

LIFESPAN OF SAT TECHNIQUES

Mathias Fleury and **Daniela Kaufmann**

University of Freiburg and TU Wien

Pragmatics of SAT

Alghero, Italy

July 4, 2023

✉ `daniela.kaufmann@tuwien.ac.at`

Idea of our study

Effect of enabling and disabling features in a SAT solver.

Idea of our study

Effect of enabling and disabling features in a SAT solver.

■ **1 Solver:** CaDiCaL

■ **4 Techniques:**

- ☐ Blocked Clause Elimination,
- ☐ Variable elimination with Bound Increase,
- ☐ On-the-fly self subsumption,
- ☐ Vivification

■ **4835 Benchmarks:** SAT Competition from 2009-2022, split by year.

CaDiCaL

Armin Biere, Katalin Fazekas, Mathias Fleury, & Maximillian Heisinger (2020).

CaDiCaL, Kissat, Paracooba, Plingeling and Treengeling Entering the SAT Competition 2020.

In Proc. of SAT Competition 2020 – Solver and Benchmark Descriptions (pp. 51–53).

- We do not alter the default setting of the other features of CaDiCaL
- OTFS is not contained in CaDiCaL and was implemented by us

Techniques under investigation

1. Blocked clause elimination (BCE)

- Removes blocked clauses $l \vee C$ from problems
- $l \vee C$ is blocked when all resolvents on l are tautological.

Usage level: low

not activated by default in CaDiCaL,

not implemented in Minisat, Glucose, MapleSAT.

Techniques under investigation

2. Variable elimination with Bound Increase (BVE^+)

- Simplifies formula by resolving all clauses on a given literal
- Elimination may slowly increase the number of clauses
- In CaDiCaL elimination of a literal may generate up to 16 new clauses

Usage level: high

implemented in CaDiCaL, Kissat and recent MapleSAT solvers

Techniques under investigation

3. On-the-fly Self Subsumption (OTFS)

- Strengthens clauses during conflict analysis
- Conflict clause $L \vee C$, reason clause $\neg L \vee D \rightarrow C \vee D$
- OTFS checks whether $C \vee D$ is included in C . If yes, L is removed from $L \vee C$.

Usage level: medium

implemented in Kissat,

not implemented in CaDiCaL, Minisat, Glucose, MapleSAT

Techniques under investigation

4. Vivification

- Eliminates redundant literals in clauses
- Takes a clause, negates the literals and decides them one-by-one
- Allows to learn new clauses that strengthen the original clause
- Solvers have different strategies to pick clauses for vivification

Usage level: high

implemented in CaDiCaL, Kissat and recent MapleSAT solvers

Experimental Setup

1. Setting **Base**

- base
 - ☐ BCE
 - ☐ BVE⁺
 - ☐ OTFS
 - ☐ Vivification
- base+BCE
- base+BVE⁺
- base+OTFS
- base+vivify

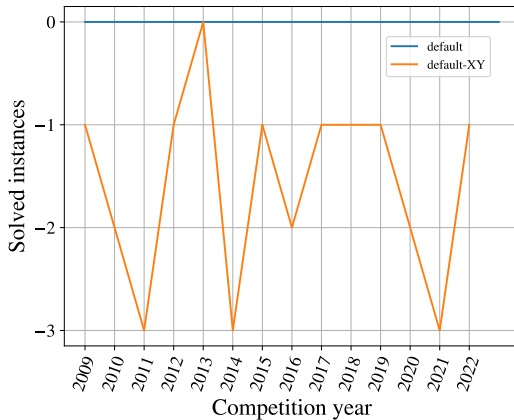
2. Setting **Default**

- default
 - ☐ BCE
 - ☐ BVE⁺
 - ☐ OTFS
 - ☐ Vivification
- default-BVE⁺
- default-OTFS
- default-vivify

3. Setting **Everything**

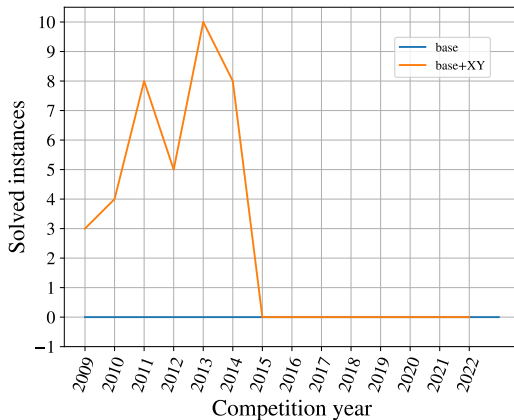
- everything
 - ☐ BCE
 - ☐ BVE⁺
 - ☐ OTFS
 - ☐ Vivification
- everything-BCE
- everything-BVE⁺
- everything-OTFS
- everything-vivify

