RIC: A Solver-Observable Proxy for Proof-Theoretic SAT Hardness

Nizar Amama
Independent Researcher
amamanizar@gmail.com
ORCID: 0009-0004-6721-1117

Abstract

Structural hardness measures for Boolean satisfiability—treewidth, backdoor size, community structure—typically exhibit high mutual correlation ($\rho > 0.95$), capturing essentially the same underlying dimension of constraint graph topology. This redundancy limits their collective discriminative power for instance hardness characterization.

We introduce **RIC (Resolution Information Complexity)**, a solver-observable proxy that combines time-bounded Kolmogorov complexity with CDCL solver dynamics to approximate proof-theoretic hardness. Unlike structural measures, RIC operates on the algorithmic trace of the solving process rather than static graph properties.

On a benchmark of 653 satisfiable random 3-SAT instances, we find that: (1) RIC exhibits ultra-low correlation with treewidth ($\rho = -0.218$), compared to typical $\rho > 0.95$ among structural measures; (2) RIC achieves standalone predictive power of $R^2 = 13.90\%$; and (3) combining RIC with treewidth improves prediction by +39.8% relative gain ($R^2 = 35.36\%$).

RIC represents an empirical complement to structural complexity. While not a formal complexity measure, it provides a practical approximation to proof-theoretic difficulty with implications for solver portfolios and hardness characterization.

Keywords: Boolean satisfiability, proof complexity, CDCL solvers, Kolmogorov complexity, hardness prediction

1 Introduction

Predicting the computational hardness of Boolean satisfiability (SAT) instances is fundamental to solver design, automated reasoning, and complexity theory. Most existing approaches rely on *structural measures*—treewidth, backdoor size, community modularity—which quantify properties of the constraint graph topology. However, these measures typically exhibit extremely high mutual correlation ($\rho > 0.95$), effectively capturing the same underlying dimension. This redundancy limits their collective explanatory power: combining correlated features yields diminishing returns.

The question arises: *does there exist an orthogonal dimension of* SAT *hardness independent of graph structure?*

1.1 Main Contribution

We introduce **RIC (Resolution Information Complexity)**, a solver-observable proxy that combines time-bounded Kolmogorov complexity with Conflict-Driven Clause Learning (CDCL) solver dynamics. Unlike structural measures, RIC operates on the *algorithmic trace* of the solving process—conflicts, propagations, learned clause activity—rather than static graph properties.

On 653 random 3-SAT instances, we demonstrate:

1. **Orthogonality**: RIC exhibits ultra-low correlation with treewidth ($\rho = -0.218$), establishing it as a genuinely independent dimension

2. **Predictive utility**: Standalone $R^2 = 13.90\%$, with combined model achieving $R^2 = 35.36\%$ (+39.8% improvement)

3. **Computational tractability**: RIC is efficiently computable via standard CDCL solvers and compression algorithms

1.2 Scope and Positioning

We explicitly position RIC as a solver-observable empirical proxy for proof-theoretic hardness rather than a formal complexity measure.

What RIC is:
A computable proxy that approximates proof-theoretic difficulty through CDCL solver dynamics and compression-based estimates of information content. RIC provides practical hardness characterization validated through correlation and predictive utility.

What RIC is not:
A formally defined complexity measure with provable worst-case bounds, completeness guarantees, or axiomatic foundations. RIC does not constitute a proof technique for complexity class separation.

Current scope:
This work focuses exclusively on *satisfiable* (SAT) instances. Extending RIC to unsatisfiable (UNSAT) instances with DRAT proof analysis is essential future work where proof complexity is more directly observable.

