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

Mixed-Integer Program (MIP)

A MIP is a problem of the form:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & l \leq x \leq u \\ & x \in \mathbb{Z}^{\mathcal{I}} \times \mathbb{R}^{C}. \end{aligned} \tag{1}$$

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

Mixed-Integer Program (MIP)

A MIP is a problem of the form:

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & l \leq x \leq u \\ & x \in \mathbb{Z}^{\mathcal{I}} \times \mathbb{R}^{C}. \end{aligned} \tag{1}$$

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

- 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):

$$\mathbb{Z}^{\mathcal{I}} \rightsquigarrow \mathbb{R}^{\mathcal{I}}$$

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

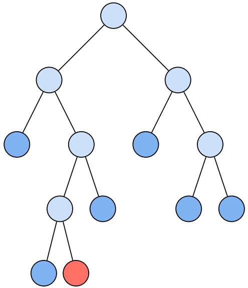
Table of Contents

Table of Contents

Conflict Analysis in MIP

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

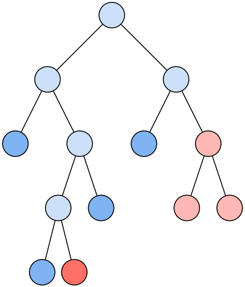
- extract a shorter explanation



Conflict Analysis in MIP

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

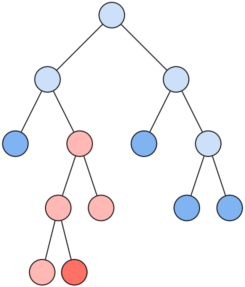
- extract a shorter explanation
- that prunes other parts of the tree



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



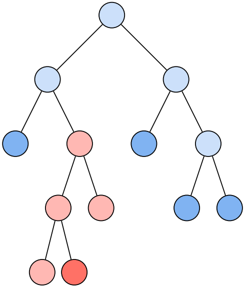
Conflict Analysis in MIP

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:

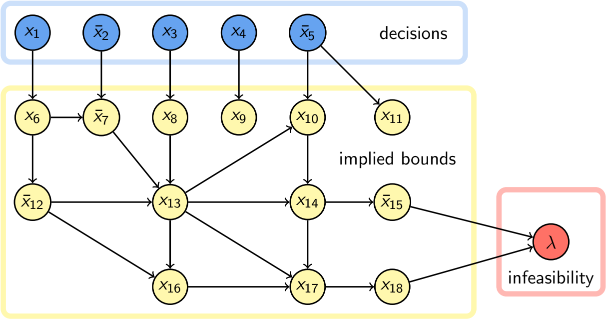
- Propagation
- LP relaxation
- Bound exceeding LP



Conflict Graph Analysis [Achterberg, 2007]

Similar to [Marques-Silva and Sakallah, 1996]

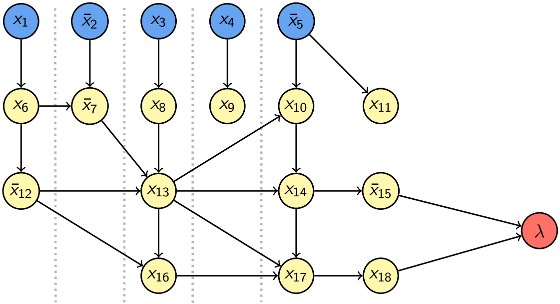
- The sequence of assignments and implications is captured by a directed implication graph



Conflict Graph Analysis [Achterberg, 2007]

Similar to [Marques-Silva and Sakallah, 1996]

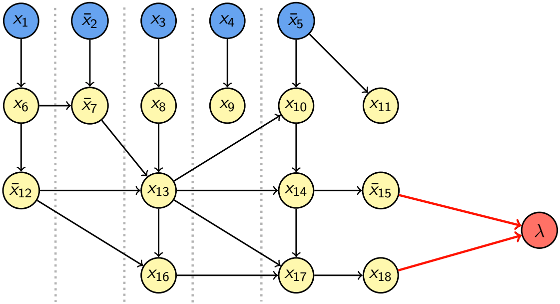
- The sequence of assignments and implications is captured by a directed implication graph
- Each cut that separates the decision nodes from λ yields a conflict (FUIP, ...)



