive mathematics.

Newton's mechanics: need for approximate descriptions and the resulting mathematical objects. *Theoretically*, Newton's equations are all we need to describe Newton's physical world. However, from the *practical* viewpoint, the corresponding number of particles is too large – e.g., we have 10^{23} atoms in each macro-volume. Even modern computers, no matter how fast they are, cannot handle that many computations. So, we need to simplify the above description.

First, to describe the dynamics of a single particle i, we cannot realistically use the positions of all the other particles to predict how the location $x_i(t)$ changes. Instead, we must use a simpler description that would capture the effect of all these particles. This description is known as a *field*. For example, the gravity field describes the joint effect of all the attracting particles – without us having to specify which part of the attractive force comes from which particle.

Second, since we have too many particles, we cannot describe the state of all of them, we can only describe their averages – e.g., the density of a body at a given location instead of the exact location of each particle.

Finally, since our description is inevitably approximate, we often cannot describe the exact dynamics, we can only make approximate predictions. In precise terms, instead of the exact value, we take into account that many different values are possible, and we can predict the probabilities of different values.

From this viewpoint:

- We need *functions* to describe densities and fields;
- We need *probability distributions* to describe uncertainty – probability distributions on numbers and on functions.

Resulting challenge for constructive mathematics. We need to describe functions, and we need to describe probability distributions.

Functions can be naturally describe in constructive mathematics, but probability distributions are not so easy to describe – even probability distributions corresponding to a single random variable. This difficulty is related to the fact that in constructive mathematics, every function is continuous – informally, if we have an algorithm that is applicable to all computable real numbers, then the resulting function can be proven to be continuous.[10]

This continuity creates a challenge when we try to describe probability distributions in constructive terms. For example, a natural way to describe a probability distribution is by describing its cumulative distribution function $F(x) = \text{Prob}(X \le x)$. This function is continuous for, e.g., a normal distribution, but it is clearly discontinuous for a random variable X which takes the value 0 with probability 1. For this random variable, using the probability density function (pdf) will not help, since the corresponding pdf is not defined when $x = 0$.

The situation is even more complex for random *functions* – i.e., probability measures on the class of functions.

Relativity theory. Modern physics made the description of the physical world even more complex.

This complexity started with General Relativity, in which the space-time is a general *manifold*.

[10]This result make physical sense: in real life, if we process real values which are obtained with a higher and higher degree of accuracy by performing more and more accurate measurements, then we should be able to return the result at some point, before we know the detailed value of the inputs x – which is exactly what continuity is about.

Already Markov showed how to describe manifolds in constructive terms, but manifolds with singularities are a challenge.

Quantum physics. Quantum physics leads to yet another class of objects. Specifically, in quantum mechanics, to describe a single particle, instead of a single 3-D vector x, we need a *wave function*, i.e., a complex-valued function $\psi(x)$ which assigns to each possible location x an "amplitude". We can then estimate the probability density of a particle at location x as $|\psi(x)|^2$. To handle quantum mechanics, we therefore need to extend the traditional constructive theory from real-valued to *complex-valued* functions.

The situation becomes even more complex in quantum field theory, where instead of a function $f(x)$, we need a *functional*, i.e., a mapping $\psi(f)$ which assigns a complex value to each function f. To describe the dynamics of such states, we need *operators* which map functions into functions, etc.

In relativistic gravity, the state of the world is a manifold M with functions defined on this manifold. So, in quantum gravity, we need a wave function $\psi(M)$ which assigns a value to each such manifold M.

Non-separable spaces: an additional problem. Some of these constructions lead to *non-separable* spaces, i.e., spaces which do not have everywhere dense countable subsets. This a big problem for constructive mathematics, since usually, in constructive spaces, each object is approximated by objects represented by a finite number of symbols. There are countably many such objects, as a result of which all usual constructive spaces are separable.

Summarizing: we need to describe:

- Probability distributions,
- Manifolds with singularities,
- Functions of complex variables,
- Objects of higher order (functionals, operators, etc.), especially objects that form non-separable spaces.

9. Collaboration With Other Disciplines Was Encouraged

Collaboration is needed. Complex objects come from disciplines such as physics. Thus, to generate an adequate constructive version of the corresponding notions, it is important to collaborate with researchers from other disciplines.

Such a collaboration, and, more generally, interest in other disciplines was welcomed and encouraged.

Students were encouraged to take classes outside their discipline. Once we started working on our Master's theses, there was no formal requirement to take any classes outside the discipline (this is an arrangement very typical for Master's programs in the academic world). However, Shanin always emphasized that while there was no *requirement* to take classes outside math, a student will be considered a true gentleman or a true lady if he or she takes a year-long class or two semester-long classes elsewhere. (I myself took General Relativity.)

Seminars enhanced collaboration. At Sergey Maslov's seminar on systems, we would hear talks by linguists, historians, geoscientists, even writers and poets. We all loved it.

Conferences provided another opportunity. For example, at a school on computational complexity at a ski resort in Tsahkazdor, Armenia, during non-logical talks, many participants would quietly leave to enjoy the great skiing weather, while we – logicians from St. Petersburg – would stay and enjoy the good "intellectual weather".

Sergey Maslov often valued these non-logical talks even more than the more technical ones. In Tsahkadzor, he described his opinion with a rhyme: "Ia priehal v Tsakhadzor rasshiriat' svoy krugozor" ("I came to Tsahkadzor to broaden my horizon").

10. First Challenge: Dealing With Complex Objects in Constructive Mathematics – Main Results

As a result of collaboration with researchers from other disciplines, constructive mathematicians from St. Petersburg came up with constructive representations of the corresponding complex objects. Let us list the corresponding representations one by one.

Probability distributions. For random variables, a constructive description of *probability distributions* was proposed by Nikolay Kossovsky [53, 54].

For random processes, the corresponding description was given in [56, 57], of the example of historically the first Wiener measure – a probability measure that describes Brownian motion.

Manifolds and, more generally, metric and pseudo-metric spaces. For *manifolds*, an important result was obtained by Zhenya Dantsin: he proved the constructive version of *Sard's Lemma*, according to which the critical values of a smooth function f from one manifold to another has Lebesgue measure 0.

Some results about constructive non-smooth metric and pseudo-metric spaces – presented at the seminar but not published at that time – later appeared in [18, 61, 70, 72, 74].

In particular, for our results about space-time models (later published in [72]) Dima Grigoriev and I received first prize at the department's best student paper competition.

An interesting aspect of studying general metric spaces is estimating their size. A natural way to estimate the size of a metric space S is to use the characteristic like *ε-entropy*, which is defined as the smallest number of points such that every point from S is ε-close to one of the these points. This characteristic takes only integer values and thus it is a discontinuous (hence, not computable) function of ε. A constructive way to describe ε-entropy and other similar characteristics is given in [59] (see also [63]).

