Preprocessing in SAT-Based Multi-Objective Combinatorial Optimization

Christoph Jabs Jeremias Berg Hannes Ihalainen Matti Järvisalo

HIIT, Department of Computer Science, University of Helsinki, Finland

August 29th 2023 @ CP'23

Problem Setting: Multi-Objective MaxSAT

SAT / MaxSAT / MO-MaxSAT

Constraints

$$\{(a_1 \lor a_2), (b_1 \lor b_2), (a_1 \lor b_1)\\ (a_1 \lor b_2), (a_2 \lor b_1), (a_2 \lor b_2),\\ (a_3 \lor a_4)\}$$

Objectives

$$O_1 = a_1 + a_2 + a_3 + a_4$$
$$O_2 = b_1 + b_2 + b_3$$

MO-MaxSAT Applications

- Interpretable classifiers (accuracy vs. size)
- Package upgradeability (newly installed vs. out of date vs. removed vs. . . .)
- . . .

Preprocessing

Preprocessing

Applying simplification rules to the instance before solving

- Central in SAT: practice and theory
- Recent promising results in MaxSAT
- Advancing to multi-objective setting

Contributions

Extend SAT and MaxSAT preprocessing to multi-objective (MO-)MaxSAT

Redundancy Notions

- Defining redundant operations
- Analysis of expressiveness

Lift (Max)SAT Techniques

Practical preprocessing for MO-MaxSAT

Empirical Evaluation

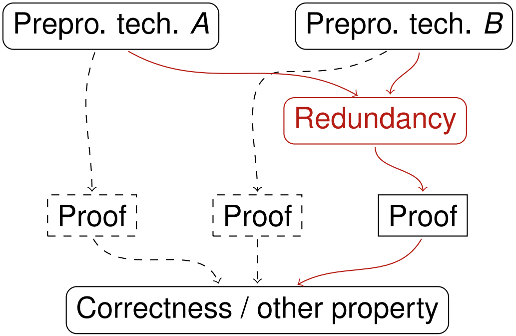
- Open-source preprocessor
- Impact on instance size
- Impact on MO-MaxSAT solvers

Links to all materials:

- `christophjabs.info/cp23`

The Central Idea: Redundancy Notions

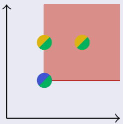
- Formal definition of redundant operations
- Unified proofs of correctness
- Unified analysis of expressiveness
- Established concept in SAT and MaxSAT [Heule et al. 2020; Ihalainen et al. 2022]



Multi-Objective Optimality

Dominating solutions

- ● dominates ●
- ● *weakly* dominates ●



Pareto-Optimality

Every non-dominated solution is Pareto-optimal

Prepro Correctness Criterion

Preserve the non-dominated set
(Pareto-optimal costs)

Redundant Clauses in General — Example

Constraints:

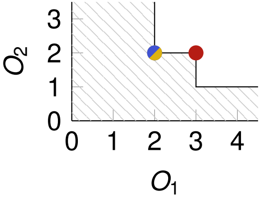
$$\{(a_1 \vee a_2), (b_1 \vee b_2), (a_1 \vee b_1) \\ (a_1 \vee b_2), (a_2 \vee b_1), (a_2 \vee b_2) \\ (a_3 \vee a_4)\}$$

Objectives:

$O_1 = a_1 + a_2 + a_3 + a_4$
$O_2 = b_1 + b_2 + b_3$

Redundant clause:

$C = (\neg a_2 \vee \neg b_2)$



	a_1	a_2	a_3	a_4	b_1	b_2	b_3	C
τ_1	0	1	1	0	1	1	0	0
τ_2	1	0	1	0	1	1	0	1
τ_3	1	1	1	0	1	1	0	0

Redundant Clauses in General

Intuition

Clause that does not change the non-dominated set of an instance

Prepro Correctness Criterion

Sequence of adding/removing redundant clauses preserves the non-dominated set

Reconstructible Clauses — Example

Constraints:

$\{(a_1 \vee a_2), (b_1 \vee b_2), (a_1 \vee b_1)$
$(a_1 \vee b_2), (a_2 \vee b_1), (a_2 \vee b_2)$
$(a_3 \vee a_4)\}$

