Improving Conflict Analysis in MIP Solvers by Pseudo-Boolean Reasoning

Gioni Mexi  Timo Berthold  Ambros Gleixner  Jakob Nordström

ZIB
ZUSE INSTITUTE BERLIN

14th Pragmatics of SAT International Workshop
July 4, 2023, Alghero, Italy
A MIP is a problem of the form:

\[
\begin{align*}
\min_{x \in \mathbb{R}^n} \quad & c^T x \\
\text{s.t.} \quad & Ax \geq b \\
& l \leq x \leq u \\
& x \in \mathbb{Z}^I \times \mathbb{R}^C.
\end{align*}
\] (1)

\(A \in \mathbb{R}^{m \times n}, \ b \in \mathbb{R}^m, \ l, u \in \mathbb{R}^n\)
Mixed-Integer Program (MIP)

A MIP is a problem of the form:

\[
\begin{align*}
\min_{x \in \mathbb{R}^n} & \quad c^T x \\
\text{s.t.} & \quad Ax \geq b \\
& \quad l \leq x \leq u \\
& \quad x \in \mathbb{Z}^I \times \mathbb{R}^C.
\end{align*}
\]  

(1)

\[A \in \mathbb{R}^{m \times n}, \ b \in \mathbb{R}^m, \ l, u \in \mathbb{R}^n\]

- 0-1 Integer Program (IP):
  \[\mathcal{I} = [n], \ l_i = 0, \ u_i = 1 \forall i \in \mathcal{I}\]

- Mixed 0-1 IP:
  \[\mathcal{I} \subset [n], \ l_i = 0, \ u_i = 1 \forall i \in \mathcal{I}\]

- Linear Programming (LP) Relaxation of (1):
  \[\mathbb{Z}^\mathcal{I} \leadsto \mathbb{R}^\mathcal{I}\]
Motivation

- Current conflict analysis in MIP:
  - as in CDCL SAT solvers (e.g., [Marques-Silva and Sakallah, 1996])
  - operates on clauses extracted from the linear constraints
Motivation

- Current conflict analysis in MIP:
  - as in CDCL SAT solvers (e.g., [Marques-Silva and Sakallah, 1996])
  - operates on clauses extracted from the linear constraints

- Pseudo-Boolean (PB) solvers [Chai and Kuehlmann, 2005]
  - extend conflict analysis to operate directly on linear constraints.
Motivation

- Current conflict analysis in MIP:
  - as in CDCL SAT solvers (e.g., [Marques-Silva and Sakallah, 1996])
  - operates on clauses extracted from the linear constraints
- Pseudo-Boolean (PB) solvers [Chai and Kuehlmann, 2005]
  - extend conflict analysis to operate directly on linear constraints.

Can MIP benefit from PB conflict analysis?
This talk:
- Integration of PB conflict analysis for 0–1 integer programs into MIP
- Extend the algorithm by using cuts from the MIP literature
- Implement the algorithm in the MIP solver SCIP
Table of Contents

Conflict Analysis in MIP

Pseudo Boolean Conflict Analysis

Computational Results

Conclusion
Table of Contents

Conflict Analysis in MIP

Pseudo Boolean Conflict Analysis

Computational Results

Conclusion
Goal: When branch-and-bound reaches an infeasible subproblem, analyze the infeasibility to
- extract a shorter explanation
**Goal:** When branch-and-bound reaches an infeasible subproblem, analyze the infeasibility to

- extract a shorter explanation
- that prunes other parts of the tree
Goal: When branch-and-bound reaches an infeasible subproblem, analyze the infeasibility to

- extract a shorter explanation
- that prunes other parts of the tree
- also in backtracking
Conflict Analysis in MIP

Goal: When branch-and-bound reaches an infeasible subproblem, analyze the infeasibility to

- extract a shorter explanation
- that prunes other parts of the tree
- also in backtracking

Reasons for infeasibility:

- Propagation
- LP relaxation
- Bound exceeding LP
Conflict Graph Analysis [Achterberg, 2007]

Similar to [Marques-Silva and Sakallah, 1996]

- The sequence of assignments and implications is captured by a directed implication graph
Conflict Graph Analysis [Achterberg, 2007]

Similar to [Marques-Silva and Sakallah, 1996]

- The sequence of assignments and implications is captured by a directed implication graph
- Each cut that separates the decision nodes from $\lambda$ yields a conflict (FUFP, . . . )
Conflict Graph Analysis [Achterberg, 2007]

Similar to [Marques-Silva and Sakallah, 1996]

- The sequence of assignments and implications is captured by a directed implication graph
- Each cut that separates the decision nodes from $\lambda$ yields a conflict (FUIP, …)

Variable assignment $\{x_{15}, x_{18}\}$ responsible for the conflict
Resolve: $x_{18}$
Conflict Graph Analysis [Achterberg, 2007]

Similar to [Marques-Silva and Sakallah, 1996]

- The sequence of assignments and implications is captured by a directed implication graph
- Each cut that separates the decision nodes from $\lambda$ yields a conflict (FUIP, ...)

