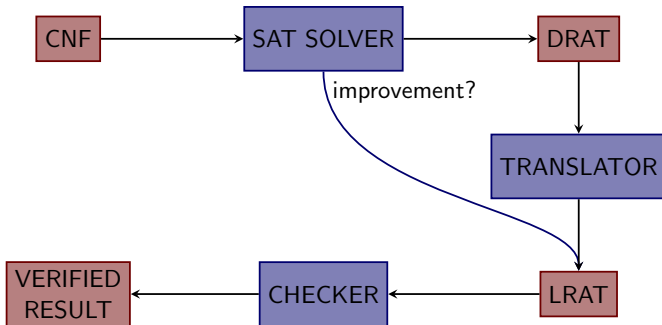


Faster LRAT Checking than Solving with CaDiCaL

Florian Pollitt, Mathias Fleury, Armin Biere

26th International Conference on Theory and Applications of
Satisfiability Testing

Solving And Checking Unsatisfiable Instances



Related Work

- DRAT - The DRAT format and DRAT-trim checker - Marijn J. H. Heule - [Heu16]
- LRAT - Efficient Certified RAT Verification - Luís Cruz-Filipe, Marijn J. H. Heule, Warren A. Hunt, Matt Kaufmann, Peter Schneider-Kamp - [Cru+17]
- FRAT - A Flexible Proof Format for SAT Solver-Elaborator Communication - Seulkee Baek, Mario Carneiro, Marijn J. H. Heule - [BCH22]
- cake_lpr: Verified Propagation Redundancy Checking in CakeML - Yong Kiam Tan, Marijn J. H. Heule, Magnus O. Myreen - [THM21]

Comparing Proof Formats

DIMACS

p cnf 2 4

1 2 0

1 -2 0

-1 2 0

-1 -2 0

DRAT [Heu16]

1 0

d 1 2 0

d 1 -2 0

2 0

d -1 2 0

0

FRAT [BCH22]

o 1 1 2 0

o 2 1 -2 0

o 3 -1 2 0

o 4 -1 -2 0

a 5 1 0 1 1 2 0

d 1 1 2 0

d 2 1 -2 0

a 6 2 0

d 3 -1 2 0

a 7 0 1 5 6 4 0

f 4 -1 -2 0

f 5 1 0

f 6 2 0

f 7 0

LRAT [Cru+17]

5 1 0 1 2 0

5 d 1 2 0

6 2 0 5 3 0

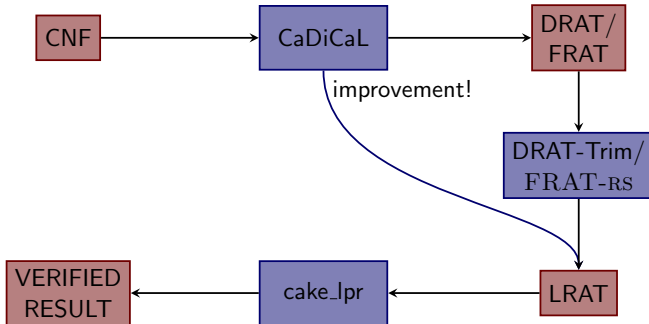
6 d 3 0

7 0 5 6 4 0

Comparing Proof Formats

	verified checker	checker simplicity	solver speed	checker speed	solver simplicity
LRAT [Cru+17]	✓	✓	✓	✓	✗
FRAT [BCH22]	✗	✗	✓	-	-
DRAT [Heu16]	✗	✗	✓	✗	✓

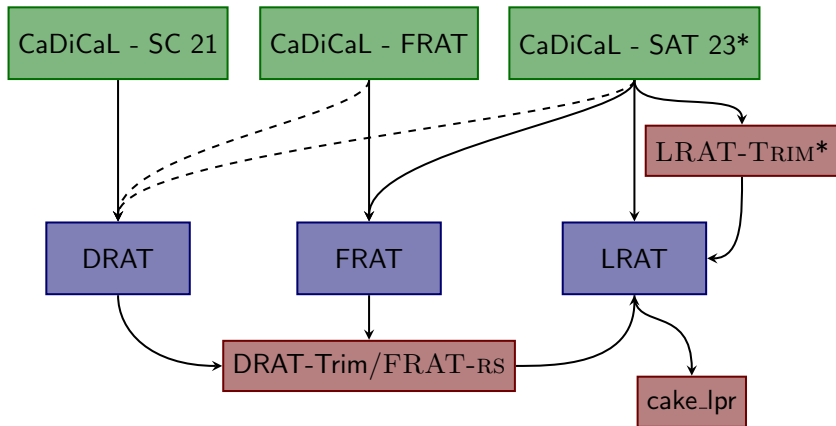
What are we doing again



Implementation

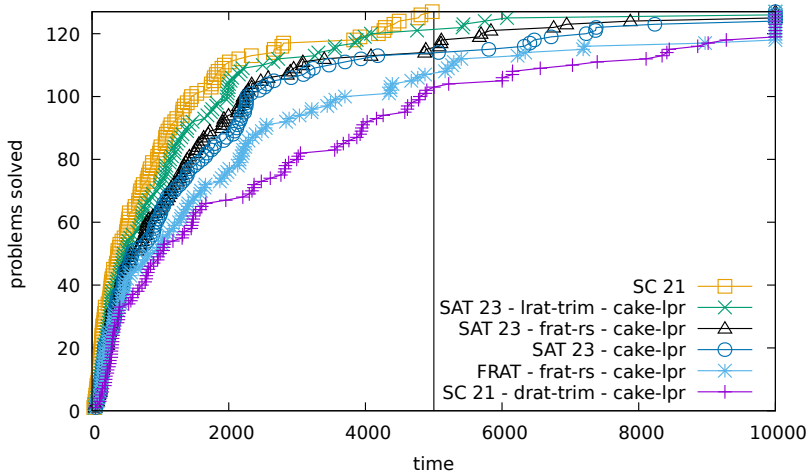
- clause learning: analyze propagated clauses
- inprocessing: implicit propagation/resolution
- equivalent literal substitution: cycles/spanning trees