Hypothesis 1 - Disabling features decreases number of solved instances



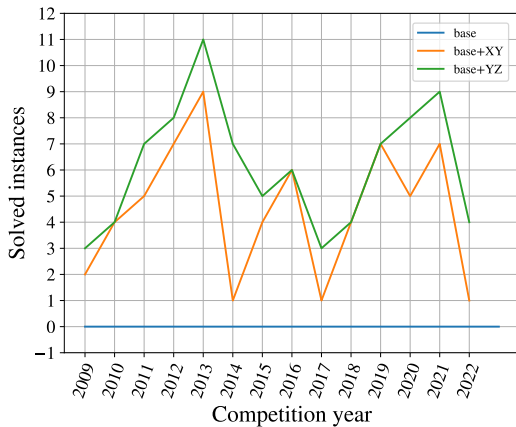
In default and everything setting turning off a feature leads to fewer solved instances.

Hypothesis 2 - Life span of techniques is limited



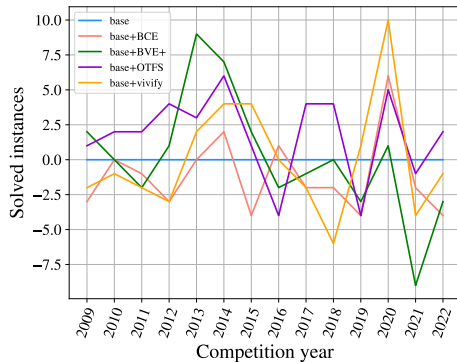
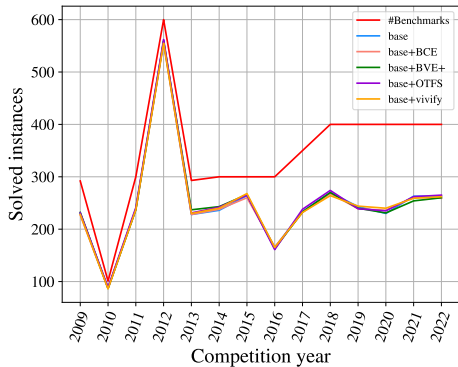
XY is published in 2014, based on success on benchmarks from 2013. XY becomes mainstream in following year → benchmarks with XY-feature are not considered anymore in the competition.

Hypothesis 3 - Features simulate each other

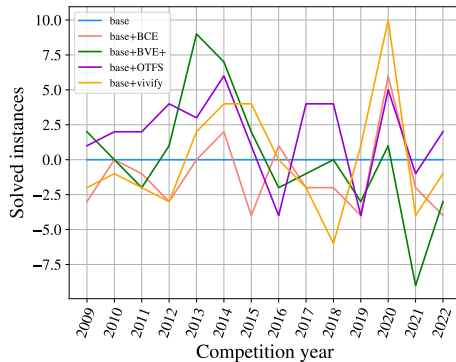
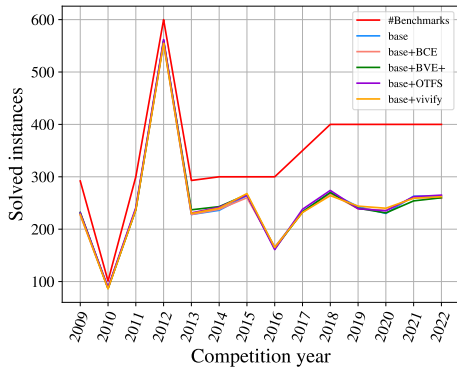


Feature YZ subsumes feature XY.