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 λ yields a conflict (FUIP, ...)

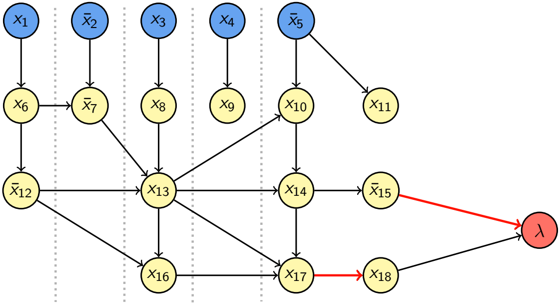


Variable assignment $\{\bar{x}_{15}, x_{18}\}$ responsible for the conflict
Resolve: x_{18}

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 λ yields a conflict (FUIP, ...)

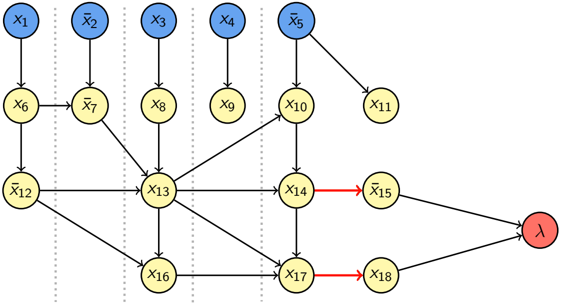


Variable assignment $\{\overline{x}_{15}, x_{17}\}$ responsible for the conflict
Resolve: $\overline{x}_{15}$

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 λ yields a conflict (FUIP, ...)



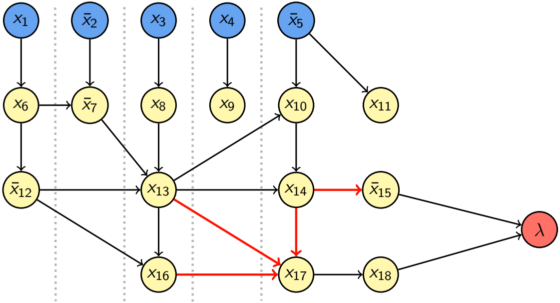
Variable assignment $\{x_{14}, x_{17}\}$ responsible for the conflict

Resolve: x_{17}

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 λ yields a conflict (FUIP, ...)

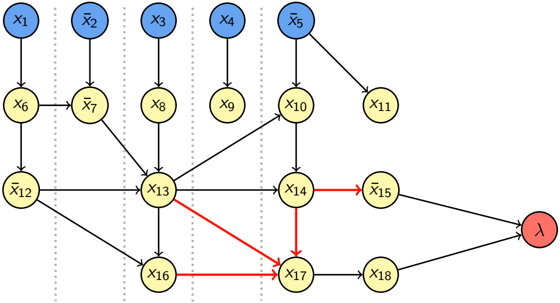


Variable assignment $\{x_{13}, x_{14}, x_{16}\}$ responsible for the conflict

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 λ yields a conflict (FUIP, ...)



Learned clause: $\overline{x}_{13} \vee \overline{x}_{14} \vee \overline{x}_{16}$
$\rightsquigarrow (1 - x_{13}) + (1 - x_{14}) + (1 - x_{16}) \geq 1$

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 $(x_1 \geq 1) \vee (x_3 \leq 0) \vee (x_7 \leq 11)$
- 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

Preliminaries I

- 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 ρ, map from literals to 0 (falsified) or 1 (true)

Preliminaries I

- 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 ρ, map from literals to 0 (falsified) or 1 (true)
- The slack of a PB constraint under a partial assignment ρ: is defined as

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

If the slack is negative $\implies$ conflict

Preliminaries I

- 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 ρ, map from literals to 0 (falsified) or 1 (true)
- The slack of a PB constraint under a partial assignment ρ: is defined as

