Local proof transformations for flexible interpolation and proof reduction

N. Sharygina

Formal Verification and Security Group

University of Lugano

June 21, 2011

Outline

Outline

Outline

Outline

Outline

Background

Formal Verification in Lugano, Switzerland

- Program Verification

Background

Formal Verification in Lugano, Switzerland

- Program Verification
  - Model checking code (LoopFrog, Synergy, SatAbs (with Oxford), FunFrog), ANSI-C
  - Efficient decision procedures as computational engines of verification (OpenSMT)

Background

Formal Verification in Lugano, Switzerland

- Program Verification
  - Model checking code (LoopFrog, Synergy, SatAbs (with Oxford), FunFrog), ANSI-C
  - Efficient decision procedures as computational engines of verification (OpenSMT)
- Abstractions

Background

Formal Verification in Lugano, Switzerland

- Program Verification
  - Model checking code (LoopFrog, Synergy, SatAbs (with Oxford), FunFrog), ANSI-C
  - Efficient decision procedures as computational engines of verification (OpenSMT)

- Abstractions
  - Program Summarization [ATVA'08], [ASE'09]
    - Avoids fix-point computation by constructing symbolic abstract transformers instead
    - Performs sound over-approximation of (unbounded) loops
    - Precision is tuned by selection of abstract domains
    - Exploits efficiency of SAT/SMT solvers

Background

Formal Verification in Lugano, Switzerland

- Program Termination [CAV'10, TACAS'11]
  - Integration of Loop Summarization with Termination Analysis
  - Compositional Transition Invariants avoid all paths computation of termination checks
  - Simple abstract domains are used for termination checks

- Program Termination [CAV'10, TACAS'11]
  - Integration of Loop Summarization with Termination Analysis
  - Compositional Transition Invariants avoid all paths computation of termination checks
  - Simple abstract domains are used for termination checks
- Synergy of Abstractions [STTT'10]
  - Interleaves precise and over-approximated abstractions
  - Reduces CEGAR iterations
  - Removes multiple counterexamples within a single refinement step
  - Localizes precise abstraction/refinement to relevant parts of the program

Background

Formal Verification in Lugano, Switzerland

- Model checking mobile code [IFM'08], [JFAC'10]
  - Specification language for security policies
  - Formalization of mobile code distribution net
  - Location-specific abstractions and model checking of security policies

Background

Formal Verification in Lugano, Switzerland

- Model checking mobile code [IFM'08], [JFAC'10]
  - Specification language for security policies
  - Formalization of mobile code distribution net
  - Location-specific abstractions and model checking of security policies
- Boolean and Theory Reasoning (SMT)
  - Procedure for bit-vector extraction and concatenation [ICCAD'09]
    - Reduces formulae to the theory of equality to avoid, when possible, expensive reduction to SAT

Background

Formal Verification in Lugano, Switzerland

- Model checking mobile code [IFM'08], [JFAC'10]
  - Specification language for security policies
  - Formalization of mobile code distribution net
  - Location-specific abstractions and model checking of security policies
- Boolean and Theory Reasoning (SMT)
  - Procedure for bit-vector extraction and concatenation [ICCAD'09]
    - Reduces formulae to the theory of equality to avoid, when possible, expensive reduction to SAT
  - Generation of explanations in theory propagation [MEMOCODE'10]
    - Computes explanations on demand by reusing the consistency check algorithm for a generic theory T.

Background

Formal Verification in Lugano, Switzerland

- Boolean and Theory Reasoning (SMT)
  - Generation of interpolants (for QF EUF, RDL)
  - Proof manipulation for interpolation [ICCAD'10]
  - Proof reduction [HVC'10]

- Boolean and Theory Reasoning (SMT)
  - Generation of interpolants (for QF EUF, RDL)
  - Proof manipulation for interpolation [ICCAD'10]
  - Proof reduction [HVC'10]
  - Solver, *OpenSMT*, combines MiniSAT2 SAT-Solver with state-of-the-art decision procedures for QF EUF, LRA, BV, RDL, IDL
    - *Extensible*: the SAT-to-theory interface facilites design and plug-in of new decision procedures
    - *Incremental*: suitable for incremental verification
    - *Open-source*: available under GPL license
    - *Efficient*: currently the fastest open-source SMT Solver for QF UF, IDL, RDL, LRA according to SMT-Comp'10.

- Boolean and Theory Reasoning (SMT)
  - Generation of interpolants (for QF EUF, RDL)
  - Proof manipulation for interpolation [S.F. Rollini, R. Bruttomesso, N. Sharygina, A. Tsitovich, ICCAD'10]
  - Resolution proof reduction [S.F. Rollini, R. Bruttomesso, N. Sharygina, HVC'10]

Outline

1. Background
2. Motivation and Related Work
3. Contribution
   - Proof Transformation for Interpolation and Reduction
4. Summary and Future Work

- Resolution proofs find application in several ambits

- Resolution proofs find application in several ambits
    - Interpolation-based model checking
    - Abstraction techniques
    - Unsatisfiable core extraction in SAT/SMT
    - Automatic theorem proving

- Resolution proofs find application in several ambits
  - Interpolation-based model checking
  - Abstraction techniques
  - Unsatisfiable core extraction in SAT/SMT
  - Automatic theorem proving

- Problems

- Resolution proofs find application in several ambits
  - Interpolation-based model checking
  - Abstraction techniques
  - Unsatisfiable core extraction in SAT/SMT
  - Automatic theorem proving

- Problems
  - Clean structure of proofs is required for interpolation generation

Proof Transformation and Reduction

Motivation

- Resolution proofs find application in several ambits
  - Interpolation-based model checking
  - Abstraction techniques
  - Unsatisfiable core extraction in SAT/SMT
  - Automatic theorem proving
- Problems
  - Clean structure of proofs is required for interpolation generation
  - Size affects efficiency
  - Size can be exponential w.r.t. input formula

Notation

Interpolation

- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]

Notation
Interpolation

- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]
  - $A \Rightarrow I$, $I \wedge B$ unsatisfiable

Notation

- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]
  - $A \Rightarrow I$, $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B

Notation

Interpolation

- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]
  - $A \Rightarrow I$, $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B
  - I as over-approximation A conflicting with B

- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]
  - $A \Rightarrow I$, $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B
  - I as over-approximation A conflicting with B
- Example

- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]
  - $A \Rightarrow I$, $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B
  - I as over-approximation A conflicting with B

- Example
  - $A \triangleq (\overline{p} \vee \overline{q}) \wedge (p \vee \overline{q})$ $\qquad B \triangleq (q \vee \overline{r}) \wedge (q \vee r)$

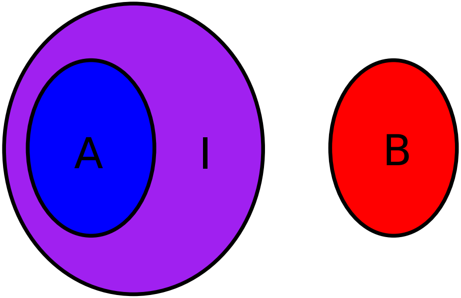
- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]
  - $A \Rightarrow I$, $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B
  - I as over-approximation A conflicting with B

- Example
  - $A \triangleq (\overline{p} \vee \overline{q}) \wedge (p \vee \overline{q})$ $\qquad B \triangleq (q \vee \overline{r}) \wedge (q \vee r)$
  - Interpolant $\overline{q}$

- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]
  - $A \Rightarrow I$, $I \wedge B$ unsatisfiable
  - I defined over common symbols of A and B
  - I as over-approximation A conflicting with B

- Example
  - $A \triangleq (\overline{p} \vee \overline{q}) \wedge (p \vee \overline{q})$ $\qquad B \triangleq (q \vee \overline{r}) \wedge (q \vee r)$
  - Interpolant $\overline{q}$
  - $A \Rightarrow \overline{q}$ $\qquad \overline{q} \wedge B$ unsatisfiable

- Craig's interpolant I for unsatisfiable conjunction of formulae $A \wedge B$ [Craig57]
  - I as over-approximation A conflicting with B



Interpolation

Background

- Applications in symbolic model checking

- Applications in symbolic model checking
  - Bounded model checking: approximate cheaper reachability set computation [McMillan03]

- Applications in symbolic model checking
  - Bounded model checking: approximate cheaper reachability set computation [McMillan03]
  - Predicate abstraction refinement based on spurious behaviors [Henzinger04]

- Applications in symbolic model checking
  - Bounded model checking: approximate cheaper reachability set computation [McMillan03]
  - Predicate abstraction refinement based on spurious behaviors [Henzinger04]
  - Property-based transition relation approximation [Jhala05]

- Applications in symbolic model checking
  - Bounded model checking: approximate cheaper reachability set computation [McMillan03]
  - Predicate abstraction refinement based on spurious behaviors [Henzinger04]
  - Property-based transition relation approximation [Jhala05]
- Forementioned applications involve

- Applications in symbolic model checking
    - Bounded model checking: approximate cheaper reachability set computation [McMillan03]
    - Predicate abstraction refinement based on spurious behaviors [Henzinger04]
    - Property-based transition relation approximation [Jhala05]
- Forementioned applications involve
    - Problem encoding into logic (SAT, SMT)

- Applications in symbolic model checking
  - Bounded model checking: approximate cheaper reachability set computation [McMillan03]
  - Predicate abstraction refinement based on spurious behaviors [Henzinger04]
  - Property-based transition relation approximation [Jhala05]
- Forementioned applications involve
  - Problem encoding into logic (SAT, SMT)
  - Problem solving by means of resolution based engines (SAT solvers, SMT solvers)

SAT and SMT
Background

- Satisfiability (SAT)

- Satisfiability (SAT)
    - Example

$A \triangleq (\overline{p} \vee \overline{q}) \wedge (p \vee \overline{q}) \qquad B \triangleq (q \vee \overline{r}) \wedge (q \vee r)$

- Satisfiability (SAT)
  - Example

    $A \triangleq (\overline{p} \vee \overline{q}) \wedge (p \vee \overline{q}) \qquad B \triangleq (q \vee \overline{r}) \wedge (q \vee r)$

- Satisfiability Modulo Theories (SMT): more expressivity than boolean logic

SAT and SMT
Background

- Satisfiability (SAT)
  - Example

    $A \triangleq (\overline{p} \vee \overline{q}) \wedge (p \vee \overline{q}) \qquad B \triangleq (q \vee \overline{r}) \wedge (q \vee r)$

- Satisfiability Modulo Theories (SMT): more expressivity than boolean logic
  - Timed automata, hybrid systems, ...

