Symmetry handling in binary programs through propagation

Jasper van Doornmalen Christopher Hojny

Eindhoven University of Technology, Department of Mathematics and Computer Science, Combinatorial Optimization

Symmetries in binary programs

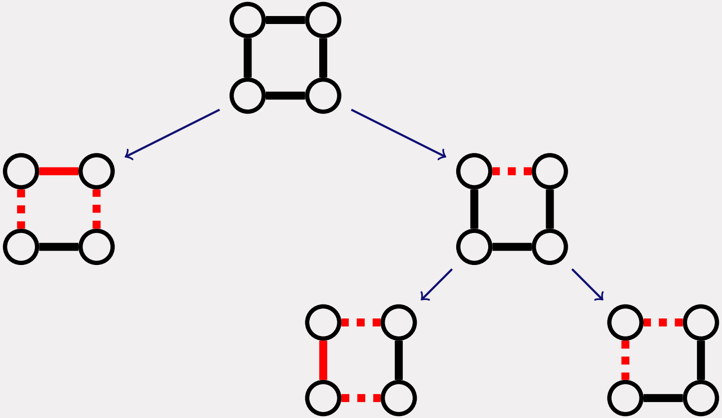
$$\begin{aligned} \text{Minimize } & c^\top x \\ \text{Subject to } & Ax \leq b \\ & x \in \{0,1\}^n \end{aligned}$$

Definition

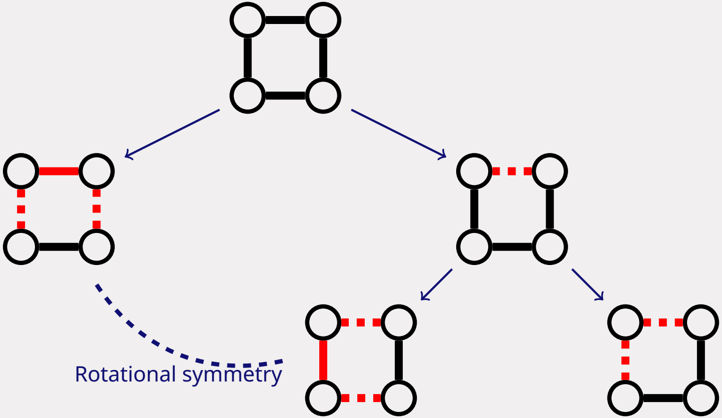
A permutation γ on $\{1, \ldots, n\}$ is a *symmetry for the program* if for all $x \in \{0,1\}^n$ holds $c^\top x = c^\top \gamma(x)$ and $Ax \leq b \iff A\gamma(x) \leq b$, where $\gamma(x) := (x_{\gamma^{-1}(1)}, \cdots, x_{\gamma^{-1}(n)})$.

- The symmetries of a program define a group.
- In this presentation: Γ is a symmetry group of the program.

Symmetry in Branch-and-Bound



Symmetry in Branch-and-Bound

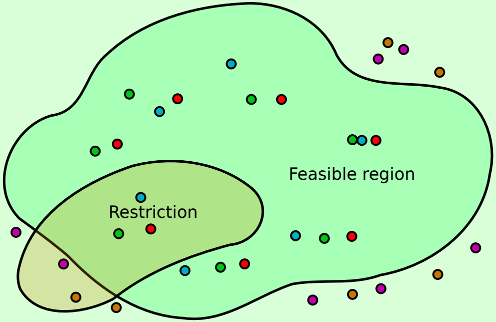


Previous work on symmetry handling in MILP

- Symmetry breaking inequalities [Friedman, 2007], [Kaibel and Loos, 2010], [Liberti, 2010], [Kaibel et al., 2011], [Liberti, 2013], [Hojny and Pfetsch, 2018], [Hojny, 2020], [Hojny et al., 2021+];
- Branching [Ostrowski et al., 2007], [Ostrowski et al., 2015];
- Propagation [Margot, 2002], [Margot, 2003], [Ostrowski et al., 2007], [Bendotti et al., 2021].

Solution approach

- Given: A problem symmetry group Γ of the program;
- Restrict region: Symmetrical solutions have (at least) a single representative;
- Enforce: by *propagation*.