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

  If the slack is negative $\implies$ conflict
- Generalized resolution rule: [Hooker, 1988]
  $\rightsquigarrow$ linear combination of two constraints that cancels a variable

Example Generalized Resolution

$$C_{\text{reason}}: \quad x_1 + x_2 + 2x_3 \geq 2$$
$$C_{\text{confl}}: \quad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3$$

$\rho = \left\{ x_1 \overset{\text{dec.}}{=} 0, x_3 \overset{C_{\text{reason}}}{=} 1 \right\} \Rightarrow$ Conflict with C_{confl}

Example Generalized Resolution

$$C_{\text{reason}}: \quad x_1 + x_2 + 2x_3 \geq 2$$
$$C_{\text{confl}}: \quad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3$$

$$\rho = \left\{ x_1 \overset{\text{dec.}}{=} 0, x_3 \overset{C_{\text{reason}}}{=} 1 \right\} \Rightarrow \text{Conflict with } C_{\text{confl}}$$

Resolving on x_3:

$$\text{resolve}\,\{x_3\} \; \frac{x_1 + x_2 + 2x_3 \geq 2 \qquad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3}{2x_1 + x_2 + x_4 + x_5 \geq 3}$$

Does not explain infeasibility since it has non-negative slack

Example Generalized Resolution

$$C_{\text{reason}}: \quad x_1 + x_2 + 2x_3 \geq 2$$
$$C_{\text{confl}}: \quad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3$$

$$\rho = \left\{ x_1 \overset{\text{dec.}}{=} 0, x_3 \overset{C_{\text{reason}}}{=} 1 \right\} \Rightarrow \text{Conflict with } C_{\text{confl}}$$

Resolving on x_3:

$$\text{resolve}\{x_3\} \ \frac{x_1 + x_2 + 2x_3 \geq 2 \qquad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3}{2x_1 + x_2 + x_4 + x_5 \geq 3}$$

Does not explain infeasibility since it has non-negative slack

- 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 ℓ_j:

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

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

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

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

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

Example Generalized Resolution

$$C_{\text{reason}}: \quad x_1 + x_2 + 2x_3 \geq 2$$
$$C_{\text{confl}}: \quad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3$$

$\rho = \left\{ x_1 \overset{\text{dec.}}{=} 0, x_3 \overset{C_{\text{reason}}}{=} 1 \right\} \Rightarrow$ Conflict with C_{confl}

Example Generalized Resolution

$$C_{\text{reason}}: \quad x_1 + x_2 + 2x_3 \geq 2$$
$$C_{\text{confl}}: \quad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3$$

$$\rho = \left\{ x_1 \overset{\text{dec.}}{=} 0, x_3 \overset{C_{\text{reason}}}{=} 1 \right\} \Rightarrow \text{Conflict with } C_{\text{confl}}$$

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

$$\text{weaken } \{x_2\} \quad \frac{x_1 + x_2 + 2x_3 \geq 2}{x_1 + 2x_3 \geq 1}$$

$$\text{saturate } \frac{x_1 + 2x_3 \geq 1}{x_1 + x_3 \geq 1}$$

$$\text{resolve } \{x_3\} \quad \frac{x_1 + x_3 \geq 1 \qquad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3}{3x_1 + x_4 + x_5 \geq 3}$$

Example Generalized Resolution

$$C_{\text{reason}}: \quad x_1 + x_2 + 2x_3 \geq 2$$
$$C_{\text{confl}}: \quad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3$$

$$\rho = \left\{ x_1 \overset{\text{dec.}}{=} 0, x_3 \overset{C_{\text{reason}}}{=} 1 \right\} \Rightarrow \text{Conflict with } C_{\text{confl}}$$

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

$$\text{weaken } \{x_2\} \; \frac{x_1 + x_2 + 2x_3 \geq 2}{x_1 + 2x_3 \geq 1}$$
$$\text{saturate } \frac{x_1 + 2x_3 \geq 1}{x_1 + x_3 \geq 1}$$
$$\text{resolve } \{x_3\} \; \frac{x_1 + x_3 \geq 1 \qquad x_1 + 2\overline{x}_3 + x_4 + x_5 \geq 3}{3x_1 + x_4 + x_5 \geq 3}$$

