6-1-2021

Green Computing: Three Examples of How Non-Trivial Mathematical Analysis Can Help

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

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

Green Computing: Three Examples of How Non-Trivial Mathematical Analysis Can Help

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

Abstract

Environment-related problems are extremely important for mankind, the fate of humanity itself depends on our ability to solve these problems. These problems are complex, we cannot solve them without using powerful computers. Thus, in the environmental research, environment-related computing is one of the main computing-related research directions. Another direction is related to the fact that computing itself can be (and currently is) harmful for the environment. How to make computing more environment-friendly, how to move towards green computing – this is the second important direction. A third direction is motivated by the very complexity of environmental systems: it is difficult to predict how environment will change. This seemingly negative difficulty can be reformulated in more positive terms: that by observing environmental processes, we can find values which are difficult to compute. A similar observation about quantum processes has led to successful quantum computing, so why not use environmental processes themselves to help us compute? This is the third direction in environment-related computing research. In this paper, we provide three examples that show that in all three directions, non-trivial mathematical analysis can help.

1 Computing and Environment: Three Directions

Computing and environment: three directions. Environmental issues are critical for the survival of humankind. These issues are also very complex, with complex dependencies that require computers to solve. This is probably the most important relation between computing and environment: that we need to use computers to solve complicated environment-related problems.

However, there are other relations between environment and computing. One of the most important relations is that computing itself contributes to

environmental problems – e.g., by consuming a large amount of energy, energy that mostly comes from the non-renewable energy sources like fossils. To make computing less damaging to the environment, to move towards environment-friendly computing is the second important direction.

Finally, there is a third direction, motivated, in particular, by the success of quantum computing; see, e.g., [41]. This success comes largely from the fact that many quantum effects are complex and difficult to predict. This means that by launching some quantum processes and observing the results, we can really fast get the values whose computing on a regular computer would take a long time. In other words, for the corresponding functions, using actual quantum processes leads to faster computation than using a regular computer. It turns out that many of such functions are practically useful – and this, in a nutshell, is what quantum computing is largely about. But environmental processes are also very complex, very difficult to predict – often even more difficult to predict that typical quantum processes. So a natural idea is to use environmental processes for computations. This is the third direction – which may sound promising, but which, at this moment of time, is the least explored.

What we do in this paper. In this paper, we provide three examples that show how non-trivial mathematical analysis can help in all three of these directions: environment-related computing, environment-friendly computing, and environment as computing.

2 Equidecomposability: an Example of Environment-Related Computing

Recycling as an important part of environmental research. There are many important problems in environment-related computing – and we ourselves, together with our students, have participated in solving some of these problems; see, e.g., [4, 8, 15, 16, 23, 24, 34, 35, 44, 45].

In this section, we will concentrate only on one these problems – namely, on one of the most fundamental ones. This problem is related to the fact that one of the main reasons why our civilization spends a lot of energy – and thus, contributes to the worsening of the environmental situation – is that we constantly produce new things: clothes, cars, computers, etc. Most of the manufacturing processes consists of several stages on each of which we spend some energy. For example, to produce a computer, we first need energy to extract the ores from the mines, then we need energy to extract the corresponding metals from the ores, we need energy to produce silicon and plastic, then we need energy to combine them into parts, etc.

Each gadget has a limited period of use: cars break down, clothes deteriorate, computers become too slow and obsolete in comparison with the new designs, etc. To save energy – and thus, to save the environment – it is desirable to recycle these gadgets – i.e., to use them to minimize the number of stages need to produce new gadgets. For example, if we melt used aluminum cans into new

2

ones, we do not need energy to extract aluminum from the aluminum ore (and this requires, by the way, a lot of energy).

Recycling is one of the most important issues in environment-related practice.

When is recycling most efficient. The largest amount of savings in recycling is when the material is largely intact, and the only difference between what we have and what we want is the shape of this material. Sometimes, this is part of the manufacturing: for example, when we tailor clothes, we start with material, cut it into appropriate pieces, and sew them together. Sometimes, this is part of recycling: if we have some old clothes in which most material is intact, a tailor can cut it and make new clothes. If we have pieces of clothes which are too small to be useful by themselves, we can sew them together to make a quilt. If we have a metal sheet, and we need to make a tube, we can cut this sheet into pieces and weld them together to form a tube. This is the type of recycling – we will call it *geometric recycling* – that we will analyze in this section.

Geometric recycling as a mathematical problem. Let us describe this situation in precise geometric terms. We are given an object of certain shape. In mathematical terms, a shape is simply a set of 3-D points, i.e., a subset of the 3-D space $P \subset \mathbb{R}^3$. We want to use this object to design an object of a different shape Q. Ideally, we should simply cut the object A into several disjoint pieces

$$P = P_1 \cup \ldots \cup P_p, \tag{1}$$

then move these pieces around, so that they form sets $Q_1, \ldots, Q_p$, and combine these sets Q_i into the desired shape Q:

$$Q = Q_1 \cup \ldots \cup Q_p. \tag{2}$$

Of course, a generic set is not always a realistic description of a part: for example, we can have weird sets like the set of all rational numbers. T make sure that we consider only practically implementable sets, we need to restrict ourselves, e.g., to polytopes – they approximate any smooth or even non-smooth objects with any needed accuracy, and thus, from the practical viewpoint, represent all possible shapes. So, in the above problem, we assume that the given sets P and Q are polytopes, and that we are looking for polytopes P_i and Q_i.

