EINDHOVEN UNIVERSITY OF TECHNOLOGY

Symmetry Handling in SCIP

An Overview

Christopher Hojny

Let’s SCIP it!, Zuse Institute Berlin, November 4, 2022

20 years of SCIP – 5 years of symmetry handling

```
commit e45f24868cf7a18c79880afeef84af57902c5f83
Author: Marc Pfetsch <pfetsch@mathematik.tu-darmstadt.de>
Date:   Sat Sep 30 13:35:31 2017 +0200

    add first version of symmetry constraint handler
```

20 years of SCIP – 5 years of symmetry handling

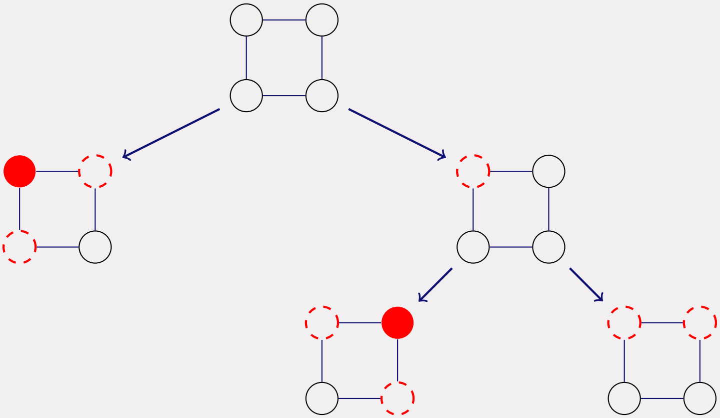
```
commit e45f24868cf7a18c79880afeef84af57902c5f83
Author: Marc Pfetsch <pfetsch@mathematik.tu-darmstadt.de>
Date:   Sat Sep 30 13:35:31 2017 +0200

    add first version of symmetry constraint handler
```

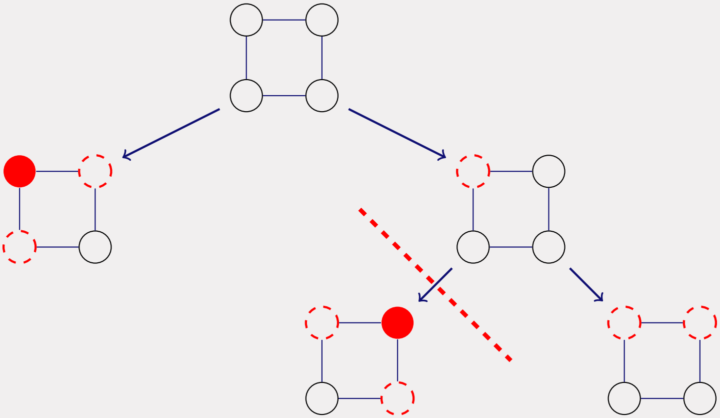
```
commit 8bcb00fab87f41d9b5a5614b72dd2f3ae900b998
Author: Marc Pfetsch <m.pfetsch@tu-bs.de>
Date:   Sat Sep 26 11:31:20 2009 +0000

    - first version (should work for partitioning)
```

Symmetry in Branch-and-Bound



Symmetry in Branch-and-Bound



Outline

Symmetry Detection

General Variables

Binary Variables
- Orbital Fixing
- Symmetry Handling Constraints

User Interaction

Problem Setting

We consider optimization problems of type

$$\max\{c^\top x \ : \ f(x,y) \leq 0,\ (x,y) \in \mathbb{Z}^n \times \mathbb{R}^p\},$$

where $f \colon \mathbb{Z}^n \times \mathbb{R}^p \to \mathbb{R}^m$.

Problem Setting

We consider optimization problems of type

$$\max\{c^\top x \,:\, f(x,y) \leq 0,\ (x,y) \in \mathbb{Z}^n \times \mathbb{R}^p\},$$

where $f\colon \mathbb{Z}^n \times \mathbb{R}^p \to \mathbb{R}^m$.

Some Definitions

- The action of a permutation $\gamma \in \mathcal{S}_n \times \mathcal{S}_p$ on $(x,y) \in \mathbb{Z}^n \times \mathbb{R}^p$ is

$$\gamma(x,y) = (x_{\gamma^{-1}(1)}, \dots, x_{\gamma^{-1}(n)}, y_{\gamma^{-1}(1)}, \dots, y_{\gamma^{-1}(p)}).$$

- A permutation $\gamma \in \mathcal{S}_n \times \mathcal{S}_p$ is a symmetry if, for every $(x,y) \in \mathbb{Z}^n \times \mathbb{R}^p$,
  - $f(x,y) \leq 0$ iff $f(\gamma(x,y)) \leq 0$, and
  - $c^\top(x,y) = c^\top\gamma(x,y)$.