- Now the slack is negative $\rightsquigarrow$ conflict invariant is preserved

Conflict Analysis Algorithm

- First introduced in [Chai and Kuehlmann, 2005]

Algorithm: Generalized Resolution Conflict Analysis

Input : conflict constraint C_{confl}, falsifying partial assignment ρ

Output : learned conflict constraint C_{learn}

$C_{\text{learn}} \leftarrow C_{\text{confl}}$

while C_{learn} not asserting **and** $C_{\text{learn}} \neq \perp$ **do**

  $\ell_r \leftarrow$ literal last assigned on ρ

  if ℓ_r propagated **and** $\bar{\ell}_r$ occurs in C_{learn} **then**

    $C_{\text{reason}} \leftarrow \texttt{reason}(\ell_r, \rho)$

    $C_{\text{reason}} \leftarrow \texttt{reduce}(C_{\text{reason}}, C_{\text{learn}}, \ell_r, \rho)$

    $C_{\text{learn}} \leftarrow \texttt{resolve}(C_{\text{learn}}, C_{\text{reason}}, \ell_r)$

  $\rho \leftarrow \rho \setminus \{\ell_r\}$

return C_{learn}

Conflict Analysis Algorithm

- First introduced in [Chai and Kuehlmann, 2005]

Algorithm: Generalized Resolution Conflict Analysis

Input : conflict constraint C_{confl}, falsifying partial assignment ρ
Output : learned conflict constraint C_{learn}

$C_{\text{learn}} \leftarrow C_{\text{confl}}$
while C_{learn} not asserting **and** $C_{\text{learn}} \neq \perp$ **do**
    $\ell_r \leftarrow$ literal last assigned on ρ
    if ℓ_r propagated **and** $\bar{\ell}_r$ occurs in C_{learn} **then**
        $C_{\text{reason}} \leftarrow \texttt{reason}(\ell_r, \rho)$
        $C_{\text{reason}} \leftarrow \texttt{reduce}(C_{\text{reason}}, C_{\text{learn}}, \ell_r, \rho)$
        $C_{\text{learn}} \leftarrow \texttt{resolve}(C_{\text{learn}}, C_{\text{reason}}, \ell_r)$
    $\rho \leftarrow \rho \setminus \{\ell_r\}$
return C_{learn}

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

Reduction Algorithm

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

Algorithm: Saturation-based Reduction Algorithm

Input : conflict constraint C_{confl}, reason constraint C_{reason}, literal to resolve ℓ_r, partial assignment ρ

Output : reduced reason C_{reason}

while $\mathit{slack}((\texttt{resolve}(C_{\text{reason}}, C_{\text{confl}}, \ell_r)), \rho) \geq 0$ **do**
- $\ell_j \leftarrow$ non falsified literal in $C_{\text{reason}} \setminus \{\ell_r\}$
- $C_{\text{reason}} \leftarrow \texttt{weaken}(C_{\text{reason}}, \ell_j)$
- $C_{\text{reason}} \leftarrow \texttt{saturate}(C_{\text{reason}})$

return C_{reason}

Reduction Algorithm

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

Algorithm: Saturation-based Reduction Algorithm

Input : conflict constraint C_{confl}, reason constraint C_{reason}, literal to resolve ℓ_r, partial assignment ρ

Output : reduced reason C_{reason}

while $\mathit{slack}((\texttt{resolve}(C_{\text{reason}}, C_{\text{confl}}, \ell_r)), \rho) \geq 0$ **do**
    $\ell_j \leftarrow$ non falsified literal in $C_{\text{reason}} \setminus \{\ell_r\}$
    $C_{\text{reason}} \leftarrow \texttt{weaken}(C_{\text{reason}}, \ell_j)$
    $C_{\text{reason}} \leftarrow \texttt{saturate}(C_{\text{reason}})$

return C_{reason}

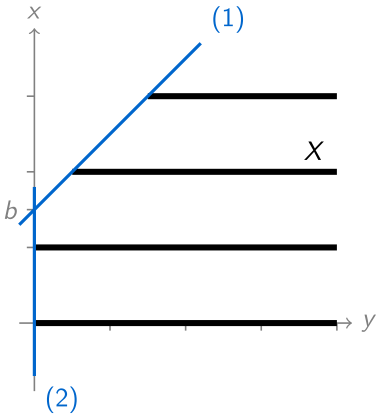
- Division (CG) can be used instead of saturation [Elffers and Nordström, 2018]
- Incomparable in terms of strength [Gocht et al., 2019]