SAT and SMT
Background

- Satisfiability (SAT)
  - Example

    $A \triangleq (\overline{p} \vee \overline{q}) \wedge (p \vee \overline{q}) \qquad B \triangleq (q \vee \overline{r}) \wedge (q \vee r)$

- Satisfiability Modulo Theories (SMT): more expressivity than boolean logic
  - Timed automata, hybrid systems, ...
  - Arbitrary precision arithmetic, data structures ...

SAT and SMT
Background

- Satisfiability (SAT)
  - Example

    $A \triangleq (\overline{p} \vee \overline{q}) \wedge (p \vee \overline{q}) \qquad B \triangleq (q \vee \overline{r}) \wedge (q \vee r)$

- Satisfiability Modulo Theories (SMT): more expressivity than boolean logic
  - Timed automata, hybrid systems, ...
  - Arbitrary precision arithmetic, data structures ...
  - Example

    $A \triangleq (5x - y \leq 1) \wedge (y - 5x \leq -1) \qquad B \triangleq (y - 5z \leq 3) \wedge (5z - y \leq -2)$

SAT and SMT

Proofs and Solving Engines

- $A \wedge B$ unsatisfiable: certificate of unsatisfiability

- $A \wedge B$ unsatisfiable: certificate of unsatisfiability
  - Propositional proof of unsatisfiability
  - Generated by logging steps at solving time

- $A \wedge B$ unsatisfiable: certificate of unsatisfiability
  - Propositional proof of unsatisfiability
  - Generated by logging steps at solving time
- DPLL SAT solver [Davis60,62]

SAT and SMT
Proofs and Solving Engines

- $A \wedge B$ unsatisfiable: certificate of unsatisfiability
  - Propositional proof of unsatisfiability
  - Generated by logging steps at solving time
- DPLL SAT solver [Davis60,62]
  - Search space boolean assignments
  - Backtracking

- $A \wedge B$ unsatisfiable: certificate of unsatisfiability
  - Propositional proof of unsatisfiability
  - Generated by logging steps at solving time
- DPLL SAT solver [Davis60,62]
  - Search space boolean assignments
  - Backtracking
- SMT solver

- $A \wedge B$ unsatisfiable: certificate of unsatisfiability
  - Propositional proof of unsatisfiability
  - Generated by logging steps at solving time
- DPLL SAT solver [Davis60,62]
  - Search space boolean assignments
  - Backtracking
- SMT solver
  - DPLL SAT solver
  - Theory solver

- Interpolant I for unsatisfiable conjunction of formulae $A \wedge B$

- Interpolant I for unsatisfiable conjunction of formulae $A \wedge B$

- State-of-the-art approach [Pudlák97, McMillan04]

Interpolation
Generation

- Interpolant I for unsatisfiable conjunction of formulae $A \wedge B$

- State-of-the-art approach [Pudlák97, McMillan04]
  - Derivation of unsatisfiability resolution proof of $A \wedge B$

- Interpolant I for unsatisfiable conjunction of formulae $A \wedge B$

- State-of-the-art approach [Pudlák97, McMillan04]
  - Derivation of unsatisfiability resolution proof of $A \wedge B$
  - Computation of I from proof structure in linear time

Resolution System

Background

- Literal p $\overline{p}$

Resolution System
Background

- Literal $p \quad \overline{p}$
- Clause $p \vee \overline{q} \vee r \vee \ldots \rightarrow p\overline{q}r\ldots$ Empty clause $\bot$

Resolution System
Background

- Literal $p \quad \overline{p}$
- Clause $p \vee \overline{q} \vee r \vee \ldots \rightarrow p\overline{q}r\ldots$ Empty clause $\perp$
- Input formula $(p \vee q) \wedge (r \vee \overline{p}) \ldots \rightarrow \{pq, r\overline{p}\}$

Resolution System
Background

- Literal $p \quad \overline{p}$
- Clause $p \vee \overline{q} \vee r \vee \ldots \rightarrow p\overline{q}r\ldots$ Empty clause $\bot$
- Input formula $(p \vee q) \wedge (r \vee \overline{p}) \ldots \rightarrow \{pq, r\overline{p}\}$
- Resolution rule $$\frac{pC \qquad \overline{p}D}{CD}\, p$$

Antecedents: pC $\overline{p}D$ Resolvent: CD Pivot: p

Resolution System
Background

- Literal p $\overline{p}$
- Clause $p \vee \overline{q} \vee r \vee \ldots \rightarrow p\overline{q}r\ldots$ Empty clause $\bot$
- Input formula $(p \vee q) \wedge (r \vee \overline{p}) \ldots \rightarrow \{pq, r\overline{p}\}$
- Resolution rule $\dfrac{pC \qquad \overline{p}D}{CD}\, p$

  Antecedents: pC $\overline{p}D$ Resolvent: CD Pivot: p
- Resolution proof of unsatisfiability of a set of clauses S

Resolution System
Background

- Literal $p \quad \overline{p}$
- Clause $p \vee \overline{q} \vee r \vee \ldots \rightarrow p\overline{q}r\ldots$ Empty clause $\bot$
- Input formula $(p \vee q) \wedge (r \vee \overline{p}) \ldots \rightarrow \{pq, r\overline{p}\}$
- Resolution rule $$\frac{pC \qquad \overline{p}D}{CD}\ p$$

  Antecedents: $pC\ \overline{p}D$ Resolvent: CD Pivot: p

- Resolution proof of unsatisfiability of a set of clauses S
  - Tree
  - Leaves as clauses of S
  - Intermediate nodes as resolvents
  - Root as unique empty clause

Resolution Proofs
SAT

- $A \triangleq \{\overline{p}\overline{q}, p\overline{q}\}$ $B \triangleq \{q\overline{r}, qr\}$

Resolution Proofs

SAT

- $A \triangleq \{\overline{p}\,\overline{q},\ p\overline{q}\}$ $\qquad B \triangleq \{q\overline{r},\ qr\}$
- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\,\overline{q} \qquad p\overline{q}}{\overline{q}}\ p \qquad\qquad \dfrac{q\overline{r} \qquad qr}{q}\ r}{\bot}\ q$$

Interpolant Generation

SAT [Pudlák97]

- Computation of interpolant I for $A \wedge B$ from proof structure

Interpolant Generation

SAT [Pudlák97]

- Computation of interpolant I for $A \wedge B$ from proof structure
- Partial interpolant for leaf

Interpolant Generation
SAT [Pudlák97]

- Computation of interpolant I for $A \wedge B$ from proof structure
- Partial interpolant for leaf
- Partial interpolant for resolvent
  - Pivot
  - Partial interpolants for antecedents

- Computation of interpolant I for $A \wedge B$ from proof structure
- Partial interpolant for leaf
- Partial interpolant for resolvent
  - Pivot
  - Partial interpolants for antecedents
- Partial interpolant for $\bot$ is I

Interpolant Generation

SAT [Pudlák97]

- $A \triangleq \{\overline{p}\,\overline{q}, p\overline{q}\}$ $\quad$ $B \triangleq \{q\overline{r}, qr\}$

- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\,\overline{q} \qquad p\overline{q}}{\overline{q}}\;p \qquad \dfrac{q\overline{r} \qquad qr}{q}\;r}{\bot}\;q$$

Interpolant Generation

SAT [Pudlák97]

- $A \triangleq \{\overline{p}\,\overline{q}, p\overline{q}\}$ $\qquad$ $B \triangleq \{q\overline{r}, qr\}$

- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\,\overline{q}\ \{\bot\} \qquad p\overline{q}\ \{\bot\}}{\overline{q}}\,p \qquad \dfrac{q\overline{r} \qquad qr}{q}\,r}{\bot}\,q$$

- $A \triangleq \{\overline{p}\,\overline{q}, p\overline{q}\}$ $\qquad$ $B \triangleq \{q\overline{r}, qr\}$

- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\,\overline{q}\ \ \{\bot\} \qquad\quad p\overline{q}\ \ \{\bot\}}{\overline{q}}\ p \qquad \dfrac{q\overline{r}\ \ \{\top\} \qquad\quad qr\ \ \{\top\}}{q}\ r}{\bot}\ q$$