Objectives:

$O_1 = a_1 + a_2 + a_3 + a_4$
$O_2 = b_1 + b_2 + b_3$

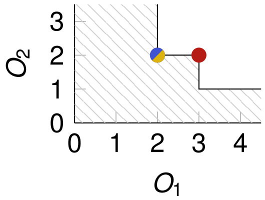
Reconstructible clause:

$C = (\neg a_4)$

Witness:

$\omega = \{a_3, \neg a_4\}$

Reconstruction:
Assign literals in ω



	a_1	a_2	a_3	a_4	b_1	b_2	b_3	C
τ_1	1	0	0	1	1	1	0	0
τ_2	1	0	1	0	1	1	0	1
τ_3	1	0	1	1	1	1	0	0

(Literal-)Reconstructible Clauses

Intuition

Redundant clause for which a "invalidated" solution can be efficiently reconstructed/repaired

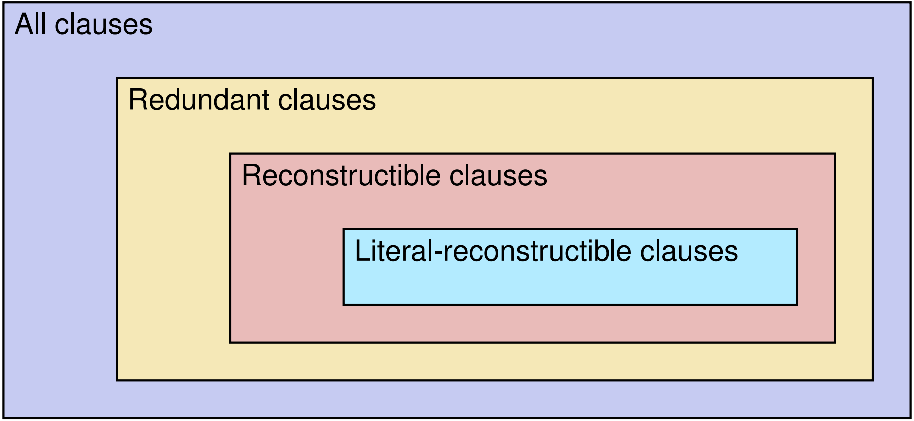
Reconstruction

Assigning all literals in the single witness to true

Literal-Reconstructible

Reconstructible clause with a single literal as witness

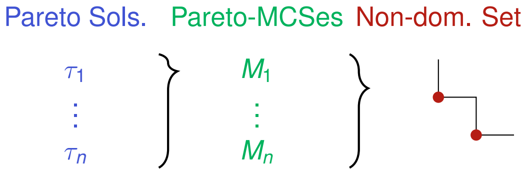
Relationship of Redundancy Notions



Pareto Minimal Correction Sets

[Terra-Neves et al. 2017]

- **Correction set**: relaxing the objectives so that a solution with no additional cost exists
- **Pareto-minimal correction set**: correction set corresponding to Pareto-optimal solution



Prepro Correctness Criterion

At least one Pareto-MCS per non-dominated cost tuple, but not all need to be preserved

(Literal-)Reconstructible Clauses and Pareto-MCSes

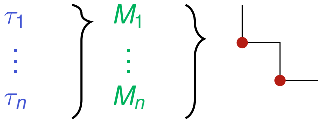
Literal-Reconstructible Clauses

… preserve Pareto-MCSes

Reconstructible Clauses

… can add/remove Pareto-MCSes

- Literal-reconstructible preprocessing techniques from single-objective MaxSAT are directly applicable
- Reconstructible preprocessing techniques can add/remove redundant Pareto-MCSes



MaxPre 2.1

MaxSAT preprocessor MaxPre 2 extended to MO-MaxSAT.
Includes techniques that...

- ... are captured by literal-reconstructible clauses
- ... are captured by reconstructible clauses
- ... alter objectives

Available at `bitbucket.org/coreo-group/maxpre2`.