Base - Summary

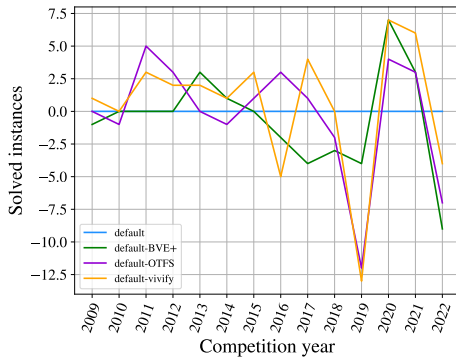
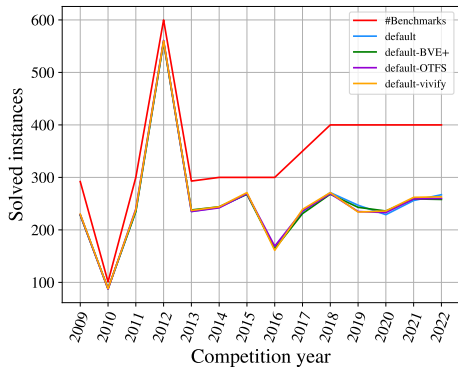


Base - Summary

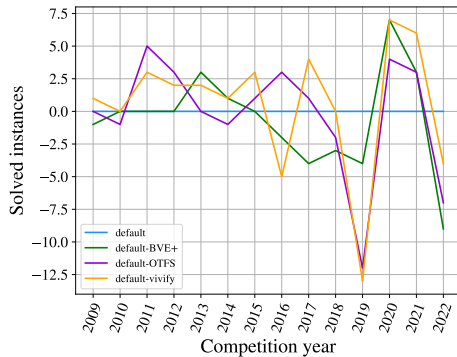
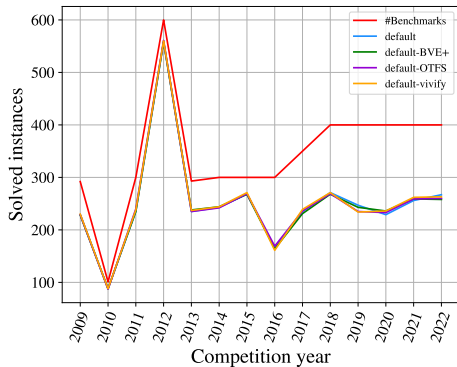


Life span is not limited → Hypothesis 2 is not true.

Default - Summary

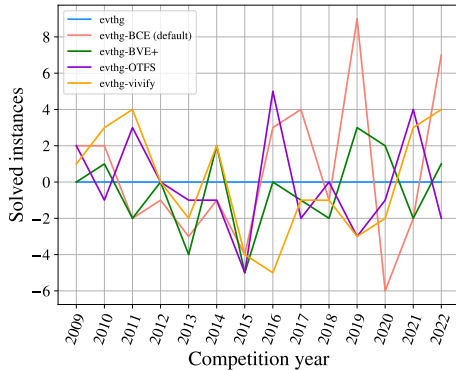
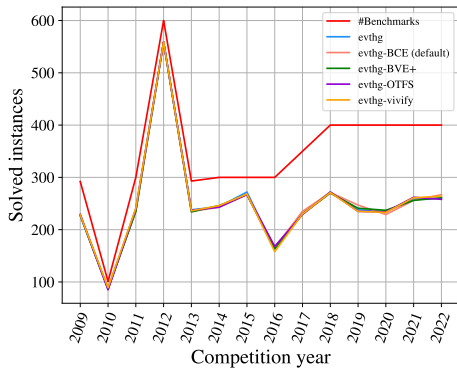


Default - Summary

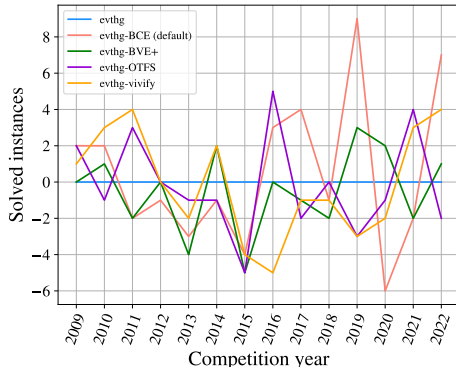
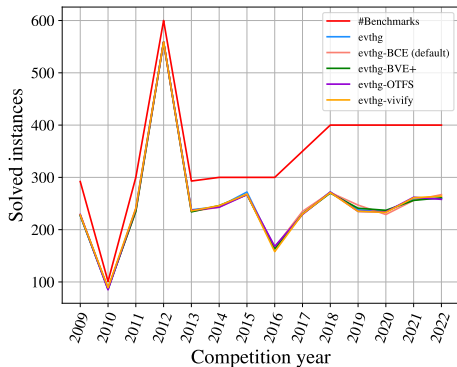


We do not see a complete drop in solved benchmarks nor a subsumption.