- $A \triangleq \{\overline{p}\,\overline{q}, p\overline{q}\}$ $\quad B \triangleq \{q\overline{r}, qr\}$
- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\,\overline{q}\ \{\bot\} \qquad p\overline{q}\ \{\bot\}}{\overline{q}\ \{\bot \vee \bot\}}\,p \qquad \dfrac{q\overline{r}\ \{\top\} \qquad qr\ \{\top\}}{q}\,r}{\bot}\,q$$

Interpolant Generation
SAT [Pudlák97]

- $A \triangleq \{\overline{p}\overline{q}, p\overline{q}\}$ $\qquad$ $B \triangleq \{q\overline{r}, qr\}$

- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\overline{q}\;\{\bot\} \qquad p\overline{q}\;\{\bot\}}{\overline{q}\;\{\bot\}}\,p \qquad \dfrac{q\overline{r}\;\{\top\} \qquad qr\;\{\top\}}{q}\,r}{\bot}\,q$$

- $A \triangleq \{\overline{p}\overline{q}, p\overline{q}\}$ $\quad B \triangleq \{q\overline{r}, qr\}$

- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\overline{q}\ \{\bot\} \qquad p\overline{q}\ \{\bot\}}{\overline{q}\ \{\bot\}}\,p \qquad \dfrac{q\overline{r}\ \{\top\} \qquad qr\ \{\top\}}{q\ \{\top \wedge \top\}}\,r}{\bot}\,q$$

- $A \triangleq \{\overline{p}\overline{q}, p\overline{q}\}$ $B \triangleq \{q\overline{r}, qr\}$
- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\overline{q}\ \{\bot\} \qquad p\overline{q}\ \{\bot\}}{\overline{q}\ \{\bot\}}\,p \qquad \dfrac{q\overline{r}\ \{\top\} \qquad qr\ \{\top\}}{q\ \{\top\}}\,r}{\bot}\,q$$

Interpolant Generation

SAT [Pudlák97]

- $A \triangleq \{\overline{p}\overline{q}, p\overline{q}\}$ $B \triangleq \{q\overline{r}, qr\}$

- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\overline{q}\ \{\bot\} \qquad p\overline{q}\ \{\bot\}}{\overline{q}\ \{\bot\}}\,p \qquad \dfrac{q\overline{r}\ \{\top\} \qquad qr\ \{\top\}}{q\ \{\top\}}\,r}{\bot\ \{(\bot \vee \overline{q}) \wedge (\top \vee q)\}}\,q$$

Interpolant Generation

SAT [Pudlák97]

- $A \triangleq \{\overline{p}\overline{q}, p\overline{q}\}$ $\quad$ $B \triangleq \{q\overline{r}, qr\}$
- Proof of unsatisfiability

$$\dfrac{\dfrac{\overline{p}\overline{q}\ \{\bot\} \qquad p\overline{q}\ \{\bot\}}{\overline{q}\ \{\bot\}}\,p \qquad \dfrac{q\overline{r}\ \{\top\} \qquad qr\ \{\top\}}{q\ \{\top\}}\,r}{\bot\ \{\overline{q}\}}\,q$$

Resolution Proofs

SMT

- $A \triangleq \{\overbrace{\boxed{(5x - y \leq 1)}}^{p}, \overbrace{\boxed{(y - 5x \leq -1)}}^{q}\}$ $B \triangleq \{\overbrace{\boxed{(y - 5z \leq 3)}}^{r}, \overbrace{\boxed{(5z - y \leq -2)}}^{s}\}$

- $A \triangleq \{ \overbrace{\boxed{(5x - y \leq 1)}}^{p}, \overbrace{\boxed{(y - 5x \leq -1)}}^{q} \}$ $B \triangleq \{ \overbrace{\boxed{(y - 5z \leq 3)}}^{r}, \overbrace{\boxed{(5z - y \leq -2)}}^{s} \}$

- Theory lemmata

- $A \triangleq \{\overbrace{\boxed{(5x - y \leq 1)}}^{p}, \overbrace{\boxed{(y - 5x \leq -1)}}^{q}\}$ $B \triangleq \{\overbrace{\boxed{(y - 5z \leq 3)}}^{r}, \overbrace{\boxed{(5z - y \leq -2)}}^{s}\}$

- Theory lemmata

  - LIA: $\overbrace{\boxed{(x - z \leq 0)}}^{t}\overbrace{\boxed{(x - z \geq 1)}}^{u}$

Resolution Proofs
SMT

- $A \triangleq \{\overbrace{\boxed{(5x - y \leq 1)}}^{p}, \overbrace{\boxed{(y - 5x \leq -1)}}^{q}\}$ $B \triangleq \{\overbrace{\boxed{(y - 5z \leq 3)}}^{r}, \overbrace{\boxed{(5z - y \leq -2)}}^{s}\}$

- Theory lemmata
  - LIA: $\overbrace{\boxed{(x - z \leq 0)}}^{t}\overbrace{\boxed{(x - z \geq 1)}}^{u}$
  - LRA: $\overbrace{\boxed{(5x - y \not\leq 1)}}^{\overline{p}}\overbrace{\boxed{(y - 5z \not\leq 3)}}^{\overline{r}}\overbrace{\boxed{(x - z \not\geq 1)}}^{\overline{u}}$

Resolution Proofs
SMT

- $A \triangleq \{\overbrace{(5x - y \leq 1)}^{p}, \overbrace{(y - 5x \leq -1)}^{q}\}$ $B \triangleq \{\overbrace{(y - 5z \leq 3)}^{r}, \overbrace{(5z - y \leq -2)}^{s}\}$

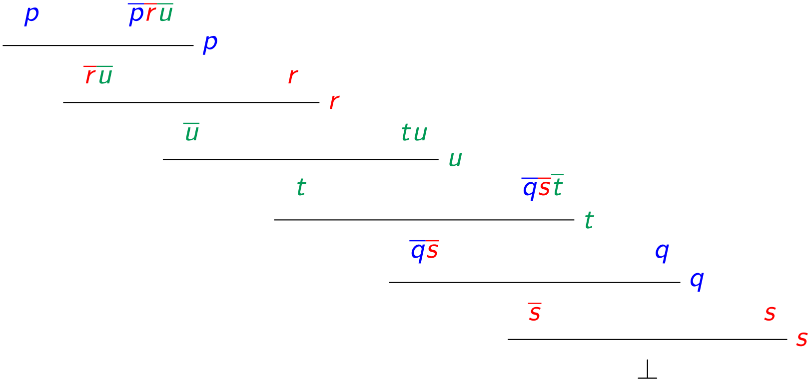
- Theory lemmata
  - LIA: $\overbrace{(x - z \leq 0)}^{t}\ \overbrace{(x - z \geq 1)}^{u}$
  - LRA: $\overbrace{(5x - y \not\leq 1)}^{\overline{p}}\ \overbrace{(y - 5z \not\leq 3)}^{\overline{r}}\ \overbrace{(x - z \not\geq 1)}^{\overline{u}}$
  - LRA: $\overbrace{(y - 5x \not\leq -1)}^{\overline{q}}\ \overbrace{(5z - y \not\leq -2)}^{\overline{s}}\ \overbrace{(x - z \not\leq 0)}^{\overline{t}}$

Resolution Proofs

SMT

- $A \triangleq \{p, q\}$ $\quad B \triangleq \{r, s\}$ $\quad L \triangleq \{tu, \overline{p}\overline{r}\overline{u}, \overline{q}\overline{s}\overline{t}\}$

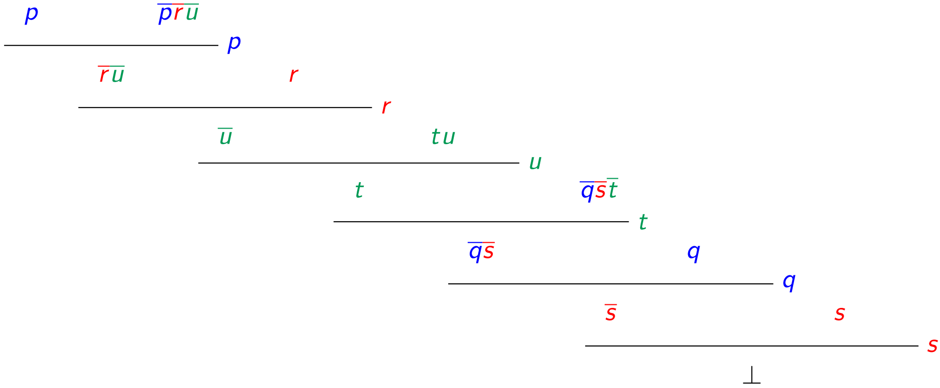
- $A \triangleq \{p, q\}$ $\quad$ $B \triangleq \{r, s\}$ $\quad$ $L \triangleq \{tu, \overline{p}\,\overline{r}\,\overline{u}, \overline{q}\,\overline{s}\,\overline{t}\}$
- Proof of unsatisfiability



Interpolant Generation

SMT

- $A \triangleq \{p, q\}$ $\quad$ $B \triangleq \{r, s\}$ $\quad$ $L \triangleq \{tu, \overline{p}\,\overline{r}\,\overline{u}, \overline{q}\,\overline{s}\,\overline{t}\}$
- Proof of unsatisfiability



