Revisiting GlucosER in a Modern Solver

Zachary Battleman
Carnegie Mellon University, Pittsburgh, Pennsylvania, US

Florian Pollitt
University Freiburg, Freiburg, Germany

Mathias Fleury
University Freiburg, Freiburg, Germany

Yakir Vizel
Technion, Israel Institute of Technology, Israel

Marijn J. H. Heule
Carnegie Mellon University, Pittsburgh, Pennsylvania, US

Armin Biere
University Freiburg, Freiburg, Germany

Randal E. Bryant
Carnegie Mellon University, Pittsburgh, Pennsylvania, US

1 Introduction

One of the main weaknesses of the conflict-driven clause learning paradigm is its foundation on resolution, a fairly weak proof system [5]. To remedy this, "inprocessing" techniques enable the controlled application of stronger proof systems, yielding performance gains. One such proof system, extended resolution (ER) [9] is the focus of this work. ER expands the resolution proof system with the ability to add definitions of the form $z \leftrightarrow \ell_1 \land \ell_2$ for a fresh variable z. There are many examples of problems that have exponential lowerbounds on resolution proofs, but polynomial proofs with ER [4–6]. Intuitively, the definitions allow the proofs to be greatly reduced by removing the need for "duplicate" steps. However, despite ER's theoretical benefits, using definitions effectively remains an open challenge.

One of the first attempts towards leveraging ER in practice was the solver GlucosER [1]. In this work, we reimplemented its definition learing strategy in the modern solver CaDiCaL. Our experiments show that this tool, CaDiCaL-ER, is very effective on certain families of formulas and crucially demonstrates greater benefit than what can be explained by general solver improvements since the release of GlucosER.

2 CaDiCaL-ER, CaDiCaL with Extended Resolution

GlucosER's strategy for learning definitions is fairly simple. Whenever two consecutive clauses of the form $\overline{\ell_1} \lor C$ and $\overline{\ell_2} \lor C$ are learned, it introduces the definition $z \leftrightarrow \ell_1 \lor \ell_2$ Future learned clauses of the form $\ell_1 \lor \ell_2 \lor C$ are replaced by $z \lor C$. This technique is integrated into conflict analysis during CDCL as a recursive post-processing step for learned clauses [1].

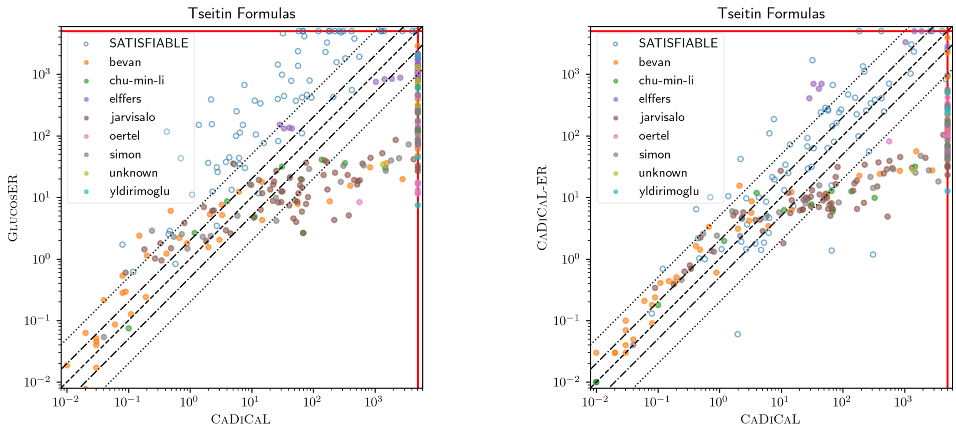
To make this approach practical, we must carefully manage the database of definitions. If too many definitions are deleted or too few are added, then the solver does not benefit from this technique. However, if too many definitions are added or too few are deleted, the solver becomes overwhelmed by both useless definitions whose propogations do not result in conflicts and long propagation chains of definitions involving other definitions. While the overhead introduced by this technique can be large, especially if the definition database is not tuned properly, this strategy is very effective on certain benchmark families.

