Logical Collapse: Why the P vs NP Problem Cannot Be Resolved

Kevin Wells, Ph.D.
Assistant Professor, School of Education
The University of Southern Mississippi
Kevin.E.Wells@usm.edu

July 13, 2025

Abstract

This paper argues that the P vs NP problem is not merely unsolved, but structurally unsolvable. The problem demands a universal binary answer, either P = NP or P $\neq$ NP, despite the existence of counterexamples that contradict both possibilities. Binary problems exhibit structural equivalence between solving and verifying, which implies P = NP holds in those cases. Structured problems, including multinomial and continuous classes, exhibit a necessary asymmetry between solving and checking, implying P $\neq$ NP. These two realities are logically incompatible with any universal claim. Therefore, the P vs NP problem, as stated, collapses under its own assumptions.

1 Introduction

The P vs NP problem is positioned as a central unsolved challenge in theoretical computer science. Formally, it asks whether every problem whose solution can be verified in polynomial time can also be solved in polynomial time. This framing suggests a clean binary outcome: either all NP problems can be solved in P (P = NP), or they cannot (P $\neq$ NP).

This paper presents the argument that such a framing is not just difficult, it is logically incoherent. I show that both possible answers to the problem are falsified by concrete counterexamples. As a result, the problem cannot be resolved under its current definition.

2 Binary Problems Imply P = NP

Binary decision problems represent a structural class where solving and verifying are indistinguishable. For example, in 2-SAT, the algorithm that finds a solution also verifies it. The solution space is discrete and constrained, and the logic required to solve is identical to that required to check.

Therefore, for binary problems:

$$\text{Solve} = \text{Check} \Rightarrow \text{P} = \text{NP (in this subclass)}$$

This is not a philosophical analogy. It is a structural identity. Binary problems logically satisfy P = NP.

3 Complex Problems Imply P $\neq$ NP

In contrast, structured problems with more complex solution spaces, such as graph coloring, scheduling, or real-valued optimization, exhibit a fundamental asymmetry between solving and checking. It is computationally easy to verify a candidate solution, but far harder to find one.

Therefore, in these problems:

$$\text{Check} \Rightarrow \text{P} \neq \text{NP (in this subclass)}$$

The asymmetry is not an artifact of limited algorithmic technique, it is a result of the underlying structural complexity.

4 The Logical Contradiction

The P vs NP problem demands a universal binary answer. But each direction of that binary question is logically falsified by an existing counterexample:

- If P = NP, then complex problems like 3-coloring violate the claim.
- If P $\neq$ NP, then binary problems like 2-SAT violate the claim.

This leads to the following contradiction:

- P= NP is false (disproven by structured problems)
- P $\neq$ NP is false (disproven by binary problems)

Therefore, no universal truth is possible under the current binary framing. The problem is formally unsolvable.

5 Epilogue: The Carnival Ring Toss

There is a reason no one has solved the P vs NP problem. And it is not because we are not clever enough, or because the right algorithm has not been written yet. It is because the problem, as framed, cannot be solved. You ca not prove P = NP, because complex problems structurally violate that equality. You can not prove P $\neq$ NP, because binary problems structurally satisfy it. One valid counterexample breaks each side of the universal claim. It is not a riddle. It is a rigged game. Like a carnival ring toss:

- The rules sound simple.
- The target looks hittable
- But the rim is just slightly too wide, the ring just slightly too light, and the framing just lightly too wrong.

You spend years refining your throw, only to find that the game is not designed to be won. It is designed to make you think you could have won if only you had tried a little harder.

That is what the P vs NP problem has become: A mathematically elegant, logically broken challenge that punishes structural clarity and rewards compliance with an ill-posed question. The only honest move left is to call the contradiction by name and stop pretending the ring ever fit the bottle.

> *"Strange game. The only winning move is not to play."*
> — WOPR, *WarGames* (1983)

Keywords: computational complexity, P vs NP, logical contradiction, structural asymmetry, decision theory, falsifiability, computational logic