To perform the above procedure, we need, given two polytopes P and Q, to find the polytopes P_i and Q_i that satisfy the conditions (1) and (2) and for which each polytope Q_i can be obtained from the corresponding polytope P_i by using some shifts and rotations; we will denote this by $P_i \sim Q_i$. Polytopes P and Q for which such divisions are possible – i.e., which can be composed of "equal" parts $P_i \sim Q_i$ – are called *equidecomposable*. For each pair (P, Q), the first natural question is: is such a decomposition possible at all?

Of course, rotations and shifts do not change the volume V, so we have $V(P_i) = V(Q_i)$ for all i and thus, $V(P) = V(Q)$. Thus, only sets of equal volume can be equidecomposable. So, this question makes sense only for polytopes of equal volume.

If such a decomposition of P and Q *is* possible, the next natural question is: what is the smallest number of parts p needed for this process – since the fewer cuts and "glue-ings" we need, the less energy we will spend on this process.

What is known: an early history of this mathematical question. For polygons, the answer to a similar question has been known for some time: every two polygons of equal area re equidecomposable. Since every polygon can be represented as a union of disjoint triangles, this result can be proven if we prove that every triangle is equidecomposable with a right rectangle (2-D box) one of whose sides is equal to 1. This shows that each polygon is equidecomposable with a right rectangle whose one side is 1 and another side is equal to the polygon's area – and thus, every two polygons with equal area are indeed equidecomposable.

In the 3-D cases, it is reasonable to try the same idea. We can similarly represent each polytope as a union of tetrahedra (= triangular pyramids), but it is not clear whether each tetrahedron is equidecomposable with a similar 3-D box. The famous mathematician Gauss, universally recognized as the top mathematician of his time – of Gaussian elimination, Gaussian distribution, etc. fame – formulated an open question of whether a regular tetrahedron is equidecomposable with a cube of the same volume.

This question became one of Hilbert's problems. Gauss's question resurfaced in 1900, when, in preparation for the 1900 International Congress of Mathematics, mathematicians asked David Hilbert – universally recognized as the top mathematician of his time – to prepare the list of most important challenges that the 19 century mathematics was leaving for the 20 century mathematicians to solve. Hilbert prepared a list of 23 problems, one of which – Problem 3 – repeated Gauss's question about equidecomposability.

There is an unclear story about this problem, since this problem was the first to be solved – it was solve by Hilbert's student Dehn in the same year [12, 13]: namely, Dehn proved that a regular tetrahedron is *not* equidecomposable with the cube of the same volume. This suspiciously fast solution – other problems took decades to solve – and the fact that it was solved by Hilbert's own student gave rise to a rumor that the problem was actually solved *before* Hilbert's talk, but Hilbert included it in the list of his problems – as supposedly still an open problem – to boost the status of his student. Whether this rumor is true or not, this result was very impressive: an answer to a challenging problem that Gauss himself, named the King of Mathematicians, tried to solve and could not.

Comment. It should be mentioned that Dehn's results have been later simplified and generalized; see, e.g., [5, 6, 26, 46].

Algorithmic aspects of Dehn's result. Dehn's result was a purely mathematical result: both Gauss and Hilbert emphasized this problem not because of recycling applications – this came later (see, e.g., [29]) – but because of their interest in axiomatic description of geometry (and related non-Euclidean geometries). Specifically, in the 2-D case, since every polygon is equidecomposable with a square, it is easy to axiomatically describe what is an area $A(P)$ of

a polygon P: it is sufficient to assume:

- that for a square of side a, the area is a^2, and

- that for two polygons that have no common interior points, $A(P \cup Q) = A(P) + A(Q)$.

What Dehn proved was that in the 3-D case, similar two conditions:

- that the volume of a cube with side a is a^3 and

- that $V(P \cup Q) = V(P) + V(Q)$,

are not sufficient to uniquely determine the volume on the set of all polytopes. In addition to the usual volume, there are other functions $V'(P)$ that satisfy these two properties.

Dehn's proof was purely mathematical, he did not describe how to compute any of these alternative functions $V'(P)$ – and he did not even formulate such a question because at that time, the notions of what is computable and what is not have not yet been formalized; this was only done by Turing in the 1930s [49]. However, nowadays, when computability is precise defined (see, e.g., [33]), a natural question is: are these alternative functions $V'(A)$ computable?