Problem Setting

We consider optimization problems of type

$$\max\{c^\top x \,:\, f(x,y) \leq 0,\ (x,y) \in \mathbb{Z}^n \times \mathbb{R}^p\},$$

where $f\colon \mathbb{Z}^n \times \mathbb{R}^p \to \mathbb{R}^m$.

Some Definitions

- The action of a permutation $\gamma \in \mathcal{S}_n \times \mathcal{S}_p$ on $(x,y) \in \mathbb{Z}^n \times \mathbb{R}^p$ is

$$\gamma(x,y) = (x_{\gamma^{-1}(1)}, \dots, x_{\gamma^{-1}(n)}, y_{\gamma^{-1}(1)}, \dots, y_{\gamma^{-1}(p)}).$$

- A permutation $\gamma \in \mathcal{S}_n \times \mathcal{S}_p$ is a symmetry if, for every $(x,y) \in \mathbb{Z}^n \times \mathbb{R}^p$,
  - $f(x,y) \leq 0$ iff $f(\gamma(x,y)) \leq 0$, and
  - $c^\top(x,y) = c^\top\gamma(x,y)$.

Bad News: Already for MIPs, finding all symmetries is NP-hard.

Formulation Symmetries (Salvagnin 2005)

Consider MIP $\max\{c^\top x \,:\, Ax \le b,\ x \in \mathbb{Z}^n\}$

$$
\begin{aligned}
\max\ & x_1 + x_2 \\
& x_1 \qquad + 2x_3 \le 3 \\
& \qquad x_2 + 2x_3 \le 3
\end{aligned}
$$

Formulation Symmetries (Salvagnin 2005)

Consider MIP $\max\{c^\top x \,:\, Ax \leq b,\; x \in \mathbb{Z}^n\}$

$\gamma \in \mathcal{S}_n$ is a formulation symmetry if there exists a permutation π of the rows of (A, b) such that

- $\gamma(c) = c$,
- $\pi(b) = b$,
- $A_{\pi^{-1}(i),\gamma^{-1}(j)} = A_{i,j}$.

$$\begin{aligned} \max\, x_1 + x_2 & \\ x_1 \qquad + 2x_3 &\leq 3 \\ x_2 + 2x_3 &\leq 3 \end{aligned}$$

Formulation Symmetries (Salvagnin 2005)

Consider MIP $\max\{c^\top x \,:\, Ax \leq b,\ x \in \mathbb{Z}^n\}$

$\gamma \in \mathcal{S}_n$ is a formulation symmetry if there exists a permutation π of the rows of (A, b) such that

- $\gamma(c) = c$,
- $\pi(b) = b$,
- $A_{\pi^{-1}(i),\gamma^{-1}(j)} = A_{i,j}$.

$$\begin{aligned} \max\ & x_1 + x_2 \\ & x_1 \phantom{{}+x_2} + 2x_3 \leq 3 \\ & \phantom{x_1 +{}} x_2 + 2x_3 \leq 3 \end{aligned}$$

construct auxiliary graph

Formulation Symmetries (Salvagnin 2005)

Consider MIP $\max\{c^\top x \,:\, Ax \leq b,\ x \in \mathbb{Z}^n\}$

$\gamma \in \mathcal{S}_n$ is a formulation symmetry if there exists a permutation π of the rows of (A, b) such that

- $\gamma(c) = c$,
- $\pi(b) = b$,
- $A_{\pi^{-1}(i),\gamma^{-1}(j)} = A_{i,j}$.

$$\begin{aligned} \max\ & x_1 + x_2 \\ & x_1 \quad\;\; + 2x_3 \leq 3 \\ & \quad\;\; x_2 + 2x_3 \leq 3 \end{aligned}$$

construct auxiliary graph

TU/e

Formulation Symmetries (Salvagnin 2005)

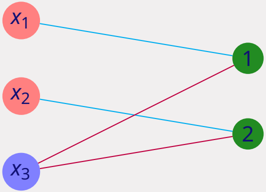
Consider MIP $\max\{c^\top x \,:\, Ax \leq b,\ x \in \mathbb{Z}^n\}$