Functions of complex variables. Several problems related to functions of *complex variables* were handled in Bishop's book [7].

Significant further progress was made by Vladimir Orevkov; see, e.g., [100].

Objects of higher type. A general constructive description of *objects of higher type* – functional, operators, etc. – was proposed by Victor Chernov in [14] (see also [15]).[11]

Non-separable spaces. A constructive approach to *non-separable spaces* was developed, with Victor Chernov's guidance, by our French research visitor Maurice Margenstern [85], based on the example of the space of almost periodic functions.[12]

General set-theoretic objects. An even more general scheme – including constructive versions of all objects of the *set-theoretic* hierarchy – was described in an unpublished paper by Michael Gelfond and Vladimir Lifschitz. Their constructive version of set theory was based on the standard ZF.[13]

11. Second Challenge: Algorithms Dealing With Not-Necessarily-Computable Objects

As we have mentioned, to process real-life data, we need algorithms which can process non-constructive objects as well.

Random sequences. For example, according to quantum physics, sequences of observations are not computable, they are *random* (with respect to some computable probability measure).[14]

If we simply allow random sequences (in the formal sense proposed by Kolmogorov and Martin-Löf; see, e.g., [81]), then we get a theory which is very similar to standard constructive mathematics; this was proven by Leonid Levin [80].[15]

---

[11] It is worth mentioning that the resulting approach turned out be similar to the approach proposed in a somewhat different context by Yuri Ershov (see, e.g., [20]).

[12] Almost periodic functions were invented by Harald Bohr, a mathematician brother of the Nobelist physicist Niels Bohr.

[13] Shortly after that, another version of constructive set theory – this time based on type theory – was proposed by Per Martin-Löf [91] (see also [29, 99, 112]). Since Martin-Löf did not need to deal with the more complex axioms of ZF, his theory is much clearer and simpler than the Gelfond and Lifschitz's version – which is probably one of the reasons the reason why they never published their version.

[14] In [47], it is shown that such non-algorithmic sequences are intuitively justified. Without them, discrete transition processes (e.g., radioactive decay) would potentially lead to devices checking whether a given Turing machine halts or not.

[15] It is worth mentioning that when he presented this work in St. Petersburg, he drew a target on his flyer – expecting that in this center of constructive mathematics, he would be attacked for suggesting that non-constructive sequences are possible.

Need to go beyond random sequences. A restriction to random sequences makes sense if we believe quantum physics to be the ultimate theory of the universe. But since most physicists think that any theory may be later modified, a better idea may be *not* to impose such theory-specific restriction on possible inputs, and consider all possible real numbers as inputs.

General inputs. For an algorithm to be able to handle general inputs, a computable function $f(x)$ should use only approximate values of x, but *not* – as traditional constructive mathematics – the code of the algorithm which computes consecutive approximations to x.

- Some such "approximation-only" algorithms were presented in Bishop's book [7].
- A general description of such algorithms for objects of arbitrary type is given in above-cited Chernov's papers [14, 15].
- Vladimit Lifschitz, in [82], provided a formalism in which such generic number can be described in constructive terms – as "fillings".

Later, this field of research crystallized as *computable analysis*; see, e.g., [103, 118].

12. Third Challenge: Need For General Ways For Analyzing Problems, Towards General Constructivity Proofs

Another challenge was to find general proofs of constructivity - which would replace previous time-consuming case-by-case proofs.

Almost negative statements. This activity started with statements that *do not contain "or" or existential quantifiers* – statements which should, intuitively, be equally valid in the traditional and in constructive mathematics. However, the actual proof turned out not to be easy; this was done by Michael Gelfond [26, 27, 28].

This class includes integral equalities and inequalities, inequalities and equalities involving max and min, and many other useful mathematical statements.

Statements containing strict inequalities. It turned out (see, e.g., [64]) that this class can be easily extended to statements which contain existence.

Terms T describing such statements can be obtained from variables (ranging over a given interval $[0, 1]$) and variable functions by using:

- Addition, subtraction, multiplication, max, min,
- Substitution of a computable constant instead of a variable,
- An operation $f(x_1, \ldots, x_n) \to \min\limits_t f(t, x_2, \ldots, x_n)$,
- An operation $f(x_1, \ldots, x_n) \to \max\limits_t f(t, x_2, \ldots, x_n)$,
- An operation $f(x_1, \ldots, x_n) \to a(f(x_1, \ldots, x_n))$ for a computable function $a(t)$ satisfying the Lipschitz condition,
- An integration operation $f(x_1, \ldots, x_n) \to \int_0^{x_1} f(t, x_2, \ldots, x_n)\, dt$.

Conditions are obtained from inequalities of the type $T > 0$ by using $\vee$, &, and quantifiers over real numbers.

It turns out that if such a condition is classically true, then it is true for some rational values of the variables and piecewise-linear functions with rational coefficients – and is, thus, constructively true.

Uniqueness implies computability. A more non-trivial class of classical statements which are automatically constructively true are statements about the existence of roots. In general, the fact that a computable function can be proven to have a root does not make this root algorithmically computable, but if this root is *unique*, then it is computable.

This result was first proven by D. Lacombe [79] for functions of one or several real variables defined on a bounded set. It was extended to general constructive compact spaces by Vladimir Lifschitz in [82]. Variations and applications of this result can be found in [60, 61, 62, 66, 75].

This approach was later developed by Ulrich Kohlenbach (see, e.g., [44, 45, 46]).

13. Fourth Challenge: When an Algorithm Is Possible, Is It Feasible? From Constructive Mathematics to Feasible (Polynomial-Time) Mathematics

Some algorithms of constructive mathematics are not feasible. An exhaustive-search algorithm that we outlined in the previous section is a typical example of algorithms generated by constructive mathematics.

Most of these algorithms take time which is exponential in terms of the input size (or even longer). Already for $n \approx 300$, the corresponding 2^n time becomes longer than the lifetime of the Universe – so these algorithms are not feasible even for reasonable-size inputs; see, e.g., [25, 65, 75, 101].

What is feasible? What happens if we only allow feasible algorithms? To answer this question, we need to have a formal definition of feasibility.

The current definition identifies feasible algorithms with algorithms that execute in polynomial time. It is well known that this definition is not perfect (but since no better one is known, researchers use it):

- For example, an algorithm that takes computation time $t(n) = 10^{300} \cdot n$ on inputs of size n is clearly not feasible, but it is a polynomial-time (even linear-time) algorithm.
- On the other hand, an algorithm which requires time $t(n) = \exp(10^{-9} \cdot n)$ is clearly feasible – at least for all inputs up to a dozen Gigabytes – but is not a polynomial-time algorithm.