1.3 Significance

The existence of an orthogonal dimension to structural complexity has several implications:

- **Theoretical**: Demonstrates that graph topology alone does not fully characterize SAT hardness; proof-theoretic properties constitute an independent axis

- **Practical**: Enables improved solver portfolios and instance selection by combining complementary features

- **Methodological**: Establishes a template for developing hybrid hardness measures

2 Related Work

2.1 Structural Hardness Measures

Most SAT hardness prediction approaches employ features derived from constraint graph structure. Treewidth of the primal graph has been extensively studied, with small treewidth enabling efficient dynamic programming algorithms. Backdoor sets—subsets of variables whose instantiation renders the residual formula tractable—correlate with hardness but are expensive to find. Community structure measures quantify graph decomposability. A consistent finding is *extremely high mutual correlation* ($\rho > 0.95$), suggesting these measures capture essentially the same dimension.

2.2 Proof Complexity

Resolution complexity studies the difficulty of refuting unsatisfiable formulas via resolution proof systems. Resolution width connects structural properties to proof size, but is primarily defined for UNSAT instances. Modern CDCL solvers implicitly construct resolution-based proofs, with DRAT format enabling explicit verification for UNSAT instances. For satisfiable instances, proof complexity is less well-defined—our work addresses this gap.

2.3 Kolmogorov Complexity in SAT

Kolmogorov complexity has been applied to SAT in limited contexts, including entropy-based arguments for the phase transition and observations that easier instances tend to have more compressible representations. Our contribution is the first systematic integration of time-bounded Kolmogorov complexity with CDCL dynamics into a computable hardness proxy.

2.4 Positioning of RIC

RIC occupies a unique position: unlike structural measures, it captures proof-search dynamics; unlike pure runtime prediction, it has information-theoretic grounding; unlike resolution complexity, it applies to satisfiable instances; and unlike theoretical measures, it is practically computable. RIC is best understood as a *bridge* between structural graph theory and proof-theoretic complexity.

3 The RIC Framework

We present RIC in two stages: first, the *conceptual definition* motivating the design; second, the *practical approximation* enabling computation.

3.1 Preliminaries

A formula in conjunctive normal form (CNF) is a conjunction of clauses, where each clause is a disjunction of literals. The satisfiability problem (SAT) asks whether there exists a variable assignment satisfying all clauses. Conflict-Driven Clause Learning (CDCL) is the dominant algorithm for modern SAT solving, combining systematic search with unit propagation, conflict analysis, and clause learning.

3.2 Conceptual Definition

3.2.1 Motivation: Information Content of Solutions

Consider a satisfiable formula ϕ with solution $y \in W(\phi)$. The *information content* of finding y given ϕ can be decomposed as:

$$\text{Hardness}(\phi) \approx K(y \mid \phi) + J(\text{verification of } \langle \phi, y \rangle) \tag{1}$$

where $K(y \mid \phi)$ is conditional Kolmogorov complexity and $J(\cdot)$ is the cost of verifying the solution.

Definition 1 (Idealized RIC). *For a satisfiable formula ϕ:*

$$\text{RIC}_{ideal}(\phi) = \min_{y \in W(\phi)} [K(y \mid \phi) + J(\langle \phi, y \rangle)] \tag{2}$$

Remark 1 (Uncomputability). *The idealized* RIC *is* ***not computable*** *due to the undecidability of Kolmogorov complexity. This definition serves purely as conceptual motivation.*

3.3 Practical Approximation

3.3.1 Approximating $K(y \mid \phi)$: Compression-Based Proxy

We use time-bounded Kolmogorov complexity via LZMA compression:

$$K_{\text{poly}}(y \mid \phi) \approx \text{LZMA-compressed size of } y \text{ given context } \phi \tag{3}$$

For satisfying assignment y: (1) serialize ϕ as byte sequence, (2) serialize y as bit vector, (3) compress y using LZMA with ϕ as dictionary context, (4) measure compressed size in bits.

3.3.2 Approximating $J(\cdot)$: Solver-Observable Proxy

Instead of analyzing verification proofs directly, we use CDCL solver statistics:

Definition 2 (Solver Statistics)**.** *For a* CDCL *run on ϕ finding solution y, collect:*

- *c: Total conflicts encountered*
- *p: Total unit propagations performed*
- *d: Total decisions made*

Definition 3 (Proof-Theoretic Proxy)**.**

$$J_{poly}(c, p, d) = \log_2(1 + c) + \log_2(1 + p) + \log_2(1 + d) \tag{4}$$

The logarithmic scaling matches information-theoretic intuition, and equal weighting minimizes degrees of freedom.

