Leveraging Linear and Mixed Integer Programming for SMT

Tim King\textsuperscript{1}  Clark Barrett\textsuperscript{1}  Cesare Tinelli\textsuperscript{2}

\textsuperscript{1}New York University
\textsuperscript{2}The University of Iowa

October 23, 2014
**APPROACH**

- Floating point LP/MIP solver within SMT to:
  1. Reseed the Simplex solver
  2. Replay an MIP proof
APPRAOCH

- Floating point LP/MIP solver within SMT to:
  1. Reseed the Simplex solver
  2. Replay an MIP proof

- Philosophy
  - Solve hard/unsolved problems
  - Augment SMT solver
  - Minimize changes in search by external solver
Table of Contents

Simplex Background

Reseeding Simplex

Replaying MIP Proofs

Empirical Results

Conclusion
Is there a satisfying assignment, \( a : \mathcal{X} \rightarrow \mathbb{R} \), that makes,

\[
\begin{align*}
    x + y & \geq 1 \\
    x - y & \geq 0 \\
    4x - y & \leq 2
\end{align*}
\]

evaluate to true?
**Decision Procedure for QF_LRA**

**Quantifier Free Linear Real Arithmetic**

Is there a satisfying assignment, \( a : \mathcal{X} \to \mathbb{R} \), that makes,

\[
\begin{align*}
    x + y & \geq 1 \\
    x - y & \geq 0 \\
    4x - y & \leq 2
\end{align*}
\]

evaluate to true?

\[
\begin{bmatrix}
    a_x \\
    a_y
\end{bmatrix} = \begin{bmatrix}
    \frac{1}{2} \\
    \frac{1}{2}
\end{bmatrix}
\]
VISUALLY

\[
\begin{align*}
    x + y & \geq 1 \\
    x - y & \geq 0 \\
    4x - y & \leq 2
\end{align*}
\]

\[
\begin{bmatrix}
    a_x \\
    a_y
\end{bmatrix} = \begin{bmatrix}
    \frac{1}{2} \\
    \frac{1}{2}
\end{bmatrix}
\]
Preprocessing

- Introduce a fresh $s_i$ for each $\sum T_{i,j} \cdot x_j$

- Literals are of the form:

$$\land \left( s_i = \sum x_j T_{i,j} \cdot x_j \right) \land \land l_i \leq x_i \leq u_i$$

and $s_i$ appears in exactly 1 equality.

- Collect into: $T\mathcal{X} = 0$ and $l \leq \mathcal{X} \leq u$
**Basic, Nonbasic, & Tableau**

- Every row in $T$ is solved for a variable $x_i$

$$x_i = \sum_{x_j \in \mathcal{N}} T_{i,j}x_j$$

- Not solved for variables are **nonbasic** $(x_j \in \mathcal{N})$

- Set of solved for variables are **basic** $(x_i \in \mathcal{B})$
Changing the assignment to $j \in \mathcal{N}$ is easy

**procedure** `UPDATE(j, \delta)`

\[ a_j \leftarrow a_j + \delta \]

**for all** basic $x_i$ **do**

\[ a_i \leftarrow a_i + T_{i,j} \cdot \delta \]
**Updating Nonbasic Variables**

Changing the assignment to $j \in \mathcal{N}$ is easy

**procedure** `UPDATE(j, \delta)`

\[
\begin{align*}
a_j & \leftarrow a_j + \delta \\
\text{for all} \; \text{basic} \; x_i \; \text{do} \\
& a_i \leftarrow a_i + T_{i,j} \cdot \delta
\end{align*}
\]

**Add the Invariant**

The nonbasic variables satisfy their bounds.
PIVOT\((i, j)\)

Move Variables In/Out of \(B\)

**Preconditions**

Given \(x_i\) basic, \(x_j\) nonbasic, and \(T_{i,j} \neq 0\), PIVOT\((i, j)\) makes \(x_i\) nonbasic and \(x_j\) basic.
## PIVOT\((i, j)\)

**Move Variables In/Out of** \(B\)

### Preconditions

Given \(x_i\) basic, \(x_j\) nonbasic, and \(T_{i,j} \neq 0\), **PIVOT\((i, j)\)** makes \(x_i\) nonbasic and \(x_j\) basic.