Mixed Integer Rounding (MIR)

Introduced in [Marchand and Wolsey, 2001]

Elementary mixed integer set:

$$X := \{\ (x, y) \in \mathbb{Z} \times \mathbb{R} : \quad x \le b + y \quad (I) \quad y \ge 0 \quad (II)\ \}$$

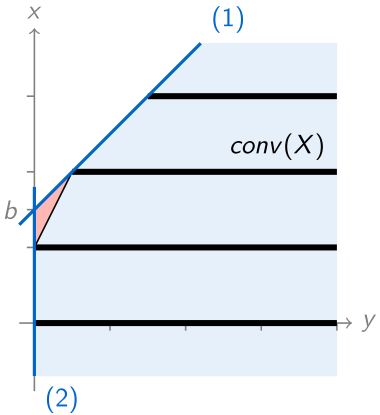


Mixed Integer Rounding (MIR)

Introduced in [Marchand and Wolsey, 2001]

Elementary mixed integer set:

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

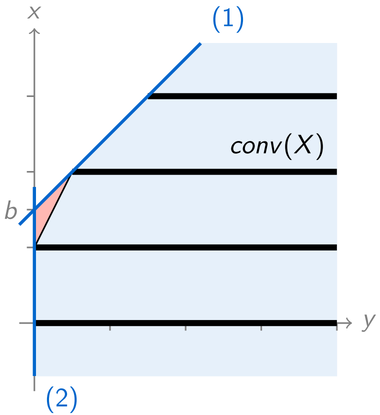


Inequalities (I) and (II) do not suffice to describe $conv(X)$.

Mixed Integer Rounding (MIR)

Disjunctive argument:

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



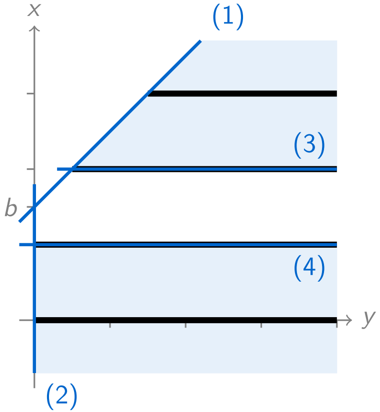
Mixed Integer Rounding (MIR)

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:

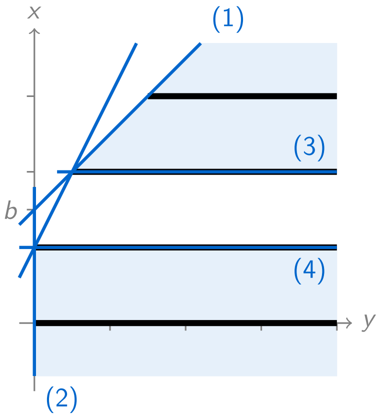
- X^1: Add $x \geq \lceil b \rceil$ (III)
- X^2: Add $x \leq \lfloor b \rfloor$ (IV)



Mixed Integer Rounding (MIR)

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

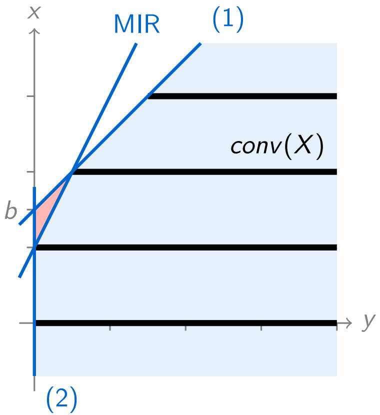
$$\underbrace{x \leq \lfloor b \rfloor + \tfrac{1}{1-f_b} y}_{(I)+f_b(III) \text{ and } (II)+(1-f_b)(IV)}$$