- $A \triangleq \{p, q\} \qquad B \triangleq \{r, s\} \qquad L \triangleq \{tu, \overline{p}\,\overline{r}\,\overline{u}, \overline{q}\,\overline{s}\,\overline{t}\}$
- Proof of unsatisfiability

$$\dfrac{p \ \{\bot\} \qquad \overline{p}\,\overline{r}\,\overline{u}}{\overline{r}\,\overline{u}}\ p$$

$$\dfrac{\overline{r}\,\overline{u} \qquad r}{\overline{u}}\ r$$

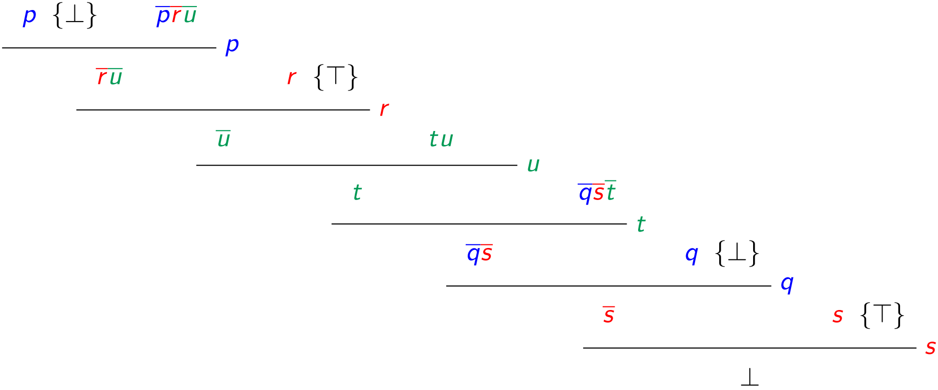
$$\dfrac{\overline{u} \qquad tu}{t}\ u$$

$$\dfrac{t \qquad \overline{q}\,\overline{s}\,\overline{t}}{\overline{q}\,\overline{s}}\ t$$

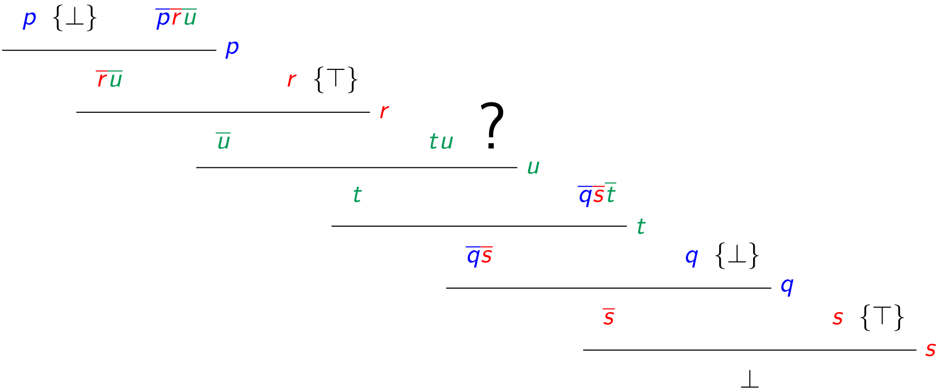
$$\dfrac{\overline{q}\,\overline{s} \qquad q \ \{\bot\}}{\overline{s}}\ q$$

$$\dfrac{\overline{s} \qquad s}{\bot}\ s$$

- $A \triangleq \{p, q\}$ $\quad B \triangleq \{r, s\}$ $\quad L \triangleq \{tu, \bar{p}\bar{r}\bar{u}, \bar{q}\bar{s}\bar{t}\}$
- Proof of unsatisfiability



- $A \triangleq \{p, q\}$ $B \triangleq \{r, s\}$ $L \triangleq \{tu, \overline{p}\,\overline{r}\,\overline{u}, \overline{q}\,\overline{s}\,\overline{t}\}$
- Proof of unsatisfiability



Interpolation
Challenge

- State-of-the-art approach [Pudlák97, McMillan04]

- State-of-the-art approach [Pudlák97, McMillan04]
  - Derivation of unsatisfiability proof of $A \wedge B$
  - Computation of interpolant from proof structure in linear time

- State-of-the-art approach [Pudlák97, McMillan04]
  - Derivation of unsatisfiability proof of $A \wedge B$
  - Computation of interpolant from proof structure in linear time
- Restriction

- State-of-the-art approach [Pudlák97, McMillan04]
  - Derivation of unsatisfiability proof of $A \wedge B$
  - Computation of interpolant from proof structure in linear time
- Restriction
  - Need for proof not to contain AB-mixed predicates

A-local B-local AB-common AB-mixed

- State-of-the-art approach [Pudlák97, McMillan04]
  - Derivation of unsatisfiability proof of $A \wedge B$
  - Computation of interpolant from proof structure in linear time
- Restriction
  - Need for proof not to contain AB-mixed predicates

A-local B-local AB-common AB-mixed

$A \triangleq \{\boxed{(5x - y \leq 1)}, \ldots\}$ $B \triangleq \{\boxed{(y - 5z \leq 3)}, \ldots\}$

- State-of-the-art approach [Pudlák97, McMillan04]
  - Derivation of unsatisfiability proof of $A \wedge B$
  - Computation of interpolant from proof structure in linear time
- Restriction
  - Need for proof not to contain AB-mixed predicates

A-local B-local AB-common AB-mixed

$A \triangleq \{ \boxed{(5x - y \leq 1)}, \ldots \}$ $B \triangleq \{ \boxed{(y - 5z \leq 3)}, \ldots \}$

$L \triangleq \{ \boxed{(x - z \leq 0)}, \ldots \}$

- Need for proof not to contain AB-mixed predicates

Interpolation

Possible Solutions

- Need for proof not to contain AB-mixed predicates

- Tune solvers to avoid generating AB-mixed predicates [Cimatti08,Beyer08]

- Need for proof not to contain AB-mixed predicates
- Tune solvers to avoid generating AB-mixed predicates [Cimatti08,Beyer08]
- Transform proof to remove AB-mixed predicates

Proof Transformation

Motivation

- Proof transformation approach

Proof Transformation

Motivation

- Proof transformation approach
- Motivation: more flexibility by decoupling SMT solving and interpolant generation

- Proof transformation approach
- Motivation: more flexibility by decoupling SMT solving and interpolant generation
- Motivation: standard SMT techniques can require addition of AB-mixed predicates

Proof Transformation

Motivation

- Proof transformation approach
- Motivation: more flexibility by decoupling SMT solving and interpolant generation
- Motivation: standard SMT techniques can require addition of AB-mixed predicates
  - Theory reduction via Lemma on Demand [DeMoura02, Barrett06]

    Reduction of AX to EUF

    Reduction of LIA to LRA

    Ackermann's Expansion
  - Theory combination via DTC [Bozzano05]

Outline

Outline

1. Background
2. Motivation and Related Work
3. Contribution
   - Proof Transformation for Interpolation and Reduction
4. Summary and Future Work

Contribution

Proof Transformation Framework

- Proof rewriting framework based on local rules

Contribution
Proof Transformation Framework

- Proof rewriting framework based on local rules
- Isolation of AB-mixed predicates into subtrees

Contribution

Proof Transformation Framework

- Proof rewriting framework based on local rules
- Isolation of AB-mixed predicates into subtrees
- Removal of AB-mixed subtrees

Contribution

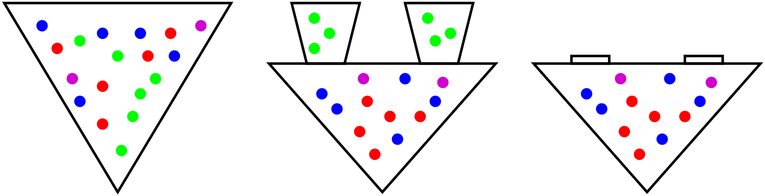
Proof Transformation Framework

- Proof rewriting framework based on local rules
- Isolation of AB-mixed predicates into subtrees
- Removal of AB-mixed subtrees
- No more AB-mixed predicates, proof still valid

(a) Initial proof: A-local, B-local, AB-common, AB-mixed

(b) Transformed proof: AB-mixed predicates isolated into subtrees

(c) Final proof: AB-mixed subtrees removed, new leaves are theory lemmata



(a) (b) (c)

Proof Transformation

Advantages

- No more AB-mixed predicates, new leaves are theory lemmata

Proof Transformation
Advantages

- No more AB-mixed predicates, new leaves are theory lemmata
- Easy combination of SMT and interpolation techniques

Proof Transformation
Advantages

- No more AB-mixed predicates, new leaves are theory lemmata
- Easy combination of SMT and interpolation techniques
  - Theory reduction, theory combination without restrictions

Proof Transformation

Advantages

- No more AB-mixed predicates, new leaves are theory lemmata
- Easy combination of SMT and interpolation techniques
  - Theory reduction, theory combination without restrictions
  - Interpolant generation for propositional resolution proofs of unsatisfiability [Pudlák97]

Proof Transformation

Advantages

- No more AB-mixed predicates, new leaves are theory lemmata
- Easy combination of SMT and interpolation techniques
  - Theory reduction, theory combination without restrictions
  - Interpolant generation for propositional resolution proofs of unsatisfiability [Pudlák97]
  - (Partial) interpolant generation for theory (combination) lemmata [Yorsh05]

Proof Transformation Framework

Features

- Local rewriting rules

Proof Transformation Framework

Features

- Local rewriting rules

- Rule context

$$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q$$

Proof Transformation Framework

Features

- Local rewriting rules
- Rule context

$$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q$$

- Exhaustiveness up to symmetry

Proof Transformation Framework

Local Rewriting Rules

- $$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q \quad \Rightarrow \quad \dfrac{\dfrac{pqC \qquad \overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$$