29th International Conference on Theory and Applications of Satisfiability Testing (SAT 2026).
Editors: Alexey Ignatiev and Stefan Szeider; Article No. 36; pp. 36:1–36:3
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

3 Experiments

The experiments were run on Bridges-2 at the Pittsburgh Supercomputing Center [3] with AMD EPYC 7742 CPUs, 5000 s time limit and 256 GB of memory shared among 64 solvers. We present two families of formulas on which CaDiCaL-ER performed notably well.

The first family is the "SAT Competition Tseitin Formula" family as defined by the Global Benchmark Database [8]. We see a general improvement for unsatisfiable formulas in both GlucosER and CaDiCaL-ER.



(a) GlucosER vs. CaDiCaL

(b) CaDiCaL-ER vs. CaDiCaL

Figure 1 A comparison of CaDiCaL to GlucosER (left) and CaDiCaL-ER (right) on Tseitin formulas.

The second family is the Mutilated Chessboard problem, a classic, difficult problem for automated reasoning tools [7]. The problem asks whether it is possible to use 2×1 dominoes to tile an $n \times n$ chessboard with its top-left and bottom-right corners removed. CaDiCaL-ER demonstrates significantly improved performance compared to both CaDiCaL and GlucosER.

Table 2 Runtime in seconds on Mutilated Chessboard formulas.

n	CaDiCaL	GlucosER	CaDiCaL-ER
17	2616	901	90
18	1485	3442	143
19	> 5000	> 5000	203
20	> 5000	> 5000	245

4 Conclusion

This work revisits the ideas of GlucosER in the modern solver CaDiCaL. Our experiments crucially show that the technique synergizes with modern solver developments independently of general improvements. For future work, we would like to better understand precisely why CaDiCaL-ER performs so well on the Mutilated Chessboard formulas as well as revisit other techniques to see if they similarly benefit from modern solver developments.

References

1 Gilles Audemard, George Katsirelos, and Laurent Simon. A restriction of extended resolution for clause learning SAT solvers. *Proceedings of the AAAI Conference on Artificial Intelligence*, 24(1):15–20, Jul. 2010. doi:10.1609/aaai.v24i1.7553.

2 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.

3 Shawn T. Brown, Paola Buitrago, Edward Hanna, Sergiu Sanielevici, Robin Scibek, and Nicholas A. Nystrom. *Bridges-2: A Platform for Rapidly-Evolving and Data Intensive Research*, pages 1–4. Association for Computing Machinery, New York, NY, USA, 2021.

4 Sam Buss and Jakob Nordström. Proof complexity and SAT solving. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability - Second Edition*, volume 336 of *Frontiers in Artificial Intelligence and Applications*, pages 233–350. IOS Press, 2021. doi:10.3233/FAIA200990.

5 Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44(1):3650, 1979. doi:10.2307/2273702.

6 Armin Haken. The intractability of resolution. *Theoretical Computer Science*, 39:297–308, 1985. Third Conference on Foundations of Software Technology and Theoretical Computer Science. doi:10.1016/0304-3975(85)90144-6.

7 Marijn J. H. Heule, Benjamin Kiesl, and Armin Biere. Clausal proofs of mutilated chessboards. In Julia M. Badger and Kristin Yvonne Rozier, editors, *NASA Formal Methods*, pages 204–210, Cham, 2019. Springer International Publishing.

8 Ashlin Iser and Christoph Jabs. Global benchmark database. In Supratik Chakraborty and Jie-Hong Roland Jiang, editors, *27th International Conference on Theory and Applications of Satisfiability Testing, SAT 2024, Pune, India, August 21-24, 2024*, volume 305 of *LIPIcs*, pages 18:1–18:10. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.SAT.2024.18.

9 G. S. Tseitin. *On the Complexity of Derivation in Propositional Calculus*, pages 466–483. Springer Berlin Heidelberg, Berlin, Heidelberg, 1983. doi:10.1007/978-3-642-81955-1_28.