An answer to this question was only obtained in a 1980 paper [27] (see also [28]), where it was proven that the only *computable* function $V(A)$ satisfying the above two properties is the usual volume. Interestingly, the result was based on complex mathematics – namely, on homological algebra ideas and results, see [6, 46].

Relation to logic. One of the areas in which computability has been actively studied is logic – logic was one of the first areas where a non-computability results were proven, and it remains one of the main areas in which researchers study what is computable, and if yes, how fast this can be computed. From this viewpoint, it is important to mention papers [10, 32] that connected equidecomposability and logics.

Can we algorithmically check whether two polytopes are equidecomposable: precise question and a partial answer. It is reasonable to consider polyhedra which can be constructed by geometric constructions. It is well known that for such polyhedra, all vertices have *algebraic* coordinates (i.e., values which are roots of polynomials with integer coefficients); see, e.g., [11].

So, the above question can be formulated in the following precise form (see, e.g., [39]): is there an *algorithm* for checking whether two given polyhedra with algebraic coordinates are equidecomposable?

A partial answer to this question – described in [32] – follows from the fact that many related formulas can be described in the following *first order theory of real numbers* (also known as *elementary geometry*): a theory in which:

- *variables* run over real numbers,

- *terms* t, t' are obtained from variables by addition and multiplications,

- *elementary formulas* are of the type $t = t'$, $t < t'$, $t \leq t'$, and

- arbitrary formulas are constructed from the elementary ones by adding logical connectives & ("and"), $\vee$ ("or"), $\neg$ ("not"), and quantifiers $\forall x$ and $\exists x$.

A well-known result by Tarski [47] is that this theory of decidable, i.e., that there exists an algorithm which, given a formula from this language, decides whether this formula holds. The original Tarski's algorithm required an unrealistically large amount of computation time; however, later, faster algorithms have been invented; see, e.g., [3, 38].

For existential formulas of the type $\exists x_1 \ldots \exists x_m F(x_1, \ldots, x_m)$, these algorithms not only return "true" or "false " – when the formula is true, they actually return some values x_i which make the formula $F(x_1, \ldots, x_m)$ true.

Let us apply this result to our problem. Each polyhedron P_i can be decomposed into tetrahedra. So, without losing generality, we can assume that P can be decomposed into tetrahedra which can be then moved one-by-one and reassembled into Q.

We say that P and Q are *n-equidecomposable* if they can be both decomposed into finitely many pair-wise congruent tetrahedra $P_i \sim Q_i$ in such a way that a total number of all the vertices of all these tetrahedra does not exceed n.

Let's fix a coordinate system. Then a tetrahedron is described by the coordinates of its 4 vertices, i.e., by 12 (finitely many) real numbers. Let's denote these 12 numbers by a 12-dimensional vector $\vec{x}$. The coordinates of the vertices of the tetrahedra which form a decomposition are also real numbers (no more than $3n$ of them, because there are no more than n vertices). Congruence can be expressed as equality of all the distances, which, in its turn, is equivalent to equality of their squares. So, it is expressible by an elementary formula of elementary geometry.

The fact that P and Q are n-equidecomposable (we will denote it by $s_n(P,Q)$) means that there exist a decomposition of P and a decomposition of Q which are pair-wise congruent. Therefore, $s_n(P,Q)$ is constructed from elementary formulas by applying existential quantifiers: $s_n(P,Q) \leftrightarrow \exists x_1 \ldots \exists x_{m_n} F_n$. Hence, $s_n(P,Q)$ is also a formula of elementary geometry. Thus, for every n, we can check whether $s_n(P,Q)$ holds or not.

If we know that P and Q are equidecomposable, this means that they are n-equidecompsable for some n. We can thus (algorithmically) check the formula $s_1(P,Q)$, $s_2(P,Q)$, $\ldots$, until we find the first n for which the existential formula $s_n(P,Q) \equiv \exists x_1 \ldots \exists x_{m_n} F_n(x_1, \ldots, x_{m_n})$ holds.

For this n, as we have mentioned, the elementary geometry algorithms also return the values x_i for which $F_n(x_1, \ldots, x_{m_n})$ holds, i.e., the coordinates of the tetrahedra which form the desired decompositions of P and Q.

Comment. It is known that algorithms for deciding elementary geometry cannot be very fast: e.g., for purely existential formulas like the ones we use, there is an exponential lower bound a^n for the number of computational steps.

However, this is OK, because a similar exponential lower bound exists for constructing the corresponding decompositions – even for polygons; see, e.g., [29].

Can we algorithmically check whether two polytopes are equidecomposable: final answer. If two polyhedra are equidecomposable, then, by applying the above algorithm for $n = 1, 2, \ldots$, we will eventually find the corresponding decompositions. However, the above algorithm does not inform us when the polyhedra are not equidecomposable – in this case, the above algorithm will simply run forever and never stop.

It turns out that we *can* algorithmically decide whether two given polyhedra are equidecomposable or not; see [31]. However, this algorithm requires a lot of non-trivial mathematics to describe, see, e.g., [2, 9, 14, 20, 21].