- **Take** \(x_i\)'s row
  \[
x_i = T_{i,j}x_j + \sum T_{i,k}x_k
  \]

- **Solve for** \(x_j\)
  \[
x_j = \frac{1}{T_{i,j}}x_i + \sum -\frac{T_{i,k}}{T_{i,j}}x_k
  \]

- **Replace** \(x_j\) everywhere else in \(T\)
TABLEAU EXAMPLE

\[ x + y \geq 1 \]
\[ x - y \geq 0 \]
\[ 4x - y \leq 2 \]
TABLEAU EXAMPLE

$T\mathbf{x} = 0$ is equivalent to

\begin{align*}
s_1 &= x + y \\
s_2 &= x - y \\
s_3 &= 4x + y
\end{align*}

$s_1 \geq 1 \land s_2 \geq 0 \land s_3 \leq 2$

$B = \{s_1, s_2, s_3\}, \mathcal{N} = \{x, y\}$
**Simplex for DPLL(T)** [DdM06]

\[
\textbf{while } \neg(l \leq a \leq u) \textbf{ do } \\
\quad \text{for all } i \in B, \text{ row } i \text{ is } x_i = \sum T_{i,f} x_j \\
\quad \text{if } \exists i \in B \text{ s.t. } a_i > u_i, \text{ and } \sum T_{i,j} x_j \text{ is minimized then } \\
\quad \quad \text{return a row conflict from row } i \\
\quad \text{else } \\
\quad \quad \text{select some basic } x_i \text{ s.t. } a_i > u_i \\
\quad \quad \text{select } x_j \text{ from } \sum T_{i,j} \cdot x_j \\
\quad \quad \text{Update the assignment of } x_j \text{ s.t. } a_i \leftarrow u_i \\
\quad \text{PIVOT}(i, j) \quad \triangleright O(|T|) \\
\]

Ignoring \( a_i < l_i \) cases
**Row Conflicts**

- Suppose \( \forall T_{i,j} > 0. a_j = l_j \) and \( \forall T_{i,j} < 0. a_j = u_j. \)
- Then \( \sum T_{i,j} x_j \geq \sum T_{i,j} a_j \) (or minimized)
**Row Conflicts**

- Suppose $\forall T_{i,j} > 0. a_j = l_j$ and $\forall T_{i,j} < 0. a_j = u_j$.
- Then $x_i = \sum T_{i,j} x_j \geq \sum T_{i,j} a_j = a_i$ (or minimized)
Suppose $\forall T_{i,j} > 0. \ a_j = l_j$ and $\forall T_{i,j} < 0. \ a_j = u_j$.

Then $x_i = \sum T_{i,j} x_j \geq \sum T_{i,j} a_j = a_i$ (or minimized)

$\ a_i > u_i \geq x_i \geq a_i \models \text{false}$
SIMPLEX FOR DPLL(\(\mathcal{T}\))

Observations

- Simplex searches for \(a\)'s that are against bounds
- Pivoting is expensive
- Most checks need few pivots [KBD13]
SUM-OF-INFEASIBILITIES SIMPLEX [KBD13]
# Table of Contents

- Simplex Background
- Reseeding Simplex
- Replaying MIP Proofs
- Empirical Results
- Conclusion
LEVERAGING LP

- SOI Simplex added optimization to Simplex for DPLL(T)
- Linear Programming solvers perform both
  - feasibility checking and
  - optimization
LEVERAGING LP

- SOISimplex added optimization to Simplex for DPLL(T)

- Linear Programming solvers perform both
  - feasibility checking and
    - optimization

- Decades of research: *fast by SMT standards*
LEVERAGING LP

- SOISimplex added optimization to Simplex for DPLL(T)
- Linear Programming solvers perform both
  - feasibility checking and
  - optimization
- Decades of research: fast by SMT standards
- Tend to use floating point (FP)
- Both Sat/Unsat answers are unsound
CAN SMT LEVERAGE LP?

- **Trusting** LP solver [YM06]
- Check each $T$-conflict used [FNORC08]
- **FORCEDPIVOT procedure** [CBdOM12, Mon09]
CAN SMT LEVERAGE LP?

- **T**rusting LP solver [YM06]
- Check each \( \bar{T} \)-conflict used [FNORC08]
- **FORCEDPIVOT** procedure [CBdOM12, Mon09]
- All use LP solver as main \( \text{QF}_L\text{RA} \) solver
Our Approach

- Call an external off-the-shelf untrusted Simplex LP solver
- Reseed the state of the exact precision solver
- Only when it is likely to help
- Implemented with GLPK
Reseeding the Simplex State

When $R$-relaxation is hard

1. Construct a FP problem from exact

   $T\chi = 0, \ l \leq \chi \leq u \implies \tilde{T}\chi = 0, \ \tilde{l} \leq \chi \leq \tilde{u}$

