



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

WITH THE POWER
OF KNOWLEDGE
– FOR THE WORLD

Certifying Pareto Optimality in Multi-Objective Maximum Satisfiability

Christoph Jabs¹ Jeremias Berg¹ Bart Bogaerts² Matti Järvisalo¹

¹University of Helsinki, Finland, ²KU Leuven & Vrije Universiteit Brussel, Belgium

August 11th 2025 @ PoS, published at TACAS'25

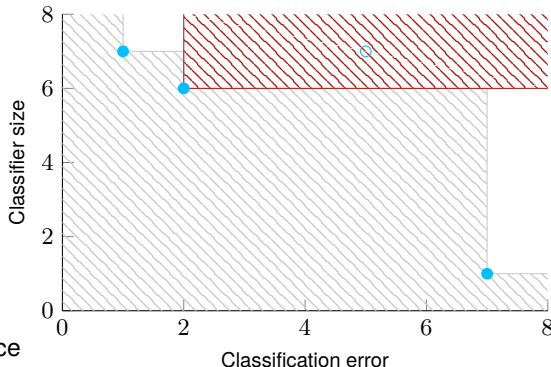


Multi-Objective Optimization

How to deal with conflicting objectives



- ▶ Housing: distance vs. price
- ▶ Aviation: assurance level vs. independence
- ▶ Decision tree: accuracy vs. interpretability



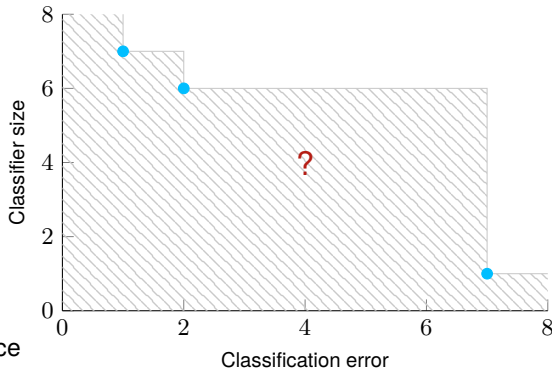


The Problem

Can you trust my solver?



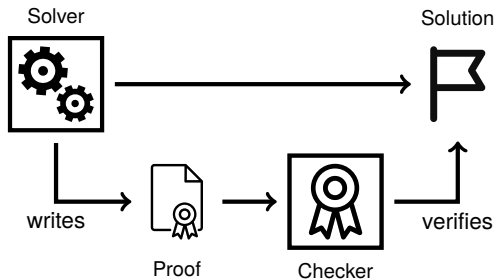
- ▶ Housing: distance vs. price
- ▶ Aviation: assurance level vs. independence
- ▶ Decision tree: accuracy vs. interpretability





Proof Logging

If we can't trust the solver, can we trust the result?



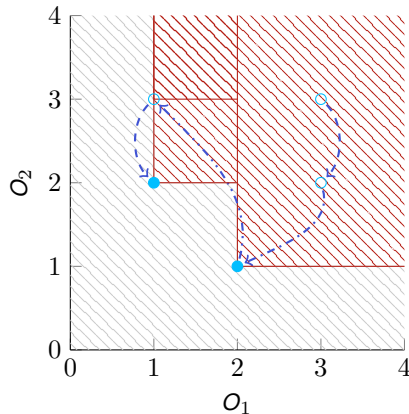
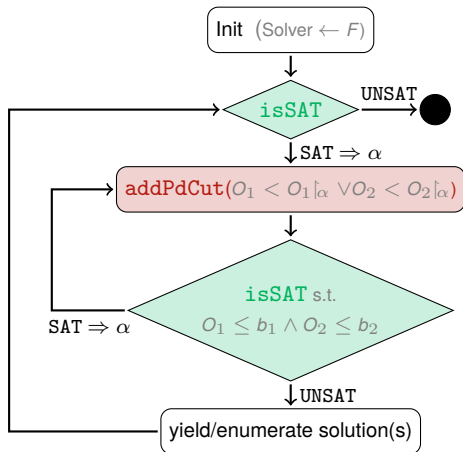
Paradigm	Proof format
SAT	DRAT
MaxSAT	VeriPB
SMT	Various formats
	...
MO-MaxSAT	This talk



The P -Minimal Algorithm

Multi-objective solution-improving search

[Soh et al. 2017]





The VERIPB Proof System

Pseudo-Boolean cutting planes proofs

[Bogaerts et al. 2023]