Variable assignment $\{\overline{x}_{15}, x_{17}\}$ responsible for the conflict
Resolve: $\overline{x}_{15}$
Conflict Graph Analysis [Achterberg, 2007]

Similar to [Marques-Silva and Sakallah, 1996]

- The sequence of assignments and implications is captured by a directed implication graph
- Each cut that separates the decision nodes from $\lambda$ yields a conflict (FUIP, ...)

Variable assignment $\{x_{14}, x_{17}\}$ responsible for the conflict

Resolve: $x_{17}$
Similar to [Marques-Silva and Sakallah, 1996]

- The sequence of assignments and implications is captured by a directed implication graph.
- Each cut that separates the decision nodes from $\lambda$ yields a conflict (FUIP, ...)

Variable assignment $\{x_{13}, x_{14}, x_{16}\}$ responsible for the conflict.
Conflict Graph Analysis [Achterberg, 2007]

Similar to [Marques-Silva and Sakallah, 1996]

- The sequence of assignments and implications is captured by a directed implication graph
- Each cut that separates the decision nodes from $\lambda$ yields a conflict (FUIP, ...)

\[
\begin{align*}
x_1 & \quad \bar{x}_2 & \quad x_3 & \quad x_4 & \quad \bar{x}_5 \\
x_6 & \quad \bar{x}_7 & \quad x_8 & \quad x_9 & \quad x_{10} \\
\bar{x}_{12} & \quad x_{13} & & & \quad x_{14} \quad \bar{x}_{15} \\
& \quad x_{16} & \quad x_{17} & \quad x_{18} & \quad \lambda
\end{align*}
\]

Learned clause: $\bar{x}_{13} \lor \bar{x}_{14} \lor \bar{x}_{16}$

$\Rightarrow (1 - x_{13}) + (1 - x_{14}) + (1 - x_{16}) \geq 1$
Conflict Graph Analysis in MIP [Achterberg, 2007]

- Technical issues: non-binary variables
  - Conflict graph: bound changes instead of variable assignments
  - Conflict clause $\rightarrow$ conflict constraint (bound disjunction)
    - e.g., conflict constraint $(x_1 \geq 1) \lor (x_3 \leq 0) \lor (x_7 \leq 11)$

- What if the reason for infeasibility is the LP relaxation?
  - Find “smaller” subset of bound changes that leads to the infeasible LP
  - Start conflict graph analysis
  - (Alternative: use LP duality theory [Witzig et al., 2019])
Table of Contents

Conflict Analysis in MIP

Pseudo Boolean Conflict Analysis

Computational Results

Conclusion
A pseudo-Boolean constraint is a 0–1 integer linear inequality

\[ \sum_{i \in \mathcal{N}} a_i \ell_i \geq b, \]

\( a_i \in \mathbb{Z}_{\geq 0} \) for all \( i \in \mathcal{N} \), \( b \in \mathbb{Z}_{\geq 0} \)

\( \ell_i \) denote literals, which can be either \( x_i \) or its negation \( \overline{x}_i = 1 - x_i \).

A partial assignment \( \rho \), map from literals to 0 (falsified) or 1 (true)
A pseudo-Boolean constraint is a 0–1 integer linear inequality

$$\sum_{i \in \mathcal{N}} a_i \ell_i \geq b,$$

$a_i \in \mathbb{Z}_{\geq 0}$ for all $i \in \mathcal{N}$, $b \in \mathbb{Z}_{\geq 0}$

$l_i$ denote literals, which can be either $x_i$ or its negation $\overline{x}_i = 1 - x_i$.

A partial assignment $\rho$, map from literals to 0 (falsified) or 1 (true)

The slack of a PB constraint under a partial assignment $\rho$: is defined as

$$\text{slack}(C, \rho) := \sum_{\{i \in \mathcal{N} : \rho(i) \neq 0\}} a_i - b.$$

If the slack is negative $\implies$ conflict
A pseudo-Boolean constraint is a 0–1 integer linear inequality

\[ \sum_{i \in \mathcal{N}} a_i \ell_i \geq b, \]

\( a_i \in \mathbb{Z}_{\geq 0} \) for all \( i \in \mathcal{N} \), \( b \in \mathbb{Z}_{\geq 0} \)

\( \ell_i \) denote literals, which can be either \( x_i \) or its negation \( \bar{x}_i = 1 - x_i \).