$\gamma \in \mathcal{S}_n$ is a formulation symmetry if there exists a permutation π of the rows of (A, b) such that

- $\gamma(c) = c$,
- $\pi(b) = b$,
- $A_{\pi^{-1}(i),\gamma^{-1}(j)} = A_{i,j}$.

$$\begin{aligned} \max\ & x_1 + x_2 \\ & x_1 \phantom{{}+x_2} + 2x_3 \leq 3 \\ & \phantom{x_1 +{}} x_2 + 2x_3 \leq 3 \end{aligned}$$

construct auxiliary graph



Formulation Symmetries (Salvagnin 2005)

Consider MIP $\max\{c^\top x \,:\, Ax \leq b,\; x \in \mathbb{Z}^n\}$

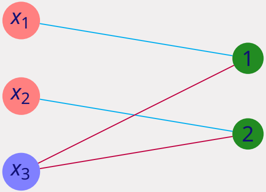
$\gamma \in \mathcal{S}_n$ is a formulation symmetry if there exists a permutation π of the rows of (A, b) such that

- $\gamma(c) = c$,
- $\pi(b) = b$,
- $A_{\pi^{-1}(i),\gamma^{-1}(j)} = A_{i,j}$.

Graph automorphism codes can be used to detect formulation symmetries.

$$\begin{aligned} \max\, x_1 + x_2 & \\ x_1 + 2x_3 &\leq 3 \\ x_2 + 2x_3 &\leq 3 \end{aligned}$$

construct auxiliary graph



Implementation in SCIP

- different symmetry detection graphs for MIPs and MINLPs
- automorphism code `bliss`
- returns list of generators of symmetry group
- possibility to limit number of generators

```
(0.3s) symmetry computation started: requiring (bin +, int -, cont +), (fixed: bin -, int +, cont -)
(0.5s) symmetry computation finished: 9 generators found (max: 1500, log10 of symmetry group size: 6.6)
```

General Variables

Basic Idea to Handle Symmetries

General Idea

Handle symmetries by sorting solutions, discard solutions that are not maximal in this sorting.

Basic Idea to Handle Symmetries

General Idea

Handle symmetries by sorting solutions, discard solutions that are not maximal in this sorting.

Basic Idea, e.g., Liberti 2012

- select variable x_i
- compute the orbit $\mathrm{orbit}(\Gamma, i) = \{\gamma(i) : \gamma \in \Gamma\}$
- add inequalities

$$x_i \geq x_j, \qquad j \in \mathrm{orbit}(\Gamma, i)$$

- we call i the leader and j the follower of the cut

Basic Idea to Handle Symmetries

General Idea

Handle symmetries by sorting solutions, discard solutions that are not maximal in this sorting.

Basic Idea, e.g., Liberti 2012

- select variable x_i
- compute the orbit $\mathrm{orbit}(\Gamma, i) = \{\gamma(i) : \gamma \in \Gamma\}$
- add inequalities

$$x_i \geq x_j, \qquad j \in \mathrm{orbit}(\Gamma, i)$$

- we call i the leader and j the follower of the cut
- **Pro:** very simple inequalities
- **Con:** amount of handled symmetries very limited

Cuts from Schreier-Sims Table (SST Cuts)

Improved Idea (Liberti and Ostrowski 2014, Salvagnin 2018)

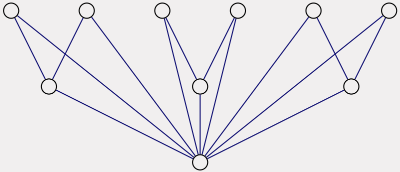
Iteratively add the simple inequalities.

Cuts from Schreier-Sims Table (SST Cuts)

Improved Idea (Liberti and Ostrowski 2014, Salvagnin 2018)

Iteratively add the simple inequalities.

1. initialize $\Gamma' \leftarrow \Gamma$

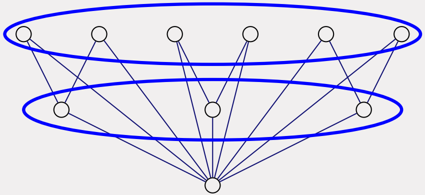


Cuts from Schreier-Sims Table (SST Cuts)

Improved Idea (Liberti and Ostrowski 2014, Salvagnin 2018)

Iteratively add the simple inequalities.