- ▶ Pseudo-Boolean Constraints
- ▶ Derive constraints by linear combination
- ▶ Redundant constraints
- ▶ Exclude solutions after finding them
- ▶ Single-objective optimization
- ▶ Preorder for expressing preference

minimize:

$$1x_1 + 2x_2 + 1x_3$$

prefer:

$\overline{x}_4$ over x_4



[A] $3x_1 + 2x_2 + 1x_3 \geq 3$

[B] $\overline{x}_1 + x_4 \geq 1$

[A+3B] $2x_2 + 1x_3 + 3x_4 \geq 3$

[red] $1x_1 + 1x_2 + 2x_5 \geq 2$

~~[red]~~ ~~$1x_4 \geq 1$~~

[solx] exclude x_1, x_2, x_3, x_4, x_5

[soli] better than $x_1, x_2, x_3, x_4, \overline{x}_5$



Multi-Objective Proof Setup

What VERIPB can already give us

hard constraints

prefer:

according to
Pareto optimality



reasoning steps from SAT solver
MO-specific reasoning

conclude as UNSAT

Guarantee

For each non-dominated point at least one solution explicitly appears in the proof

Syntactic restrictions

- ▶ First step in proof must load the Pareto order
- ▶ Order must never be changed



Encoding Pareto Dominance as a VERIPB Order

Telling VERIPB about the objectives

Given $O_1, \dots, O_p$

Required VERIPB order:

formula that is true iff α (weakly) dominates β

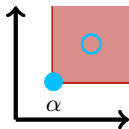
VERIPB Pareto order

$$O_i|_{\alpha} \leq O_i|_{\beta}, \quad \text{for } i = 1, \dots, p$$



Certifying Pareto Dominance Cuts

The building block for all algorithms



$$O_1 < O_1|_{\alpha} \vee O_2 < O_2|_{\alpha}$$

1. Reified objective CNF encoding

$$w_1 \Leftrightarrow O_1 < O_1|_{\alpha}$$

$$w_2 \Leftrightarrow O_2 < O_2|_{\alpha}$$

2. Map each weakly dominated solution to α
(Redundant with α as witness)
3. Log α and thereby exclude it
4. Derive PD cut by combining previous two constraints

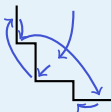


Proof Logging MO-MaxSAT Algorithms

Putting everything together

P-Minimal

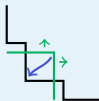
[Soh et al. 2017]



- ▶ SAT solver reasoning
- ▶ CNF objective encodings
- ▶ PD cuts

Lower-Bounding

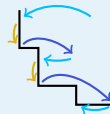
[Cortes et al. 2023]



Upper-bounds irrelevant
→ same as *P*-Minimal

BiOPTSAT

[Jabs et al. 2024]

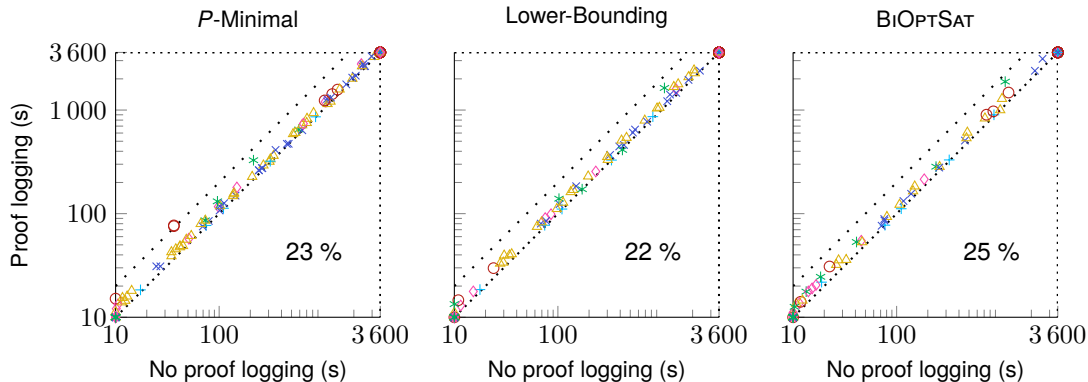


- ▶ Derive lower-bound on first objective
- ▶ Certify PD cut
- ▶ Strengthen PD cut based on known lower-bound



Proof Logging Overhead

How expensive is this





Limitations and Future Work

What next?