A partial assignment \( \rho \), map from literals to 0 (falsified) or 1 (true)

The slack of a PB constraint under a partial assignment \( \rho \): is defined as

\[ \text{slack}(C, \rho) := \sum_{\{i \in \mathcal{N} : \rho(i) \neq 0\}} a_i - b. \]

If the slack is negative \( \implies \) conflict

Generalized resolution rule: [Hooker, 1988]

\( \leadsto \) linear combination of two constraints that cancels a variable
Example Generalized Resolution

\[ C_{\text{reason}} : \ x_1 + x_2 + 2x_3 \geq 2 \]
\[ C_{\text{confl}} : \ x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3 \]
\[ \rho = \{ x_1 \overset{\text{dec.}}{=} 0, \ x_3 \overset{C_{\text{reason}}}{=} 1 \} \Rightarrow \text{Conflict with } C_{\text{confl}} \]
Example Generalized Resolution

\[ C_{\text{reason}} : \ x_1 + x_2 + 2x_3 \geq 2 \]
\[ C_{\text{confl}} : \ x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3 \]
\[ \rho = \left\{ x_1 = 0, \ x_3 = 1 \right\} \Rightarrow \text{Conflict with } C_{\text{confl}} \]

Resolving on \( x_3 \):

\[
\text{resolve } \{x_3\} \quad \frac{x_1 + x_2 + 2x_3 \geq 2}{2x_1 + x_2 + x_4 + x_5 \geq 3} \quad \frac{x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3}{2x_1 + x_2 + x_4 + x_5 \geq 3}
\]

Does not explain infeasibility since it has non-negative slack
Example Generalized Resolution

\[
\begin{align*}
C_{\text{reason}} : & \quad x_1 + x_2 + 2x_3 \geq 2 \\
C_{\text{confl}} : & \quad x_1 + 2x_3 + x_4 + x_5 \geq 3 \\
\rho = \left\{ x_1 \overset{\text{dec.}}{=} 0, x_3 \overset{\text{reason}}{=} 1 \right\} \Rightarrow \text{Conflict with } C_{\text{confl}}
\end{align*}
\]

Resolving on \( x_3 \):

\[
\text{resolve} \{ x_3 \} \quad \frac{x_1 + x_2 + 2x_3 \geq 2}{2x_1 + x_2 + x_4 + x_5 \geq 3}
\]

Does not explain infeasibility since it has non-negative slack

▷ Issue: the reason does not propagate tightly over the reals
▷ Can we make the reason constraint propagate tightly?
Techniques used to reduce the slack of the reason

- Weakening non falsified literals $\ell_j$:

  $$\text{weaken}(\sum_{i \in \mathcal{N}} a_i \ell_i \geq b, \ell_j) = \sum_{i \neq j \in \mathcal{N}} a_i \ell_i \geq b - a_j$$

- Cut Rules:
  - Saturation (Coef. Tightening):

    $$\text{saturate}(\sum_{i \in \mathcal{N}} a_i \ell_i \geq b) = \sum_{i \in \mathcal{N}} \min\{a_i, b\} \ell_i \geq b$$

  - Division (Chvatal-Gomory) by $d > 0$:

    $$\text{divide}(\sum_{i \in \mathcal{N}} a_i \ell_i \geq b, d) = \sum_{i \in \mathcal{N}} \left\lceil \frac{a_i}{d} \right\rceil \ell_i \geq \left\lceil \frac{b}{d} \right\rceil$$
Example Generalized Resolution

\[
C_{\text{reason}} : \quad x_1 + x_2 + 2x_3 \geq 2 \\
C_{\text{confl}} : \quad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3 \\
\rho = \{ x_1 \overset{\text{dec.}}{=} 0, x_3 \overset{\text{reason}}{=} 1 \} \Rightarrow \text{Conflict with } C_{\text{confl}}
\]
Example Generalized Resolution

\[ C_{\text{reason}} : \quad x_1 + x_2 + 2x_3 \geq 2 \]
\[ C_{\text{confl}} : \quad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3 \]
\[
\rho = \left\{ x_1 \overset{\text{dec.}}{=} 0, \quad x_3 \overset{\text{reason}}{=} 1 \right\} \Rightarrow \text{Conflict with } C_{\text{confl}}
\]