Remaining open questions. Can we always check equidecomposability in exponential time? When can we check it faster?

What if instead of looking for a perfect solution – equidecomposability – we look for an approximate solution, in which some small volume of the material can be wasted?

3 Reversibility: an Example of Environment-Friendly Computing

Why reversibility. Computers consume a large proportion of the world's energy, thus contributing to the environmental problems. How can we decrease this amount? Some decrease can be achieved by engineering, but there is also a fundamental reason for energy consumption and for the fact that computers warm up their environment. This is related to statistical physics and thermodynamics (see, e.g., [17, 48]), according to which every irreversible process generates heat. Specifically, the amount of heat Q generated by each process is proportional to the product $T \cdot S$ of the temperature T and the increase in entropy S – and this increase is, in its turn, proportional to the numbers of bits of information lost in the process.

And irreversibility is ubiquitous in computing: computers are formed from logical gates such as "or"- and "and"-gates, and the corresponding operations are not reversible. For example, when we know the result $y = x_1 \mathbin{\&} x_2$ of applying the "and"-gate to the inputs x_1 and x_2, we cannot uniquely reconstruct the inputs: when $y = 0$, this could mean $(x_1, x_2) = (0, 0)$, this could mean $(x_1, x_2) = (0, 1)$, or this could mean $(x_1, x_2) = (1, 0)$. So, to decrease computer's contribution to warming, we need to make sure that all operations are reversible.

This is, by the way, what happens in quantum computers [41], since on the microscopic level of quantum processes, all processes are reversible – both Newton's equations of classical physics and Schroedinger's equations of quantum physics are reversible [17, 48].

7

Reversibility of computing with real numbers: general discussion.
Most of the time, computer process real numbers – i.e., measurement results.
What does reversibility means in this case? This problem was analyzed in [18]
(see also [19]); let us provide a reasonably detailed (and somewhat clarified)
description of this paper's results.

First, we should have at least as many outputs as we have inputs – otherwise,
to reconstruct the inputs, we will have fewer equations than unknowns, and we
know that in this case, this reconstruction problem usually has many solution.

This property is necessary, but not sufficient. Let us explain why. Indeed,
each measurement result x_i has some accuracy δ_i. Measurement results differing
by less than δ_i may correspond to the same actual values of the corresponding
quantity. So, in fact, while we can have many different tuples $(x_1, \ldots, x_n)$, in
reality, many of these tuples are indistinguishable. The set of all possible tuples
is thus divided into boxes of volume proportional to $\delta_1 \cdot \ldots \cdot \delta_n$, and the number
of substantially different tuples is equal to the volume of the whole domain of
possible value divided by the volume of this box.

Each transformation transforms an input domain into an output domain.
Reversibility means that if we know the output, we can uniquely reconstruct
the input. Thus, for a transformation to be reversible, there needs to be a
one-to-one correspondence between input boxes and output boxes. So, when
an input domain consists of several input boxes, the domain obtained after the
transformation should consist of exactly the same number of output boxes. Since
– at least locally – all input boxes have the same volume, this means that the
volume of the domain is proportional to the number of such boxes. Similarly,
the output volume is proportional to the number of output boxes. Thus, the
volume of the output domain should be proportional to the volume of the input
domain. It is known that for small domains, the output volume is equal to the
input volume multiplied by the determinant

$$D(x_1, \ldots, x_n) \stackrel{\text{def}}{=} \det \left\| \frac{\partial f_i}{\partial x_j}(x_1, \ldots, x_n) \right\|$$

of the matrix of partial derivatives. Thus, the reversibility requirement means
that this determinant should be constant: $D(x_1, \ldots, x_n) = \text{const}$.

Reversibility of computing with real numbers: first case. Let us first
consider the case when, in the original transformation, the number of outputs
is the same as the number of inputs:

$$(x_1, \ldots, x_n) \mapsto (f_1(x_1, \ldots, x_n), \ldots, f_n(x_1, \ldots, x_n)).$$

Of course, for many such real-life transformations – e.g. for a frequent trans-
formation of all inputs into a log-scale $(x_1, \ldots, x_n) \mapsto (\ln(x_1), \ldots, \ln(x_n))$ – the
determinant $D(x_1, \ldots, x_n)$ is not a constant. To make it constant, we need to
add at least one auxiliary variable u, i.e., replace the original transformation
with a new transformation

$$(x_1, \ldots, x_n, u) \mapsto (f_1(x_1, \ldots, x_n), \ldots, f_n(x_1, \ldots, x_n), v_f(x_1, \ldots, x_n, u))$$

8

for an appropriate function $v_f(x_1, \ldots, x_n, u)$.

Which auxiliary function should be choose? As shown in [18], the requirement that for the new enlarged transformation the corresponding $(n + 1)$-dimensional determinant is equal to a constant c implies that

$$v_f(x_1, \ldots, x_n, u) = \frac{c \cdot u}{\det \left\| \dfrac{\partial f_i}{\partial x_j}(x_1, \ldots, x_n) \right\|} + C$$

for some constant C.