2. Call untrusted LP Simplex solver on $\tilde{T}, \tilde{l}, \tilde{u}$

3. Get back FP $\tilde{a}$ and $\tilde{B}$

4. Convert $(\tilde{a} : \chi \to \mathbb{F})$ into $(a^{massage} : \chi \to \mathbb{Q})$

5. RESEED$(a^{massage}, \tilde{B})$ to get a new $a$ and $T$

6. Call SMT’s trusted $\mathbb{Q}$ Simplex solver
**CONCERNS WHEN IMPORTING \( \tilde{a} \)**

\[
y = -\frac{2}{3} x + \frac{1}{3} s \quad s \geq 1
\]

\[
\begin{bmatrix}
a_x \\
a_y \\
a_s
\end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{3} \\ 1 \end{bmatrix}
\]

Suppose \( a_y = \frac{1}{3} - \epsilon \). Then \( a_s < 1 \).
**Concerns when importing \( \tilde{a} \)**

\[
\begin{align*}
    y &= -\frac{2}{3}x + \frac{1}{3}s & s \geq 1 \\
    \begin{bmatrix}
        a_x \\
        a_y \\
        a_s
    \end{bmatrix} &= \begin{bmatrix}
        0 \\
        \frac{1}{3} \\
        1
    \end{bmatrix}
\end{align*}
\]

Suppose \( a_y = \frac{1}{3} - \epsilon \). Then \( a_s < 1 \).

- Fix it with Simplex?
- Flipping coins on tightly satisfied inequalities
- Simplex generates tight solutions
Massaging Assignments
Floats to Rationals

\[
\begin{align*}
   r & \leftarrow \text{DIOAPPROX}(\tilde{a}_i, D) \\
   \text{if } \quad |r - a_i| \leq \epsilon \quad \text{then } r & \leftarrow a_i \\
   \text{if } \quad x \in X_\mathbb{Z} \quad \text{and } \quad |r - \lfloor r \rfloor| \leq \epsilon \quad \text{then } r & \leftarrow \lfloor r \rfloor \\
   \text{if } \quad r > u_i \quad \text{or } \quad |r - u_i| \leq \epsilon \quad \text{then } r & \leftarrow u_i \\
   \text{else if } \quad r < l_i \quad \text{or } \quad |r - l_i| \leq \epsilon \quad \text{then } r & \leftarrow l_i \\
   a_i^{\text{massage}} & \leftarrow r
\end{align*}
\]

Magic $D = 2^{28}$
MASSAGING ASSIGNMENTS
FLOATS TO RATIONALS

\[
\begin{align*}
    r & \leftarrow \text{DIOAPPROX}(\tilde{a}_i, D) \\
    \text{if} \quad |r - a_i| & \leq \epsilon \text{ then } r \leftarrow \lfloor r \rfloor \\
    \text{if } x \in X_\mathbb{Z} \text{ and } |r - \lfloor r \rfloor| & \leq \epsilon \text{ then } r \leftarrow \lfloor r \rfloor \\
    \text{if } r > u_i \text{ or } |r - u_i| & \leq \epsilon \text{ then } r \leftarrow u_i \\
    \text{else if } r < l_i \text{ or } |r - l_i| & \leq \epsilon \text{ then } r \leftarrow l_i \\
    a_i^{\text{massage}} & \leftarrow r
\end{align*}
\]

Magic \( D = 2^{28} \)
Reseeding Simplex \((a^{\text{massage}}, \tilde{\mathcal{B}})\)

\[\text{for all } j \in \mathcal{N} \text{ do } \text{UPDATE } x_j \text{ s.t. } a_j \leftarrow a_j^{\text{massage}}\]

repeat
  if any row conflict then return Unsat
  if \(l \leq a \leq u\) then return Sat
select \(i, k\) s.t. \(k \in \tilde{\mathcal{B}}, i \not\in \tilde{\mathcal{B}}, T_{i,k} \neq 0,\) and \(a_i > u_i (\ldots)\)
  if found \(x_i\) and \(x_k\) then
    PIVOT\((i, k)\) and \text{UPDATE}(i, \cdot) \text{ s.t. } a_i \leftarrow a_i^{\text{massage}}
  \text{else}
    \text{return Unknown} \quad \triangleright \tilde{\mathcal{B}} \text{ is not valid basis}
until \(\mathcal{N} \cap \tilde{\mathcal{B}} = \emptyset\)
return Unknown \quad \triangleright \text{Call SMT's simplex solver}
Reseeding Simplex \((a^{massage}, \tilde{B})\): Abstract