Another problem with this definition is that the division into polynomial time and non-polynomial time is somewhat heuristic, motivated more by examples of feasible and non-feasible algorithms than by a deep theoretical analysis. This problem was somewhat eliminated by Vladimir Sazonov who showed that this this division can be reformulated in less heuristic logical terms [106].

What if we only allow feasible algorithms? So what happens if we only allow feasible algorithms – i.e., using the modern formalization of feasiblity, algorithms that require polynomial time? Several results along these lines have been developed in [55]; see also [63] and later comments by Yuri Gurevich [33].

It turns out that this feasible analysis is even more negative that that of the usual constructive mathematics: while addition and multiplication of computable numbers are still feasible, almost everything else is NP-hard:

- Integration,
- Computing the maximum of a computable function,
- Even computing $\sin(x)$ or $\exp(x)$ of a value in a floating point format.[16]

Most of these results were later covered by a thorough analysis presented in a monograph by Ker-I Ko [43]. In addition to negative results, this book contains many interesting efficient algorithms; for example, algorithms for analytical functions, integration (and many other operations) are feasible.

However, in the 1970s, feasible analysis was not welcomed too much. The seminar's opinion was that if the goal was to make constructive mathematics closer to computational practice, this goal failed.

Interval computations as applied constructive mathematics. Much more successful was another approach to make constructive mathematics more realistic. Namely, Yuri Matiyasevich observed that while algorithms of constructive mathematics assume that we have inputs known with increasing accuracy, in practice, the accuracy is fixed. At any given moment of time, we only have a single measurement result $\tilde{x}$, corresponding to the currently available accuracy Δ; see, e.g., [104]. As a result, the only information that we have about the (unknown) actual value x of the measured quantity is that it belongs to the interval $\mathbf{x} = [\tilde{x} - \Delta, \tilde{x} + \Delta]$. Given a data processing algorithm $y = f(x_1, \ldots, x_n)$ and intervals $\mathbf{x}_1, \ldots, \mathbf{x}_n$ corresponding to the inputs, we must therefore describe the corresponding range of possible values of y. This problem is called the problem of *interval computations*, or *interval analysis*.

In this problem, techniques borrowed from constructive mathematics work so well that many researchers – including Yu. V. Matiyasevich himself – consider interval analysis Applied Constructive Mathematics. Interval analysis has numerous practical applications ranging from robotics to planning spaceship trajectories to chemical engineering; see, e.g., [19, 37, 39, 40, 41, 75, 95].

The main idea of interval computations can be traced to Norbert Wiener [119, 120]. Its algorithms were developed by Ramon Moore in the late 1950s and early 1960s. Yuri Matiyasevich boosted this area by organizing conferences and by helping to launch a journal – then called *Interval Computations* – which remains,

[16]Fixed point and floating point formats have to be treated separately, since the transition from floating point to fixed point requires, in general, exponential time.

under the new, somewhat more general title *Reliable Computing*, the main journal of the interval computations community.

Can other physical ideas make computations feasible? If computations are not feasible on existing computers, maybe computers using some novel physical phenomena can make these problems feasibly solvable?

This indeed turned to be true.

- For example, if causality-violating processes ("time machines") are possible, then we can solve many NP-hard problems in polynomial time [50].
- We can achieve a similar speed-up if in our space-time, the volume of a sphere grows exponentially with the radius – as it does, e.g., in Lobachevsky space – see [76, 86, 96].
- Other schemes of this type are described in [1, 51, 75].

14. Fifth Challenge: What If No General Algorithm Is Possible?

If a general problem is not computable, can we relax it to make it computable? For example, if – as in Specker's sequence – the limit is not computable in the usual sense, in what sense is it computable?

This ideas was pioneered already by N. A. Shanin, who developed several notions of constructive pseudo-numbers; the whole hierarchy of such notions was developed and analyzed by Boris Kushner [78] – and we have already mentioned even more general Lifschitz's "fillings" [82].

With respect to this question, it is important to distinguish between:

- Problems which are "almost" computable and
- Problems which are strongly non-computable.

It turns out that in many cases, we can abstract from the specifics of a problem and describe this difference on the level of logic, by introducing an additional operation of *strong negation.* This idea was pioneered by N. N. Vorobiev in [115] and later developed by Bishop [7] and by Yuri Gurevich [32].

For example, while a negation to the statement $x > 0$ is the statement $x \leq 0$, a strong negation would mean $x < 0$. In this case, there is no algorithmic way to distinguish between $x \leq 0$ or $x > 0$, but we can easily distinguish between $x > 0$ and $x < 0$: it is sufficient to compute x with sufficient accuracy.

The idea of strong negation – in which, instead of a *single* property, we consider a *pair* of properties which are strong negations to each other – enables us to re-introduce the duality between "and" and "or" [7, 122], duality that is present in classical logic but which is missing in the traditional constructive logic.

If a general algorithm is not possible, can we find a reasonable subclass of problems for which the solution is algorithmically possible? Such classes are known. For example, for a general computable function that takes values of different signs at different sides of the interval, it is not possible to algorithmically find a root.

However, if we restrict ourselves to computable analytical functions, the root can always be computed.

Interestingly. a restriction to functions described by analytical *expressions* does not help: most algorithmically unsolvable problems remain algorithmically unsolvable; see, e.g., [49].

Another idea is, instead of all *mathematically* possible inputs, to only allow inputs which are *physically* possible. As a reasonable formalization of physical possibility, we can take, e.g. physicists's belief that events with very small probabilities cannot occur. This may sound strange, but this is exactly the belief behind a much more intuitive conclusion that a cold kettle, when placed on a cold stove, will never start boiling by itself – in spite of all the molecular motion which can theoretically lead to such phenomena.

This idea was first analyzed in [48, 22] by appropriately modifying the Kolmogorov/Martin-Löf's algorithmic definition of randomness [81]. This idea was further developed in [38, 67, 68, 69, 71, 73]. If turns out that under such a physics-motivated limitation, most negative results of constructive analysis disappear – and the corresponding problems become algorithmically solvable [68, 73].

What if we use novel physical phenomena? Maybe, if a problem is not computable, the use of some novel physical phenomenon can make this problems algorithmically solvable.

This is indeed possible. For example, as shown in [50], if we have access to flawless time machine, and either time or space are potentially infinite, then we can compute problems from the class Δ^1_1 – way beyond the usual computability.

Another – probably more realistic – idea is to take into account that, according to physicists, no physical theory is perfect, every theory will eventually encounter situations when this theory will need to be modified; see, e.g., [21]. Interestingly, a natural formalization of this idea leads to the possibility of computing functions which are usually considered not to be computable [52, 73, 121]. In other words, using observations of the physical world (looking at the tea leaves?) can enhance our computational abilities.