- $$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q \quad \Rightarrow \quad \dfrac{\dfrac{pqC \qquad \overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$$

- Pivots swapping

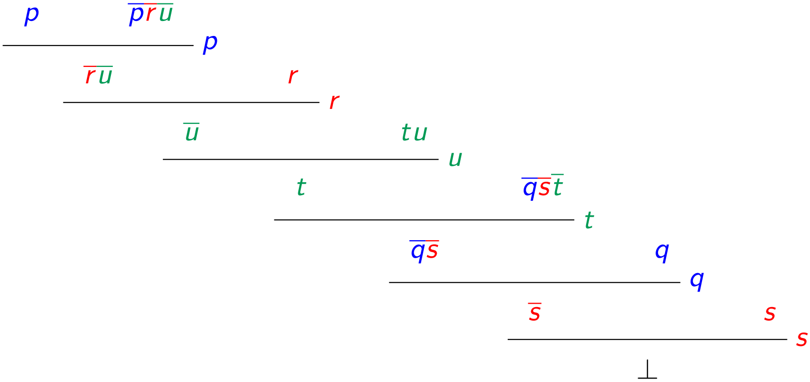
Proof Transformation Framework

Local Rewriting Rules

- $$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q \quad \Rightarrow \quad \dfrac{\dfrac{pqC \qquad \overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$$

- Pivots swapping
- AB-mixed predicates isolation into subtrees

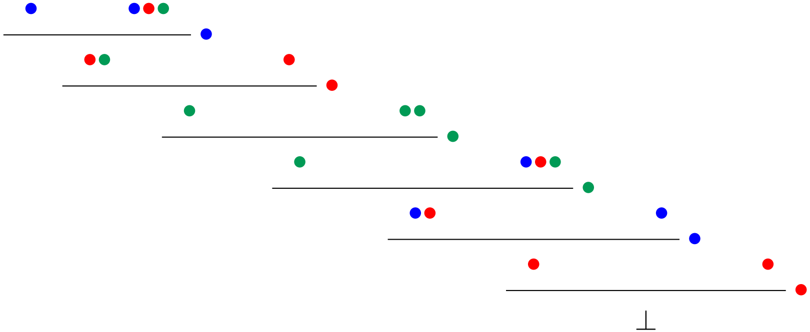
- $A \triangleq \{p, q\}$ $\quad B \triangleq \{r, s\}$ $\quad L \triangleq \{tu, \overline{p}\,\overline{r}\,\overline{u}, \overline{q}\,\overline{s}\,\overline{t}\}$
- Proof of unsatisfiability