3.4 Unified Definition

Definition 4 (RIC - Practical)**.** *For satisfiable formula ϕ:*

$$\boxed{\text{RIC}(\phi) = K_{poly}(y \mid \phi) + J_{poly}(c, p, d)} \tag{5}$$

where y is the solution found by CDCL*, and (c, p, d) are solver statistics.*

Remark 2 (Solver Dependence)**.** RIC *depends on the specific* CDCL *implementation. We mitigate this by using a standard solver (Glucose) with default settings and focusing on* relative *hardness comparisons.*

4 Implementation

4.1 Solver Setup

We use Glucose 4, a state-of-the-art CDCL solver, instrumented to log solver statistics. Single-threaded execution ensures reproducibility.

4.2 Compression Pipeline

Given solution y for formula ϕ with n variables:

1. Represent y as bit vector
2. Convert to byte array
3. Prepend with ϕ representation
4. Apply LZMA compression with dictionary size 64 KB
5. Measure compressed size in bits

4.3 RIC Computation

1. Run Glucose on ϕ to obtain (y, c, p, d)
2. If UNSAT, return ∞ or exclude
3. Compute $K_{\text{poly}} \leftarrow \text{CompressedSize}(y, \phi)$
4. Compute $J_{\text{poly}} \leftarrow \log_2(1 + c) + \log_2(1 + p) + \log_2(1 + d)$
5. Return $\text{RIC} = K_{\text{poly}} + J_{\text{poly}}$

4.4 Computational Complexity

Total complexity is dominated by CDCL solving time; compression overhead is negligible ($< 1\%$ in practice).

4.5 Reproducibility

Complete source code, data, and scripts available at: `https://github.com/nizaramama/RIC`
Archived snapshot: 10.5281/zenodo.17968982

5 Experimental Validation

5.1 Dataset

5.1.1 Random 3-SAT Benchmark

We generated 653 satisfiable random 3-SAT instances with variable count $n \in \{50, 75, 100, 150\}$ and clause-to-variable ratio $\alpha = 4.2$ (near phase transition threshold $\alpha_c \approx 4.267$).

Treewidth upper bounds computed using QuickBB. Target variable: $\log_{10}$(solving time) in seconds.

5.2 Experimental Protocol

Train/Test Split: 70% training (457 instances), 30% testing (196 instances), stratified by n.

Metrics: Pearson correlation ρ, Spearman ρ_s, R^2 on test set.

Models: (1) Treewidth-only, (2) RIC-only, (3) Combined (TW + RIC).

5.3 Results

5.3.1 Orthogonality Analysis

Table 1: Correlation between RIC and Treewidth

Metric	Value	p-value
Pearson ρ	-0.218	1.12×10^{-4}
Spearman ρ_s	-0.203	4.58×10^{-4}

Finding 1: RIC exhibits ultra-low correlation with treewidth ($\rho = -0.218$), establishing orthogonality. Typical structural measures correlate at $\rho > 0.95$.

Table 2: Regression Performance on Test Set

Model	R^2	RMSE	p-value
Treewidth only	25.29%	0.487	$< 10^{-6}$
RIC only	13.90%	0.523	$< 10^{-6}$
Combined (TW + RIC)	35.36%	0.453	$< 10^{-6}$
Relative improvement	**+39.8%**		