Evaluation Setup

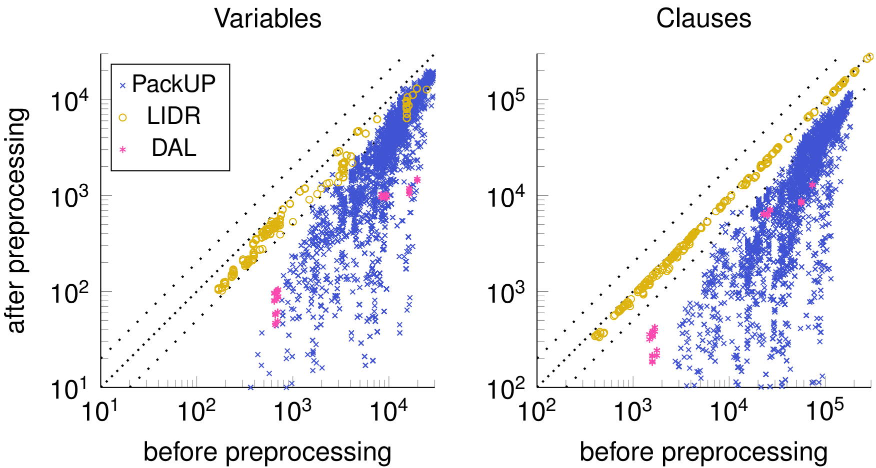
Benchmark domains

- PackUP: package upgradeability
- LIDR: learning interpretable decision rules
- DAL: aircraft development assurance level

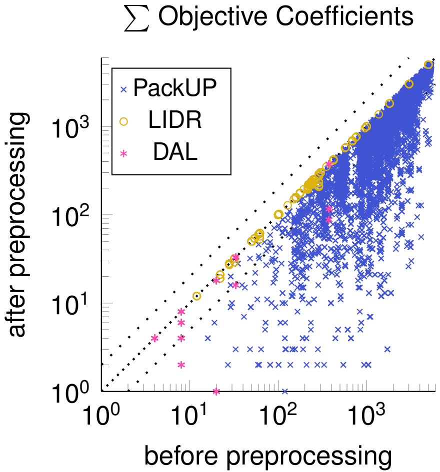
MO-MaxSAT solvers

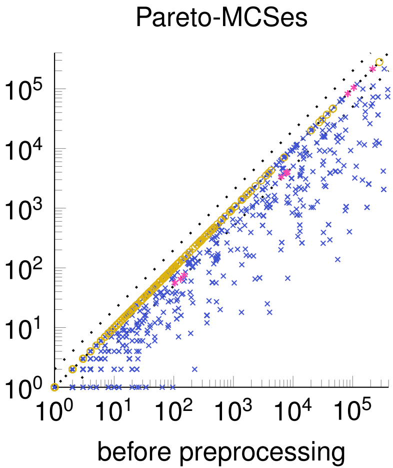
- BiOptSat: bi-objective [Jabs et al. 2022]
- Scuttle: multi-objective (P-minimal) [Soh et al. 2017]
- CLM: multi-objective [Cortes et al. 2023]
- leximaxIST: leximax [Cabral et al. 2022]

Impact on Instance Statistics



Impact on Instance Statistics (cont.)





Impact on Solver Runtime

# inst.	PackUP 3692 (1420*)		LIDR 366		DAL 96	
Solver	Δ #	$\Delta \sum t$	Δ #	$\Delta \sum t$.	Δ #	$\Delta \sum t$
BiOptSat*	+27	−13.7	+3	−18.3	–	–
Scuttle+	+6	−40.4	−1	+7.2	+1	−0.6
CLM+	−5	+14.5	±0	+1.4	−7	+4.7
LeximaxIST†	+71	−166.1	±0	+1.2	±0	+0.9

Δ #: difference in number of solved instances

$\Delta \sum t$: difference in cumulative runtime over solved instances in 10^3 seconds

*bi-objective, +multi-objective, †leximax optimization

Conclusions

Summary

- Preprocessing multi-objective MaxSAT
- Defining redundant operations
- Analysis of expressiveness of redundancy notions
- Lifting preprocessing techniques to MO-MaxSAT
- Empirical evaluation to show potential

Further Research

- MO-MaxSAT-specific prepro techniques
- Prepro techniques focused on reducing objective ranges