Comment. The value of C is irrelevant: we can always re-scale the auxiliary variable u into $u' = u - C$ and thus, make this constant equal to 0.

Reversibility of computing with real numbers: second case. What if the number of outputs is smaller than the number of inputs? In this case, we need to add additional outputs, to preserve reversibility. For example, a binary operation $(x_1, x_2) \mapsto f(x_1, x_2)$ has to be replaced with a transformation $(x_1, x_2) \mapsto (f(x_1, x_2), g(x_1, x_2))$ for an appropriate auxiliary function $g(x_1, x_2)$. The reversibility requirement – that the volume must be preserved – takes the form

$$\frac{\partial f}{\partial x_1} \cdot \frac{\partial g}{\partial x_2} - \frac{\partial f}{\partial x_2} \cdot \frac{\partial g}{\partial x_1} = \text{const.}$$

In this case, there is no general analytical expression for the solution, but there are explicit expressions for the usual binary operations $f(x_1, x_2)$. For example, for addition $f(x_1, x_2) = x_1 + x_2$, one of the possible solutions is subtraction $g(x_1, x_2) = x_1 - x_2$; for multiplication $f(x_1, x_2) = x_1 \cdot x_2$, we can take $g(x_1, x_2) = \ln(x_1/x_2)$, etc.

Comment. In this section, mathematics is much simpler than in the previous one, but still solving partial differential equations is not exactly trivial mathematics.

4 Gaia Hypothesis Made Rational: an Example of Using Environment for Computing

Physics vs. study of environment. The main difference between physics and study of environment is that:

- in physics, we usually have a theory that explains all – or at least almost all – observations, that predict most important future events, while

- in the study of environment, no matter how more complex we make our models, we are far from accurate predictions.

To some extent, the same is true in physics: theories change, they are updated. After Newton, most physicists were under the impression that Newtonian physics would be sufficient to explain all the world's phenomena – and until

the last 19 century, it was mostly sufficient. However, new observations lead to the need to replace Newtonian physics with more accurate relativistic and quantum approaches. Many physicists actually believe that in physics too, no matter what model we propose, no matter what law of nature we discover that fits all known observations, eventually a new observation will appear that will be inconsistent with this law – and thus, the law will need to be modified; see, e.g., [17, 48]. In this section, we will describe the consequences of this widely spread belief.

How this belief affects computations. The above belief means, in particular, that the sequence of observations is not computable – if it was computable, i.e., if there existed an algorithm that predicts all observations, then this algorithm would not need any eventual modifications and would, thus, contradict this belief.

In other words, according to this belief, simply observing the world can help us come up with values that cannot come from any computations – i.e., observations have the ability beyond computations. Since observations have this beyond-usual-computations ability, it is reasonable to expect that using these observations in computations can speed up the computation process. This indeed turned out to be the case – although, similarly to the previous two sections, this conclusion requires some non-trivial mathematics.

To describe the exact result, we need to briefly recall several notions from computational complexity; for details, see, e.g. [33, 43]. One of the main tasks of computational complexity is to distinguish between:

- *feasible* algorithms – i.e., algorithms that require reasonable computation time for inputs of reasonable length – and

- algorithms which require so much computation time that they are not practically possible: e.g., that require, for inputs of length 100, longer time than the lifetime of the Universe.

How to clearly separate feasible from non-feasible algorithms is still an open problem. At present, the best formalization – that, in most case, adequately describes feasibility – is to define an algorithm A to be feasible if for there is a polynomial $P(n)$ such that for each input of size n, this algorithm requires time bounded by $P(n)$.

The next natural question is which problems can be solved by feasible algorithms. To formalize this question, we need to formalize what is a problem. In most practical problems, while the problem itself may be difficult to solve, it is feasible to check whether a candidate for a solution is indeed a solution. For example:

- it is difficult to come up with a proof of a mathematical statement, but

- once someone gives us a text that is supposed to be a detailed proof, it is straightforward to check each step and confirm (or not) whether this is indeed a correct proof, with no unproven gaps.

The class of such problems – for which, once we have a candidate for a solution, we can feasibly check whether it is indeed a solution – is denoted by NP. For some of these problems, there is a feasible algorithm; the class of all such problems is denoted by P.

Whether all problems can be feasibly solved, i.e., whether P is equal to NP, is not known: this is a famous open problem. What *is* known is that in the class NP there are problems which are the most difficult in this class – in the sense that every other problem from the class NP can be feasible reduced to each of these problems. Such problems are known as *NP-complete*. Historically the first example of an NP-complete problem is the following *propositional satisfiability* (SAT) problem:

- given a propositional formula, i.e., an expression obtained from the Boolean (true-false) variables by using "and", "or", and "not",

- find the values of these variables that make this formula true – or return a message that this formula is always false.

