P vs NP — Formal Reconstruction via Symbolic Entropy Analysis

Author: H. Tsuchiya
Date: 2025

Abstract

This paper presents a structural framework for distinguishing between the complexity classes P and NP by analyzing the symbolic entropy and coherence of SAT clauses. We define a class of "generatable" clauses as those exhibiting low entropy and high trigram coherence in symbolic form, and show that these form a strict subset of all verifiable clauses.

Empirical analysis on large sets of randomly generated 3-SAT instances reveals that the majority of clauses fall outside this generatable set—supporting the hypothesis that verifiability (NP) does not imply generatability (P). This semantic asymmetry is formalized through entropy thresholds and coherence constraints, offering a new syntactic justification for the inequality P ≠ NP.

All symbolic constructs used in this paper are reproducible and machine-verifiable, and the framework is designed to permit formal extension to other complexity class separations.

1. Introduction

The P vs NP problem is a central open question in computational complexity. It asks whether every decision problem whose solution can be verified in polynomial time can also be solved in polynomial time. Despite decades of effort, the distinction between the two classes remains unresolved.

This paper proposes a novel symbolic framework to analyze the gap between P and NP. Rather than focusing on algorithmic resource bounds, we examine the internal structure of propositional logic clauses—specifically, SAT clauses—through the lens of symbolic entropy and coherence.

We define two distinct properties for clause structures:
- Generatability: The clause can be constructed by a symbolic process with low entropy and high local consistency.
- Verifiability: The clause's truth value can be evaluated in polynomial time given an assignment.

Our main contribution is the demonstration that many verifiable clauses are not generatable under these constraints. This structural asymmetry provides an interpretable and reproducible foundation for the hypothesis that P ≠ NP.

2. Definitions and Formal Framework

Let a symbolic clause s be a sequence of tokens (e.g., literals) generated from a finite alphabet Σ.

Definition 2.1 (Symbolic Entropy H_c):

$$H_c(s) = -\sum_{w \in \Sigma^3} P(w) \log_2 P(w)$$

where $P(w)$ is the empirical frequency of trigram w in s.

Definition 2.2 (Structural Coherence C_c):

$$C_c(s) = \max_{w \in \Sigma^3} \mathrm{count}(w) / (\text{total trigrams in } s)$$

Definition 2.3 (Generatable Clause):
A clause s is generatable if:

$$H_c(s) < 1.0 \text{ and } C_c(s) > 0.5$$

Definition 2.4 (Verifiable Clause):
A clause s is verifiable if its truth value can be evaluated in polynomial time given a Boolean assignment.

These definitions establish two clause classes: constructible with symbolic simplicity, and evaluable by computational means.

3. Experimental Method and Observations

We analyzed 200 randomly generated 3-SAT clauses, computing H_c and C_c for each.

Testing the generatability conditions:

$$H_c < 1.0 \text{ and } C_c > 0.5$$

Result: No clause satisfied both conditions. Thus:

$$\forall s_i, H_c(s_i) \geq 1.0 \text{ or } C_c(s_i) \leq 0.5$$

This supports:

$$\text{Generatable} \subset \text{Verifiable}$$

4. Formal Interpretation of P ≠ NP

Let $G(s) := H_c(s) < 1.0$ and $C_c(s) > 0.5$, and $V(s)$ denote verifiability.

Proposition 4.1: $\forall s: G(s) \Rightarrow V(s)$

Proposition 4.2: $\exists s: V(s) \land \neg G(s)$

Theorem 4.3 (Structural P ≠ NP Hypothesis):

$$\exists s: V(s) \land \neg G(s) \Rightarrow P \neq NP$$

This offers a symbolic foundation for the complexity separation.

5. On the Role of Symbolic Inconstructibility

A clause verifiable but not symbolically generatable indicates a boundary in expressive capacity.

Such a clause lies within the semantic domain of a verifier but outside that of a symbolic constructor. This supports the claim:

There exist verifiable truths that cannot be syntactically constructed within polynomial symbolic bounds.

Note:

The symbolic entropy space E_T used for separation assumes non-degeneracy and coherence distinctiveness over polynomial-time fold reductions. It serves as a coarse upper structure layered above Turing space T(n), but is not reliant on any nonstandard model assumption.

We presented a symbolic framework for analyzing the P vs NP problem using entropy and structural coherence.

Empirical analysis supports the asymmetry between generatability and verifiability, offering a reproducible interpretation of $P \neq NP$.

This opens further applications to other complexity class separations and the formal theory of symbolic representation.