An interesting aspect of this problem again goes back to logic:

- If – by virtue of some physical phenomena – we are able to algorithmically solve some class of problems,
- What other classes of problems will we then be able to solve?

In [58], we describe which classes of problems imply the ability to algorithmically solve all the problems from analysis.

Questions of this type were later described, in a very general way, by Harvey Friedman who pioneered the whole area of *reverse mathematics*; see, e.g., [24, 110].

15. This Was Really a Boom

This was a boom. We had many interesting results, we had many great ideas. Gena Davydov once compared this period with Boldino Autumn, a most productive period in the life of the famous Russian poet Alexander Pushkin.[17]

We were optimistic. Vladimir Lifchitz was very optimistic that in a few years, to most mathematicians,

- A natural question after proving an existence theorem would be – can we effectively produce the resulting object?
- A natural way to answer this question would be to use tools from constructive mathematics.

I am euphoric, Vladimir liked to say, and I am not afraid to use this word – and this is how most of us felt.

We were recognized. Other departments felt that logic and constructive mathematics were booming. In addition to Matiyasevich's world-wide recognition, there were many other recognitions on a smaller scale:

- Vladimit Lifschitz got several research prizes,
- Dima Grigoriev and I shared a prize for the best student research paper, etc.

I remember how at a game theory seminar (which I, by the way, continued to attend), Nikolay Vorobiev encouraged the attendees to submit their paper for the best paper competition: OK, so Vladimir Lifschitz would get the first prize, but we could still aim for the second and third place prizes.

Researchers approached us suggesting collaboration. Motivated by our successes, more and more researchers from other disciplines started discussing topics of possible collaboration with us, especially physicists. Let me give two examples.

First example: use of global and local properties of analytical functions in physics. Leonid Khalfin, a physicist from St. Petersburg, had an interesting idea related to the use of complex numbers in quantum physics.

- Physicists gladly use the "global" effects of analyticity, such as the possibility to estimate complex integrals by using only the function's behavior over singularities.
- However, physicists rarely use the "local" properties of analyticity, for which there is often no physical meaning.

In classical mathematics, global and local properties are provably equivalent. However, Khalfin conjectured that, since local properties do not seem to correspond to any meaningful (observable) properties, maybe a proper constructive version of the theory – which explicitly limits us to potentially observable quantities – will enable us to separate the global and local properties, and to enjoy the useful effects of global properties without having to assume the local ones.

[17] Luckily, our reasons for boom were different from Pushkin's: he got stuck in the village of Boldino due to the quarantine caused by the deadly cholera epidemic.

The usual constructive mathematics does not help here, since in it, global and local properties are still equivalent. However, I still believe that if we limit ourselves to only feasible algorithms, maybe such a goal can be achieved.

Second example: attempts to use quantum effects to speed up computations. Andrei A. Grib, another physicist from St. Petersburg, helped us explore the possible use of quantum effects in computations. In this research, we were inspired by a question formulated by George Kreisel: if we use quantum effects,

- Can we compute something that we could not compute before?
- Can we compute some things faster than what we could compute before?

Our analysis only lead to preliminary results, but we were proud that we were part of the general intellectual atmosphere that had led to the current boom in quantum computing algorithms (see, e.g., [98]), which has already generated famous results:

- Grover's algorithm that searches in an array of size n in time $O(\sqrt{n})$ [30, 31] and
- Shor's algorithm [109] that factors large integers in time polynomial in this integer's bit length and can, thus, potentially break most current codes – specially the RSA code underlying security on the web, the code which is based on the inherent difficulty of such factoring.

16. Rebels in Science, Rebels in Life: Not Everything Was Perfect

We were rebels. Being in constructive mathematics in the community of mathematicians means going against the grain. Not surprisingly, folks who are rebels in their professional life were rebels in their politics as well.[18] Let me give a few examples.

Shanin resigned from the university as a protest. When the University, in violation of all its rules, rejected Zhenya Danstin's candidacy for the PhD program – and it was very clear to everyone that his Jewish origin was the only reason – Shanin officially resigned from the university.

This was a usual tactic under the tsars, when one could gain private employment, but Shanin is the only professor I know who resigned from the Soviet University as a protest.

Contacts with "enemies of the people" were encouraged. In 1970, Revolt Pimenov, leader of the space-time seminar, was arrested for reading and distributing "illegal" books (Orwell, Solzhenitsyn, etc.), and for these "anti-Soviet activities", he was sentenced to exile to the Far North Republic of Komi. I – and many others – kept in touch with him. When the time came for my University-required practicum, I expressed my desired to work with Pimenov in Komi Republic.

[18] It is not that everyone else willingly supported the Communist regime: when the first reasonably free elections where held in St. Petersburg in 1989, most communist candidates convincingly lost. However, many logicians went further than many others in their resistance.

Shanin, who was required to approve (or not) our practicum plans, asked only one question: “What will you practice there? Science or anti-Soviet activities?”. He was happy with my honest answer “both”, and to the Komi Republic I went – to the shock of local folks who were surprised to see a student of the prestigious St. Petersburg University officially sent to work with an exiled “enemy of the people”.[19]

Comment. It is not that everyone was against socialism as an idea – socialist Sweden was, to many of us then, a good example of how social equality can be established without shooting and jailing political opponents. Some logicians even kept a rosy image of Lenin as a true defender of the people. However, everyone was openly and fearlessly appalled by the violations of human rights that were ubiquitous during the communist dictatorship years.

Shanin expressed his disapproval of the authorities. When Solzhenitsyn was exiled, in violation of many international treaties signed by the Soviet Union, Shanin made a loud protest statement at the beginning of the seminar.

At that time, I considered such behavior normal, but later, when I moved to Novosibirsk (where such behavior was unheard of), and when I learned of cases when people were fired and jailed for such public protests, I realized how unusually brave St. Petersburg logicians were.

Maslov fired, probably killed. The endings were not always good. When in 1978, the communists staged a political “process” against the physicist and human right defender Yuri Orlov, Sergei Maslov wrote a letter to Brezhnev condemning the unfair closed trial as a violation of Soviet laws and many treaties signed by the Soviet Union, he was promptly fired from his teaching job.

Since he continued his political activities in spite of the continuous threats from the communists, it is quite possible that the KGB helped organize a suspicious car accident that killed him in 1982.

17. Why Were We Not As Successful As We Hoped? Maybe There Is Still Hope

What went wrong? We were so optimistic, we were so successful, so what went wrong? Why is constructive mathematics still not exactly mainstream?

Political reasons. Of course, there were reasons beyond our control. We lived under a totalitarian dictatorship. Journals and conference proceedings were all regulated by the state – and just like sausages were often difficult to buy, paper was scarce too. As a result, most published papers were short and thus, inevitably, not easy to read – which did not help their understandability.