Mixed Integer Rounding (MIR)

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

$$\underbrace{x \leq \lfloor b \rfloor + \tfrac{1}{1-f_b} y}_{\text{MIR inequality}}$$



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

$$\sum_{i \in I_1} \left\lceil \frac{a_i}{d} \right\rceil \ell_i + \sum_{i \in I_2} \left(\left\lfloor \frac{a_i}{d} \right\rfloor + \frac{f(a_i/d)}{f(b/d)} \right) \ell_i \geq \left\lceil \frac{b}{d} \right\rceil, \tag{1}$$

where

$$I_1 = \{ i \in \mathcal{N} \, : \, f(a_i/d) \geq f(b/d) \text{ or } f(a_i/d) \in \mathbb{Z} \},$$

$$I_2 = \{ i' \in \mathcal{N} \, : \, f(a_{i'}/d) < f(b/d) \text{ and } f(a_{i'}/d) \notin \mathbb{Z} \},$$

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

MIR Reduction

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

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

MIR Reduction

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

1. weakening all non-falsified literal not divisible by a_r, and
2. Applying MIR on C_{reason} with divisor $d = a_r$
   $\rightsquigarrow$ 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):
   $\rightsquigarrow x_1 + x_2 + x_3 \geq 1$
2. MIR-based reduction:
   $\rightsquigarrow \frac{0.2}{0.8}x_1 + \frac{0.6}{0.8}x_2 + x_3 \geq 1$

MIR Reduction

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

1. weakening all non-falsified literal not divisible by a_r, and
2. Applying MIR on C_{reason} with divisor $d = a_r$
   $\rightsquigarrow$ 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):
   $\rightsquigarrow x_1 + x_2 + x_3 \geq 1$
2. MIR-based reduction:
   $\rightsquigarrow \frac{0.2}{0.8}x_1 + \frac{0.6}{0.8}x_2 + x_3 \geq 1$

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

Table of Contents

Experimental Setup

Some implementation details:

- PB conflict analysis can be generalized for constraints with real coefficients. However, foating-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, foating-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.
  $\rightsquigarrow$ use a large and diverse test set of instances and multiple seeds.
- 195 pure 0-1 models from the MIPLIB2017 collection $\times$ 5 seeds.

Computational Results

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

Computational Results

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

- "MIR" leads always to smaller search trees

Computational Results

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

- "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

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

- "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

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.

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

Achterberg, T. (2007).
Conflict analysis in mixed integer programming.
Discrete Optimization, 4(1):4–20.

Chai, D. and Kuehlmann, A. (2005).
A fast pseudo-Boolean constraint solver.
volume 24, pages 305–317.
Preliminary version in DAC '03.

Elffers, J. and Nordström, J. (2018).
Divide and conquer: Towards faster pseudo-Boolean solving.
In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI '18)*, pages 1291–1299.

Gocht, S., Nordström, J., and Yehudayoff, A. (2019).
On division versus saturation in pseudo-Boolean solving.
In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI '19)*, pages 1711–1718.

Hooker, J. N. (1988).
Generalized resolution and cutting planes.
Annals of Operations Research, 12(1):217–239.

Le Berre, D. and Parrain, A. (2010).
The Sat4j library, release 2.2.
Journal on Satisfiability, Boolean Modeling and Computation, 7(2-3):59–64.

Marchand, H. and Wolsey, L. A. (2001).
Aggregation and mixed integer rounding to solve mips.
Operations research, 49(3):363–371.

Marques-Silva, J. P. and Sakallah, K. A. (1996).
GRASP—a new search algorithm for satisfiability.
In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD '96)*, pages 220–227.

Witzig, J., Berthold, T., and Heinz, S. (2019).
Computational aspects of infeasibility analysis in mixed integer programming.
Technical Report 19-54, ZIB, Takustr. 7, 14195 Berlin.