- ▶ Efficient checking PBOxIDE
- ▶ Support other orders than Pareto, e.g., lexicmax (requires auxiliary variables)
- ▶ Proofs for enumeration of all Pareto-optimal solutions
- ▶ Proper integration in VERIPB
 - ▶ Automatically generate order from objectives
 - ▶ Distinguish between optimal and sub-optimal solutions



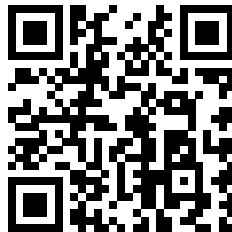
Proof Logging for Multi-Objective MaxSAT

Summary and conclusions

Paper, slides, code, and contact:

christophjabs.info/pos25

- ▶ MO-MaxSAT certificates that all non-dominated points were discovered
- ▶ Proofs in VERIPB format without extension
- ▶ Low overhead for proof logging
- ▶ Open-source implementation





Bibliography I



Bogaerts, Bart et al. (2023). 'Certified Dominance and Symmetry Breaking for Combinatorial Optimisation.'. In: *JAIR* 77, pp. 1539–1589. doi: [10.1613/jair.1.14296](https://doi.org/10.1613/jair.1.14296).



Cortes, João et al. (2023). 'New Core-Guided and Hitting Set Algorithms for Multi-Objective Combinatorial Optimization'. 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 II*. Ed. by Sriram Sankaranarayanan and Natasha Sharygina. Vol. 13994. Lecture Notes in Computer Science. Springer, pp. 55–73. doi: [10.1007/978-3-031-30820-8_7](https://doi.org/10.1007/978-3-031-30820-8_7).



Jabs, Christoph et al. (2024). 'From Single-Objective to Bi-Objective Maximum Satisfiability Solving'. In: *J. Artif. Intell. Res.* 80, pp. 1223–1269. doi: [10.1613/JAIR.1.15333](https://doi.org/10.1613/JAIR.1.15333). URL: <https://doi.org/10.1613/jair.1.15333>.



Soh, Takehide et al. (2017). 'Solving Multiobjective Discrete Optimization Problems with Propositional Minimal Model Generation'. In: *CP*. Ed. by J. Christopher Beck. Vol. 10416. Lecture Notes in Computer Science. Springer, pp. 596–614. doi: [10.1007/978-3-319-66158-2_38](https://doi.org/10.1007/978-3-319-66158-2_38).



Vandesande, Dieter et al. (2022). 'QMaxSATpb: A Certified MaxSAT Solver'. In: *Logic Programming and Nonmonotonic Reasoning - 16th International Conference, LPNMR 2022, Genova, Italy, September 5-9, 2022, Proceedings*, pp. 429–442. doi: [10.1007/978-3-031-15707-3_33](https://doi.org/10.1007/978-3-031-15707-3_33). URL: https://doi.org/10.1007/978-3-031-15707-3_33.

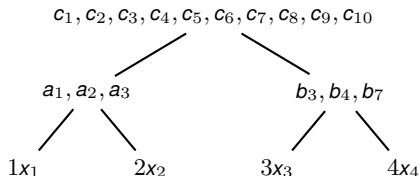


Certifying the Generalized Totalizer Encoding

[Vandesande et al. 2022]

Certified reified pseudo-Boolean CNF encodings

$$O = 1x_1 + 2x_2 + 3x_3 + 4x_4$$



1. Semantic definition of variables

$$a_2 \Leftrightarrow 1x_1 + 2x_2 \geq 2$$

(redundant because fresh variable)

2. Derive clauses from semantic definitions

$$(a_2^{\rightarrow} + b_3^{\rightarrow} + c_5^{\leftarrow} := (\overline{a_2} \vee \overline{b_3} \vee c_5))$$

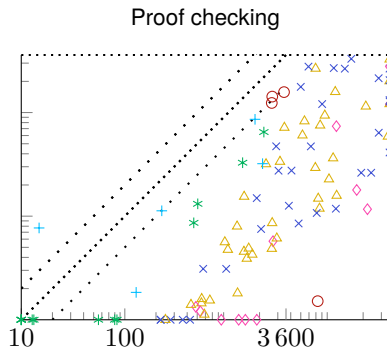
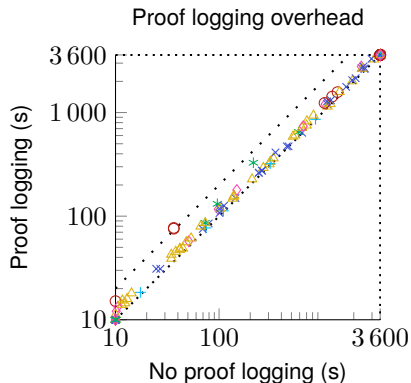
Our contribution

- ▶ This also works for the *generalized* totalizer
- ▶ Adapting process for complex core-boosted encodings



Proof Checking Performance

VERIPB still needs a lot of optimization work



Logging overhead:

23.3%

Checking:

47× longer