Reduction LIA to LRA

Transformation

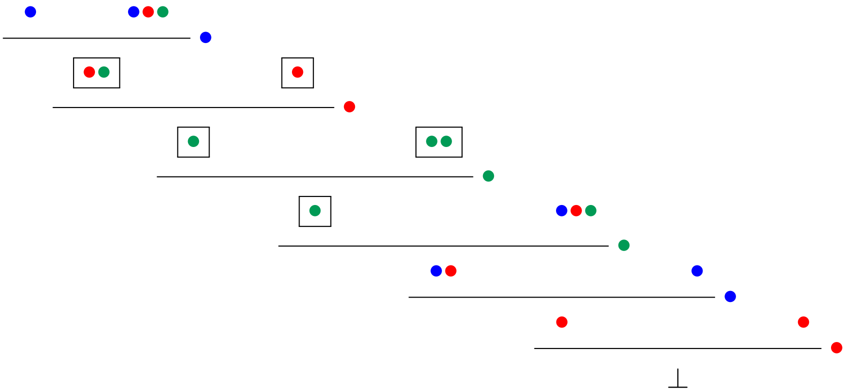
- Proof of unsatisfiability



Reduction LIA to LRA

Transformation

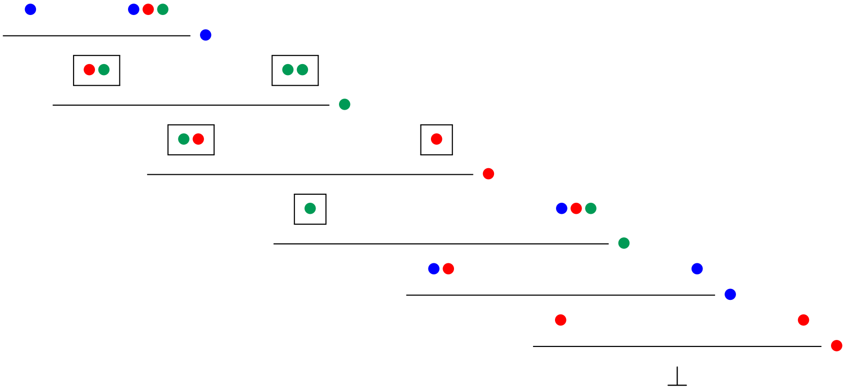
- Proof of unsatisfiability



Reduction LIA to LRA

Transformation

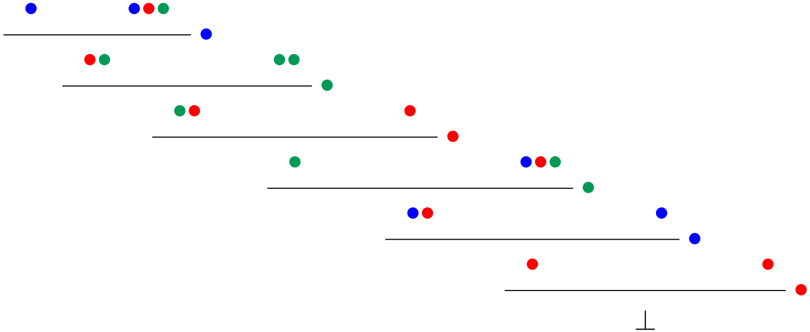
- Proof of unsatisfiability



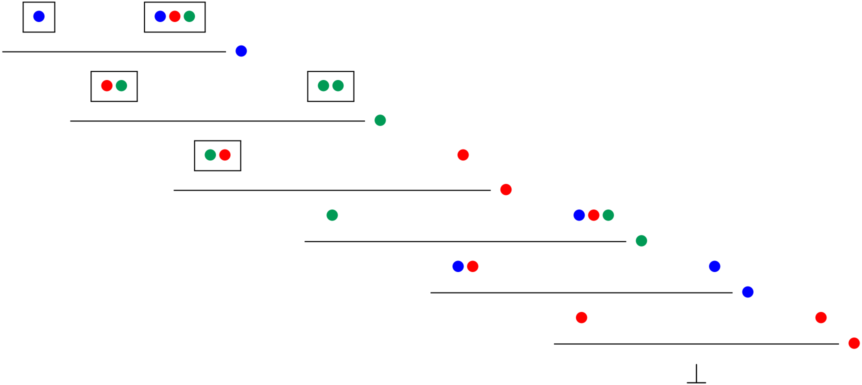
Reduction LIA to LRA

Transformation

- Proof of unsatisfiability



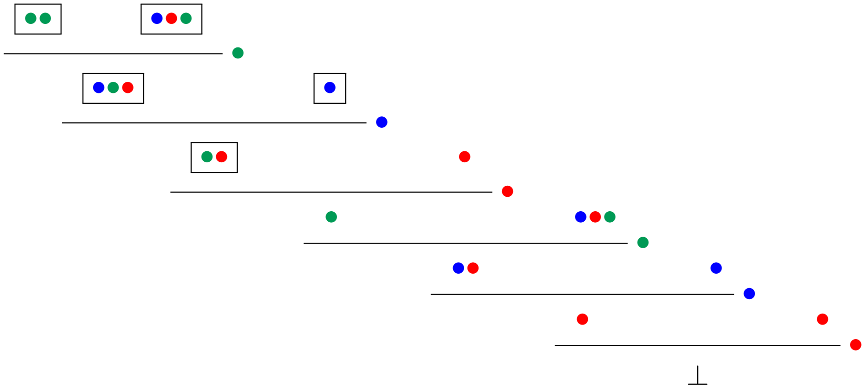
- Proof of unsatisfiability



Reduction LIA to LRA

Transformation

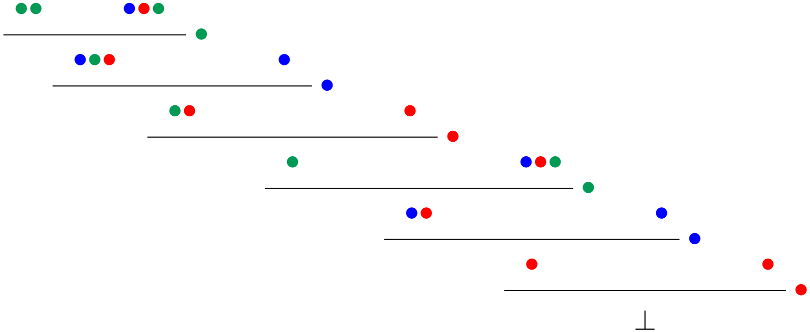
- Proof of unsatisfiability



Reduction LIA to LRA

Transformation

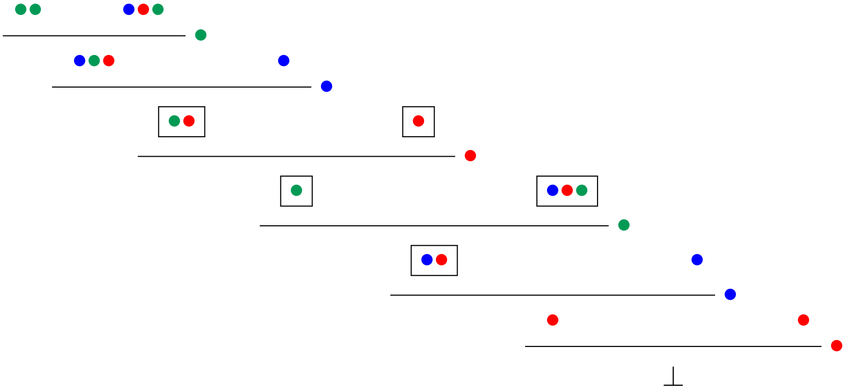
- Proof of unsatisfiability



Reduction LIA to LRA

Transformation

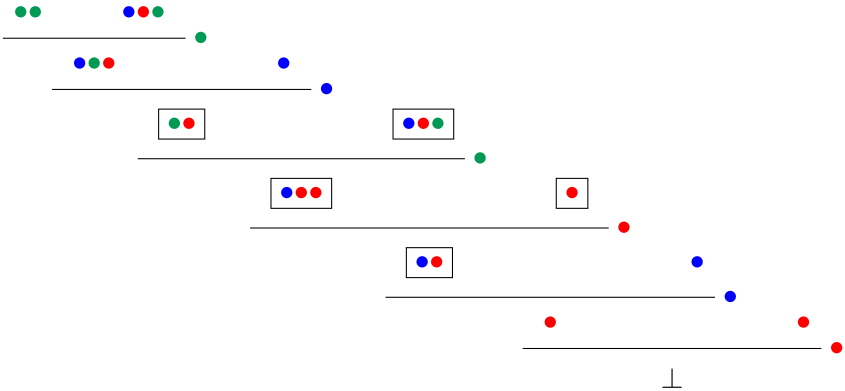
- Proof of unsatisfiability



Reduction LIA to LRA

Transformation

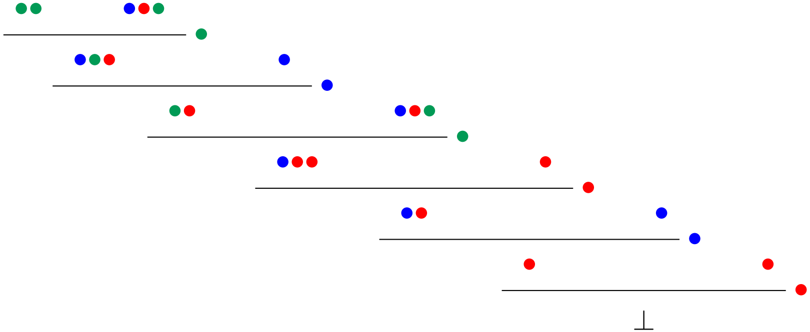
- Proof of unsatisfiability



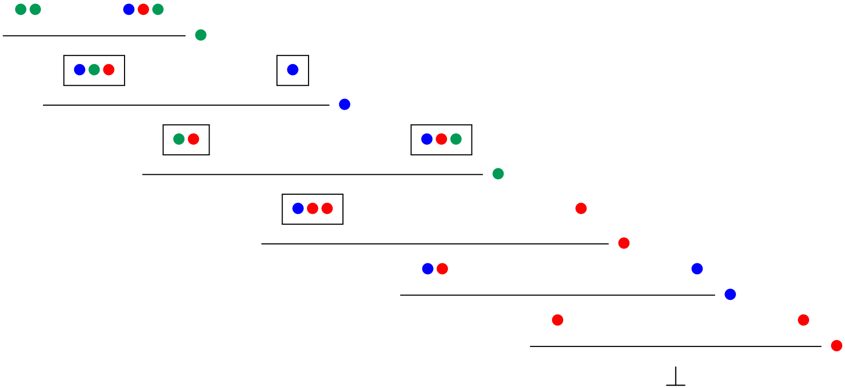
Reduction LIA to LRA

Transformation

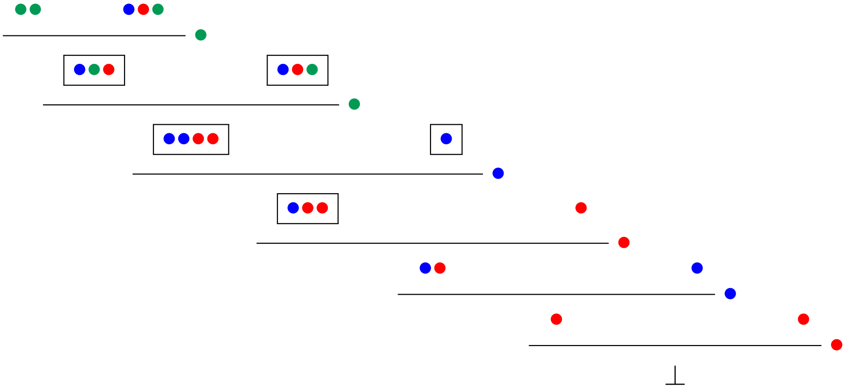
- Proof of unsatisfiability



- Proof of unsatisfiability



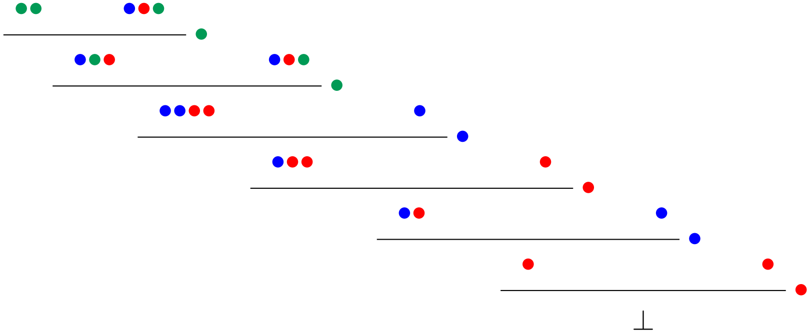
- Proof of unsatisfiability



Reduction LIA to LRA

Transformation

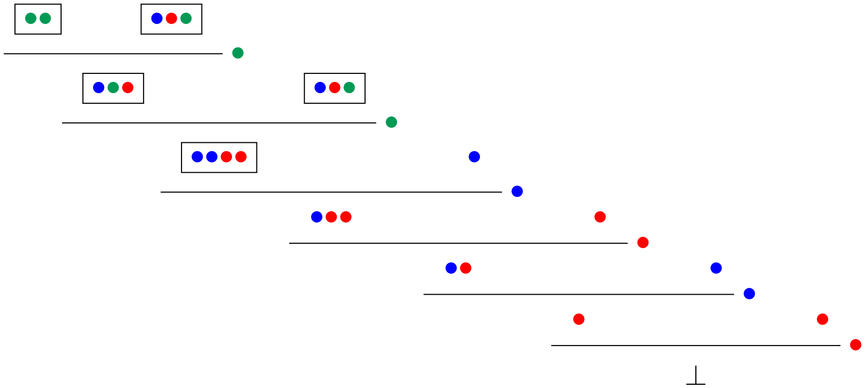
- Proof of unsatisfiability



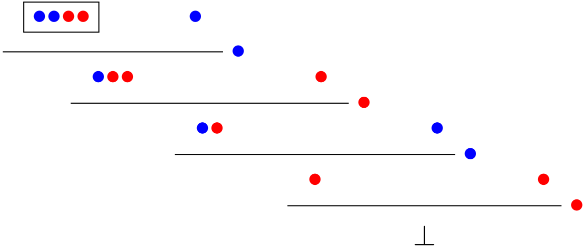
Reduction LIA to LRA

Transformation

- Proof of unsatisfiability



- Proof of unsatisfiability



Proof Transformation Framework

Considerations

- Potential drawbacks

Proof Transformation Framework

Considerations

- Potential drawbacks
    - Overhead w.r.t. solving time

- Potential drawbacks
  - Overhead w.r.t. solving time
  - Increase of proof size

Transformation Framework

Features

- Local rewriting rules

Transformation Framework
Features

- Local rewriting rules
  - B reduction
  - A perturbation

Transformation Framework

Features

- Local rewriting rules
  - B reduction
  - A perturbation

- Rule context

$$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q$$

Transformation Framework

Features

- Local rewriting rules
  - B reduction
  - A perturbation

- Rule context

$$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q$$

- Exhaustiveness up to symmetry

- B rules

$B1$	$$\dfrac{\dfrac{pqC \qquad \overline{p}qD}{qCD}\,p \qquad p\overline{q}E}{pCDE}\,q \quad \Rightarrow \quad \dfrac{pqC \qquad p\overline{q}E}{pCE}\,q$$

Transformation Framework

Local rewriting rules

- B rules

$B1$	$\dfrac{\dfrac{pqC \quad \overline{p}qD}{qCD}\,p \qquad p\overline{q}E}{pCDE}\,q \quad \Rightarrow \quad \dfrac{pqC \quad p\overline{q}E}{pCE}\,q$

- Redundancy as reintroduction variable after elimination

- B rules

$B1$	$\dfrac{\dfrac{pqC \qquad \overline{p}qD}{qCD}\,p \qquad p\overline{q}E}{pCDE}\,q \quad \Rightarrow \quad \dfrac{pqC \qquad p\overline{q}E}{pCE}\,q$