Paper, slides, code, poster, and contact: `christophjabs.info/cp23`

Bibliography I

- Cabral, Miguel et al. (2022). 'SAT-Based Leximax Optimisation Algorithms'. In: *25th International Conference on Theory and Applications of Satisfiability Testing, SAT*. Ed. by Kuldeep S. Meel and Ofer Strichman. Vol. 236. LIPIcs. Schloss Dagstuhl—Leibniz-Zentrum für Informatik, 29:1–29:19. DOI: 10.4230/LIPIcs.SAT.2022.29.
- 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*. Ed. by Sriram Sankaranarayanan and Natasha Sharygina. Vol. 13994. LNCS. Springer, pp. 55–73. DOI: 10.1007/978-3-031-30820-8_7.
- Heule, Marijn J. H. et al. (2020). 'Strong Extension-Free Proof Systems'. In: *J. Autom. Reason.* 64.3, pp. 533–554. DOI: 10.1007/s10817-019-09516-0.
- Ihalainen, Hannes et al. (2022). 'Clause Redundancy and Preprocessing in Maximum Satisfiability'. In: *Automated Reasoning—11th International Joint Conference, IJCAR*. Ed. by Jasmin Blanchette et al. Vol. 13385. LNCS. Springer, pp. 75–94. DOI: 10.1007/978-3-031-10769-6_6.
- Jabs, Christoph et al. (2022). 'MaxSAT-Based Bi-Objective Boolean Optimization'. In: *25th International Conference on Theory and Applications of Satisfiability Testing, SAT*. Ed. by Kuldeep S. Meel and Ofer Strichman. Vol. 236. LIPIcs. Schloss Dagstuhl—Leibniz-Zentrum für Informatik, 12:1–12:23. DOI: 10.4230/LIPIcs.SAT.2022.12.

Bibliography II

Soh, Takehide et al. (2017). 'Solving Multiobjective Discrete Optimization Problems with Propositional Minimal Model Generation'. In: *Principles and Practice of Constraint Programming—23rd International Conference, CP*. Ed. by J. Christopher Beck. Vol. 10416. LNCS. Springer, pp. 596–614. DOI: 10.1007/978-3-319-66158-2_38.

Terra-Neves, Miguel et al. (2017). 'Introducing Pareto Minimal Correction Subsets'. In: *Theory and Applications of Satisfiability Testing—20th International Conference, SAT*. Ed. by Serge Gaspers and Toby Walsh. Vol. 10491. LNCS. Springer, pp. 195–211. DOI: 10.1007/978-3-319-66263-3_13.

Literal-Reconstructible Clauses — Example

Constraints:

$$\{(a_1 \lor a_2), (b_1 \lor b_2), (a_1 \lor b_1) \\ (a_1 \lor b_2), (a_2 \lor b_1), (a_2 \lor b_2) \\ (a_3 \lor a_4)\}$$

Objectives:

$O_1 = a_1 + a_2 + a_3 + a_4$
$O_2 = b_1 + b_2 + b_3$

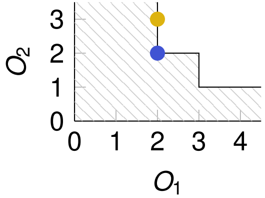
Literal-reconstructible clause:

$C = (\neg b_3)$

Reconstruct on:

$l = \neg b_3$

Reconstruction:
Assign l



	a_1	a_2	a_3	a_4	b_1	b_2	b_3	C
τ_1	1	0	1	0	1	1	1	0
τ_2	1	0	1	0	1	1	0	1

Impact on Solver Runtime — Summary Statistics — All Solvers

	PackUP		LIDR		DAL	
# inst.	3692 (1420)		366		96	
Solver	Δ #	$\Delta \sum t$	Δ #	$\Delta \sum t.$	Δ #	$\Delta \sum t$
BIOPTSAT (bi-objective optimization)						
LSU	$+27$	-13.7	$+3$	-18.3	–	–
CG	$+5$	-6.4	± 0	-5.5	–	–
Hybrid	$+5$	-13.6	± 0	-4.9	–	–
SCUTTLE	$+6$	-40.4	-1	$+7.2$	$+1$	-0.6
CLM						
CG	-5	$+14.5$	± 0	$+1.4$	-7	$+4.7$
IHS	-19	-68.7	-19	-3.3	-7	-0.1
LEXIMAXIST (leximax optimization)						
LSU	$+71$	-166.1	± 0	$+1.2$	± 0	$+0.9$
CG	$+3$	-0.7	$+2$	-5.7	$+1$	-3.2