1. initialize $\Gamma' \leftarrow \Gamma$
2. select variable i affected by Γ'
3. add inequalities $x_i \geq x_j$ for all $j \in \mathrm{orbit}(\Gamma, i)$

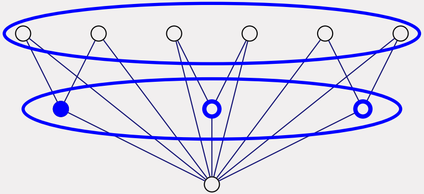


Cuts from Schreier-Sims Table (SST Cuts)

Improved Idea (Liberti and Ostrowski 2014, Salvagnin 2018)

Iteratively add the simple inequalities.

1. initialize $\Gamma' \leftarrow \Gamma$
2. select variable i affected by Γ'
3. add inequalities $x_i \geq x_j$ for all $j \in \mathrm{orbit}(\Gamma, i)$
4. replace $\Gamma' \leftarrow \{\gamma \in \Gamma' : \gamma(i) = i\}$

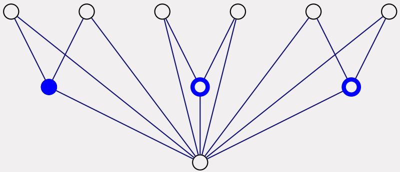


Cuts from Schreier-Sims Table (SST Cuts)

Improved Idea (Liberti and Ostrowski 2014, Salvagnin 2018)

Iteratively add the simple inequalities.

1. initialize $\Gamma' \leftarrow \Gamma$
2. select variable i affected by Γ'
3. add inequalities $x_i \geq x_j$ for all $j \in \mathrm{orbit}(\Gamma, i)$
4. replace $\Gamma' \leftarrow \{\gamma \in \Gamma' : \gamma(i) = i\}$
5. if $\Gamma' = \{\mathrm{id}\}$, terminate, otherwise go to Step 2

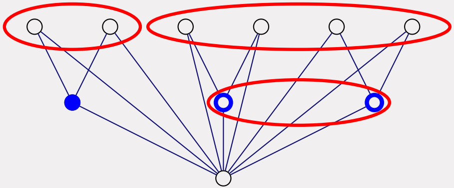


Cuts from Schreier-Sims Table (SST Cuts)

Improved Idea **(Liberti and Ostrowski 2014, Salvagnin 2018)**

Iteratively add the simple inequalities.

1. initialize $\Gamma' \leftarrow \Gamma$
2. select variable i affected by Γ'
3. add inequalities $x_i \geq x_j$ for all $j \in \mathrm{orbit}(\Gamma, i)$
4. replace $\Gamma' \leftarrow \{\gamma \in \Gamma' : \gamma(i) = i\}$
5. if $\Gamma' = \{\mathrm{id}\}$, terminate, otherwise go to Step 2



Parameterizing SST Cuts

many degrees of freedom in generating SST cuts, among others,

- type of variable x_i:
  - binary
  - general integer
  - continuous
- type of orbit:
  - orbit of maximum size
  - orbit of minimum size
  - orbit with most conflicts

Numerical Results I – Minimum Orbit

variable type						
bin	int	cont	orbit	symresack	time	#opt
					1107.4	126
x			min		-3.1 %	128
	x		min		±0 %	127
		x	min		±0 %	125
	x	x	min		±0 %	126
x	x	x	min		-1.5 %	127

Setup:

- MIP solver: SCIP 8.0.0.2
- LP solver: SoPlex 6.0.0.2
- test set: MIPLIB 2017 benchmark (240 instances)
- time limit: 2 h per instance

Numerical Results II – Maximum Orbit

variable type						
bin	int	cont	orbit	symresack	time	#opt
					1107.4	126
x			max		-7.0 %	130
	x		max		-0.3 %	127
		x	max		+0.2 %	125
	x	x	max		-0.2 %	126
x	x	x	max		-2.2 %	129

Numerical Results III – Maximum Conflict Orbit

variable type						
bin	int	cont	orbit	symresack	time	#opt
					1107.4	126
x			conf		-7.1 %	129
x	x	x	conf		-5.2 %	128

Binary Variables

Binary Programs

We consider

$$\max\{c^\top x \,:\, Ax \leq b,\ x \in \{0,1\}^n\}.$$

Assume that symmetry group Γ of binary program is known.

