SATISFIABILITY-PRESERVING REASONING IN SOFTWARE VERIFICATION

ADRIÁN REBOLA-PARDO TU WIEN

TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

Microsoft

logics Logical Methods in Computer Science

Supported by FWF W1255-N23 and Microsoft Research through its PhD Programme

MOTIVATION

Mission Software verification

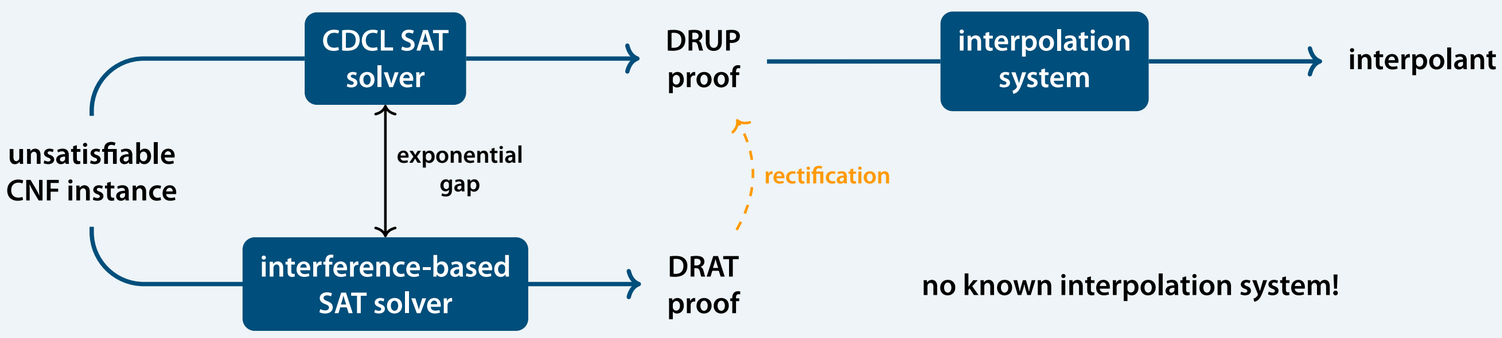
Method Interpolation-based model checking

Interpolants Special formulas extracted from proofs

Problem State-of-the-art proofs do not admit interpolation

Goal Interpolants from satisfiability-preserving proofs

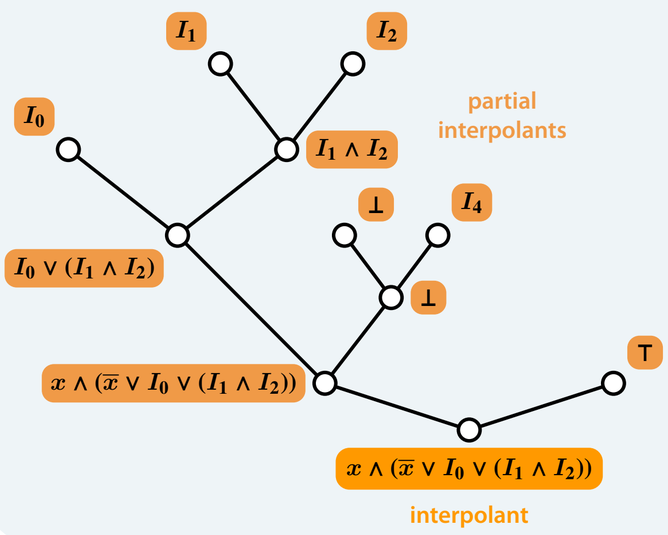
Interpolants **cannot be generated** from proofs obtained from state-of-the-art SAT solvers