Everything - Summary



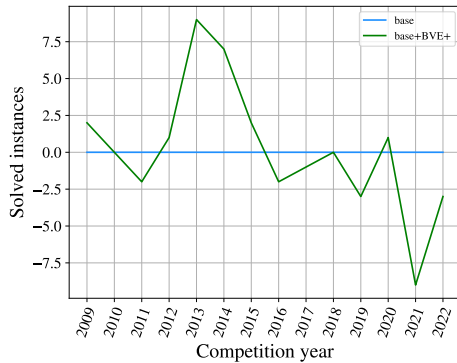
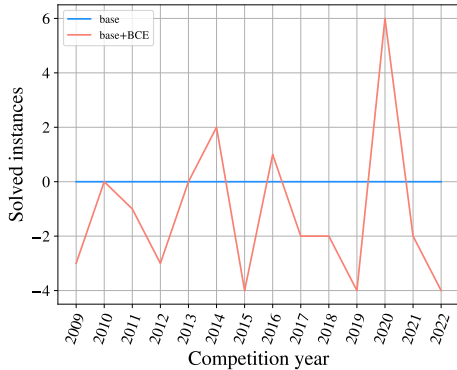
Everything - Summary



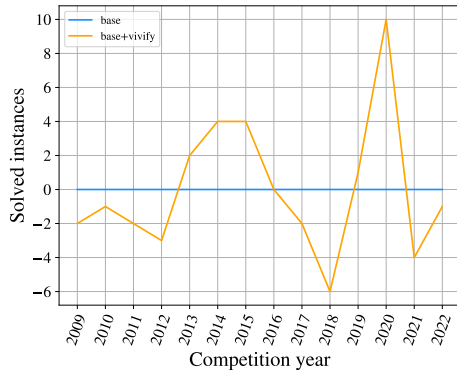
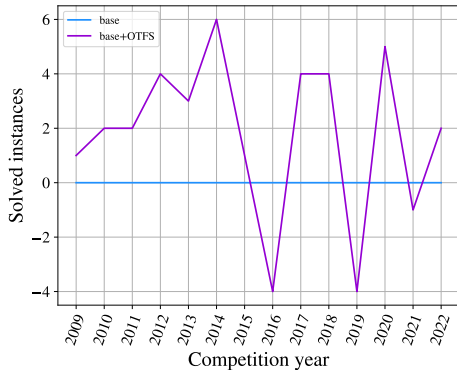
We do not see a complete drop in solved benchmarks nor a subsumption →
Hypotheses 1 and 3 are not true.

BASE

Base - BCE & BVE⁺



Base - OTFS & vivify



Base - CDF

Base vs. Base+BCE

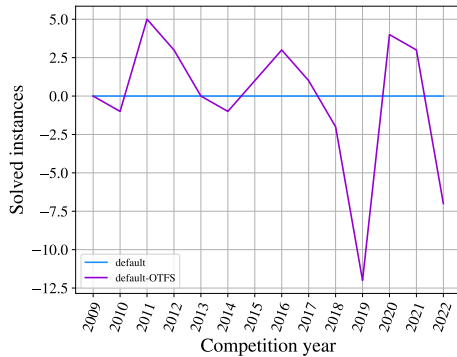
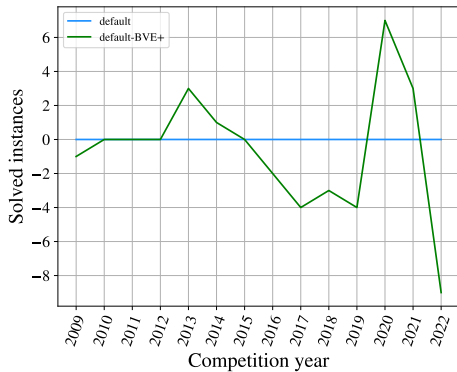
Base vs. Base+BVE⁺