Weaken non-falsified variables in \( C_{\text{reason}} \) other than \( x_3 \):

weaken \( \{x_2\} \)
\[
\frac{x_1 + x_2 + 2x_3 \geq 2}{x_1 + 2x_3 \geq 1}
\]
saturate
\[
\frac{x_1 + 2x_3 \geq 1}{x_1 + x_3 \geq 1}
\]
resolve \( \{x_3\} \)
\[
\frac{x_1 + x_3 \geq 1}{3x_1 + x_4 + x_5 \geq 3}
\]
\[
\frac{x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3}{3x_1 + x_4 + x_5 \geq 3}
\]

\( \Rightarrow \) Conflict with \( C_{\text{confl}} \)
Example Generalized Resolution

Weaken non-falsified variables in $C_{\text{reason}}$ other than $x_3$:

Weaken $\{x_2\}$
\[
\frac{x_1 + x_2 + 2x_3 \geq 2}{x_1 + 2x_3 \geq 1}
\]

Saturate
\[
\frac{x_1 + 2x_3 \geq 1}{x_1 + x_3 \geq 1}
\]

Resolve $\{x_3\}$
\[
\frac{x_1 + x_3 \geq 1}{3x_1 + x_4 + x_5 \geq 3}
\]

Now the slack is negative $\Rightarrow$ conflict invariant is preserved.
Conflict Analysis Algorithm

First introduced in [Chai and Kuehlmann, 2005]

**Algorithm:** Generalized Resolution Conflict Analysis

**Input:** conflict constraint $C_{\text{confl}}$, falsifying partial assignment $\rho$

**Output:** learned conflict constraint $C_{\text{learn}}$

1. $C_{\text{learn}} \leftarrow C_{\text{confl}}$
2. while $C_{\text{learn}}$ not asserting and $C_{\text{learn}} \neq \perp$ do
3.   $\ell_r \leftarrow$ literal last assigned on $\rho$
4.   if $\ell_r$ propagated and $\overline{\ell_r}$ occurs in $C_{\text{learn}}$ then
5.     $C_{\text{reason}} \leftarrow \text{reason}(\ell_r, \rho)$
6.     $C_{\text{reason}} \leftarrow \text{reduce}(C_{\text{reason}}, C_{\text{learn}}, \ell_r, \rho)$
7.     $C_{\text{learn}} \leftarrow \text{resolve}(C_{\text{learn}}, C_{\text{reason}}, \ell_r)$
8.     $\rho \leftarrow \rho \setminus \{\ell_r\}$
9. return $C_{\text{learn}}$
Conflict Analysis Algorithm

- First introduced in [Chai and Kuehlmann, 2005]

**Algorithm:** Generalized Resolution Conflict Analysis

**Input**: conflict constraint $C_{confl}$, falsifying partial assignment $\rho$

**Output**: learned conflict constraint $C_{learn}$

```plaintext
1. $C_{learn} \leftarrow C_{confl}$
2. **while** $C_{learn}$ not asserting and $C_{learn} \neq \bot$ **do**
   3. $\ell_r \leftarrow$ literal last assigned on $\rho$
   4. **if** $\ell_r$ propagated and $\bar{\ell}_r$ occurs in $C_{learn}$ **then**
      5. $C_{reason} \leftarrow$ reason($\ell_r, \rho$)
      6. $C_{reason} \leftarrow$ reduce($C_{reason}, C_{learn}, \ell_r, \rho$)
      7. $C_{learn} \leftarrow$ resolve($C_{learn}, C_{reason}, \ell_r$)
      8. $\rho \leftarrow \rho \setminus \{\ell_r\}$
3. **return** $C_{learn}$
```

- Sat4j [Le Berre and Parrain, 2010]
- RoundingSAT [Elffers and Nordström, 2018]
Reduction Algorithm

▶ Goal: Make the reason constraint propagate tightly
⇝ Linear combination with $C_{\text{confl}}$ remains infeasible (our invariant holds)

**Algorithm:** Saturation-based Reduction Algorithm

**Input:**
- conflict constraint $C_{\text{confl}}$
- reason constraint $C_{\text{reason}}$
- literal to resolve $\ell_r$
- partial assignment $\rho$

**Output:**
- reduced reason $C_{\text{reason}}$

1. **while** $\text{slack}((\text{resolve}(C_{\text{reason}}, C_{\text{confl}}, \ell_r)), \rho) \geq 0$ **do**

2. \[ l_j \leftarrow \text{non falsified literal in } C_{\text{reason}} \setminus \{\ell_r\} \]