SCIP has two dedicated symmetry handling techniques for binary variables:

- orbital fixing
- symmetry handling constraints

Common Ground: for each set of equivalent solutions, it is sufficient to compute a representative solution

Graph Coloring

Input

- undirected graph $G = (V, E)$
- positive integer k

Task

find maximum induced subgraph of G that admits proper k-coloring

$$\max \sum_{v \in V} \sum_{j=1}^{k} x_{vj}$$

$$\begin{aligned} \sum_{j=1}^{k} x_{vj} &\leq 1, && v \in V, \\ x_{uj} + x_{vj} &\leq 1, && \{u, v\} \in E,\ j \in \{1, \dots, k\}, \\ x_{vj} &\in \{0, 1\}, && v \in V,\ j \in \{1, \dots, k\} \end{aligned}$$

Graph Coloring

Input

- undirected graph $G = (V, E)$
- positive integer k

Task

find maximum induced subgraph of G that admits proper k-coloring

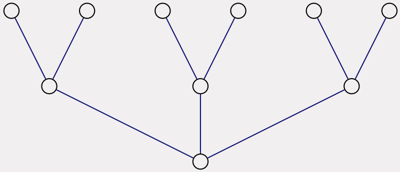
$$\begin{aligned} \max \sum_{v \in V} \sum_{j=1}^{k} x_{vj} & & \\ \sum_{j=1}^{k} x_{vj} \le 1, & \quad & v \in V, \\ x_{uj} + x_{vj} \le 1, & \quad & \{u, v\} \in E,\ j \in \{1, \dots, k\}, \\ x_{vj} \in \{0, 1\}, & \quad & v \in V,\ j \in \{1, \dots, k\} \end{aligned}$$

Symmetries

- color symmetries $\rightsquigarrow$ column permutations
- graph symmetries $\rightsquigarrow$ row permutations

Orbital Fixing (Margot 2003, Ostrowski 2009)

Idea: based on branching decisions, exclude non-representative solutions by fixing variables



Steps

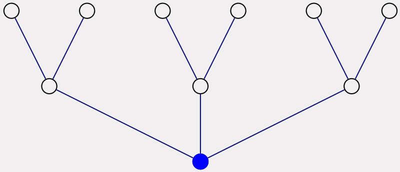
1. $I \leftarrow$ all variables fixed to 1 by branching
2. compute $\Gamma' = \{\gamma \in \Gamma : \gamma(I) = I\}$
3. for each $i \in \{1, \dots, n\}$, compute $O = \{\gamma(i) : \gamma \in \Gamma\}$
4. if one variable in O is fixed to 0 (resp. to 1), fix all variables in O to 0 (resp. to 1)

Orbital Fixing (Margot 2003, Ostrowski 2009)

Idea: based on branching decisions, exclude non-representative solutions by fixing variables

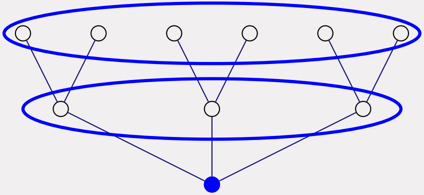
Steps

1. $I \leftarrow$ all variables fixed to 1 by branching
2. compute $\Gamma' = \{\gamma \in \Gamma \,:\, \gamma(I) = I\}$
3. for each $i \in \{1, \dots, n\}$,
   compute $O = \{\gamma(i) \,:\, \gamma \in \Gamma\}$
4. if one variable in O is fixed to 0 (resp. to 1), fix all variables in O to 0 (resp. to 1)



Orbital Fixing (Margot 2003, Ostrowski 2009)

Idea: based on branching decisions, exclude non-representative solutions by fixing variables



Steps

1. $I \leftarrow$ all variables fixed to 1 by branching
2. compute $\Gamma' = \{\gamma \in \Gamma \,:\, \gamma(I) = I\}$
3. for each $i \in \{1, \ldots, n\}$,
   compute $O = \{\gamma(i) \,:\, \gamma \in \Gamma\}$
4. if one variable in O is fixed to 0 (resp. to 1), fix all variables in O to 0 (resp. to 1)

Symmetry Handling Constraints

Main Idea: for each set of symmetric solutions, forbid solutions which are not lexicographically maximal

- Friedman 2007: for every $\gamma \in \Gamma$

$$\sum_{i=1}^{n} 2^{n-i} x_i \geq \sum_{i=1}^{n} 2^{n-i} \gamma(x)_i$$

Symmetry Handling Constraints

Main Idea: for each set of symmetric solutions, forbid solutions which are not lexicographically maximal