Base vs. Base+OTFS

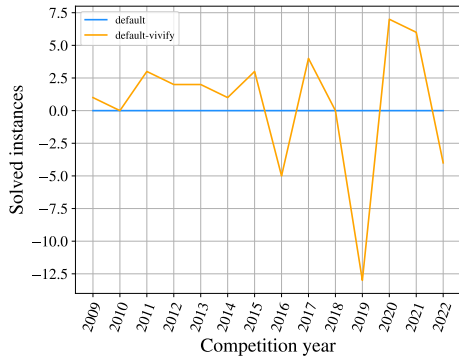
Base vs. Base+vivify

DEFAULT

Default - BVE⁺ & OTFS



Default – vivify



Default - CDF

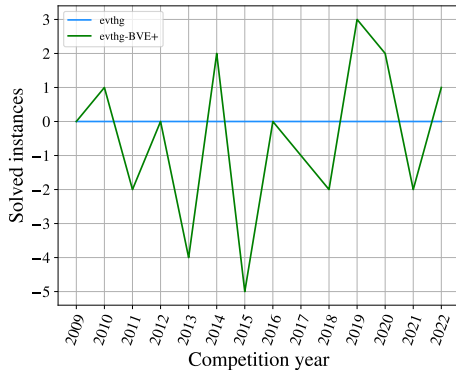
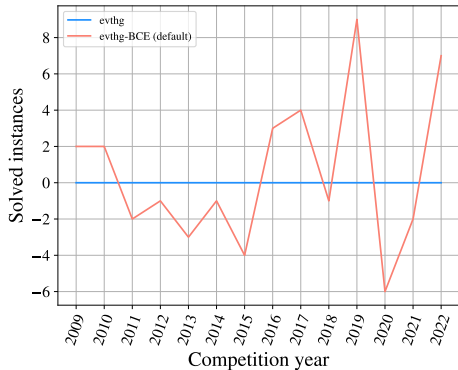
Default vs. Default-BVE⁺

Default vs. Default-OTFS

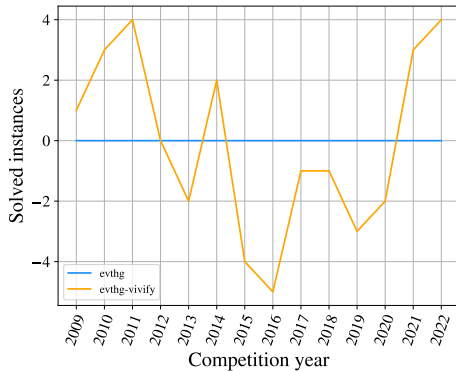
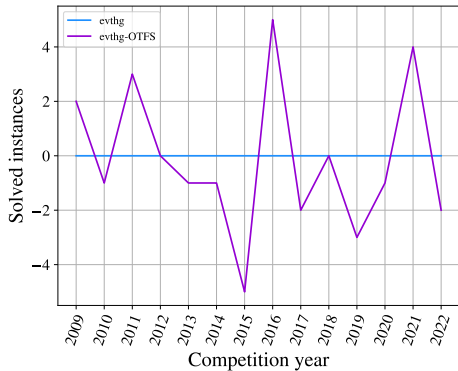
Default vs. Default-vivify

EVERYTHING

Everything - BCE & BVE⁺



Everything - OTFS & vivify



Everything - CDF

Everything vs. Everything-BCE

Everything vs. Everything-BVE⁺

Everything vs. Everything-OTFS

Everything vs. Everything-vivify

ANALYSIS

Blocked Clause Elimination

Bounded variable elimination

On-the-fly self subsumption

Vivification

Conclusion & Outlook

- Disabling features decreases number of solved instances → not confirmed
- Life span of techniques is limited → not confirmed
- Features simulate each other → not confirmed

Conclusion & Outlook

- Disabling features decreases number of solved instances → not confirmed
- Life span of techniques is limited → not confirmed
- Features simulate each other → not confirmed

Current evaluation does not consider . . .

- . . . that benchmarks have been used in more than one year.
- . . . the families of the benchmarks.
- . . . a full causal analysis nor Monte-Carlo grid method.

Conclusion & Outlook

- Disabling features decreases number of solved instances → not confirmed
- Life span of techniques is limited → not confirmed
- Features simulate each other → not confirmed

Current evaluation does not consider ...

- ... that benchmarks have been used in more than one year.
- ... the families of the benchmarks.
- ... a full causal analysis nor Monte-Carlo grid method.

What would be an interesting evaluation?

LIFESPAN OF SAT TECHNIQUES

Mathias Fleury and **Daniela Kaufmann**

University of Freiburg and TU Wien

Pragmatics of SAT

Alghero, Italy

July 4, 2023

✉ `daniela.kaufmann@tuwien.ac.at`