Pull in \(a^{massage}\) on \(N\)

repeat

One Simplex for DPLL(\(T\)) round

Select leaving \(x_i\) from \(\neg \tilde{B}\)

Select entering \(x_j\) from \(N \cap \tilde{B}\)

until \(N \cap \tilde{B} = \emptyset\) or fail

Call SMT’s simplex solver
**Reseeding Simplex** \((a^{\text{massage}}, \tilde{B})\): Abstract

Pull in \(a^{\text{massage}}\) on \(\mathcal{N}\)

repeat

One Simplex for DPLL(\(T\)) round
Select leaving \(x_i\) from \(\neg \tilde{B}\)
Select entering \(x_j\) from \(\mathcal{N} \cap \tilde{B}\)

until \(\mathcal{N} \cap \tilde{B} = \emptyset\) or fail
Call SMT’s simplex solver
RESEEDING SIMPLEX \( (a^{\text{massage}}, \bar{B}) \): ABSTRACT

Pull in \( a^{\text{massage}} \) on \( \mathcal{N} \)
repeat
  One Simplex for DPLL(\( T \)) round
  Select leaving \( x_i \) from \( \neg \bar{B} \)
  Select entering \( x_j \) from \( \mathcal{N} \cap \bar{B} \)
until \( \mathcal{N} \cap \bar{B} = \emptyset \) or fail
  Call SMT’s simplex solver
\textbf{MOVE} \langle \text{QF\_LRA + LP} \rangle \rightarrow \langle \text{QF\_LIRA + MIP} \rangle

- Partition variables $\mathcal{X}$ into $\mathcal{X}_R \cup \mathcal{X}_Z$
**MOVE** \(\langle QF\_LRA + LP\rangle \rightarrow \langle QF\_LIRA + MIP\rangle\)

- Partition variables \(\mathcal{X}\) into \(\mathcal{X}_R \cup \mathcal{X}_Z\)
- \(\mathbb{R}\)-relaxation treat all \(\mathcal{X}\) as \(\mathcal{X}_R\)
- \(a\) is \(\mathbb{Z}\)-compatible if \(\forall x_i \in \mathcal{X}_Z\), then \(a_i \in \mathbb{Z}\)
**MOVE** \( \langle \text{QF}_L \text{RA} + \text{LP} \rangle \rightarrow \langle \text{QF}_L \text{IRA} + \text{MIP} \rangle \)

- Partition variables \( \mathcal{X} \) into \( \mathcal{X}_R \cup \mathcal{X}_Z \)
- \( \mathbb{R} \)-relaxation treat all \( \mathcal{X} \) as \( \mathcal{X}_R \)
- \( a \) is \( \mathbb{Z} \)-compatible if \( \forall x_i \in \mathcal{X}_Z, \text{then } a_i \in \mathbb{Z} \)
- MIP is new for DPLL(\( T \))
RETURNING TO THE EXAMPLE

\[
\begin{align*}
4x - y &\leq 2 \\
4x - y &\leq 2
\end{align*}
\]

\[
\begin{bmatrix}
a_x \\
a_y
\end{bmatrix} = \begin{bmatrix}
\frac{1}{2} \\
\frac{1}{2}
\end{bmatrix}
\]

\(\mathbb{R}\)-feasible
not
\(\mathbb{Z}\)-compatible
BRANCHES AND CUTS

REFINING Z-INFEASIBLE ASSIGNMENTS

- **Branch:**

  \[
  x_i \in \mathcal{X}_\mathbb{Z} \quad \alpha \in \mathbb{R} \\
  x_i \leq \lfloor \alpha \rfloor \lor x_i \geq \lceil \alpha \rceil
  \]

- **Cut:** $\sum c_i x_j \geq d$ such that
  - $\{l_i\} \models_{\mathbb{R}\mathbb{Z}} \sum c_j x_j \geq d$
  - $\{l_i\} \not\models_{\mathbb{R}} \sum c_j x_j \geq d$
  - $\{x_j = a_j\} \not\models \sum c_j x_j \geq d$ (*)
**Branches and Cuts**

**Visually**

Branch: \( y \geq 1 \lor y \leq 0 \)

Cut: \( \{ \cdots \} \models_{\text{RZ}} x \geq 1 \)
Branch-and-cut Solvers
Most SMT solvers and many MIP solvers

