Review of: "Mathematics Is Physical"

Sergio Miguel Tomé[1]

1 Universidad de Salamanca

Potential competing interests: No potential competing interests to declare.

The article claims that mathematics is physical. To support that claim, the author analyzes the concept of Turing Machine, the concepts of classical and quantum information and the Gödel incompleteness theorem.

Despite the author's good intentions, the article suffers from weakness, vague claims, and mistakes.

Firstly, the idea presented in the article has been proposed for many different researchers. The article does not cite any of the previous researchers that have made similar claims.   There are hundreds of articles addressing this topic, the idea is not original. I suggest that the author should read carefully [1] to know the state of the art about the topic that he addresses.

The author writes “What is discussed in the paper highlights a fundamental principle that mathematics is ultimately influenced by the underlying physical entities”. Pag. 21

The author does not formulate the principle, and the claim is vague. In the literature, a principle has been already clearly enunciated and reasoned that connects mathematics and physics; it is the principle of computability [1]. The principle of computability explains the relation between computation and physics and it gives an explanation to the issue about the unreasonable effectiveness of mathematics noted and highlighted by Eugene Wigner [2]. Also, the author does not address the opposite position to the author's position, Platonism. There is not a critique or discussion about Platonism, which is defended by several mathematicians.

The author has different kinds of mistakes in his claims.

The author writes:

“The Turing machine was introduced in 1937, more than a decade after the establishment of quantum mechanics. However, Turing, his contemporaries, and subsequent scientists did not realize that these computational models assumed that the information directly processed by the machines was recorded on media that could be read or copied accurately at once.” pag. 11.

Firstsly, Turing was trying to resolve the Entscheidungsproblem (and he did it). He developed the Turing machine to address the Entscheidungsproblem and no other issues. Also, Turing was the first in considering computer models more powerful than the Turing machine, the O-machine [3]. Secondly, David Deutsch defined the quantum Turing machine in 1985 [4,5]. Currently, none has built a quantum Turing machine to the best of my knowledge. We don't know if it is physically possible. This is an open question, and there is a debate about it with different opinions [1].

The author says:

“Computers are finite physical entities” pag. 18

The author makes the same mistake that other physicists who have written on this subject have made in misunderstanding the scope of the concept of computation [1]. In the theory of computation, there exist a huge number of computational models with many different features. There are theoretical computers that can handle infinite sequences of numbers [6,7]. Whether those theoretical computers can exist in our universe is an open question.

The author writes:

“In conclusion, while the preceding discussion lacks mathematical rigor, it reveals a fundamental truth: mathematicians and computers are finite physical entities that are subject to limitations.”

We don’t know if it is a fundamental truth. It is one of the possibilities. This possibility is the Tegmark–Szudzik thesis [1] but there are also other possibilities.

The author writes:

“As a result, there will always be mathematical propositions that are beyond the reach of human...” pag. 18.

However, Gödel's theorems (I and II) show that humans can go beyond the limits of the axiomatic method.

The author writes:

“The proof presented here demonstrates that the number of programs is uncountably infinite, and that this is the fundamental reason why it is impossible to determine whether any program halts or not.”

That claim is false. The number of programs is not uncountably infinite. The number of programs is countably infinite. The impossibility of a program to determine if a program halts or runs forever is because the number of functions is uncountably infinite and the number of programs is countably infinite. There doesn't exist a program that carries out the function that is defined in the halting problem.

This article cannot be published in the current state. It must be corrected, and it requires to cite the literature already published about this topic.

I recommend the author to read up on this topic because there is a large amount of literature before rewriting the article for proper citation.

[1] Miguel-Tomé S, Sánchez-Lázaro ÁL, Alonso-Romero L. Fundamental Physics and Computation: The Computer-Theoretic Framework. *Universe*. 2022; 8(1):40. https://doi.org/10.3390/universe8010040

[2] Wigner, E.P. The Unreasonable Effectiveness of Mathematics in the Natural Sciences. *Commun. Pure Appl.*

Math. **1960**, *13*, 1–14.

[3] Turing, A.M. Systems of Logic Based on Ordinals. Proc. Lond. Math. Soc. 1939, s2-45, 161–228.

[4] Deutsch, D. Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. Proc. R. Soc. Lond. A Math.

Phys. Eng. Sci. 1985, 400, 97–117.

[5] Deutsch, D. Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. Proc. R. Soc. Lond. A Math.

Phys. Eng. Sci. 1985, 400, 97–117.

[6] Beggs, E.; Cortez, P.; Costa, J.; Tucker, J. Classifying the computational power of stochastic physical oracles. Int. J. Unconv. Comput. 2018, 14, 59–90.

[7] Carl, M. Ordinal Computability: An Introduction to Infinitary Machines; de Gruyter: Berlin, Germany