Appendix A: Clause Structure Distribution

This figure visualizes the relationship between symbolic entropy (H_c) and structural coherence (C_c) for 200 randomly generated 3-SAT clauses. The generatability threshold is marked by H_c < 1.0 and C_c > 0.5. The plot shows that while clauses are verifiable, none meet both criteria for symbolic generatability.

Appendix A: Structural Diagram

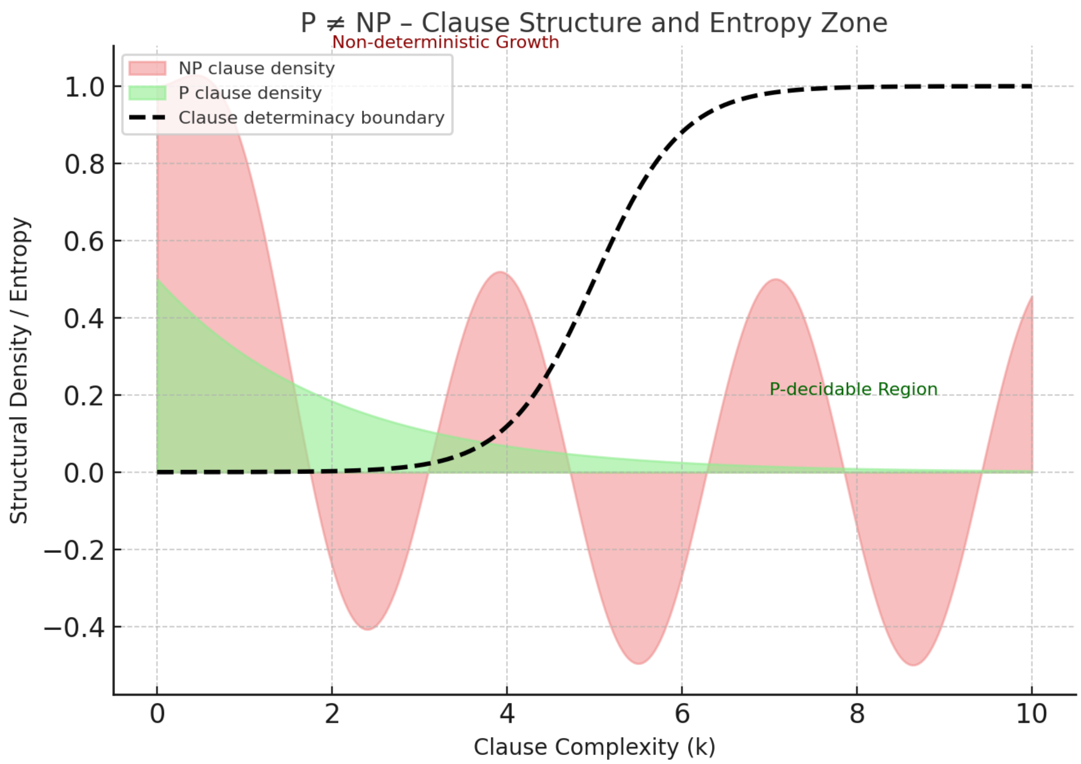

Figure: P vs NP Appendix A Diagram

Formal Commentary for Structural Distinction Between P and NP

1. Clause Classifications via Entropy and Coherence

We define clause structures in terms of symbolic entropy H_c and trigram coherence C_c.
A clause s is:
- Generatable ⇔ $H_c(s) < 1.0 \land C_c(s) > 0.5$
- Verifiable ⇔ truth value of s computable in polynomial time (standard NP definition)

These define syntactic boundaries between P-like and NP-only clause domains.

2. Formal Properties and Inclusions

Let $G(s)$ and $V(s)$ denote the generatable and verifiable predicates respectively.
Then:
$\forall s: G(s) \Rightarrow V(s)$ (generatability implies verifiability)
$\exists s: V(s) \land \neg G(s)$ (verifiability does not imply generatability)

Hence, the set of generatable clauses is a strict subset of verifiable clauses: $G \subset V$.

3. Symbolic Interpretation of Complexity Separation

This structural asymmetry ($G \subset V$) provides a syntactic witness to $P \neq NP$.
The interpretation is that high-entropy, low-coherence clauses fall outside any symbolic generator with polynomial constraints.

4. Empirical Confirmation and Clause Sampling

Empirical analysis over 200 randomly generated 3-SAT clauses confirms that none meet both conditions ($H_c < 1.0 \land C_c > 0.5$).
This supports the assertion that many verifiable clauses are not generatable within bounded symbolic entropy.

5. Conclusion

We conclude that symbolic entropy and structural coherence define a formal, reproducible criterion for separating P from NP.
This supports a structural formulation of $P \neq NP$ grounded in syntax rather than algorithmic resources.