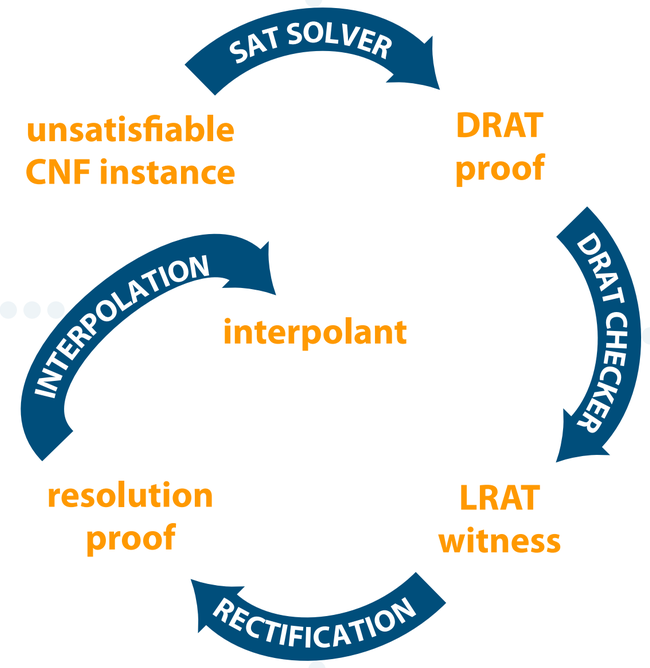


Idea *rectify* DRAT proofs into resolution-like proofs

An off-the-shelf **interpolation system** is used

Partial interpolants are recursively generated





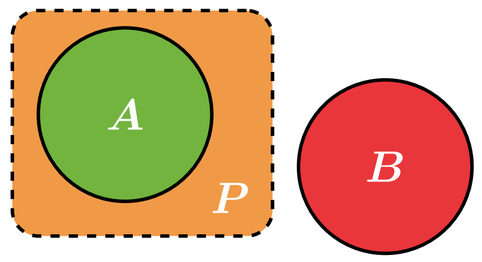
INTERPOLANTS

Use for model checking overapproximating set of reachable states

$A \wedge B$ unsatisfiable, P interpolant

$A \models P \models \neg B$

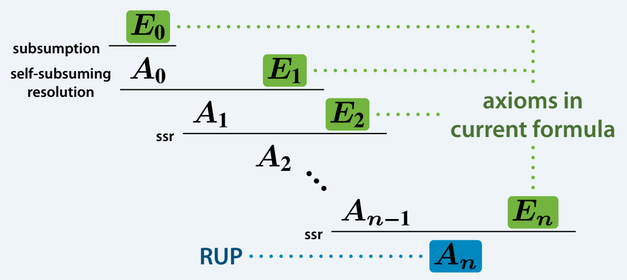
$\mathrm{var}(P) \subseteq \mathrm{var}(A) \cap \mathrm{var}(B)$



DRAT proofs contain **satisfiability-preserving** inferences

Clause deletion

RUP clause introduction



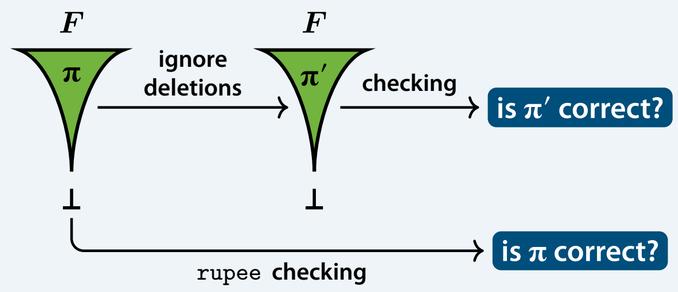
RAT clause introduction

C is a RAT in F upon some literal $l \in C$

$\Updownarrow$

$C \otimes_l D$ is a RUP in F, for all clauses $D \in F$

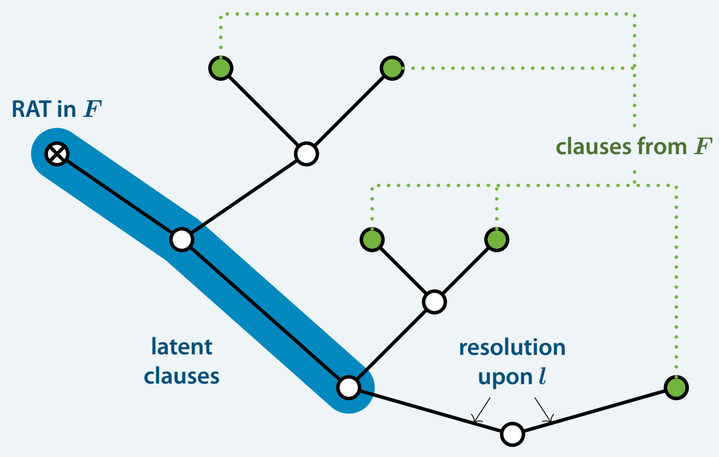
Clause deletion is a source of **incompleteness** in DRAT proof checkers



