9-2015

Constructive Mathematics Is Seemingly Simple but There Are Still Open Problems: Kreisel's Observation Explained

Olga Kosheleva
University of Texas at El Paso, olgak@utep.edu

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

Comments:
Technical Report: UTEP-CS-15-68
Published in *Journal of Innovative Technology and Education*, 2015, Vol. 2, No. 1, pp. 51-56.

Constructive Mathematics Is Seemingly Simple but There Are Still Open Problems: Kreisel's Observation Explained

Olga Kosheleva and Vladik Kreinovich
Universiy of Texas at El Paso
500 W. University
El Paso, TX 79968, USA
olgak@utep.edu, vladik@utep.edu

Abstract

In his correspondence with Grigory Mints, the famous logician Georg Kreisel noticed that many results of constructive mathematics seem easier-to-prove than the corresponding classical (non-constructive) results – although he noted that these results are still far from being simple and the corresponding open problems are challenging. In this paper, we provide a possible explanation for this empirical observation.

1 Constructive Mathematics and Kreisel's Observation: A Brief Introduction to the Problem

Main objectives of science and engineering. In this paper, we deal with Kreisel's observation about the so-called *constructive mathematics.* In order to understand what is constructive mathematics, we need to recall what are the main objectives of science and engineering.

One of the main objectives of science is to predict the future state of the world: we want to predict tomorrow's weather, we want to predict what will happen to a building during an earthquake of a certain strength, etc.

Once we learn to *predict* what will happen in the future, the next task is how to *change* the future so that it will be more beneficial to us. Generally speaking, this is already the task not for science but for engineering. One of the main objective of engineering is:

- to design objects with that will make it possible to achieve the desired result, and
- to come up with control strategies that will lead to this result.

For example, if our objective is to land on Mars, we must:

- design an appropriate spaceship, and
- develop a control strategy that will enable us to land this spaceship on Mars.

Need for constructive mathematics. To solve a problem of science and/or engineering, we usually describe, in precise terms (i.e., in terms of mathematics) all our knowledge about the corresponding domains, and we use this knowledge to come up with appropriate predictions, constructions, and/or control strategies.

For example, to come up with an appropriate prediction, we usually describe the dynamics of the world in terms of the corresponding system of (partial) differential equations, and then we solve this system of equations to come up with the corresponding prediction.

The problem with this approach is that we are interested in *algorithms* for predictions, but traditional mathematics does not deal with algorithms. Moreover, it is often not even easy to formulate the existence of an algorithm.

The closest we get to algorithms in the traditional mathematics is when we prove *existence*: the existence of a solution, the existence of a construction with the desired property, or the existence of a control strategy that satisfies a given objective.

- Sometimes, the existence proof is straightforward, we prove the existence of the desired object by explicitly constructing it, so that from this proof, we can explicitly extract the corresponding algorithm.
- However, often, proofs are indirect – e.g., such as many proofs by contradiction. In this case, even after we have *proved* the existence of a solution, it is still not clear how to actually *construct* the corresponding solution.

It is therefore important to extend the traditional mathematics, so that it will be able to describe not only what is *theoretically* possible, but also to explicitly describe what is *algorithmically* constructible. Such an extension is known as *constructive mathematics.*

What is constructive mathematics: a brief introduction. In constructive mathematics, existence means that we have an explicit algorithm that enables us to construct the corresponding object. For example, in constructive mathematics, the statement

$$\forall n\, \exists m\, P(m,n)$$

means that there exists an algorithm e such that for every n, the property $P(m,n)$ holds for $m=e(n)$:

$$\exists e\, \forall n\, P(e(n),n).$$

Traditional (non-algorithmic) existence can also be describe in constructive mathematics, usually as a double negation $\neg\neg\exists n\, P(n)$.

Comment. It should be mentioned that in the traditional mathematics, the double negation of a statement is equivalent to the statement itself, since the law of excluded middle $P \vee \neg P$ is true for every statement P. In constructive mathematics, however, disjunction $A \vee B$ is viewed as a particular case of existence and is thus, interpreted as the fact that we know exactly which one of the two statements A and B is true. Since arithmetic is not decidable, we cannot always tell whether P or $\neg P$ is true. Thus, in constructive mathematics, the law of excluded middle is no longer always true, and thus, the double negation of a statement is, in general, not equivalent to the original statement.

More information about constructive mathematics can be found, e.g., in [1, 2, 3, 4, 6].

Kreisel's observation. In his unpublished 1973 correspondence with Grigory Mints, the famous logician George Kreisel noticed [5] that many results of constructive mathematics seem easier to prove that the corresponding results of traditional (non-constructive mathematics) – although, of course, not so much easier that we can actually solve all the corresponding open problems.

Why this observation is puzzling. Kreisel's observation seem counterintuitive:

- the goal of constructive mathematics is to go *beyond* the traditional one, to not only prove the existence, but also to provide the corresponding algorithms;
- so, at first glance, the problems of constructive mathematics should be *harder-to-solve* than the corresponding problem of the traditional (non-constructive) mathematics;
- however, in many cases, as Kreisel has correctly observed, these problems are *easier to solve.*

What we do in this paper. In this paper, we provide a possible explanation for Kreisel's seemingly counter-intuitive observation.