- Redundancy as reintroduction variable after elimination

- Subproof simplification

Transformation Framework

Local rewriting rules

- B rules

$B1$	$\dfrac{\dfrac{pqC \qquad \overline{p}qD}{qCD}\,p \qquad p\overline{q}E}{pCDE}\,q \quad \Rightarrow \quad \dfrac{pqC \qquad p\overline{q}E}{pCE}\,q$

- Redundancy as reintroduction variable after elimination
- Subproof simplification
- Subproof root strengthening

Transformation Framework

Local rewriting rules

- A rules

$A2$	$\dfrac{\dfrac{pqC \quad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q \quad \Rightarrow \quad \dfrac{\dfrac{pqC \quad \overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$

- A rules

$A2$	$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q \quad \Rightarrow \quad \dfrac{\dfrac{pqC \qquad \overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$

- Pivots swapping

- A rules

$A2$	$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q \quad \Rightarrow \quad \dfrac{\dfrac{pqC \qquad \overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$

  - Pivots swapping

  - Topology perturbation

- A rules

$A2$	$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q \quad \Rightarrow \quad \dfrac{\dfrac{pqC \qquad \overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$

- Pivots swapping
- Topology perturbation
- Redundancies exposure

Local rewriting rules

$A1$	$\dfrac{\dfrac{pqC \qquad \overline{p}qD}{qCD}\,p \qquad \overline{q}E}{CDE}\,q$	$\Rightarrow$	$\dfrac{\dfrac{pqC \qquad \overline{q}E}{pCE} \qquad \dfrac{\overline{q}E \qquad \overline{p}qD}{\overline{p}DE}\,q}{CDE}\,p$
$A2$	$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{q}E}{CDE}\,q$	$\Rightarrow$	$\dfrac{\dfrac{pqC \qquad \overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$
$B1$	$\dfrac{\dfrac{pqC \qquad \overline{p}qD}{qCD}\,p \qquad p\overline{q}E}{pCDE}\,q$	$\Rightarrow$	$\dfrac{pqC \qquad p\overline{q}E}{pCE}\,q$
$B2$	$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qDC}\,p \qquad p\overline{q}E}{pCDE}\,q$	$\Rightarrow$	$\dfrac{\dfrac{pqC \qquad p\overline{q}E}{pCE}\,q \qquad \overline{p}D}{CDE}\,p$
$B2'$	$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qDC}\,p \qquad p\overline{q}E}{pCDE}\,q$	$\Rightarrow$	$\dfrac{pqC \qquad p\overline{q}E}{pCE}\,q$
$B3$	$\dfrac{\dfrac{pqC \qquad \overline{p}D}{qCD}\,p \qquad \overline{p}\overline{q}E}{\overline{p}CDE}\,q$	$\Rightarrow$	$\overline{p}D$

- opensmt

- opensmt
  - C++ open-source SMT solver developed at USI
  - Fastest open-source solver in SMT-comp 2009, 2010 for various logics

Evaluation

Framework and Benchmarks

- opensmt
  - C++ open-source SMT solver developed at USI
  - Fastest open-source solver in SMT-comp 2009, 2010 for various logics

- Benchmarks

- opensmt
    - C++ open-source SMT solver developed at USI
    - Fastest open-source solver in SMT-comp 2009, 2010 for various logics
- Benchmarks
    - SMT: SMT-LIB library
    - Academic and industrial problems

Evaluation

Experimental results over QF_UFIDL

Group	#	$\#AB$	$\%_{time}$	$\%_{nodes}$	$\%_{edges}$
RDS	2	7	93%	2%	2%
EufLaAr	2	103	91%	30%	26%
pete	6	4	33%	8%	9%
pete2	56	17	59%	27%	32%
uclid	8	11	64%	37%	42%
Overall	74	17	59%	26%	30%

- # — number of benchmarks solved
- $\#AB$ — average number of AB-mixed predicates in proof
- $\%_{time}$ — average time overhead
- $\%_{nodes}$, $\%_{edges}$ — average difference in proof size

Comparison

- RecyclePivots (closest related work) [Strichman'08]

Comparison

- RecyclePivots (closest related work) [Strichman'08]
  - **Pros**
    Global information
    Fast and effective
  - **Cons**
    Cannot expose redundancies

Comparison

- RecyclePivots (closest related work) [Strichman'08]
  - **Pros**
    Global information
    Fast and effective
  - **Cons**
    Cannot expose redundancies
- Rule-based approach

Comparison

- RecyclePivots (closest related work) [Strichman'08]
  - **Pros**
    Global information
    Fast and effective
  - **Cons**
    Cannot expose redundancies

- Rule-based approach
  - **Pros**
    Flexibility in rules application
    Flexibility in amount of transformation
    Can expose redundancies
  - **Cons**
    Local information

Implementation

Reduction Algorithm

- Based on a sequence of proof traversals (e.g. topological order)

Implementation
Reduction Algorithm

- Based on a sequence of proof traversals (e.g. topological order)
- Parameterized in number of traversals and time limit

- Based on a sequence of proof traversals (e.g. topological order)
- Parameterized in number of traversals and time limit
- Examination non-leaf clauses

Implementation
Reduction Algorithm

- Based on a sequence of proof traversals (e.g. topological order)
- Parameterized in number of traversals and time limit
- Examination non-leaf clauses
  - Pivot in both antecedents $\longrightarrow$ update, match context, apply rule

$$\frac{qC'D' \qquad \overline{q}E'}{CDE}\,q \quad \Rightarrow \quad \frac{qC'D' \qquad \overline{q}E'}{C'D'E'}\,q \quad \Rightarrow \quad \frac{\dfrac{pqC' \qquad \overline{p}D'}{qC'D'}\,p \qquad \overline{q}E'}{C'D'E'}\,q$$

Implementation

Reduction Algorithm

- Based on a sequence of proof traversals (e.g. topological order)
- Parameterized in number of traversals and time limit
- Examination non-leaf clauses
  - Pivot in both antecedents $\longrightarrow$ update, match context, apply rule

$$\dfrac{qC'D' \qquad \overline{q}E'}{CDE}\,q \;\Rightarrow\; \dfrac{qC'D' \qquad \overline{q}E'}{C'D'E'}\,q \;\Rightarrow\; \dfrac{\dfrac{pqC' \qquad \overline{p}D'}{qC'D'}\,p \qquad \overline{q}E'}{C'D'E'}\,q$$

  - Pivot not in both antecedents $\longrightarrow$ remove resolution step

$$\dfrac{C'D' \qquad \overline{q}E'}{CDE}\,q \;\Rightarrow\; C'D'$$

- Based on a sequence of proof traversals (e.g. topological order)
- Parameterized in number of traversals and time limit
- Examination non-leaf clauses
  - Pivot in both antecedents $\rightarrow$ update, match context, apply rule

$$\frac{qC'D' \qquad \overline{q}E'}{CDE}\,q \quad\Rightarrow\quad \frac{qC'D' \qquad \overline{q}E'}{C'D'E'}\,q \quad\Rightarrow\quad \frac{\dfrac{pqC' \qquad \overline{p}D'}{qC'D'}\,p \qquad \overline{q}E'}{C'D'E'}\,q$$

  - Pivot not in both antecedents $\rightarrow$ remove resolution step

$$\frac{C'D' \qquad \overline{q}E'}{CDE}\,q \quad\Rightarrow\quad C'D'$$

- Easy combination with RecyclePivots

- Implemented in C++ and integrated with OpenSMT
- Available at **www.inf.usi.ch/phd/rollini/hvc.html**

- Implemented in C++ and integrated with OpenSMT
- Available at **www.inf.usi.ch/phd/rollini/hvc.html**

- Benchmarks

Evaluation

Framework and Benchmarks

- Implemented in C++ and integrated with OpenSMT
- Available at **www.inf.usi.ch/phd/rollini/hvc.html**

- Benchmarks
  - SMT: SMT-LIB library
  - SAT: SAT competition
  - Academic and industrial problems

Combined Approach Evaluation

Experimental results over SMT: QF_UF, QF_IDL, QF_LRA, QF_RDL

	#	Avg_{nodes}	Avg_{edges}	Avg_{core}	$T(s)$	Max_{nodes}	Max_{edges}	Max_{core}
RP	1370	6.7%	7.5%	1.3%	1.7	65.1%	68.9%	39.1%
Ratio								
0.01	1366	8.9%	10.7%	1.4%	3.4	66.3%	70.2%	45.7%
0.025	1366	9.8%	11.9%	1.5%	3.6	77.2%	79.9%	45.7%
0.05	1366	10.7%	13.0%	1.6%	4.1	78.5%	81.2%	45.7%
0.075	1366	11.4%	13.8%	1.7%	4.5	78.5%	81.2%	45.7%
0.1	1364	11.8%	14.4%	1.7%	5.0	78.8%	83.6%	45.7%
0.25	1359	13.6%	16.6%	1.9%	7.6	79.6%	84.4%	45.7%
0.5	1348	15.0%	18.4%	2.0%	11.5	79.1%	85.2%	45.7%
0.75	1341	16.0%	19.5%	2.1%	15.1	79.9%	86.1%	45.7%
1	1337	16.7%	20.4%	2.2%	18.8	79.9%	86.1%	45.7%

- *Ratio* — time threshold as fraction of solving time
- # — number of benchmarks solved
- Avg_{nodes}, Avg_{edges}, Avg_{core} — average reduction in proof size
- $T(s)$ — average transformation time in seconds
- Max_{nodes}, Max_{edges}, Max_{core} — max reduction in proof size