1. Treat all of $\mathcal{X}$ as if they were $\mathcal{X}_R$
2. Solve this $\mathbb{R}$-relaxation
3. If $\mathbb{R}$-infeasible, return $\mathbb{R}$-conflict[s]
4. If $\mathbb{R}$-relaxation is $(\text{Sat } a)$ and $a$ is $\mathbb{Z}$-compatible, return $a$
5. Try to derive the cut $\sum c_jx_j \geq d$
6. If successful, add the cut and goto (1)
7. Branch on some $x_i \in \mathcal{X}_\mathbb{Z}$ with $a_i \notin \mathbb{Z}$
Branch-and-cut Solvers

Most SMT solvers and many MIP solvers

1. Treat all of $\mathcal{X}$ as if they were $\mathcal{X}_R$

2. Solve this $\mathbb{R}$-relaxation

3. If $\mathbb{R}$-infeasible, return $\mathbb{R}$-conflict[s]

4. If $\mathbb{R}$-relaxation is $(\text{Sat } a)$ and $a$ is $\mathbb{Z}$-compatible, return $a$

5. Try to derive the cut $\sum c_j x_j \geq d$

6. If successful, add the cut and goto (1)

7. Branch on some $x_i \in \mathcal{X}_\mathbb{Z}$ with $a_i \notin \mathbb{Z}$

Heuristically limit cuts
Branch-and-cut Solvers

Most SMT solvers and many MIP solvers

1. Treat all of $\mathcal{X}$ as if they were $\mathcal{X}_R$

2. Solve this $\mathbb{R}$-relaxation

3. If $\mathbb{R}$-infeasible, return $\mathbb{R}$-conflict[s]

4. If $\mathbb{R}$-relaxation is (Sat $a$) and $a$ is $\mathbb{Z}$-compatible, return $a$

5. Try to derive the cut $\sum c_j x_j \geq d$

6. If successful, add the cut and goto (1)

7. Branch on some $x_i \in \mathcal{X}_\mathbb{Z}$ with $a_i \notin \mathbb{Z}$

Heuristically limit cuts Only at leaves in DPLL($T$)
Possible answers from MIP?

1. \(\mathbb{R}\)-infeasible
2. \(\mathbb{R}\)-feasible and \(\mathbb{Z}\)-feasible
3. \(\mathbb{R}\)-feasible and \(\mathbb{Z}\)-infeasible
4. Failure Cases
POSSIBLE ANSWERS FROM MIP?

1. \( \mathbb{R} \)-infeasible

2. \( \mathbb{R} \)-feasible and \( \mathbb{Z} \)-feasible

3. \( \mathbb{R} \)-feasible and \( \mathbb{Z} \)-infeasible

4. Failure Cases

   Just Reseed like \( \mathbb{R} \)-feasible
   If \( a \) is \( \mathbb{Z} \)-compatible \( \implies \) done!
POSSIBLE ANSWERS FROM MIP?

1. R-infeasible

2. R-feasible and Z-feasible

3. R-feasible and Z-infeasible

4. Failure Cases

Can we leverage MIP’s reasoning?
**Infeasible Branch-and-Cut Executions**

**Proof Trees**

- Leaves are $\mathbb{R}$-infeasible
- Internal nodes are branches
  
  \[ x_i \leq \lfloor \alpha \rfloor \lor x_i \geq \lceil \alpha \rceil \quad \text{if } x_i \in \mathcal{X}_Z \]
- Nodes have cuts
  \[ \{l_i\} \models_{\mathbb{R}Z} \sum c_j x_j \geq d \]
**INFEASIBLE BRANCH-AND-CUT EXECUTIONS**

**Proof Trees**

- Leaves are $\mathbb{R}$-infeasible
- Internal nodes are branches

\[ x_i \leq \lfloor \alpha \rfloor \lor x_i \geq \lceil \alpha \rceil \quad \text{if } x_i \in \mathcal{X}_Z \]

- Nodes have cuts

\[ \{l_i\} \models_{\mathbb{R}Z} \sum c_jx_j \geq d \]

Resolution to remove branches
Replaying the MIP Execution

- Instrument GLPK to print hints about: branch, unsat leaves, and derivations of cutting planes
- Repeat “the big steps” in the SMT solver
Replaying the MIP Execution

- Instrument GLPK to print hints about: branch, unsat leaves, and derivations of cutting planes
- Repeat “the big steps” in the SMT solver
- Reconstruct the Resolution+Cutting Planes proof
Replaying the MIP Execution

- Instrument GLPK to print hints about: branch, unsat leaves, and derivations of cutting planes
- Repeat "the big steps" in the SMT solver
- Reconstruct the Resolution+Cutting Planes proof
- Success is a conflict
Replaying the MIP Execution