2 Kreisel's Observation Explained

Known fact: most practically useful results can be reformulated in arithmetic terms. The usual mathematical description of differential equations and their solutions requires quantifiers over real numbers. However, it is known that from the practical viewpoint, integers are sufficient to describe most of the practically important properties and results; see, e.g., [1, 2, 3, 4, 6].

Indeed, the reason why real numbers are used in mathematics it to describe the actual values of the corresponding physical quantities. In practice, however, when we measure a quantity, we never get the exact real numbers, we get an approximate value – which, in modern days, is usually described by a binary-rational number, i.e., a number of the type $m \cdot 2^{-n}$ – which can be describe by two integers m and n.

Similarly, when, e.g., we describe continuity of a function $f(x)$, the usual definition is that for every real number $\varepsilon > 0$ there exists a real number $\delta > 0$ such that $d(x,x') \le \delta$ implies that $d(f(x), f(x')) \le \varepsilon$. However, one can easily check that this definition is equivalent to requiring that for every integer $k > 0$ there exists an integer $\ell > 0$ for which $d(x,x') \le 2^{-\ell}$ implies that $d(f(x), f(x')) \le 2^{-k}$.

Similar equivalent reformulations can be made for many other properties.

A typical practice-related problem. A typical practice-related problem is, given an input, to algorithmically produce an output. For example:

- we know the equations and the initial conditions, we need to compute the future values of the corresponding quantities;
- we know the objectives, we need to come up with a design and/or with a control strategy that lets us achieve these objectives, etc.

In terms of constructive mathematics, if we denote the input by x, the desired output by y, and the property that we want this output to satisfy by $P(x,y)$, then we want to prove the formula of the following type:

$$\forall x\, \exists y\, P(x,y). \tag{1}$$

Let us show that this constructive formula is indeed easier-to-solve than its classical counterpart.

The complexity of formulas: a brief reminder. In general, the complexity of a formula is determined by its number of quantifiers:

- formulas without quantifiers are algorithmically decidable,
- formulas with one quantifier are, in general, more difficult to decide, formulas with two quantifiers are even more difficult to decide, etc.

In general, each formula can be described in the so-called *prenex normal form*, as

$$Q_1 x_1\, Q_2 x_2\, \ldots\, Q_n x_n\, P(x_1, \ldots, x_n),$$

where $P(x_1, \ldots, x_n)$ is a quantifier-free formula, and Q_i are alternating quantifiers, so that:

- $\exists$ is always followed by $\forall$, and
- $\forall$ is always followed by $\exists$.

The class of all formulas of this type with $Q_1 = \exists$ is denoted by Σ_n, and the class of all formulas of this type with $Q_1 = \forall$ is denoted by Π_n.

Each of the two classes Σ_n and Π_n is, in general, easier-to-decide than each of the two classes Σ_{n+1} and Π_{n+1}. Some formulas can be represented in both forms Σ_n and Π_n. The resulting intersection of these two classes is usually denoted by $\Delta_n \stackrel{\text{def}}{=} \Sigma_n \cap \Pi_n$. It is known that:

- each of the two classes Σ_n and Π_n is, in general, easier-to-decide than the intersection class Δ_{n+1}, and
- the intersection class is, in general, easier-to-decide than each of the two classes Σ_{n+1} and Π_{n+1}.

Resulting explanation of Kreisel's observation. As one can see from the formula (1), its classical analogue is, in general, a formula from the class Σ_2.

On the other hand, the constructive analogue of the formula (1), on the one hand, has the same Σ_2-form, but on the other hand, by definition of constructive mathematics, also has the form

$$\exists e\, \forall x\, P(e(x), x),$$

i.e., the form Π_2.

Thus, the constructive formulas of the type (1) can be described both in the Σ_2- and in Π_2-forms and thus, belong to the intersection class Δ_2. We have already mentioned that, in general, formulas from the class Δ_2 are easier to solve than formulas from the class Π_2. This explains Kreisel's observation that results of constructive mathematics are, in general, somewhat easier to prove than the similar non-constructive results.

On the other hand, the class Δ_2 is far from being decidable: even its subclasses Σ_1 and Π_1 (with one quantifier) are not decidable. Thus, in spite of its relative simplicity, constructive mathematics still contains challenging open problems.

So, Kreisel's observation has been explained.

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 authors are thankful to all the participants of the Third St. Petersburg Days of Logic and Computability (St. Petersburg, Russia, August 24–26, 2015) for valuable discussions.

References

[1] O. Aberth, *Computable Calculus*, Academic Press, New York, 2000.

[2] M. Beeson, *Foundations of Constructive Mathematics*, Springer Verlag, Heidelberg, 1985.

[3] E. Bishop and D. Bridges, *Constructive Analysis*, Springer Verlag, Heidelberg, 1985.

[4] D. Bridges and L. Vita, *Techniques of Constructive Analysis*, Springer Verlag, Heidelberg, 2006.

[5] G. Kreisel and G. Mints, Unpublished correspondence, 1973.

[6] B. Kushner, *Lectures on Constructive Mathematical Analysis*, American Mathematical Society, Providence, Rhode Island, 1985.