3. \[ C_{\text{reason}} \leftarrow \text{weaken}(C_{\text{reason}}, l_j) \]

4. \[ C_{\text{reason}} \leftarrow \text{saturate}(C_{\text{reason}}) \]

5. **return** $C_{\text{reason}}$

Division (CG) can be used instead of saturation [Elffers and Nordström, 2018]

Incomparable in terms of strength [Gocht et al., 2019]
Reduction Algorithm

Goal: Make the reason constraint propagate tightly
\[ \rightsquigarrow \text{Linear combination with } C_{\text{confl}} \text{ remains infeasible (our invariant holds)} \]

Algorithm: Saturation-based Reduction Algorithm

\textbf{Input} : conflict constraint \( C_{\text{confl}} \), reason constraint \( C_{\text{reason}} \),
literal to resolve \( \ell_r \), partial assignment \( \rho \)

\textbf{Output} : reduced reason \( C_{\text{reason}} \)

1. \textbf{while} \( \text{slack}((\text{resolve}(C_{\text{reason}}, C_{\text{confl}}, \ell_r)), \rho) \geq 0 \) \textbf{do}
   2. \( \ell_j \leftarrow \) non falsified literal in \( C_{\text{reason}} \backslash \{\ell_r\} \)
   3. \( C_{\text{reason}} \leftarrow \text{weaken}(C_{\text{reason}}, \ell_j) \)
   4. \( C_{\text{reason}} \leftarrow \text{saturate}(C_{\text{reason}}) \)

5. \textbf{return} \( C_{\text{reason}} \)

▶ Division (CG) can be used instead of saturation [Elffers and Nordström, 2018]
▶ Incomparable in terms of strength [Gocht et al., 2019]
Mixed Integer Rounding (MIR)

Introduced in [Marchand and Wolsey, 2001]

Elementary mixed integer set:

\[ X := \{ (x, y) \in \mathbb{Z} \times \mathbb{R} : \]
\[ x \leq b + y \quad (I) \]
\[ y \geq 0 \quad (II) \} \]
Mixed Integer Rounding (MIR)

Introduced in [Marchand and Wolsey, 2001]

Elementary mixed integer set:

\[ X := \{ (x, y) \in \mathbb{Z} \times \mathbb{R} : x \leq b + y \quad (I) \quad y \geq 0 \quad (II) \} \]

Inequalities (I) and (II) do not suffice to describe \( \text{conv}(X) \).
Mixed Integer Rounding (MIR)

Disjunctive argument:

- If an inequality is valid for \( X^1 \) and for \( X^2 \), it is also valid for \( X^1 \cup X^2 \).

\[
\begin{align*}
X^1 & : x \geq \lceil b \rceil \\
X^2 & : x \leq \lfloor b \rfloor
\end{align*}
\]
Mixed Integer Rounding (MIR)

Disjunctive argument:
▶ If an inequality is valid for $X^1$ and for $X^2$, it is also valid for $X^1 \cup X^2$.

Here:
▶ $X^1$: Add $x \geq \lceil b \rceil$ (III)
▶ $X^2$: Add $x \leq \lfloor b \rfloor$ (IV)
Mixed Integer Rounding (MIR)

Inequality valid for $X^1$ and for $X^2$:

$$x \leq \lfloor b \rfloor + \frac{1}{1-f_b} y$$

\[ (I)+f_b(III) \text{ and } (II)+(1-f_b)(IV) \]
Mixed Integer Rounding (MIR)

Inequality valid for $X^1 \cup X^2 = X$:

$$x \leq \lfloor b \rfloor + \frac{1}{1-f_b} y$$

MIR inequality

Inequality (1) and (2) do not suffice to describe $\text{conv}(X)$. 
Normalized MIR Cut

Let $C : \sum_{i \in \mathcal{N}} a_i \ell_i \geq b$. The **Mixed Integer Rounding (MIR) Cut** of $C$ with divisor $d \in \mathbb{Z}_{>0}$ is given by the constraint

$$
\sum_{i \in l_1} \left\lceil \frac{a_i}{d} \right\rceil \ell_i + \sum_{i \in l_2} \left( \left\lfloor \frac{a_i}{d} \right\rfloor + \frac{f(a_i/d)}{f(b/d)} \right) \ell_i \geq \left\lceil \frac{b}{d} \right\rceil,
$$

(1)

where

$$
l_1 = \{ i \in \mathcal{N} : f(a_i/d) \geq f(b/d) \text{ or } f(a_i/d) \in \mathbb{Z} \},
$$

$$
l_2 = \{ i' \in \mathcal{N} : f(a_{i'}/d) < f(b/d) \text{ and } f(a_{i'}/d) \notin \mathbb{Z} \},
$$

and $f(\cdot) = \cdot - \lfloor \cdot \rfloor$. To obtain a normalized version of the MIR cut, we multiply both sides of the constraint by $(b \mod d)$. 