Experiments

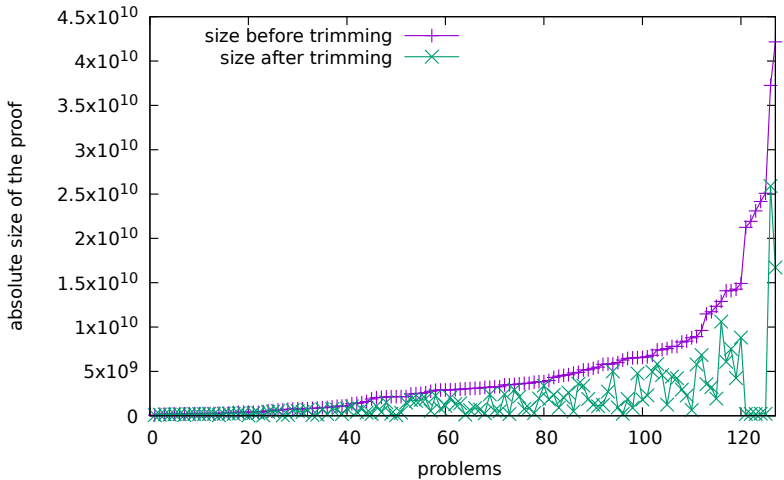


*contributions from this toolpaper

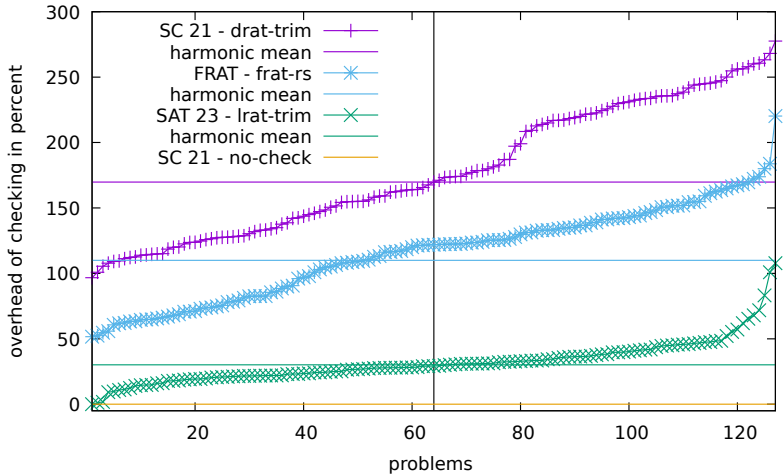
Results



Proof Trimming



Faster Checking Than Solving



Conclusion

- little overhead in solving
- big decrease in checking time
- faster checking then solving
- direct application in: Unsatisfiability Proofs for Distributed Clause-Sharing SAT Solvers - Dawn Michaelson, Dominik Schreiber, Marijn J. H. Heule, Benjamin Kiesl-Reiter, Michael W. Whalen - [Mic+23]
- future work: LRAT in more solvers, i.e., kissat or gimsatul

- [BCH22] Seulkee Baek, Mario Carneiro, and Marijn J. H. Heule. “A Flexible Proof Format for SAT Solver-Elaborator Communication”. In: *Log. Methods Comput. Sci.* 18.2 (2022). DOI: 10.46298/lmcs-18(2:3)2022.
- [Cru+17] Luís Cruz-Filipe, Marijn J. H. Heule, Warren A. Hunt, Matt Kaufmann, and Peter Schneider-Kamp. “Efficient Certified RAT Verification”. In: *Automated Deduction – CADE 26*. Ed. by Leonardo de Moura. Cham: Springer International Publishing, 2017, pp. 220–236. ISBN: 978-3-319-63046-5.

[Heu16] Marijn J. H. Heule. “The DRAT format and DRAT-trim checker”. In: *CoRR* abs/1610.06229 (2016). arXiv: 1610.06229. URL: <http://arxiv.org/abs/1610.06229>.

[Mic+23] Dawn Michaelson, Dominik Schreiber, Marijn J. H. Heule, Benjamin Kiesl-Reiter, and Michael W. Whalen. “Unsatisfiability Proofs for Distributed Clause-Sharing SAT Solvers”. In: *Tools and Algorithms for the Construction and Analysis of Systems - 29th International Conference, TACAS 2023, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2022, Paris, France, April 22-27, 2023, Proceedings, Part I*. Ed. by Sriram Sankaranarayanan and Natasha Sharygina. Vol. 13993. Lecture Notes in Computer Science. Springer, 2023, pp. 348–366. DOI: [10.1007/978-3-031-30823-9_18](https://doi.org/10.1007/978-3-031-30823-9_18). URL: https://doi.org/10.1007/978-3-031-30823-9_5C_18.

[THM21] Yong Kiam Tan, Marijn J. H. Heule, and Magnus O. Myreen. “cake_lpr: Verified Propagation Redundancy Checking in CakeML”. In: *Tools and Algorithms for the Construction and Analysis of Systems - 27th International Conference, TACAS 2021, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2021, Luxembourg City, Luxembourg, March 27 - April 1, 2021, Proceedings, Part II*. Ed. by Jan Friso Groote and Kim Guldstrand Larsen. Vol. 12652. Lecture Notes in Computer Science. Springer, 2021, pp. 223–241. DOI: 10.1007/978-3-030-72013-1_12.