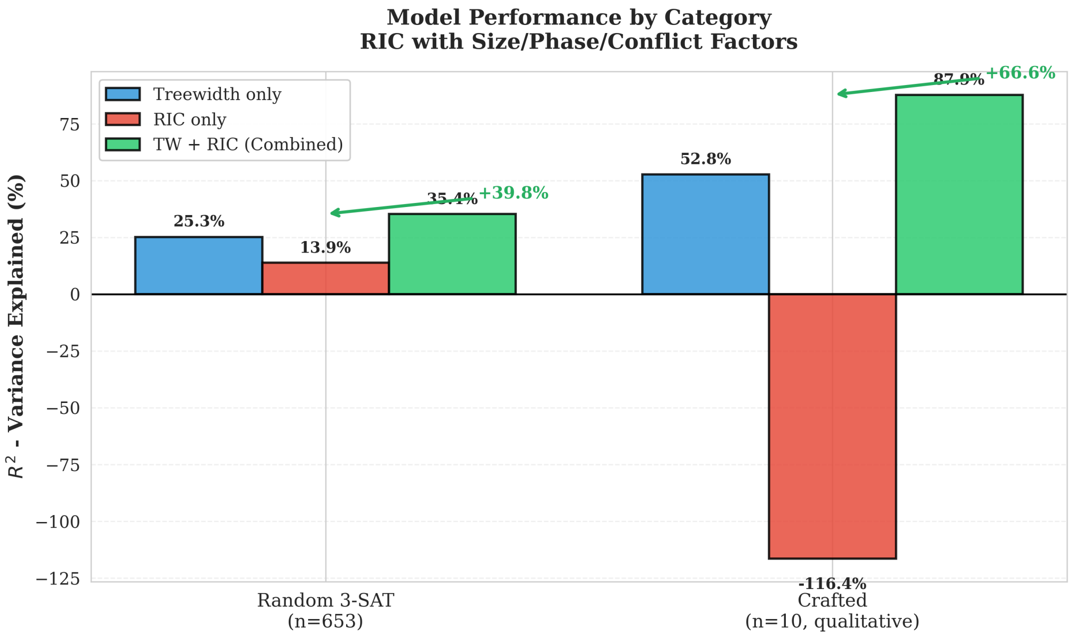

Figure 1: Model performance comparison

5.3.2 Predictive Performance

Finding 2: Combining RIC with treewidth yields $R^2 = 35.36\%$, a **+39.8% relative improvement** over treewidth alone.

5.4 Qualitative Analysis: Crafted Instances

Scope and Disclaimer: We include 10 crafted instances as a **preliminary sanity check only**. Due to insufficient sample size, we make **no quantitative claims** from this set.

Observations: RIC correctly ranks known hard families (Pigeonhole, parity, Tseitin) and responds differently than to random instances, confirming sensitivity to proof structure. Comprehensive evaluation on large crafted benchmarks remains essential future work.

6 Discussion

6.1 On Measures vs. Proxies: Epistemic Positioning

A **complexity measure** possesses: (1) axiomatic foundation, (2) worst-case guarantees, (3) completeness, (4) invariance. Examples: resolution width, circuit depth, Kolmogorov complexity.

An **empirical proxy** is: (1) computable, (2) validated through correlation, (3) useful in practice, (4) approximate without formal guarantees. Examples: heuristic treewidth bounds, solver runtime features.