Output LRAT witness

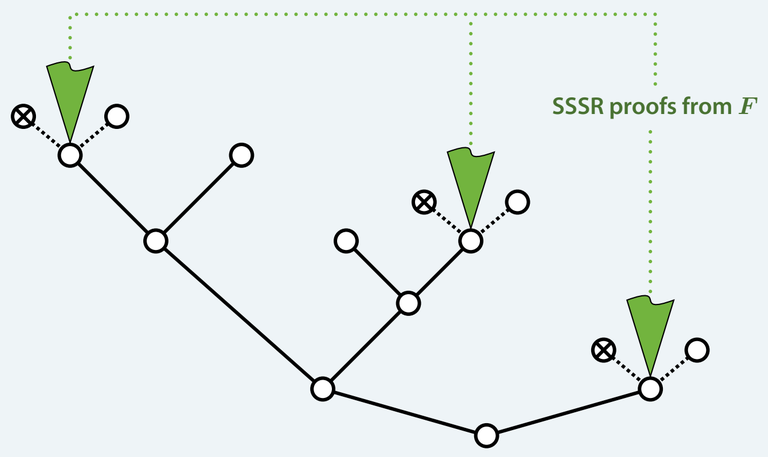
Latent clauses do not follow from premises

Theorem when literal l is eliminated by resolution, the resolvent *is* a consequence of F.

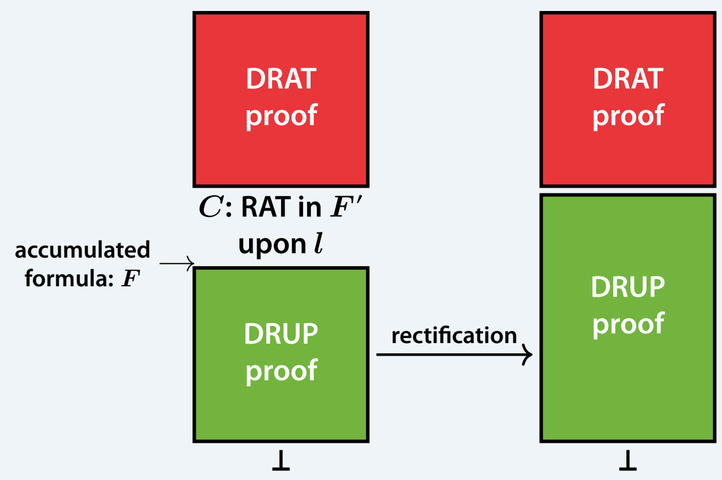


Cut elimination lifts the resolvent

Resolvents of the RAT with clauses in F are **RUPs**, so there exist **SSSR proofs** for them.



Repeat to eliminate all RATs



DRAT SEMANTICS

DRAT models **without loss of generality**

C is a RAT in F upon l

if $I \models C$, then $I \models F \wedge C$

if $I \not\models C$, then $I + l \models F \wedge C$

FUTURE WORK

Interpolants Generation from PR proofs

Model checking Mutation semantics

Bit-vectors Word-level interpolation

SAT solving Clause minimization

REFERENCES

T. Philipp, A. Rebola-Pardo, *DRAT Proofs for XOR Reasoning*. JELIA 2016

W. Forkel, T. Philipp, A. Rebola-Pardo, E. Werner *Fuzzing and Verifying RAT Refutations with Deletion Information*. FLAIRS 2017

T. Philipp, A. Rebola-Pardo, *Towards a Semantics of Unsatisfiability Proofs with Inprocessing*. LPAR 2017

L. Cruz-Filipe, A. Rebola-Pardo, A. Biere, *Complete and Efficient DRAT Proof Checking*. Submitted to CPP 2017

A. Rebola-Pardo, G. Weissenbacher, *Converting DRAT proofs to RUP proofs*. In preparation