[19]Pimenov, by the way, taught me to not be afraid of the KGB-installed electronic bugs in our homes: they already know, he said, that we are mostly against them, so they do not gain anything by hearing us say it one more time.

Travel to conferences abroad was strictly limited – I was never allowed to go to a conference abroad until 1988, when Gorbachev's perestroika was in full swing and I was allowed to attend a conference in Bulgaria – only to be not allowed to go to a conference in (still communist-controlled) Poland.

A special censorship permission was needed to send a paper abroad, even to send a letter on abstract mathematics abroad – and the permission was often denied. Mathematical letters sent to me from colleagues abroad were opened and stamped before they were delivered to me, and I was summoned to the KGB and threatened with jail because I sent a few letters with my own formulas abroad – they showed me xerox copies of my own letters.[20]

When a Western mathematician visited from abroad, he or she was under constant open surveillance. When Kip Thorne, a famous astrophysicist, visited Moscow and asked me to meet him in front of his hotel (local citizens were not allowed inside hotels for foreigners), a guy in a typical KGB "uniform" (coat, tie, white shirt) followed us wherever we went, his hand over his ear so that we would know that he was listening attentively.

Maybe we were too picky. This is all true. But, I think, there were also our own reasons. Yes, publication space was limited, but I think we were too picky in selecting what to publish, trying to be more saintly than the Pope. Too often, after a reasonable paper was presented, its reception was negative.

I remember that, at one of the seminars, when the chair desperately asked the audience for any positive remark or suggestion, someone replied that the author may consider, as a positive suggestion, a suggestion to grow upon oneself.

Many things that we considered to be not worth publishing – at least not worth publishing in detail – later turned out to be useful, and many of us later published some of it – but alas, still only a small portion of it (since everyone prefers to publish their most recent results). A lot of results and details were simply lost.

Maybe it is because our algorithms were not feasible? Maybe the problem was that the abstract algorithms that we analyzed and developed – inspired by the practical need for such algorithms – turned out be not exactly practical.

But we *were* working to make them more practical, so why did we not succeed?

Maybe we had problems communicating with people from other disciplines? Sometimes, especially when we tried to handle algorithms of interest to other disciplines such as physics, we suffered from a lack of understanding – but as Grisha Mints mentioned recently, even when understanding was there, for some mysterious reasons, the results were not as spectacular and as ground-breaking as we hoped.

[20] This was even more appalling to me, since xerox services were highly rationed, I could rarely get a copy of needed papers, but the KGB seemed to have an unlimited ability to copy everything we sent.

Constructive mathematics is alive and well. Why we did not succeed in still a mystery.

I still feel that there is a need for constructive mathematics – and there are constructive mathematicians around who are still producing interesting results (see, e.g., [2, 6, 8, 9, 10]), publishing books and papers, and organizing conferences.

Let us hope. So maybe there will be a second coming of constructivism.

Let us hope, and – more importantly – let us work together to make it happen.

References

[1] S. Aaronson, *NP-complete problems and physical reality*, ACM SIGACT News **36** (2005), 30–52.

[2] O. Aberth, *Precise Numerical Analysis Using C++*, Academic Press, New York, 1998.

[3] A. D. Alexandrov, *On Lorentz transformations*, Uspekhi Math. Nauk **5**(1) (1950), 187 (in Russian).

[4] A. D. Aleksandrov, A. N. Kolmogorov, and M. A. Lavrentev, *Mathematics: Its Content, Methods and Meaning*, Dover, New York, 1999.

[5] A. D. Alexandrov and V. V. Ovchinnikova, *Remarks on the foundations of Special Relativity*, Leningrad University Vestnik, No. 11 (1953), 94–110 (in Russian).

[6] M. J. Beeson, *Foundations of constructive mathematics*, Springer-Verlag, N.Y., 1985.

[7] E. Bishop, *Foundations of Constructive Analysis*, McGraw-Hill, New York, 1967.

[8] E. Bishop and D. S. Bridges, *Constructive Analysis*, Springer, N.Y., 1985.

[9] D.S. Bridges, *Constructive Functional Analysis*, Pitman, London, 1979.

[10] D. S. Bridges and S. L. Vita, *Techniques of Constructive Analysis*, Springer-Verlag, New York, 2006.

[11] L. E. J. Brouwer, *On the Foundations of Mathematics*, PhD Dissertation (in Dutch); Amsterdam; English translation in: A. Heyting (ed.), *L. E. J. Brouwer: Collected Works, Volume 1: Philosophy and Foundations of Mathematics*, Elsevier, Amsterdam and New York, 1975, pp. 11–101.

[12] H. Busemann, *Timelike Spaces*, PWN, Warszawa, 1967.

[13] G. S. Ceitin, I. D. Zaslavskii, and N. A. Ŝanin, *Peculiarities of constructive mathematical analysis*, in: *Proceedings of the 1966 International Congress of Mathemaricians*, Mir Publ., Moscow, 1968, pp. 253–261.

[14] V. P. Chernov, *On constructive operators of finite types*, Zapiski Nauchnth Seminarov LOMI **32** (1972), 140–147.

[15] V. P. Chernov, *Point-less and continuous mappoings*, Izvetiya Vuzov, Matematika No. 8 (1985), 51–59 (in Russian).

[16] A. Church, *An Unsolvable Problem of Elementary Number Theory*, The American Journal of Mathematics **58**(2) (1936), 345–363; reprinted in [17], pp. 88–107.

[17] M. Davis (ed.), *The Undecidable: Basic Papers on Undecidable Propositions, Unsolvable Problems and Computable Functions*, Dover, New York, 2004.

[18] G. P. Dimuro, A. C. R. Costa, and V. Kreinovich, *Modelling measurement processes as timed information processes in simplex domains*, Proceedings of the 10th IMEKO TC7 International Symposium on Advances of Measurement Science, Saint-Petersburg, Russia, June 30–July 2, 2004, Vol. 1, pp. 71–76.

[19] V. G. Dmitriev, N. A. Zheludeva, and V. Kreinovich, *Applications of interval analysis methods to estimate algorithms errors in measuring systems*, Measurement, Control, Automatization No. 1(53) (1985), 31–40 (in Russian).

[20] Yu. V. Ershov, *Theory of Numberings*, Nauka Publ., Moscow, 1977.

[21] R. Feynman, R. Leighton, M. Sands, *Feynman Lectures on Physics*, Basic Books, New York, 2005.

[22] A. M. Finkelstein and V. Kreinovich, *Impossibility of hardly possible events: physical consequences*, Abstracts of the 8th International Congress on Logic, Methodology, and Philosophy of Science, Moscow, 1987, Vol. 5, No. 2, pp. 25–27.

[23] P. C. Fishburn, *Nonlinear Preference and Utility Theory*, John Hopkins Press, Baltimore, Maryland, 1988.