- Instrument GLPK to print hints about: branch, unsat leaves, and derivations of cutting planes
- Repeat “the big steps” in the SMT solver
- Reconstruct the Resolution+Cutting Planes proof
- Success is a conflict
- Any failure can be safely dropped
CUTTING PLANES

- Instantiate a cutting plane procedure from a hint
- Derivation must tightly match to get the “same” cut
- White-box knowledge and detailed hints
- Support for Gomory (easy) and MIR (hard) cuts
Table of Contents

Simplex Background

Reseeding Simplex

Replaying MIP Proofs

Empirical Results

Conclusion
SOISimplex + Reseed + Replay Results
# SMT Solver Comparison

## QF_LRA

<table>
<thead>
<tr>
<th>set</th>
<th># inst.</th>
<th># sel.</th>
<th>solved</th>
<th>time (s)</th>
<th>solved</th>
<th>time (s)</th>
<th>solved</th>
<th>time (s)</th>
<th>solved</th>
<th>time (s)</th>
<th>solved</th>
<th>time (s)</th>
<th>solved</th>
<th>time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>QF_LRA</td>
<td>634</td>
<td>634</td>
<td>627</td>
<td>6199</td>
<td>618</td>
<td>7721</td>
<td>620</td>
<td>5265</td>
<td>612</td>
<td>10814</td>
<td>615</td>
<td>5696</td>
<td></td>
<td></td>
</tr>
<tr>
<td>latendresse</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>129</td>
<td>10</td>
<td>44</td>
<td>12</td>
<td>85</td>
<td>10</td>
<td>99</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>miplib</td>
<td>42</td>
<td>37</td>
<td>30</td>
<td>1530</td>
<td>21</td>
<td>3037</td>
<td>23</td>
<td>2730</td>
<td>17</td>
<td>5682</td>
<td>18</td>
<td>2435</td>
<td></td>
<td></td>
</tr>
<tr>
<td>total</td>
<td>-</td>
<td>41</td>
<td>34</td>
<td>1534</td>
<td>25</td>
<td>3041</td>
<td>27</td>
<td>2330</td>
<td>21</td>
<td>5684</td>
<td>22</td>
<td>2436</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(AR) = Applied either RESEED or REPLAY, $K = 1000$, & SOI+MIP is CVC4 1.4 with options
### SMT Solver Comparison

#### QF_LIA $\neg$-CONJUNCTIVE

<table>
<thead>
<tr>
<th></th>
<th>SOI+MIP</th>
<th>CVC4</th>
<th>mathsat5</th>
<th>Z3</th>
<th>altergo</th>
</tr>
</thead>
<tbody>
<tr>
<td>set</td>
<td># inst.</td>
<td># sel.</td>
<td>solved</td>
<td>time (s)</td>
<td>solved</td>
</tr>
<tr>
<td>everything</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QF_LIA</td>
<td>5882</td>
<td>5882</td>
<td>5738</td>
<td>97K</td>
<td>5540</td>
</tr>
<tr>
<td>conjuncts</td>
<td>1303</td>
<td>1303</td>
<td>1249</td>
<td>11K</td>
<td>1068</td>
</tr>
<tr>
<td>(AR) $\neg$ conjunctive</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>convert</td>
<td>319</td>
<td>282</td>
<td>208</td>
<td>9646</td>
<td>193</td>
</tr>
<tr>
<td>bofill-*</td>
<td>652</td>
<td>460</td>
<td>460</td>
<td>5401</td>
<td>458</td>
</tr>
<tr>
<td>CIRC</td>
<td>51</td>
<td>11</td>
<td>11</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>calypto</td>
<td>37</td>
<td>37</td>
<td>37</td>
<td>3</td>
<td>37</td>
</tr>
<tr>
<td>nec-smt</td>
<td>2780</td>
<td>207</td>
<td>207</td>
<td>17K</td>
<td>207</td>
</tr>
<tr>
<td>wasa</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>total</td>
<td>- 998</td>
<td>924</td>
<td>32K</td>
<td>907</td>
<td>31K</td>
</tr>
</tbody>
</table>

(AR) = Applied either RESEED or REPLAY, $K = 1000$, & SOI+MIP is CVC4 1.4 with options