Gioni Mexi et al Improving Conflict Analysis in MIP Solvers by Pseudo-Boolean Reasoning 15 / 21
MIR Reduction

For a partial assignment $\rho$ and $C_{\text{reason}} : \sum_{i \in \mathcal{N}} a_i \ell_i \geq b$ propagating a literal $\ell_r$ to 1:

1. weakening all non-falsified literal not divisible by $a_r$, and
2. Applying MIR on $C_{\text{reason}}$ with divisor $d = a_r$
   $\implies$ slack 0.

Remarks:

▶ MIR-based reduction implies Division-based reduction, e.g.,

Let $\rho = \{x_1 = 0, x_2 = 0, x_3 = 1\}$ and $C_{\text{reason}} : 2x_1 + 6x_2 + 10x_3 \geq 8$:

1. Division-based reduction (divide by 10 and apply ceiling):
   $\implies x_1 + x_2 + x_3 \geq 1$

2. MIR-based reduction:
   $\implies 0.2x_1 + 0.6x_2 + x_3 \geq 1$

▶ MIR/Division-based reduction is incomparable to Saturation-based reduction
MIR Reduction

For a partial assignment $\rho$ and $C_{\text{reason}} : \sum_{i \in \mathcal{N}} a_i \ell_i \geq b$ propagating a literal $\ell_r$ to 1:

1. weakening all non-falsified literal not divisible by $a_r$, and
2. Applying MIR on $C_{\text{reason}}$ with divisor $d = a_r$
   \[ \Rightarrow \text{slack 0}. \]

Remarks:

- MIR-based reduction implies Division-based reduction, e.g.,

Let $\rho = \{x_1 = 0, x_2 = 0, x_3 = 1\}$ and $C_{\text{reason}} : 2x_1 + 6x_2 + 10x_3 \geq 8$:

1. Division-based reduction (divide by 10 and apply ceiling):
   \[ \Rightarrow x_1 + x_2 + x_3 \geq 1 \]
2. MIR-based reduction:
   \[ \Rightarrow \frac{0.2}{0.8} x_1 + \frac{0.6}{0.8} x_2 + x_3 \geq 1 \]

Remarks:

- MIR/Division-based reduction is incomparable to Saturation-based reduction
MIR Reduction

For a partial assignment $\rho$ and $C_{\text{reason}}: \sum_{i \in N} a_i \ell_i \geq b$ propagating a literal $\ell_r$ to 1:

1. weakening all non-falsified literal not divisible by $a_r$, and
2. Applying MIR on $C_{\text{reason}}$ with divisor $d = a_r$
   $\Rightarrow$ slack 0.

Remarks:

▶ MIR-based reduction implies Division-based reduction, e.g.,

Let $\rho = \{x_1 = 0, x_2 = 0, x_3 = 1\}$ and $C_{\text{reason}}: 2x_1 + 6x_2 + 10x_3 \geq 8$:

1. Division-based reduction (divide by 10 and apply ceiling):
   $\Rightarrow x_1 + x_2 + x_3 \geq 1$

2. MIR-based reduction:
   $\Rightarrow \frac{0.2}{0.8} x_1 + \frac{0.6}{0.8} x_2 + x_3 \geq 1$

▶ MIR/Division-based reduction is incomparable to Saturation-based reduction
Table of Contents

Conflict Analysis in MIP

Pseudo Boolean Conflict Analysis

Computational Results

Conclusion
Experimental Setup

Some implementation details:

- PB conflict analysis can be generalized for constraints with real coefficients. However, floating-point arithmetic may cause numerical issues. To mitigate the risks:
  - Stop if the coefficients of the constraints span too many orders of magnitude
  - Remove variables with too small coefficients
Experimental Setup

Some implementation details:

- PB conflict analysis can be generalized for constraints with real coefficients. However, floating-point arithmetic may cause numerical issues. To mitigate the risks:
  - Stop if the coefficients of the constraints span too many orders of magnitude
  - Remove variables with too small coefficients

Setup:

- Implemented all techniques in the open source MIP solver SCIP.
- Performance variability is a key concern in MIP literature. Use a large and diverse test set of instances and multiple seeds.
- 195 pure 0-1 models from the MIPLIB2017 collection × 5 seeds.
## Computational Results