[24] H. Friedman, *Some systems of second order arithmetic and their use*, Proceedings of the International Congress of Mathematicians, Vancouver, B.C., 1974, Vol. 1, Canad. Math. Congress, Montreal, Quebec, 1975, pp. 235–242.

[25] M. E. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, 1979.

[26] M. G. Gel'fond, *On constructive pseudofunctions*, Proceedings of the Leningrad Mathematical Institute of the Academy of Sciences **16** (1969), 20–27 (in Russian), English translation: *Seminars in Mathematics*, Published by Consultants Bureau (New York-London) **16** (1971), 7–10.

[27] M. G. Gel'fond, *Relationship between the classical and constructive developments of mathematical analysis*, Proceedings of the Leningrad Mathematical Institute of the Academy of Sciences **32** (1972), 5–11 (in Russian), English translation: Journal of Soviet Mathematics **6**(4) (1976), 347–352.

[28] M. G. Gelfond, *Classes of formulas of classical analysis which are consistent with the constructive interpretation*, PhD Dissertation, Leningrad Mathematical Institute of the Academy of Sciences, 1975 (in Russian).

[29] J. G. Granström, *Treatise on Intuitionistic Type Theory*, Springer, 2011.

[30] L. K. Grover, *A fast quantum mechanical algorithm for database search*, Proc. 28th ACM Symp. on Theory of Computing, 1996, pp. 212–219.

[31] L. K. Grover, *Quantum mechanics helps in searching for a needle in a haystack*, Phys. Rev. Lett. **79**(2) (1997), 325–328.

[32] Y. Gurevich, *Intuitionistic logic with strong negation*, Studia Logica **36** (1977), 49–59.

[33] Y. Gurevich, *Platonism, constructivism, and computer proofs vs. proofs by hand*, Bulletin of EATCS (European Association for Theoretical Computer Science **57** (1995), 145–166.

[34] A. Heyting, *Die formalen Regeln der intuitionistischen Logik*, Sitzungsberichte der Preussischen Akademie der Wissenschaften, Physik-Math. Kl., 1930, 42–71 and 158–169; English translation of Part I in P. Mancosu, *From Brouwer to Hilbert: The Debate on the Foundations of Mathematics in the 1920s*, Oxford University Press, New York and Oxford, 1998, pp. 311–327.

[35] A. Heyting, *Intuitionism: An Introduction*, 3rd revised edition, North-Holland Publishing Co., Amsterdam, 1971.

[36] D. Hilbert, *Mathematical Problems, lecture delivered before the Int'l Congress of Mathematics in Paris in 1900*, translated in Bull. Amer. Math. Soc. **8** (1902), 437–479; reproduced in: F. E. Browder (ed.), *Mathematical Developments Arising from Hilbert's Problems*, American Math. Society, Providence, RI, 1976.

[37] Interval Computations website http://www.cs.utep.edu/interval-comp

[38] A. Jalal-Kamali, O. Nebesky, M. H. Durcholz, V. Kreinovich, and L. Longpré, *Towards a 'generic' notion of genericity: from typical and random to meager, shy, etc.*, *Journal of Uncertain Systems* **6***(2) (2012), 104–113.*

[39] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*, Springer-Verlag, London, 2001.*

[40] R. B. Kearfott, Rigorous Global Search: Continuous Problems*, Kluwer, Dordrecht, 1996.*

[41] R. B. Kearfott and V. Kreinovich (eds.), Applications of Interval Computations, Kluwer, Dordrecht, 1996.

[42] S. C. Kleene, "On the interpretation of intuitionistic number theory", Journal of Symbolic Logic, 1945, Vol. 10, pp. 109–124.

[43] K. Ko, Computational Complexity of Real Functions, Birkhauser, Boston, Massachusetts, 1991.

[44] U. Kohlenbach. Theorie der majorisierbaren und stetigen Funktionale und ihre Anwendung bei der Extraktion von Schranken aus inkonstruktiven Beweisen: Effektive Eindeutigkeitsmodule bei besten Approximationen aus ineffektiven Eindeutigkeitsbeweisen, Ph.D. Dissertation, Frankfurt am Main, 1990.

[45] U. Kohlenbach, Effective moduli from ineffective uniqueness proofs. An unwinding of de La Vallée Poussin's proof for Chebycheff approximation, Annals for Pure and Applied Logic, **64**(1) (1993), 27–94.

[46] U. Kohlenbach, *Applied Proof Theory: Proof Interpretations and their Use in Mathematics*, Springer Verlag, Berlin-Heidelberg, 2008.

[47] O. M. Kosheleva and V. Kreinovich, *Derivation of the probabilistic character of physics from fundamental assumptions*, Research Notes in Philosophy of Physics, University of British Columbia, No. 4, 1978, 5 pp.

[48] O. M. Kosheleva and V. Kreinovich, *On the Algorithmic Problems of a Measurement Process*, Research Reports in Philosophy of Physics, University of Toronto, Ontario, Canada, Department of Philosophy, No. 5, 1978, 63 pp.

[49] O. M. Kosheleva and V. Kreinovich, *Analytical representation of constructivistic counterexamples*, Recursive Functions Theory: Newsletter No. 21 (1979), 9 (publ. No. 230).

[50] O. M. Kosheleva and V. Kreinovich, *What can physics give to constructive mathematics*, Mathematical Logic and Mathematical Linguistics, Kalinin, 1981, pp. 117–128 (in Russian).

[51] O. M. Kosheleva and V. Kreinovich, *Space-Time Assumptions Behind NP-Hardness of Propositional Satisfiability*, Mathematical Structures and Modelling **29** (2014), 13–30.

[52] O. M. Kosheleva and S. Soloviev, *On the logic of using observable events in decision making*, Proceedings of the IX National USSR Symposium on Cybernetics, Moscow, 1981, pp. 49–51.

[53] N. K. Kossovsky, *Some Questions in the Constructive Theory of Normed Boolean Algebras*, Trudy Mat. Inst. Steklov **113** (1970), 3–38; English translation Proceedings of the Steklov Institute of Mathematics **113**(1) (1970), 1–41.

[54] N. K. Kossovsky, *Some Problems in the Constructive Probability Theory*, In: *Logic, Language, and Probability*, Springer, Netherlands, 1973, pp. 83–99.

[55] V. Kreinovich, *Remarks on the margins of Kushner's book: on the constructive mathematical analysis restricted to polynomial time algorithms*, Manuscript (in Russian), Leningrad, 1973.

[56] V. Kreinovich, *Constructive theory of Wiener measure*, Proceedings of the 3rd USSR National Conference on Mathematical Logic, Novosibirsk, USSR, 1974, pp. 116–118 (in Russian).

[57] V. Kreinovich, *Applying Wiener measure to calculations*, Proceedings of the 3rd Leningrad Conference of Young Mathematicians, Leningrad University, 1974, pp. 6–9 (in Russian).