Δ #: difference in number of solved instances

$\Delta \sum t$: difference in cumulative runtime over solved instances in 10^3 seconds

Impact on Solver Runtime — Summary Statistics — All Details

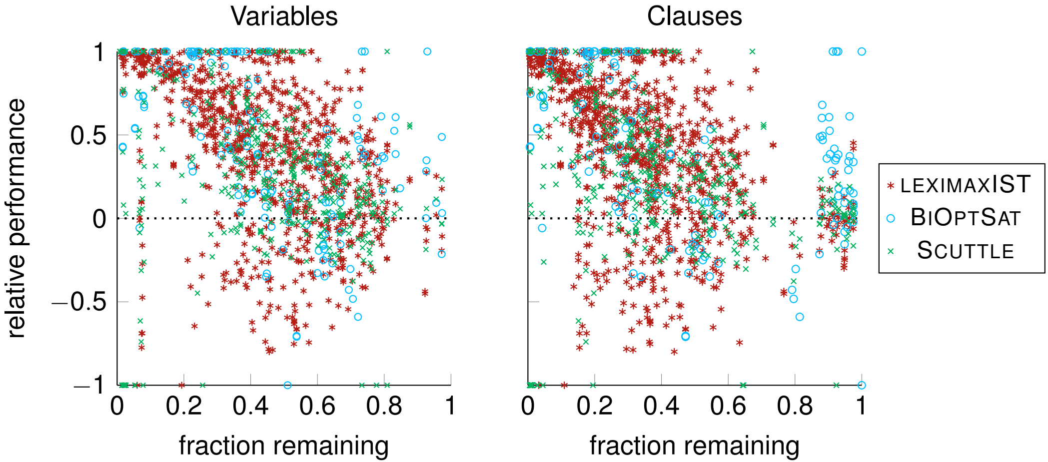
Solver	Prepro.	PackUP #	PackUP uniq.	PackUP $\sum t$	LIDR #	LIDR uniq.	LIDR $\sum t$	DAL #	DAL uniq.	DAL $\sum t$
BiOptSat (LSU)	no	1134	0	61.4	220	1	52.4	–	–	–
(bi-opt.)	yes	**1161**	27	**47.7**	**223**	4	**34.1**	–	–	–
BiOptSat (CG)	no	1154	1	40.9	222	1	43.9	–	–	–
(bi-opt.)	yes	**1159**	6	**34.5**	222	1	**38.4**	–	–	–
BiOptSat (Hybrid)	no	1154	1	46.6	222	0	40.4	–	–	–
(bi-opt.)	yes	**1159**	6	**33.0**	222	0	**35.5**	–	–	–
Scuttle	no	1772	40	284.5	**219**	1	51.3	66	0	5.9
(multi-opt.)	yes	**1778**	46	**244.1**	218	0	44.1	**67**	1	**5.3**
CLM (CG)	no	**1593**	88	**301.3**	206	2	**48.0**	**60**	7	**8.1**
(multi-opt.)	yes	1588	83	315.8	206	2	49.4	53	0	12.8
CLM (IHS)	no	**1301**	91	258.5	**134**	19	26.8	**48**	7	0.3
(multi-opt.)	yes	1282	72	189.8	115	0	23.5	41	0	0.2
LeximaxIST (LSU)	no	2276	2	434.6	224	0	**28.4**	72	0	**4.3**
(leximax-opt.)	yes	**2347**	73	**268.5**	224	0	29.6	72	0	5.2
LeximaxIST (CG)	no	2450	13	140.7	**220**	2	43.7	72	1	12.9
(leximax-opt.)	yes	**2453**	16	**140.0**	218	0	38.0	**73**	2	**9.7**

#: number of solved instances

uniq.: number of uniquely solved instances

$\sum t$: cumulative runtime over solved instances in 10^3 seconds

Impact on Solver Runtime — Per Instance



Impact on Solver Runtime — Per Instance (cont.)