AltErgo is using [BCC$^+$ 12]
## SMT Solver Comparison

### QF_LIA Conjunctive

<table>
<thead>
<tr>
<th></th>
<th>SOI+MIP</th>
<th>CVC4</th>
<th>mathsat5</th>
<th>Z3</th>
<th>altergo</th>
</tr>
</thead>
<tbody>
<tr>
<td>set</td>
<td># inst.</td>
<td># sel.</td>
<td>solved</td>
<td>time (s)</td>
<td>solved</td>
</tr>
<tr>
<td>everything</td>
<td>5882</td>
<td>5882</td>
<td>5738</td>
<td>97K</td>
<td>5540</td>
</tr>
<tr>
<td>QF_LIA</td>
<td>1303</td>
<td>1303</td>
<td>1249</td>
<td>11K</td>
<td>1068</td>
</tr>
<tr>
<td>conjuncts</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### (AR) Conjunctive

<table>
<thead>
<tr>
<th></th>
<th>SOI+MIP</th>
<th>CVC4</th>
<th>mathsat5</th>
<th>Z3</th>
<th>altergo</th>
</tr>
</thead>
<tbody>
<tr>
<td>dillig</td>
<td>233</td>
<td>189</td>
<td>189</td>
<td>49</td>
<td></td>
</tr>
<tr>
<td>milib2003</td>
<td>16</td>
<td>8</td>
<td>4</td>
<td>307</td>
<td>4</td>
</tr>
<tr>
<td>prime-cone</td>
<td>37</td>
<td>37</td>
<td>37</td>
<td>2</td>
<td>37</td>
</tr>
<tr>
<td>slacks</td>
<td>233</td>
<td>188</td>
<td>166</td>
<td>61</td>
<td>93</td>
</tr>
<tr>
<td>CAV_2009</td>
<td>591</td>
<td>424</td>
<td>424</td>
<td>69</td>
<td>346</td>
</tr>
<tr>
<td>cut_lem.</td>
<td>93</td>
<td>74</td>
<td>62</td>
<td>9581</td>
<td>64</td>
</tr>
<tr>
<td>total</td>
<td>-</td>
<td>920</td>
<td>882</td>
<td>10K</td>
<td>701</td>
</tr>
</tbody>
</table>

(AR) = Applied either RESEED or REPLAY, K = 1000, & SOI+MIP is CVC4 1.4 with options
### Comparison with Conjunctive Solvers

<table>
<thead>
<tr>
<th>Set</th>
<th># Inst.</th>
<th># Sel.</th>
<th>SOI+MIP</th>
<th>cutsat</th>
<th>scip</th>
<th>glpk</th>
</tr>
</thead>
<tbody>
<tr>
<td>conjuncts</td>
<td>1303</td>
<td>1303</td>
<td>1249</td>
<td>1018</td>
<td>1255</td>
<td>1173</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>11130</td>
<td>35330</td>
<td>7164</td>
<td>8895</td>
</tr>
<tr>
<td>(AR) conjunctive</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dillig</td>
<td>233</td>
<td>189</td>
<td>189</td>
<td>166</td>
<td>189</td>
<td>189</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>49</td>
<td>5840</td>
<td>42</td>
<td>3</td>
</tr>
<tr>
<td>miplib2003</td>
<td>16</td>
<td>8</td>
<td>4</td>
<td>307</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>307</td>
<td>146</td>
<td>17</td>
<td>295</td>
</tr>
<tr>
<td>prime-cone</td>
<td>37</td>
<td>37</td>
<td>37</td>
<td>4</td>
<td>37</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>slacks</td>
<td>233</td>
<td>188</td>
<td>166</td>
<td>61</td>
<td>161</td>
<td>101</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>61</td>
<td>6324</td>
<td>2361</td>
<td>11</td>
</tr>
<tr>
<td>CAV_2009</td>
<td>591</td>
<td>424</td>
<td>424</td>
<td>69</td>
<td>17015</td>
<td>105</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>424</td>
<td>17015</td>
<td>105</td>
<td>6</td>
</tr>
<tr>
<td>cut_lemmas</td>
<td>93</td>
<td>74</td>
<td>62</td>
<td>9581</td>
<td>15</td>
<td>72</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9581</td>
<td>1887</td>
<td>1757</td>
<td>760</td>
</tr>
<tr>
<td>total</td>
<td>-</td>
<td>920</td>
<td>882</td>
<td>10069</td>
<td>890</td>
<td>828</td>
</tr>
</tbody>
</table>