Symmetry handling using lexicographic order

Definition (Lexicographic order)

$x, y \in \{0,1\}^n$;

- $x \succ y$ if there is a $j \in \{1, \dots, n\}$ with $x_i = y_i$ for all $i < j$, and $x_j > y_j$;
- $x \succeq y$ if $x \succ y$ or $x = y$.

$$\begin{bmatrix}1\\0\\1\\1\\\vdots\end{bmatrix} \succ \begin{bmatrix}1\\0\\1\\0\\\vdots\end{bmatrix}$$

Definition (Lexicographic leader in orbit)

Let $x \in \{0,1\}^n$.
If $x \succeq \gamma(x)$ for all $\gamma \in \Gamma$, then x is the lexicographic leader in its Γ-orbit $\{\gamma(x) : \gamma \in \Gamma\}$.

Propagation

Common in CP/CSP/SAT/MINLP solvers.

"Reduce (sub)problem size by logical deduction, restricting variable bounds."

- Problem variables $x \in \{0,1\}^n$;
- Set $\mathcal{C}$ of constraints;
- $I_0, I_1 \subseteq \{1, \dots, n\}$: Variable indices fixed to 0 and 1.

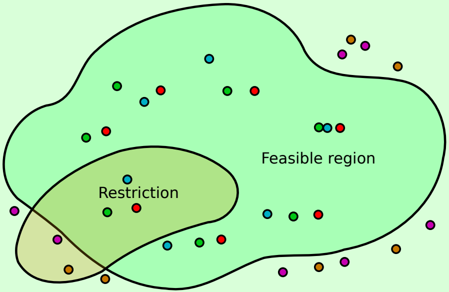
$$\mathcal{C}(I_0, I_1) := \left\{ x \in \{0,1\}^n \;:\; x \text{ satisfy constraints of } \mathcal{C}, \quad \begin{array}{l} x_i = 0 \text{ for all } i \in I_0, \\ x_i = 1 \text{ for all } i \in I_1 \end{array} \right\}$$

Goal of propagation: Find $\hat{I}_0, \hat{I}_1 \supseteq I_0, I_1$ such that $\mathcal{C}(I_0, I_1) = \mathcal{C}(\hat{I}_0, \hat{I}_1)$.
Complete: When $\hat{I}_0, \hat{I}_1$ are inclusionwise maximal.

Solution approach

- Given: A problem symmetry group Γ of the program;
- Restrict region: Symmetrical solutions have (at least) a single representative:
  - Constraint for lexicographic leaders in Γ-orbit:

$$x \succeq \gamma(x) \text{ for all } \gamma \in \Gamma.$$



- Enforce: by propagation.

Simpler case: Propagation for a single permutation

Compute complete set of fixings of $x \succeq \gamma(x)$ for a single permutation γ.

Example

$\gamma = (1,2,5)(3,4)$; Initial fixings: $x_1 = 1,\ x_3 = 0,\ x_5 = 1$.

$$x = \begin{bmatrix} x_1 = & 1 \\ x_2 = & \_ \\ x_3 = & 0 \\ x_4 = & \_ \\ x_5 = & 1 \end{bmatrix} \succeq \begin{bmatrix} x_5 = & 1 \\ x_1 = & 1 \\ x_4 = & \_ \\ x_3 = & 0 \\ x_2 = & \_ \end{bmatrix} = \gamma(x)$$

Simpler case: Propagation for a single permutation

Compute complete set of fixings of $x \succeq \gamma(x)$ for a single permutation γ.

Example

$\gamma = (1,2,5)(3,4)$; Initial fixings: $x_1 = 1,\ x_3 = 0,\ x_5 = 1$.

$$x = \begin{bmatrix} x_1 = & 1 \\ x_2 = & 1 \\ x_3 = & 0 \\ x_4 = & \_ \\ x_5 = & 1 \end{bmatrix} \succeq \begin{bmatrix} x_5 = & 1 \\ x_1 = & 1 \\ x_4 = & \_ \\ x_3 = & 0 \\ x_2 = & 1 \end{bmatrix} = \gamma(x)$$

Simpler case: Propagation for a single permutation

Compute complete set of fixings of $x \succeq \gamma(x)$ for a single permutation γ.