The paper [30] studied what can be computed if we consider "physical" algorithms, i.e., algorithms, that, in addition to usual computational step, can access (and use) unrelated observations of the physical world. In this paper, it is proven that there exists a feasible physical algorithm A with the following property. For every $\varepsilon > 0$ and for every integer n, there exists a larger integer $N > n$ for which the proportion of SAT formulas of length N that is correctly solved by this algorithm A is $\geq 1 - \varepsilon$. In other words, if we use observations, then we can feasible solve most instances of hard problems. A similar result holds for all other NP-complete problems. So, under the assumption that the physicists' belief is correct, observations indeed help computations.

Comment. The paper [30] uses yet another area of non-trivial mathematics: the notions of meager sets, also known as sets of Baire first category; see, e.g., [25, 42].

Why environment? The above result is about observations in general: we can observe stars in the sky, we can observe how elementary particles interact in an accelerator, etc. So why are we citing this result in a paper about environment-related applications? The reason is simple: to get good results, we need a sufficiently large number of observations – observations that cannot be explained by simple formulas. We can get a lot of data by observing stars in the sky or planets, but their visible motion is well described by reasonably simple algorithms: already ancients could predict solar eclipses many years ahead.

Micro-world observations are not that easy to predict, but each observation requires a costly experiment, so we do not have that many data points. We have a lot of not-easy-to-predict data points from geological sciences, but much more data comes from observing environment: for environment-related data, we have the same *spatial* variability as for the geological data, but environmental data also has *temporal* variability, so instead of 2-D or 3-D data, we now have 3-D or 4-D data.

What is Gaia hypothesis. As we have mentioned, the reason why we believe that environmental data can be helpful in computing is that this data is difficult to predict. But there is another known phenomenon which is also very difficult to predict – the behavior of human beings. In contrast to planets, we are not deterministic machines, we make choices, and these choices are difficult to predict. A natural idea is thus to conclude that the Earth is, in some reasonable sense, an intelligent being – similar to us. This, in a nutshell, is the main idea of the so-called *Gaia hypothesis*; see, e.g., [37].

At first glance, this hypothesis sounds like an unscientific mysticism; however, researchers made some sense of this hypothesis; see, e.g., [1, 36, 50]. What we propose is yet another rational interpretation of this hypothesis. Indeed, what is intelligence? It is largely the ability to solve complex problems – whether they are real-life complex problems or abstract ones. How do we know that someone has intelligence? Maybe we observe this person solving problems, maybe this person helps us solve complex problems.

From this viewpoint, since observing the environment can help us solve complex problems, we can therefore naturally say that in this sense, the Earth *is* an intelligent being – which is exactly what the Gaia hypothesis is claiming.

5 Conclusion

In this paper, we showed that non-trivial mathematics *can* be helpful in all three research directions that relate computing and environment. Based on these examples, we believe that we need other non-trivial mathematical ideas to solve other related problems. Let us jointly look for such solutions!

Acknowledgments

References

[1] O. D. N. Alcabes, S. Olson, and D. S. Abbot, "Robustness of Gaian feedbacks to climate perturbations", *Monthly Notices of the Royal Astronomical Society*, 2020, Vol. 492, pp. 2572–2577.

[2] L. Babai, R. Beals, J-y. Cai, G. Ivanyos, and E. M. Luks, *Multiplicative Equations Over Commuting Matrices*, Princeton/Rutgers DIMACS Center, Technical Report, TR 95-32, 1995.

[3] S. Basu, R. Pollack, and M.-F. Roy, *Algorithms in Real Algebraic Geometry*, Springer-Verlag, Berlin, 2006.

[4] J. Beck, *Data Processing under a Combination of Interval and Probabilistic Uncertainty and Its Application to Earth and Environmental Studies and Engineering*, Ph.D. Dissertation, Department of Computer Science, University of Texas at El Paso, 2004.

[5] D. Benko, "A new approach to Hilbert's third problem", *American Mathematical Monthly*, 2007, Vol. 114, pp. 665–676.

[6] V. G. Boltianskii, *Hilbert's Third Problem*, V. H. Winston & Sons, Washington, D.C., 1978.

[7] V. G. Boltianskii, "Combinatorial geometry", In: R. V. Gamrkrelidze (ed.), *Algebra, Topology, and Geometry*, Moscow, VINITI Publ., 1981, Vol. 19, pp. 209–274 (in Russian).

[8] J. Brady, O. Lerma, V. Kreinovich, and C. Tweedie, "Toward computing an optimal trajectory for an environment-oriented Unmanned Aerial Vehicle (UAV) under uncertainty", *Journal of Uncertain Systems*, 2015, Vol. 9, No. 2, pp. 84–94.

[9] J. Buchmann and F. Eisenbrand, "On factor refinement in number fields", *Mathematics Of Computation*, 1999, Vol. 68, No. 225, pp. 345–350.

[10] D. E. Cooke, R. Duran, A. Gates, and V. Kreinovich, "Geombinatoric problems of environmentally safe manufacturing and linear logic", *Geombinatorics*, 1994, Vol. 4, No. 2, pp. 36–47.