(AR) = Applied either RESEED or REPLAY, \( K = 1000 \), & SOI+MIP is CVC4 1.4 with options

cutsat is using [JdM11]
**QF_LIA Reseed and Replay Success Rates**

<table>
<thead>
<tr>
<th>set</th>
<th># inst.</th>
<th>solve int calls</th>
<th>Reseed attempts</th>
<th>successes</th>
<th>Replay attempts</th>
<th>successes</th>
</tr>
</thead>
<tbody>
<tr>
<td>QF_LIA</td>
<td>1806</td>
<td>3873</td>
<td>2559</td>
<td>1058</td>
<td>652</td>
<td>425</td>
</tr>
<tr>
<td>convert</td>
<td>208</td>
<td>2130</td>
<td>1356</td>
<td>1</td>
<td>178</td>
<td>3</td>
</tr>
<tr>
<td>bofill-scheduling</td>
<td>460</td>
<td>254</td>
<td>245</td>
<td>245</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CIRC</td>
<td>11</td>
<td>85</td>
<td>6</td>
<td>5</td>
<td>79</td>
<td>77</td>
</tr>
<tr>
<td>calypto</td>
<td>37</td>
<td>375</td>
<td>77</td>
<td>23</td>
<td>293</td>
<td>278</td>
</tr>
<tr>
<td>wisa</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>dillig</td>
<td>189</td>
<td>228</td>
<td>225</td>
<td>185</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>miplib2003</td>
<td>4</td>
<td>10</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>prime-cone</td>
<td>37</td>
<td>37</td>
<td>19</td>
<td>19</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>slacks</td>
<td>166</td>
<td>195</td>
<td>168</td>
<td>162</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>CAV_2009</td>
<td>424</td>
<td>469</td>
<td>459</td>
<td>414</td>
<td>8</td>
<td>7</td>
</tr>
<tr>
<td>cut_lemmas</td>
<td>62</td>
<td>89</td>
<td>0</td>
<td>0</td>
<td>65</td>
<td>33</td>
</tr>
</tbody>
</table>

Only includes solved instances
Table of Contents

Simplex Background

Reseeding Simplex

Replaying MIP Proofs

Empirical Results

Conclusion
FUTURE WORK

- Optimization Modulo Theories
- Logging and replaying FP Farkas’s lemma [NS04]
- $k$-precision FP Simplex solver for SMT [CKSW13]
**Replay & Reseed Summary**

- Integrated a floating point LP/MIP solver (GLPK) (Backup. Not the main engine!)
**Replay & Reseed Summary**

- Integrated a floating point LP/MIP solver (GLPK) (Backup. Not the main engine!)
- Reseeding Simplex
  - Helps find models and $\mathbb{R}$-relaxation conflicts
  - 1 week to implement [*]
Replay & Reseed Summary

▶ Integrated a floating point LP/MIP solver (GLPK) (Backup. Not the main engine!)

▶ Reseeding Simplex
  ▶ Helps find models and ℝ-relaxation conflicts
    ▶ 1 week to implement [*]

▶ Replaying MIP conflicts (significantly more effort) MIP must be white-box and must log proofs!
**Replay & Reseed Summary**

- Integrated a floating point LP/MIP solver (GLPK) (Backup. Not the main engine!)
- Reseeding Simplex
  - Helps find models and $\mathbb{R}$-relaxation conflicts
  - 1 week to implement [*]
- Replaying MIP conflicts (significantly more effort)
  MIP must be white-box and must log proofs!
- Overall helpful, but there are limitations

Thank you for your attention!
Replay & Reseed Summary

- Integrated a floating point LP/MIP solver (GLPK) (Backup. Not the main engine!)

- Reseeding Simplex
  - Helps find models and $\mathbb{R}$-relaxation conflicts
  - 1 week to implement [*]

- Replaying MIP conflicts (significantly more effort)
  MIP must be white-box and must log proofs!

- Overall helpful, but there are limitations

Thank you for your attention!
WHAT HAPPENED ON THE CONVERT FAMILY?

- MIP solver is wrong about feasibility too often
- Variables are in bounds up to a “dual gap”
  - Intuitively: Let $a_i$ violate $u_i$ by a little where little is scaled by the size of the numbers
- Numerically stability of floating points
- Gap is too large for $\mathbb{QF}_{\mathbb{LIA}}$ bit-extracts for $\sim m + n > 40$
  
  $$x = 2^m y + z \land z \in [0, 2^m), y \in [0, 2^n), x \in [0, 2^{m+n})$$

- Decreasing the maximum gap leads $\implies$ cycling
- Need bigger floating point numbers or more pre-processing
REFERENCES I


REFERENCES II


References III


REFERENCES IV


APPENDIX
RESOLUTION PHASE

The proof reconstruction phase uses the following heuristics:

- All up-branch conflicts are resolved with all down-branch conflicts (DP-style)
- Performed eager subsumption checking
  Pays for itself by keeping the set of conflicts small
- Non-chronological backtracks when possible
  (One branch has a conflict not involving its branch literal)