Combined Approach Evaluation

Experimental results over SMT: QF_UF, QF_IDL, QF_LRA, QF_RDL

	#	Avg_{nodes}	Avg_{edges}	Avg_{core}	$T(s)$	Max_{nodes}	Max_{edges}	Max_{core}
RP	1370	6.7%	7.5%	1.3%	1.7	65.1%	68.9%	39.1%
Ratio								
0.01	1366	8.9%	10.7%	1.4%	3.4	66.3%	70.2%	45.7%
0.025	1366	9.8%	11.9%	1.5%	3.6	77.2%	79.9%	45.7%
0.05	1366	10.7%	13.0%	1.6%	4.1	78.5%	81.2%	45.7%
0.075	1366	11.4%	13.8%	1.7%	4.5	78.5%	81.2%	45.7%
0.1	1364	11.8%	14.4%	1.7%	5.0	78.8%	83.6%	45.7%
0.25	1359	13.6%	16.6%	1.9%	7.6	79.6%	84.4%	45.7%
0.5	1348	15.0%	18.4%	2.0%	11.5	79.1%	85.2%	45.7%
0.75	1341	16.0%	19.5%	2.1%	15.1	79.9%	86.1%	45.7%
1	1337	16.7%	20.4%	2.2%	18.8	79.9%	86.1%	45.7%

- *Ratio* — time threshold as fraction of solving time
- # — number of benchmarks solved
- Avg_{nodes}, Avg_{edges}, Avg_{core} — average reduction in proof size
- $T(s)$ — average transformation time in seconds
- Max_{nodes}, Max_{edges}, Max_{core} — max reduction in proof size

Combined Approach Evaluation

Experimental results over SMT: QF_UF, QF_IDL, QF_LRA, QF_RDL

	#	Avg_{nodes}	Avg_{edges}	Avg_{core}	$T(s)$	Max_{nodes}	Max_{edges}	Max_{core}
RP	1370	6.7%	7.5%	1.3%	1.7	65.1%	68.9%	39.1%
Ratio								
0.01	1366	8.9%	10.7%	1.4%	3.4	66.3%	70.2%	45.7%
0.025	1366	9.8%	11.9%	1.5%	3.6	77.2%	79.9%	45.7%
0.05	1366	10.7%	13.0%	1.6%	4.1	78.5%	81.2%	45.7%
0.075	1366	11.4%	13.8%	1.7%	4.5	78.5%	81.2%	45.7%
0.1	1364	11.8%	14.4%	1.7%	5.0	78.8%	83.6%	45.7%
0.25	1359	13.6%	16.6%	1.9%	7.6	79.6%	84.4%	45.7%
0.5	1348	15.0%	18.4%	2.0%	11.5	79.1%	85.2%	45.7%
0.75	1341	16.0%	19.5%	2.1%	15.1	79.9%	86.1%	45.7%
1	1337	16.7%	20.4%	2.2%	18.8	79.9%	86.1%	45.7%

- *Ratio* — time threshold as fraction of solving time
- # — number of benchmarks solved
- Avg_{nodes}, Avg_{edges}, Avg_{core} — average reduction in proof size
- $T(s)$ — average transformation time in seconds
- Max_{nodes}, Max_{edges}, Max_{core} — max reduction in proof size

Combined Approach Evaluation

Experimental results over SAT

	#	Avg_{nodes}	Avg_{edges}	Avg_{core}	$T(s)$	Max_{nodes}	Max_{edges}	Max_{core}
RP	25	5.9%	6.5%	1.7%	10.8	33.1%	33.4%	30.3%
Ratio								
0.01	25	6.8%	7.9%	1.7%	32.3	34.0%	34.4%	30.5%
0.025	25	6.8%	7.9%	1.7%	32.3	34.0%	34.4%	30.5%
0.05	25	7.0%	8.2%	1.8%	40.0	34.0%	34.4%	30.5%
0.075	25	7.2%	8.4%	1.8%	49.3	34.7%	35.1%	30.5%
0.1	25	7.3%	8.4%	1.8%	60.2	34.7%	35.1%	30.5%
0.25	25	7.6%	8.8%	1.9%	125.3	39.8%	40.6%	31.7%
0.5	25	7.8%	9.1%	1.9%	243.5	41.0%	41.9%	32.1%
0.75	25	7.9%	9.3%	1.9%	360.0	41.6%	42.6%	32.1%
1	23	8.4%	9.9%	2.1%	175.6	33.1%	33.4%	30.6%

- *Ratio* — time threshold as fraction of solving time
- # — number of benchmarks solved
- Avg_{nodes}, Avg_{edges}, Avg_{core} — average reduction in proof size
- $T(s)$ — average transformation time in seconds
- Max_{nodes}, Max_{edges}, Max_{core} — max reduction in proof size

Combined Approach Evaluation

Experimental results over SAT

	#	Avg_{nodes}	Avg_{edges}	Avg_{core}	$T(s)$	Max_{nodes}	Max_{edges}	Max_{core}
RP	25	5.9%	6.5%	1.7%	10.8	33.1%	33.4%	30.3%
Ratio								
0.01	25	6.8%	7.9%	1.7%	32.3	34.0%	34.4%	30.5%
0.025	25	6.8%	7.9%	1.7%	32.3	34.0%	34.4%	30.5%
0.05	25	7.0%	8.2%	1.8%	40.0	34.0%	34.4%	30.5%
0.075	25	7.2%	8.4%	1.8%	49.3	34.7%	35.1%	30.5%
0.1	25	7.3%	8.4%	1.8%	60.2	34.7%	35.1%	30.5%
0.25	25	7.6%	8.8%	1.9%	125.3	39.8%	40.6%	31.7%
0.5	25	7.8%	9.1%	1.9%	243.5	41.0%	41.9%	32.1%
0.75	25	7.9%	9.3%	1.9%	360.0	41.6%	42.6%	32.1%
1	23	8.4%	9.9%	2.1%	175.6	33.1%	33.4%	30.6%

- *Ratio* — time threshold as fraction of solving time
- # — number of benchmarks solved
- Avg_{nodes}, Avg_{edges}, Avg_{core} — average reduction in proof size
- $T(s)$ — average transformation time in seconds
- Max_{nodes}, Max_{edges}, Max_{core} — max reduction in proof size

Combined Approach Evaluation

Experimental results over SAT

	#	Avg_{nodes}	Avg_{edges}	Avg_{core}	$T(s)$	Max_{nodes}	Max_{edges}	Max_{core}
RP	25	5.9%	6.5%	1.7%	10.8	33.1%	33.4%	30.3%
Ratio								
0.01	25	6.8%	7.9%	1.7%	32.3	34.0%	34.4%	30.5%
0.025	25	6.8%	7.9%	1.7%	32.3	34.0%	34.4%	30.5%
0.05	25	7.0%	8.2%	1.8%	40.0	34.0%	34.4%	30.5%
0.075	25	7.2%	8.4%	1.8%	49.3	34.7%	35.1%	30.5%
0.1	25	7.3%	8.4%	1.8%	60.2	34.7%	35.1%	30.5%
0.25	25	7.6%	8.8%	1.9%	125.3	39.8%	40.6%	31.7%
0.5	25	7.8%	9.1%	1.9%	243.5	41.0%	41.9%	32.1%
0.75	25	7.9%	9.3%	1.9%	360.0	41.6%	42.6%	32.1%
1	23	8.4%	9.9%	2.1%	175.6	33.1%	33.4%	30.6%

- *Ratio* — time threshold as fraction of solving time
- $\#$ — number of benchmarks solved
- Avg_{nodes}, Avg_{edges}, Avg_{core} — average reduction in proof size
- $T(s)$ — average transformation time in seconds
- Max_{nodes}, Max_{edges}, Max_{core} — max reduction in proof size

Outline

Summary

- Proof transformation

  1. Interpolation, SMT, AB-mixed predicates

Summary

- Proof transformation

  1. Interpolation, SMT, AB-mixed predicates

  2. Proof transformation framework for AB-mixed predicates removal

Summary

- Proof transformation
  1. Interpolation, SMT, AB-mixed predicates
  2. Proof transformation framework for AB-mixed predicates removal
  3. Easy combination:
     - Standard SMTs
     - State-of-the art interpolant generation procedures

Summary

- Proof transformation
  1. Interpolation, SMT, AB-mixed predicates
  2. Proof transformation framework for AB-mixed predicates removal
  3. Easy combination:
     - Standard SMTs
     - State-of-the art interpolant generation procedures
- Rule-based proof reduction

Summary

- Proof transformation
  1. Interpolation, SMT, AB-mixed predicates
  2. Proof transformation framework for AB-mixed predicates removal
  3. Easy combination:
     - Standard SMTs
     - State-of-the art interpolant generation procedures
- Rule-based proof reduction
- Pivots redundancies detection and removal

Future Work

- Exploitation of DPLL proof structure

Future Work

- Exploitation of DPLL proof structure
- Evaluation on concrete applications (e.g. interpolation)

Future Work

- Exploitation of DPLL proof structure
- Evaluation on concrete applications (e.g. interpolation)
- Rule-based control of interpolants' strength

Publications

- Proof reduction

S.F. Rollini, R. Bruttomesso and N. Sharygina
An Efficient and Flexible Approach to Resolution Proof Reduction.
HVC 2010.

- Proof manipulation for interpolation

R. Bruttomesso, S.F. Rollini, N. Sharygina and A. Tsitovich
Flexible Interpolation with Local Proof Transformations.
ICCAD 2010

Thanks for your attention!

http://www.verify.inf.usi.ch/