Experimental Results for Vampire on the Equational Theories Project

Mikoláš Janota
Czech Technical University in Prague, CIIRC, Czechia

August 22, 2025

Abstract

Equational Theories Project is a collaborative effort, which explores the validity of certain first-order logic implications of certain kind. The project has been completed but triggered further research. This report investigates how much can be automatically proven and disproven by the automated theorem prover Vampire. An interesting conclusion is that Vampire can prove all the considered implications that hold and also is able to refute a vast majority of those that do not hold.

1 Introduction

This report accompanies experiments carried out by the author. All the experiments and relevant scripts are placed on zendodo [8].

Terrence Tao proposed on his blog [10] a collaborative project, which aims to bring together mathematicians and researchers on automated reasoning and other related fields. The project aims to classify all implications of a certain type. The implications are between two universally quantified first order logic equalities that use a single binary operation—let us denote this operation as $*$. For illustration, consider the following implications.

$$(x * y) * z = x * (y * z) \to x * y = y * x \tag{1}$$

$$x * y = y * x \to (x * y) * z = x * (y * z) \tag{2}$$

$$x * y = u * w \to x * y = y * x \tag{3}$$

The first implication (1) asks if associativity implies commutativity. This implication does not hold because for instance matrix multiplication is associative but it is not commutative. One may also ask if commutativity implies associativity (2), which also does not hold because for instance $\frac{x+y}{2}$ is commutative but it is not associative. On the other hand, implication (3) does hold, because the left-hand side requires that the operation $*$ always returns the same value, and therefore it is necessarily commutative.

Table 1: Solving methods

method	command-line arguments
fmb 500i	`-i 500 -sa fmb -sas cadical`
satur 500i	`-i 500 --mode casc`
fmb 60s	`-t 60s -sa fmb -sas cadical`
satur 600s	`-t 600s --mode casc`
fmb 600s	`-t 600s -sa fmb -sas cadical`

2 Experimental Setup

We consider all the equations in `generate_eqs_list.eqs`. There are $n = 4{,}694$ equations, which means there are $n^2 - n$ possible implications pairs, giving 22,028,942 pairs. Even though many pairs could be inferred from the value of other pairs through the transitivity of implication, purposefully we do not do that. Meaning, all pairs are targeted directly and only marked as solved if the one of the prover's configurations decided its validity (refuted/proven).

The automated theorem prover Vampire [6] supports saturation-based proving [2, 3]. It also has a finite model builder [5, 7], which has the SAT solver CaDiCaL as its backend [4].[1]

The input for Vampire was generated in the TPTP format [9], in CNF, already negated and skolemized.[2] The following TPTP corresponds to the test whether commutativity implies associativity.

```
% 43 AND NOT 4512
% m(X, Y) = m(Y, X) AND NOT m(X, m(Y, Z)) = m(m(X, Y), Z)
cnf(lhs, axiom, m(X, Y) = m(Y, X)).
cnf(rhs, negated_conjecture, m(a, m(b, c)) != m(m(a, b), c)).
```

All experiments were run on a server with two AMD EPYC 7513 32-Core processors @ 3680 MHz and with 514 GB RAM with 100 jobs in parallel. The problems were tackled by Vampire 5.0.0 (Release build, commit 128f1f6ca on 2025-07-30 12:07:12 +0200) with CaDiCaL: cadical-2.1.3. Times were measured in walk-clock time.

3 Experiments

Vampire was run on all the problems with increasing time out and alternating between the finite model building mode and the saturation mode as summarized by Table 1.

Table 2 shows how many problems were solved by the individual method. Most problems are solved with the short timeout of 500 instructions. Figure 1

[1] Enabled by the options `-sa fmb -sas cadical`.
[2] The `generate_tptp.py` script is used for this.

Method	Refuted	Proven	Total
fmb 500i	13,837,151	275,209	14,112,360
satur 500i	778	7,895,986	7,896,764
fmb 60s	16,302	0	16,302
satur 600s	36	2,390	2,426
fmb 600s	28	0	28
total	13,854,295	8,173,585	22,027,880

Table 2: Overview of the Results

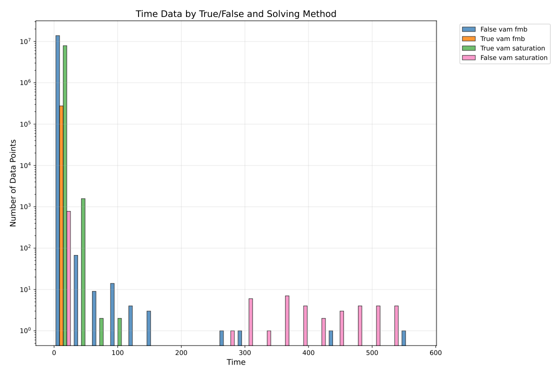

Figure 1: Histogram of the solving times.

shows the solving times organized in a histogram and divided by the solving method and the result.