- Friedman 2007: for every $\gamma \in \Gamma$

$$\sum_{i=1}^{n} 2^{n-i} x_i \geq \sum_{i=1}^{n} 2^{n-i} \gamma(x)_i$$

- SCIP's approach: consider symresack

$$P_\gamma = \operatorname{conv}\left\{x \in \{0,1\}^n \,:\, \sum_{i=1}^{n} 2^{n-i} x_i \geq \sum_{i=1}^{n} 2^{n-i} \gamma(x)_i\right\}$$

Symmetry Handling Constraints

Main Idea: for each set of symmetric solutions, forbid solutions which are not lexicographically maximal

- Friedman 2007: for every $\gamma \in \Gamma$

$$\sum_{i=1}^{n} 2^{n-i} x_i \geq \sum_{i=1}^{n} 2^{n-i} \gamma(x)_i$$

- SCIP's approach: consider symresack

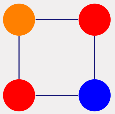
$$P_\gamma = \operatorname{conv} \left\{ x \in \{0,1\}^n \;:\; \sum_{i=1}^{n} 2^{n-i} x_i \geq \sum_{i=1}^{n} 2^{n-i} \gamma(x)_i \right\}$$

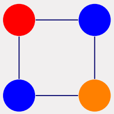
- cover inequalities can be separated in $O(n)$ time (H. and Pfetsch 2018)
- symresacks can be propagated in $O(n)$ time (van Doornmalen and H. 2022+)

Symmetric Groups

More reductions can be found by taking entire group Γ into account.

- if all variables can be sorted arbitrarily, use $x_1 \geq x_2 \geq \cdots \geq x_n$
- in graph coloring, we can sort groups (columns) of variables arbitrarily
- orbitopes sort columns of binary matrices lexicographically non-increasingly





$$\begin{pmatrix} 1 & & \\ & 1 & \\ 1 & & \\ & & 1 \end{pmatrix} \qquad \begin{pmatrix} & 1 & \\ & & 1 \\ & 1 & \\ 1 & & \end{pmatrix}$$

Orbitopes

Idea: sort columns lexicographically non-increasingly

Full Orbitopes

- no further restriction on binary matrices
- linear time propagation algorithm (Bendotti et al. 2019
- linear time separation algorithm for IP formulation (H. and Pfetsch 2018)

Packing/Partitioning Orbitopes

- each row has at most/exactly one 1-entry
- linear time propagation algorithm (Kaibel et al. 2011)
- linear time separation algorithm of facet description (Kaibel and Pfetsch 2008)

Symmetry Handling Constraints in SCIP

split symmetry group Γ into independent factors $\Gamma = \Gamma_1 \otimes \cdots \otimes \Gamma_\ell$

- orbitope detection:
  - heuristic to detect whether Γ_i can be completely handled by orbitopes
  - decide whether full or packing/partitioning orbitopes are used
- no orbitope is added?
  - scan list of generators of Γ_i
  - try to find "hidden" orbitopes
  - handle remaining generators by symresacks
  - possibly add SST cuts
- no hidden orbitopes found?
  - use orbital fixing

SCIP 8.0 $\rightsquigarrow$ running time improvement on MIPLIB 2017 benchmark testset: 16 %

User Interaction

Providing Symmetries

- providing list of symmetries currently not possible
- symmetries can be provided by three types of constraint handlers
  - symresacks enforce x is not lex. smaller than $\gamma(x)$

    symresack([x_1,...,x_n],[$\gamma(1)$,...,$\gamma(n)$]);

  - orbitopes sort columns of matrices $X \in \{0,1\}^{m \times n}$

    fullOrbitope($x_{1,1}$,...,$x_{1,n}$. ... .x_{m1},...,x_{mn});

    packOrbitope(...); partOrbitope(...);

  - orbisacks are orbitopes with two columns

    fullOrbisack(...); packOrbisack(...); partOrbisack(...);

Summary

SCIP's symmetry handling approach

- detect symmetries by detecting symmetries of auxiliary colored graph
- mixed symmetry handling strategy
  - SST cuts
  - orbitopes und symresacks
  - orbital fixing

TU/e

Summary

SCIP's symmetry handling approach

- detect symmetries by detecting symmetries of auxiliary colored graph
- mixed symmetry handling strategy
  - SST cuts
  - orbitopes und symresacks
  - orbital fixing

Thank you for your attention.