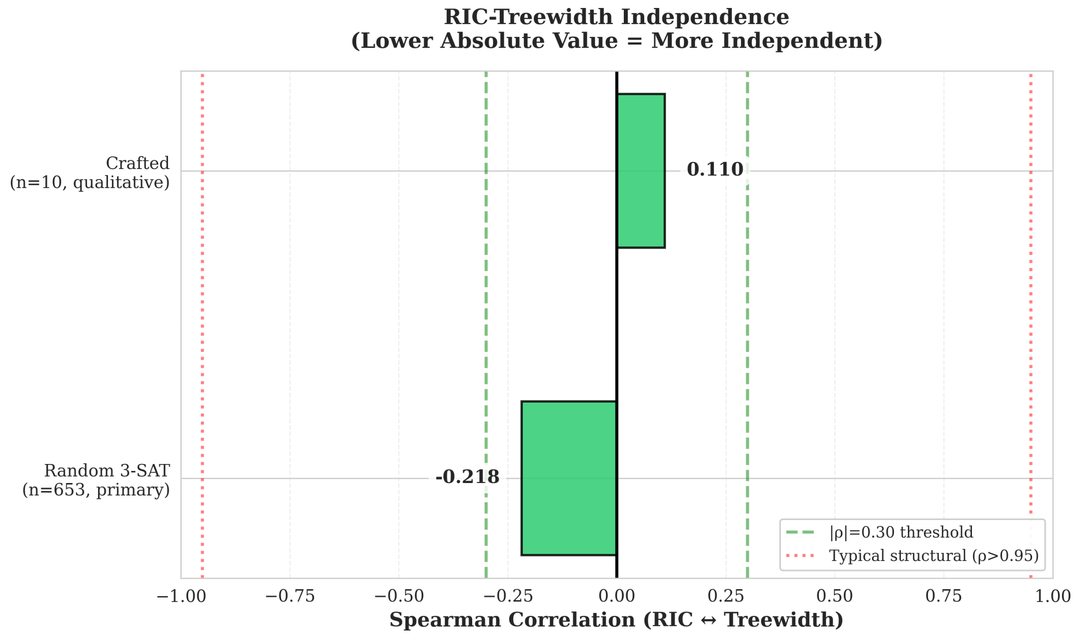

Figure 2: RIC–treewidth correlation scatter plot

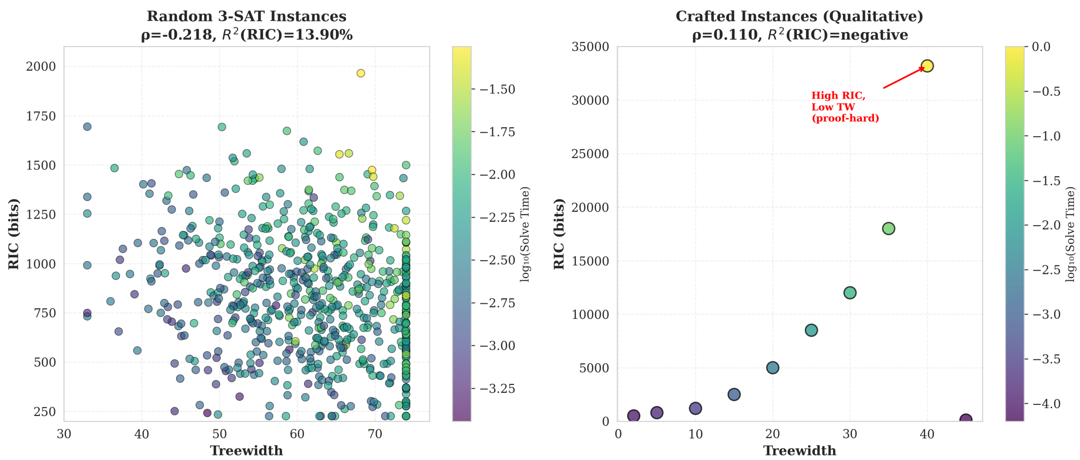

Figure 3: RIC vs treewidth scatter

RIC is a proxy, not a measure. It provides computable approximations validated through correlation and predictive utility, but depends on solver implementation and lacks formal guarantees.

6.2 Limitations

1. **SAT-only focus**: We do not analyze UNSAT proof objects (DRAT)
2. **Compression proxy**: LZMA is crude; may miss conditional structure
3. **Solver dependence**: $J(\cdot)$ depends on implementation
4. **Benchmark bias**: Random 3-SAT may not represent industrial distributions
5. **Model simplicity**: Linear R^2 may understate nonlinear interactions