As expected, the finite model builder is mainly successful in refuting implications and the saturation-based approach in proving them. However, it can also be the other way around even though it is rare.

When proving the implication $A \to B$, a saturation-based prover *can* determine that the implication does not hold. This is when the prover runs on the formula $A \wedge \neg B$ and eventually, the calculus of the prover does not enable it to derive any more clauses, and it has not derived the empty clause so far. In such case, however, we do not have a witnessing model that would show that the implication does not hold (a model of $A \wedge \neg B$). It is also not guarantee that a finite model exists if this happens.

All the problems that were not solved are marked false in The equational project [1]. This means that Vampire can prove all the implications that can be proven, and the challenge lies the implications that need to be refuted. Additional 22 problems can be derived by calculating propagating via transitivity of implication.[3] Only 310 of the undecided implications require an infinite model according to the Equational project, which indicates there is also a room for

[3]For example $511 \not\to 3079$ follows from $1120 \to 511$ and $1120 \not\to 3079$.

improvement for finite model finding. Infinite model finding of course poses a hard challenge.

Acknowledgements

The author would like to thank Martin Suda, Geoff Sutcliffe, and Chad Brown for discussions about this project. The research was supported by the Ministry of Education, Youth and Sports within the dedicated program ERC CZ under the project POSTMAN no. LL1902, by the Czech Science Foundation grant no. 25-17929X, and by the European Union under the project ROBOPROX (reg. no. CZ.02.01.01/00/22_008/0004590). This article is part of the RICAIP project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857306.

References

[1] Equational theories project. URL: https://teorth.github.io/equational_theories/dashboard/.

[2] Leo Bachmair and Harald Ganzinger. Rewrite-based equational theorem proving with selection and simplification. *Journal of Logic and Computation*, 4(3):217–247, 1994. Original formalization of the superposition calculus in a saturation framework. doi:10.1093/logcom/4.3.217.

[3] Leo Bachmair and Harald Ganzinger. Resolution theorem proving. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*, volume I, pages 19–99. Elsevier and MIT Press, 2001. Foundational description of saturation-based reasoning and the superposition calculus.

[4] Armin Biere, Tobias Faller, Katalin Fazekas, Mathias Fleury, Nils Froleyks, and Florian Pollitt. CaDiCaL 2.0. In Arie Gurfinkel and Vijay Ganesh, editors, *Computer Aided Verification - 36th International Conference, CAV 2024, Montreal, QC, Canada, July 24-27, 2024, Proceedings, Part I*, volume 14681 of *Lecture Notes in Computer Science*, pages 133–152. Springer, 2024. doi:10.1007/978-3-031-65627-9_7.

[5] Koen Claessen and Niklas Sörensson. New techniques that improve MACE-style model finding. In *Proc. of Workshop on Model Computation (MODEL)*, 2003. URL: https://web.archive.org/web/20070109072155/http://www.cs.chalmers.se/~koen/pubs/entry-model-paradox.html.

[6] Laura Kovács and Andrei Voronkov. First-order theorem proving and vampire. In Natasha Sharygina and Helmut Veith, editors, *Computer Aided Verification*, pages 1–35, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[7] Giles Reger, Martin Riener, and Martin Suda. Symmetry avoidance in mace-style finite model finding. In Andreas Herzig and Andrei Popescu, editors, *Frontiers of Combining Systems - 12th International Symposium, FroCoS 2019, London, UK, September 4-6, 2019, Proceedings*, volume 11715 of *Lecture Notes in Computer Science*, pages 3–21. Springer, 2019. doi:10.1007/978-3-030-29007-8_1.

[8] Mikoláš Janota. Experimental results for vampire on the equational theories project, 2025. doi:10.5281/zenodo.16910288.

[9] Geoff Sutcliffe. The TPTP problem library and associated infrastructure. *J. Autom. Reasoning*, 43(4):337–362, 2009. doi:10.1007/s10817-009-9143-8.

[10] Terrence Tao. A pilot project in universal algebra to explore new ways to collaborate and use machine assistance?, 2025. URL: https://terrytao.wordpress.com/2024/09/25/a-pilot-project-in-universal-algebra-to-explore-new-ways-to-collaborate-and-use-machine-assistance/.