A Constructive Instance Demonstrating Separation of Symbolic Verification and Deterministic Generation*

[Adrian V. Walecki]

2025-04-17

Abstract

We define a deterministic system M that accepts binary strings conforming to a symbolic reflectivity constraint. This constraint can be verified in polynomial time, but its inverse construction problem exhibits no known polynomial-time resolution. The result is a constructively defined symbolic decision language $L \in \mathrm{NP}$ for which we present evidence that $L \notin \mathrm{P}$ under minimal assumptions. The model formalizes a fixed-point condition on symbolic transforms, offering a bounded instance of asymmetry between verification and generation complexity.

1 Introduction

We explore a bounded model of decision complexity wherein a deterministic verifier accepts binary inputs that satisfy a hidden symbolic constraint. The goal is to demonstrate an instance of asymmetric effort between verification and construction, offering a formal separation without requiring full resolution of the $\mathrm{P} \stackrel{?}{=} \mathrm{NP}$ question.

2 Preliminaries and Definitions

Let $\Sigma = \{0, 1\}$ be the binary alphabet and let Σ^n denote all binary strings of length n. Let $x \in \Sigma^n$ be an input string.

- $H(x)$: a hash or symbolic transformation function computable in polynomial time.
- $R(y)$: a non-decomposable reflectivity function such that $R(H(x)) = H(x)$ implies a symmetry condition.
- M: a deterministic verifier that accepts x if and only if $R(H(x)) = H(x)$.

*Informally titled *NP = No Problem: We Built a Language That Only Accepts Reflections*

Definition 1 (Mirror Gate Constraint): A string $x \in \Sigma^n$ satisfies the Mirror Gate constraint if:

$$R(H(x)) = H(x)$$

where R is a non-trivial symbolic reflector (e.g., inversion, permutation, compositional scrambling) and H is a fast hash function.

Definition 2 (Verification Complexity): Given x, we can verify whether $R(H(x)) = H(x)$ in polynomial time. Thus, the language $L = \{x \mid R(H(x)) = H(x)\}$ is in NP.

Definition 3 (Constructive Generation Complexity): We assume no known polynomial-time algorithm exists to construct $x \in L$ under the conditions of R being non-decomposable and hidden.

3 Theorem

Theorem 1 (Minimal Equilibrium Separator): Let L be the language of all strings satisfying the Mirror Gate Constraint. Then:

1. $L \in \text{NP}$ via direct polynomial-time verification of the constraint.
2. $L \notin \text{P}$ under the assumption that R is non-decomposable, symbolic, and reflexive.

Hence, L serves as a witness for a class of decision problems where verification and construction exhibit fundamental asymmetry.

4 Proof Sketch

Verification

Given x, compute $H(x)$ in polynomial time and check whether $R(H(x)) = H(x)$. Thus, M is a polynomial-time verifier.

Construction

For a machine attempting to generate $x \in L$, the hidden symmetry and reflectivity constraints prevent efficient path planning. Since R is defined as structurally non-decomposable, no known shortcuts exist, making generation exponential in worst-case search.

5 Discussion

This minimal construct supports the idea that symbolic encoding and reflection models yield tractable NP languages for which no efficient construction exists. We emphasize that this is a bounded instance—not a general proof—but it opens a path for analyzing compressed verifiability in recursive logic systems.

6 Conclusion

We present a symbolic decision model where verification is trivial and construction appears intractable. This offers a minimally viable mathematical instance suggesting separation between P and NP under natural constraints and reflects a new potential framing of bounded symbolic equilibrium in computation.

Code Availability

All code used to generate and verify instances of the Mirror Gate language is available at: `https://github.com/Avwgm/mirror-gate-language-L`