[58] V. Kreinovich, *What does the law of the excluded middle follow from?*, Proceedings of the Leningrad Mathematical Institute of the Academy of Sciences **40** (1974), 37–40 (in Russian), English translation: Journal of Soviet Mathematics **8**(1) (1977), 266–271.

[59] V. Kreinovich, *Constructivization of the notions of epsilon- entropy and epsilon-capacity*, Proceedings of the Leningrad Mathematical Institute of the Academy of Sciences **40** (1974), Vol. 40, pp. 38–44 (in Russian), English translation: Journal of Soviet Mathematics **8**(3) (1977), 271–276.

[60] V. Kreinovich, *Uniqueness implies algorithmic computability*, Proceedings of the 4th Student Mathematical Conference, Leningrad University, Leningrad, 1975, pp. 19–21 (in Russian).

[61] V. Kreinovich, *Categories of Space-Time Models*, Ph.D. dissertationNovosibirsk, Soviet Academy of Sciences, Siberian Branch, Institute of Mathematics, 1979 (in Russian).

[62] V. Kreinovich, *Review of Bridges, Constructive functional analysis*, Zentralblatt für Mathematik **401** (1979), # 03027.

[63] V. Kreinovich, *Review of 'Constructivization of the notions of $\varepsilon-$entropy and $\varepsilon-$capacity' by V. Kreinovich*, Mathematical Reviews **58**(1) (1979), 16, Review No. 78.

[64] V. Kreinovich, *A reviewer's remark in a review of, 'Some relations between classical and constructive mathematics' by M. Beeson*, Mathematical reviews **58**(2) (1979), 755–756, Review No. 5094.

[65] V. Kreinovich, *Unsolvability of several algorithmically solvable analytical problems*, Abstracts Amer. Math. Soc. **1**(1) (1980), 174.

[66] V. Kreinovich, *Physics-motivated ideas for extracting efficient bounds (and algorithms) from classical proofs: beyond local compactness, beyond uniqueness*, Abstracts of the Conference on the Methods of Proof Theory in Mathematics, Max-Planck Institut für Mathematik, Bonn, Germany, June 3–10, 2007, p. 8.

[67] V. Kreinovich, *Toward formalizing non-monotonic reasoning in physics: the use of Kolmogorov complexity*, Revista Iberoamericana de Inteligencia Artificial **49** (2009), 4–20.

[68] V. Kreinovich, *Negative results of computable analysis disappear if we restrict ourselves to random (or, more generally, typical) inputs*, Mathematical Structures and Modeling **25** (2012), 100–113.

[69] V. Kreinovich, *Towards formalizing non-monotonic reasoning in physics: logical approach based on physical induction and its relation to Kolmogorov complexity*, In: E. Erdem, J. Lee, Y. Lierler, and D. Pearce (Eds.), *Correct Reasoning: Essays on Logic-Based AI in Honor of Vladimir Lifschitz*, Springer Lecture Notes in Computer Science **7265** (2012), 390–404.

[70] V. Kreinovich, G. P. Dimuro, and A. C. da Rocha Costa, *From Intervals to? Towards a General Description of Validated Uncertainty*, Catholic University of Pelotas, Brazil, Technical Report, January 2004.

[71] V. Kreinovich and A. M. Finkelstein, *Towards applying computational complexity to foundations of physics*, Notes of Mathematical Seminars of St. Petersburg Department of Steklov Institute of Mathematics **316** (2014), 63–110; reprinted in Journal of Mathematical Sciences **134** (2006), 2358–2382.

[72] V. Kreinovich and O. Kosheleva, *Computational Complexity of Determining Which Statements about Causality Hold in Different Space-Time Models*, Theoretical Computer Science **405**(1-2) (2008), 50–63.

[73] V. Kreinovich and O. Kosheleva, *Logic of Scientific Discovery: How Physical Induction Affects What Is Computable*, Proceedings of the International Interdisciplinary Conference Philosophy, Mathematics, Linguistics: Aspects of Interaction 2014 PhML'2014, St. Petersburg, Russia, April 21–25, 2014, pp. 116–127.

[74] V. Kreinovich, O. Kosheleva, S. A. Starks, K. Tupelly, G. P. Dimuro, A. C. da Rocha Costa, and K. Villaverde, *From Intervals to Domains: Towards a General Description of Validated Uncertainty, with Potential Applications to Geospatial and Meteorological Data*, Journal of Computational and Applied Mathematics **199**(2) (2007), 411–417.

[75] V. Kreinovich, A. Lakeyev, J. Rohn, and P. Kahl, *Computational Complexity and Feasibility of Data Processing and Interval Computations*, Kluwer, Dordrecht, 1998.

[76] V. Kreinovich and M. Margenstern, *In some curved spaces, one can solve NP-hard problems in polynomial time*, Notes of Mathematical Seminars of St. Petersburg Department of Steklov Institute of Mathematics **358** (2008), 224–250; reprinted in Journal of Mathematical Sciences **158**(5) (2009), 727–740.

[77] E. H. Kronheimer and R. Penrose, *On the structure of causal spaces*, Proc. Cambr. Phil. Soc. **63**(2) (1967), 481–501.

[78] B. A. Kushner, *Lectures on Constructive Mathematical Analysis*, Amer. Math. Soc., Providence, Rhode Island, 1984.

[79] D. Lacombe, *Les ensembles récursivement ouvert ou fermés, et leurs applications à l'analyse récurslve*, Compt. Rend. **245**(13) (1957), 1040–1043.

[80] L. A. Levin, *Randomness Conservation Inequalities*, Information and Control **61**(1) (1984), 15–37.

[81] M. Li and P. M. B. Vitányi, *An Introduction to Kolmogorov Complexity*, Springer-Verlag, Berlin, Hiedelberg, New York, 2008.

[82] V. A. Lifschitz, *Investigation of constructive functions by the method of fillings*, J. Soviet Math. **1** (1973), 41–47.

[83] V. Lifschitz, *Constructive assertions in an extension of classical mathematics*, The Journal of Symbolic Logic **47**(2) (1982), 359–387.

[84] R. D. Luce and R. Raiffa, *Games and Decisions: Introduction and Critical Survey*, Dover, New York, 1989.

[85] M. Margenstern, *On constructive functionals on the space of almost perodic functions*, Zapiski Nauchnykh Seminarov LOMI **40** (1974), 45–62 (in Russian).

[86] M. Margenstern and K. Morita, *NP problems are tractable in the space of cellular automata in the hyperbolic plane*, Theoretical Computer Science **259**(1-2) (2001), 99–128.

[87] A. A. Markov, *On constructive mathematics*, Trudy Mat. Inst. Steklov **67** (1962), 8–14; English translation in American Mathematical Society Translations **98**(2) (1971), 1–9.