<table>
<thead>
<tr>
<th>Settings</th>
<th>solved</th>
<th>time(s)</th>
<th># nodes</th>
<th>time quot</th>
<th>nodes quot</th>
</tr>
</thead>
<tbody>
<tr>
<td>all(975)</td>
<td>Graph</td>
<td>405</td>
<td>603.55</td>
<td>682.31</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>419</td>
<td>601.4</td>
<td>683.48</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>420</td>
<td>599.37</td>
<td>677.04</td>
<td>0.99</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>418</td>
<td>599.76</td>
<td>691.81</td>
<td>0.99</td>
</tr>
<tr>
<td>affected(286)</td>
<td>Graph</td>
<td>263</td>
<td>121.21</td>
<td>753.96</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>277</td>
<td>117.82</td>
<td>682.43</td>
<td>0.97</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>278</td>
<td>116.91</td>
<td>675.11</td>
<td>0.96</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>276</td>
<td>116.71</td>
<td>710.72</td>
<td>0.96</td>
</tr>
<tr>
<td>affected and all-optimal(254)</td>
<td>Graph</td>
<td>254</td>
<td>81.47</td>
<td>507.23</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>254</td>
<td>82.87</td>
<td>482.61</td>
<td>1.02</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>254</td>
<td>81.43</td>
<td>468.57</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>254</td>
<td>80.21</td>
<td>485.52</td>
<td>0.98</td>
</tr>
</tbody>
</table>

"MIR" leads always to smaller search trees

"MIR" vs "No Conflict Analysis" on 279 affected instances:
+ 25 solved, 27% faster, 37% smaller trees

Still requires further investigation: weakening, choose best cut, . . .
### Computational Results

<table>
<thead>
<tr>
<th>Settings</th>
<th>solved</th>
<th>time(s)</th>
<th># nodes</th>
<th>time quot</th>
<th>nodes quot</th>
</tr>
</thead>
<tbody>
<tr>
<td>all(975)</td>
<td>Graph</td>
<td>405</td>
<td>603.55</td>
<td>682.31</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>419</td>
<td>601.4</td>
<td>683.48</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>420</td>
<td>599.37</td>
<td>677.04</td>
<td>0.99</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>418</td>
<td>599.76</td>
<td>691.81</td>
<td>0.99</td>
</tr>
<tr>
<td>affected(286)</td>
<td>Graph</td>
<td>263</td>
<td>121.21</td>
<td>753.96</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>277</td>
<td>117.82</td>
<td>682.43</td>
<td>0.97</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>278</td>
<td>116.91</td>
<td>675.11</td>
<td>0.96</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>276</td>
<td>116.71</td>
<td>710.72</td>
<td>0.96</td>
</tr>
<tr>
<td>affected and all-optimal(254)</td>
<td>Graph</td>
<td>254</td>
<td>81.47</td>
<td>507.23</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>254</td>
<td>82.87</td>
<td>482.61</td>
<td>1.02</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>254</td>
<td>81.43</td>
<td>468.57</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>254</td>
<td>80.21</td>
<td>485.52</td>
<td>0.98</td>
</tr>
</tbody>
</table>

▶ “MIR” leads always to smaller search trees
### Computational Results

<table>
<thead>
<tr>
<th>Settings</th>
<th>solved</th>
<th>time(s)</th>
<th># nodes</th>
<th>time quot</th>
<th>nodes quot</th>
</tr>
</thead>
<tbody>
<tr>
<td>all(975) Graph</td>
<td>405</td>
<td>603.55</td>
<td>682.31</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Division</td>
<td>419</td>
<td>601.4</td>
<td>683.48</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>MIR</td>
<td>420</td>
<td>599.37</td>
<td>677.04</td>
<td>0.99</td>
<td>0.99</td>
</tr>
<tr>
<td>Saturation</td>
<td>418</td>
<td>599.76</td>
<td>691.81</td>
<td>0.99</td>
<td>1.01</td>
</tr>
<tr>
<td>affected(286) Graph</td>
<td>263</td>
<td>121.21</td>
<td>753.96</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Division</td>
<td>277</td>
<td>117.82</td>
<td>682.43</td>
<td>0.97</td>
<td>0.91</td>
</tr>
<tr>
<td>MIR</td>
<td>278</td>
<td>116.91</td>
<td>675.11</td>
<td>0.96</td>
<td>0.90</td>
</tr>
<tr>
<td>Saturation</td>
<td>276</td>
<td>116.71</td>
<td>710.72</td>
<td>0.96</td>
<td>0.94</td>
</tr>
<tr>
<td>affected and all-optimal(254) Graph</td>
<td>254</td>
<td>81.47</td>
<td>507.23</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Division</td>
<td>254</td>
<td>82.87</td>
<td>482.61</td>
<td>1.02</td>
<td>0.95</td>
</tr>
<tr>
<td>MIR</td>
<td>254</td>
<td>81.43</td>
<td>468.57</td>
<td>1.0</td>
<td>0.92</td>
</tr>
<tr>
<td>Saturation</td>
<td>254</td>
<td>80.21</td>
<td>485.52</td>
<td>0.98</td>
<td>0.96</td>
</tr>
</tbody>
</table>