[11] R. Courant and H. Robbins, "Geometric constructions. The algebra of number fields," Ch. 3 in *What Is Mathematics?: An Elementary Approach to Ideas and Methods*, Oxford University Press, Oxford, England, UK, 1996, pp. 117–164.

[12] M. Dehn, "Über raumgleiche Polyeder", *Nachr. Acad. Wiss. Gottingen Math.-Phys. Kl.*, 1900, pp. 345–354.

[13] M. Dehn, "Über den Rauminhalt", *Mathematische Annalen*, 1901, No. 3, pp. 465–478.

[14] H. Derksen, E. Jeandel, and P. Koiran, "Quantum automata and algebraic groups", *Journal of Symbolic Computation*, 2005, Vol. 39, pp. 357–371.

[15] S. M. Escarzaga, C. Tweedie, O. Kosheleva, and V. Kreinovich, "How to predict nesting sites and how to measure shoreline erosion: fuzzy and probabilistic techniques for environment-related spatial data processing", *Proceedings of the 2016 World Conference on Soft Computing*, Berkeley, California, May 22–25, 2016, pp. 249–252.

[16] S. M. Escarzaga, C. Tweedie, O. Kosheleva, and V. Kreinovich, "How to predict nesting sites and how to measure shoreline erosion: fuzzy and probabilistic techniques for environment-related spatial data processing", In: L. Zadeh, R. R. Yager, S. N. Shahbazova, M. Reformat, and V. Kreinovich (eds.), *Recent Developments and New Direction in Soft Computing: Foundations and Applications*, Springer Verlag, Cham, Switzerland, 2018, pp. 595–604.

[17] R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics*, Addison Wesley, Boston, Massachusetts, 2005.

[18] O. Galindo, L. Bokati, and V. Kreinovich, "Towards a more efficient representation of functions in quantum and reversible computing", *Proceedings of the Joint 11th Conference of the European Society for Fuzzy Logic and Technology EUSFLAT'2019 and International Quantum Systems Association (IQSA) Workshop on Quantum Structures*, Prague, Czech Republic, September 9–13, 2019.

[19] O. Galindo and V. Kreinovich, "For quantum and reversible computing, intervals are more appropriate than general sets, and fuzzy numbers than general fuzzy sets", *Proceedings of the Joint 11th Conference of the European Society for Fuzzy Logic and Technology EUSFLAT'2019 and International Quantum Systems Association (IQSA) Workshop on Quantum Structures*, Prague, Czech Republic, September 9–13, 2019.

[20] G. Ge, *Algorithms Related to Multiplicative Representation of Algebraic Numbers*, PhD Dissertation, Department of Mathematics, University of California at Berkeley, 1993.

[21] G. Ge, "Recognizing units in number fields", *Mathematics of Computation*, 1994, Vol. 63, No. 207, pp. 377–387.

[22] D. Hilbert, "Mathematische Probleme", *Nachrichten von der Königl. Gesellschaft der Wiss. zu Göttingen*, 1900, pp. 253–297; English translation: "Mathematical Problems, lecture delivered before the International Congress of Mathematics in Paris in 1900", translated in *Bull. Amer. Math, Soc.*, 1902, Vol. 8, pp. 437–479; reprinted in F. E. Browder (ed.), *Mathematical Developments Arising from Hilbert's Problems*, American Math. Soc., Providence, Rhode Island, 1976.

[23] A. Jaimes, C. Tweedie, V. Kreinovich, and M. Ceberio, "Scale-invariant approach to multi-criterion optimization under uncertainty, with applications to optimal sensor placement, in particular, to sensor placement in environmental research", *International Journal of Reliability and Safety*, 2012, Vol. 6, No. 1–3, pp. 188–203.

[24] A. Jaimes, C. Tweedie, T. Magoč, V. Kreinovich, and M. Ceberio, "Optimal sensor placement in environmental research: designing a sensor network under uncertainty", In: M. Beer, R. L. Muhanna, and R. L. Mullen

(Eds.), *Proceedings of the 4th International Workshop on Reliable Engineering Computing REC'2010*, Singapore, March 3–5, 2010, pp. 255–267.

[25] 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*, 2012, Vol. 6, No. 2, pp. 104–113.

[26] R. Kellerhals, "Old and new on Hilbert's third problem", In: *Proceedings of the 9th Meeting of European Women in Mathematics (EWM), Loccum, Germany, 1999*, Hindawi Pub., Cairo, 2000, pp. 179–187.

[27] O. M. Kosheleva, "Axiomatization of volume in elementary geometry", *Siberian Mathematical Journal*, 1980, Vol. 21, No. 1, pp. 106–114 (in Russian); English translation in *Siberian Mathematical Journal*, 1980, Vol. 21, pp. 78–85.

[28] O. M. Kosheleva, "Hilbert Problems (Almost) 100 Years Later (From the Viewpoint of Interval Computations)", *Reliable Computing*, 1998, Vol. 4, No. 4, pp. 399–403.