[88] A. A. Markov, *On the logic of constructive mathematics*, Vestnik Moskov. Univ., Ser. I Mat. Mekh. **25** (1970), 7–29 (in Russian).

[89] A. A. Markov, *An attempt of construction of the logic of constructive mathematics*, In: B. Kushner and N. Nagorny (eds.), *Theory of Algorithms and Mathematical Logic*, Computing Center of the Acad. of Sci., Moscow, 1976, pp. 3–31 (in Russian).

[90] A. A. Markov and N. M. Nagorny, *The Theory of Algorithms*, Reidel Publ., 1988.

[91] P. Martin-Löf, *Intuitionistic Type Theory*, Bibliopolis, Naples, Italy, 1984.

[92] Yu. V. Matiyasevich, *Diophantine character of recursively enumerable sets*, Dolkady AN SSSR **191**(2) (1970), 278–282 (in Rusian), English translation in Soviet Mathem. Doklady **11**(2) (1970), 354–358.

[93] Yu. V. Matiyasevich, *Hilbert's Tenth Problem*, MIT Press, Cambridge, Massachusetts, 1993.

[94] G. E. Mints, *Stepwise semantics of A. A. Markov*, a Supplement to the Russian translation of J. Barwise (ed.), *Handbook of Mathematical Logic* (North Holland, 1982), Nauka Publ., Moscow, 1983, pp. 348–357 (in Russian).

[95] R. E. Moore, R. B. Kearfott, and M. J. Cloud, *Introduction to Interval Analysis.* SIAM Press, Philadelphia, Pennsylviania, 2009.

[96] D. Morgenstein and V. Kreinovich, *Which algorithms are feasible and which are not depends on the geometry of space-time*, Geombinatorics **4**(3) (1995), 80–97.

[97] H. T. Nguyen, V. Kreinovich, B. Wu, and G. Xiang, *Computing Statistics under Interval and Fuzzy Uncertainty*, Springer Verlag, Berlin, Heidelberg, 2012.

[98] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge, U.K., 2000.

[99] B. Nordström, K. Petersson, and J. M. Smith, *Programming in Martin-Löf's Type Theory*, Oxford University Press, Oxford, UK, 1990.

[100] V. P. Orevkov, *A new proof of the uniqueness theorem for constructive differentiable functions of a complex variable*, Zap. Naucn. Sem. Leningrad. Otdel. Inst. Steklov. **40** (1974), 119–126 (in Russian).

[101] C. H. Papadimitriou, *Computational Complexity*, Addison Wesley, San Diego, 1994.

[102] R. I. Pimenov, *Kinematic spaces: Mathematical Theory of Space-Time*, Consultants Bureau, New York, 1970.

[103] M. B. Pour-El and J. I. Richards, *Computability in Analysis and Physics*, Springer Verlag, Berlin, 1989.

[104] S. G. Rabinovich, *Measurement Errors and Uncertainty: Theory and Practice*, Springer Verlag, Berlin, 2005.

[105] H. Raiffa, *Decision Analysis*, McGraw-Hill, Columbus, Ohio, 1997.

[106] V. Yu. Sazonov, *A logical approach to the problem P=NP*, Springer Lecture Notes in Computer Science **88** (1980), 562–575.

[107] N. A. Shanin, *On constructive understanding of mathematical statements*, Proc. Steklov Inst. of Math. **52** (1958), 226–231 (in Russian), English translation in American Mathematical Society Translations **23**(2) (1971), 109–189.

[108] N. A. Shanin, *Constructive real numbers and constructive function spaces*, Trudy Mat. Inst. Steklov **67** (1962), 15–294 (in Russian); translated into English as a monograph by the American Mathematical Society, 1968.

[109] P. Shor, *Algorithms for quantum computation: Discrete logarithms and factoring*, Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science, 1994, pp. 124–134.

[110] S. G. Simspon (ed.), *Reverse Mathematics 2001*, Association for Symbolic Logic, Wellsley, Massachusetts, 2001.

[111] E. Specker, *Nicht konstruktiv beweisbare Sätze der Analysis*, Journal of Symbolic Logic **14** (1949), 145–158.

[112] S. Thompson, *Type Theory and Functional Programming*, Addison-Wesley, 1991.

[113] A. Troelstra and D. van Dalen, *Constructivism in Mathematics. An Introduction*, Vols. I, II. North-Holland, Amsterdam, 1988.

[114] A. M. Turing, *On Computable Numbers, With An Application to the Entscheidungsproblem*, Proceedings of the London Mathematical Society, Series 2, **42** (1936–37), 230–265; corrections ibid **43** (1937), 544–546; reprinted in [17], pp. 115–153.

[115] N. N. Vorob'ev, *A constructive calculus of statements with strong negation*, Problems of the Constructive Direction in Mathematics. Part 3, Trudy Mat. Inst. Steklov **72** (1964), 195–227 (in Russian).

[116] N. N. Vorobiev, *Game Theory*, Springer, New York, 1977.

[117] N. N. Vorobiev, *Foundations of Games: Non-Cooperative Games*, Nauka Publ., Moscow, 1984 (in Russian).

[118] K. Weihrauch, *Computable Analysis*, Springer Verlag, Verlag, 2000.

[119] N. Wiener, *A contribution to the theory of relative position*, Proc. Cambridge Philos. Soc. **17** (1914), 441–449.

[120] N. Wiener, *A new theory of measurement: a study in the logic of mathematics*, Proceedings of the London Mathematical Society **19** (1921), 181–205.

[121] M. Zakharevich and O. Kosheleva, *If many physicists are right and no physical theory is perfect, then the use of physical observations can enhance computations*, Journal of Uncertain Systems **8**(3) (2014), 227–232.

[122] I. D. Zaslavsky, *Symmetric constructive logic*, Armenian Academy of Sciences, Erevan, 1978.

[123] E. C. Zeeman, *Causality implies the Lorentz group*, Journal of Mathematical Physics **5**(4) (1964), No. 4, 490–493.

Acknowledgments

This work was supported in part by the National Science Foundation grants HRD-0734825 and HRD-1242122 (Cyber-ShARE Center of Excellence) and DUE-0926721.

The author is greatly thankful to all his colleagues for valuable discussions. My special thanks to Irving Anellis, who tirelessly kept alive interest in foundations and history of logic and foundations of mathematics, especially history of login in Eastern Europe. My sincere thanks to Francine F. Abeles and Mark E. Fuller for their great idea to have a book published in Irving's memory, and for their support, encouragement, and editing help. Thank you all.

Vladik Kreinovich
Department of Computer Science
University of Texas at El Paso
500 W. University
El Paso, TX 79968, USA
e-mail: `vladik@utep.edu`