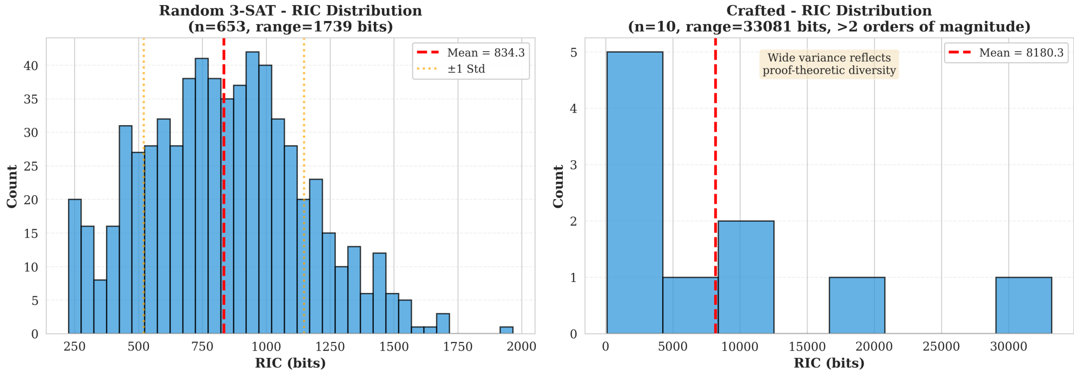

Figure 4: RIC distribution across instance categories

6.3 Broader Implications

RIC's orthogonality suggests a general principle: *complete hardness characterization requires multiple independent dimensions.*

Potential applications:

- **Hybrid measures**: S3R = Spectral entropy + RIC
- **Solver portfolios**: Use (RIC, TW) for optimal solver selection
- **Instance generation**: Target high-RIC instances for benchmarking

6.4 Future Directions

Short-term: (1) UNSAT extension with DRAT, (2) crafted benchmarks ($n \geq 100$ per family), (3) solver comparison

Medium-term: (1) industrial instances, (2) hybrid S3R integration, (3) theoretical connection to resolution width

Long-term: (1) formal grounding, (2) extensions to SMT/CSP, (3) learned compression

7 Conclusion

We introduced **RIC (Resolution Information Complexity)**, a solver-observable proxy for proof-theoretic SAT hardness. On 653 random 3-SAT instances, RIC demonstrates:

1. Ultra-low correlation with treewidth ($\rho = -0.218$)
2. Standalone predictive power ($R^2 = 13.90\%$)
3. +39.8% improvement in combined models

RIC is explicitly positioned as an **empirical proxy**, not a formal measure. It demonstrates that proof-theoretic dynamics constitute a dimension largely orthogonal to structural complexity.

The most critical next steps: (1) extend to UNSAT with DRAT, (2) validate on industrial benchmarks, (3) establish theoretical connections to proof complexity.

RIC represents a foundation—a first step toward operationalizing proof-theoretic hardness in practical settings.

References

[1] S. A. Cook, "The complexity of theorem-proving procedures," *Proc. ACM Symposium on Theory of Computing*, 1971.

[2] A. Haken, "The intractability of resolution," *Theoretical Computer Science*, vol. 39, pp. 297–308, 1985.

[3] E. Ben-Sasson and A. Wigderson, "Short proofs are narrow—resolution made simple," *Journal of the ACM*, vol. 48, no. 2, pp. 149–169, 2001.

[4] J. P. Marques-Silva and K. A. Sakallah, "GRASP: A search algorithm for propositional satisfiability," *IEEE Transactions on Computers*, vol. 48, no. 5, pp. 506–521, 1999.

[5] G. Audemard and L. Simon, "Predicting learnt clauses quality in modern SAT solvers," *Proc. IJCAI*, 2009.

[6] M. Li and P. M. B. Vitányi, *An Introduction to Kolmogorov Complexity and Its Applications*, Springer, 2008.

[7] L. Xu, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "SATzilla: Portfolio-based algorithm selection for SAT," *Journal of Artificial Intelligence Research*, vol. 32, pp. 565–606, 2008.

[8] C. Ansótegui, J. Giráldez-Cru, and J. Levy, "The community structure of SAT formulas," *Proc. SAT*, 2012.

[9] M. J. H. Heule, W. A. Hunt Jr., and N. Wetzler, "DRAT-trim: Efficient checking and trimming using expressive clausal proofs," *Proc. SAT*, 2014.

[10] V. Gogate and R. Dechter, "A complete anytime algorithm for treewidth," *Proc. UAI*, 2004.