[29] O. Kosheleva, V. Kreinovich. "Geombinatorics, computational complexity, and saving environment: let's start", *Geombinatorics*, 1994, Vol. 3, No. 3 (January), pp. 90–99.

[30] O. Kosheleva, M. Zakharevich, and V. Kreinovich, "If many physicists are right and no physical theory is perfect, then by using physical observations, we can feasibly solve almost all instances of each NP-complete problem", *Mathematical Structures and Modeling*, 2014, Vol. 31, pp. 4–17.

[31] V. Kreinovich, "Equidecomposability (scissors congruence) of polyhedra in $\mathbb{R}^3$ and $\mathbb{R}^4$ is algorithmically decidable: Hilbert's 3rd Problem revisited", *Geombinatorics*, 2008, Vol. 18, No. 1, pp. 26–34.

[32] V. Kreinovich and O. Kosheleva, "An application of logic to combinatorial geometry: how many tetrahedra are equidecomposable with a cube?". *Mathematical Logic Quarterly*, 1994, Vol. 40, No. 1, pp. 31–34.

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

[34] V. Kreinovich, L. Longpré, S. A. Starks, G. Xiang, J. Beck, R. Kandathi, A. Nayak, S. Ferson, and J. Hajagos, "Interval versions of statistical techniques, with applications to environmental analysis, bioinformatics, and privacy in statistical databases", *Journal of Computational and Applied Mathematics*, 2007, Vol. 199, No. 2, pp. 418–423.

[35] V. Kreinovich and R. Ouncharoen, "Fuzzy (and interval) techniques in the age of big data: an overview with applications to environmental science, geosciences, engineering, and medicine", *International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems*, 2015, Vol. 23, Suppl. 1, pp. 75–89.

[36] T. M. Lenton, S. J. Daines, J. G. Dyke, A. E. Nicholson, D. M. Wilkinson, and H. T. Williams, "Selection for Gaia across multiple scales", *Trends in Ecology & Evolution*, 2018, Vol. 33, No. 8, pp. 633–645.

[37] J. Lovelock, *Gaia: A New Look at Life on Earth*, Oxford University Press, Oxford, UK, 2000.

[38] B. Mishra, "Computational real algebraic geometry", in: *Handbook on Discreet and Computational Geometry*, CRC Press, Boca Raton, Florida, 1997.

[39] Y. Mohanty, "Construction of a 3/4-ideal hyperbolic tetrahedron out of ideal tetrahedra", *Discrete and Computational Geometry*, 2004, Vol. 32, pp. 117–128.

[40] W. D. Neumann, "Hilbert's 3rd problem and invariants of 3-manifolds", *Geom. Topol. Monogr.*, 1998, Vol. 1, pp. 383–411.

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

[42] J. C. Oxtoby, *Measure and Category: A Survey of the Analogies between Topological and Measure Spaces*, Springer Verlag, New York, Heidelberg, Berlin, 1980.

[43] C. Papadimitriou, *Computational Complexity*, Addison Welsey, Reading, Massachusetts, 1994.

[44] C. Servin, M. Ceberio, A. Jaimes, C. Tweedie, and V. Kreinovich, "How to describe and propagate uncertainty when processing time series: metrological and computational challenges, with potential applications to environmental studies", In: S.-M.Chen and W. Pedrycz (eds.), *Time Series Analysis, Modeling and Applications: A Computational Intelligence Perspective*, Springer Verlag, 2013, pp. 279–299.

[45] S. A. Starks and V. Kreinovich, "Environmentally-oriented processing of multi-spectral satellite images: new challenges for Bayesian methods", In: G. J. Erickson, J. T. Rychert, and C. R. Smith (eds.), *Maximum Entropy and Bayesian Methods*, Kluwer, Dordrecht, 1998, pp. 271–276.

[46] J.-P. Sydler, "Conditions nécessaires et suffisantes pour l'équivalence des polyèdres de l'espace euclidean à trois dimensions", *Comment. Math. Helv.*, 1965, Vol. 40, pp. 43–80.

[47] A. Tarski, *A Decision Method for Elementary Algebra and Geometry*, 2nd ed., Berkeley and Los Angeles, 1951, 63 pp.

[48] K. S. Thorne and R. D. Blandford, *Modern Classical Physics: Optics, Fluids, Plasmas, Elasticity, Relativity, and Statistical Physics*, Princeton University Press, Princeton, New Jersey, 2017.

[49] A. M. Turing. "On computable numbers, with an application to the Entscheidungsproblem", *Proceedings of the London Mathematical Society*, 1936/37, Vol. 42, pp. 230–265; "A correction", ibid, 1937, Vol. 43, pp. 544–546.

[50] S. A. Vakulenko, I. Sudakov, S. V. Petrovskii, and D. Lukichev, "Stability of a planetary climate system with the biosphere competing for resources", *Physics Reviews E*, 2021, Vol. 103, Paper 022202.