- “MIR” leads always to smaller search trees
- “MIR” vs “No Conflict Analysis” on 279 affected instances:
  - +25 solved, 27% faster, 37% smaller trees
# Computational Results

<table>
<thead>
<tr>
<th>Settings</th>
<th>solved</th>
<th>time(s)</th>
<th># nodes</th>
<th>time quot</th>
<th>nodes quot</th>
</tr>
</thead>
<tbody>
<tr>
<td>all(975)</td>
<td>Graph</td>
<td>405</td>
<td>603.55</td>
<td>682.31</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>419</td>
<td>601.4</td>
<td>683.48</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>420</td>
<td>599.37</td>
<td>677.04</td>
<td>0.99</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>418</td>
<td>599.76</td>
<td>691.81</td>
<td>0.99</td>
</tr>
<tr>
<td>affected(286)</td>
<td>Graph</td>
<td>263</td>
<td>121.21</td>
<td>753.96</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>277</td>
<td>117.82</td>
<td>682.43</td>
<td>0.97</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>278</td>
<td>116.91</td>
<td>675.11</td>
<td>0.96</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>276</td>
<td>116.71</td>
<td>710.72</td>
<td>0.96</td>
</tr>
<tr>
<td>affected and all-optimal(254)</td>
<td>Graph</td>
<td>254</td>
<td>81.47</td>
<td>507.23</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Division</td>
<td>254</td>
<td>82.87</td>
<td>482.61</td>
<td>1.02</td>
</tr>
<tr>
<td></td>
<td>MIR</td>
<td>254</td>
<td>81.43</td>
<td>468.57</td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>Saturation</td>
<td>254</td>
<td>80.21</td>
<td>485.52</td>
<td>0.98</td>
</tr>
</tbody>
</table>

- “MIR” leads always to smaller search trees
- “MIR” vs “No Conflict Analysis” on 279 affected instances:
  +25 solved, 27% faster, 37% smaller trees
- Still requires further investigation: weakening, choose best cut, . . .
Table of Contents

Conflict Analysis in MIP

Pseudo Boolean Conflict Analysis

Computational Results

Conclusion
Conclusion

In this work:

▶ We studied the integration of PB conflict analysis into a MIP solving framework.
▶ We strengthened the PB conflict analysis further by using MIR cuts.

Next Steps:

▶ Dynamically choose the best strengthening method?
▶ Post-process the final learned constraint - remove irrelevant variables e.g., $3x_1 + x_4 + x_5 \geq 3$ can be strengthened to $x_1 \geq 1$
▶ Complement variables (e.g., replacing $x_i$ by $1 - \bar{x}_i$) before CG/MIR
▶ Generalize to 0–1 mixed IPs

Thank you for your attention!

Questions?
mexi@zib.de
Conclusion

In this work:
- We studied the integration of PB conflict analysis into a MIP solving framework.
- We strengthened the PB conflict analysis further by using MIR cuts.

Next Steps:
- Dynamically choose the best strengthening method?
- Post-Process the final learned constraint
  - remove irrelevant variables
    e.g., $3x_1 + x_4 + x_5 \geq 3$ can be strengthened to $x_1 \geq 1$
- Complement variables (e.g., replacing $x_i$ by $1 - \bar{x}_i$) before CG/MIR
- Generalize to 0–1 mixed IPs
Conclusion

In this work:
▶ We studied the integration of PB conflict analysis into a MIP solving framework.
▶ We strengthened the PB conflict analysis further by using MIR cuts.

Next Steps:
▶ Dynamically choose the best strengthening method?
▶ Post-Process the final learned constraint
  - remove irrelevant variables
    e.g., $3x_1 + x_4 + x_5 \geq 3$ can be strengthened to $x_1 \geq 1$
▶ Complement variables (e.g., replacing $x_i$ by $1 - \bar{x}_i$) before CG/MIR
▶ Generalize to 0–1 mixed IPs

Thank you for your attention!

Questions?
mexi@zib.de