Example

$\gamma = (1,2,5)(3,4)$; Initial fixings: $x_1 = 1,\ x_3 = 0,\ x_5 = 1$.

$$x = \begin{bmatrix} x_1 = & 1 \\ x_2 = & 1 \\ x_3 = & 0 \\ x_4 = & 0 \\ x_5 = & 1 \end{bmatrix} \succeq \begin{bmatrix} x_5 = & 1 \\ x_1 = & 1 \\ x_4 = & 0 \\ x_3 = & 0 \\ x_2 = & 1 \end{bmatrix} = \gamma(x)$$

Simpler case: Propagation for a single permutation

Lemma

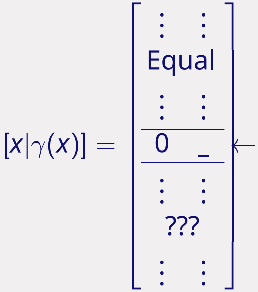
For γ permuting $\{1, \ldots, n\}$, the complete set of fixings of $x \succeq \gamma(x)$ can be found in $\mathcal{O}(n)$ time.

Proof.

Compare x_i and $\gamma(x)_i$, starting at $i = 1$, for increasing i up to n.
9 possible situations of $(x_i, \gamma(x)_i)$:

- $(0, \_)$ or $(\_, 1)$;
- $(0, 0)$ or $(1, 1)$;
- $(1, 0)$;
- $(0, 1)$;
- $(1, \_)$, $(\_, 0)$ or $(\_, \_)$.

□



TU/e

Simpler case: Propagation for a single permutation

Lemma

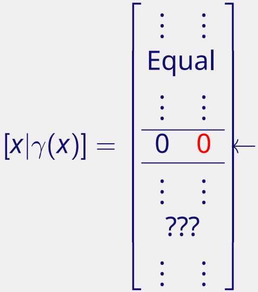
For γ permuting $\{1, \ldots, n\}$, the complete set of fixings of $x \succeq \gamma(x)$ can be found in $\mathcal{O}(n)$ time.

Proof.

Compare x_i and $\gamma(x)_i$, starting at $i = 1$, for increasing i up to n.
9 possible situations of $(x_i, \gamma(x)_i)$:

- $(0, \_)$ or $(\_, 1)$;
- $(0, 0)$ or $(1, 1)$;
- $(1, 0)$;
- $(0, 1)$;
- $(1, \_)$, $(\_, 0)$ or $(\_, \_)$.

□



Simpler case: Propagation for a single permutation

Lemma

For γ permuting $\{1, \dots, n\}$, the complete set of fixings of $x \succeq \gamma(x)$ can be found in $\mathcal{O}(n)$ time.

Proof.

Compare x_i and $\gamma(x)_i$, starting at $i = 1$, for increasing i up to n.
9 possible situations of $(x_i, \gamma(x)_i)$:

- $(0, \_)$ or $(\_, 1)$;
- $(0, 0)$ or $(1, 1)$;
- $(1, 0)$;
- $(0, 1)$;
- $(1, \_)$, $(\_, 0)$ or $(\_, \_)$. Can make $(1, 0)$. Test for $(0, 0)$ and $(1, 1)$. $\square$

$$[x|\gamma(x)] = \begin{bmatrix} \vdots & \vdots \\ \multicolumn{2}{c}{\text{Equal}} \\ \vdots & \vdots \\ \hline \_ & \_ \\ \hline \vdots & \vdots \\ \multicolumn{2}{c}{???} \\ \vdots & \vdots \end{bmatrix} \leftarrow$$

Simpler case: Propagation for a single permutation

Lemma

For γ permuting $\{1, \ldots, n\}$, the complete set of fixings of $x \succeq \gamma(x)$ can be found in $\mathcal{O}(n)$ time.

Proof.

Compare x_i and $\gamma(x)_i$, starting at $i = 1$, for increasing i up to n.
9 possible situations of $(x_i, \gamma(x)_i)$:

- $(0, \_)$ or $(\_, 1)$;
- $(0, 0)$ or $(1, 1)$;
- $(1, 0)$;
- $(0, 1)$;
- $(1, \_)$, $(\_, 0)$ or $(\_, \_)$. Can make $(1, 0)$. Test for $(0, 0)$ and $(1, 1)$. $\square$

$$[x|\gamma(x)] = \begin{bmatrix} \vdots & \vdots \\ \multicolumn{2}{c}{\text{Equal}} \\ \vdots & \vdots \\ \hline \underline{1} & \underline{0} \\ \hline \vdots & \vdots \\ \multicolumn{2}{c}{???} \\ \vdots & \vdots \end{bmatrix} \leftarrow$$

Simpler case: Propagation for a single permutation

Lemma

For γ permuting $\{1, \ldots, n\}$, the complete set of fixings of $x \succeq \gamma(x)$ can be found in $\mathcal{O}(n)$ time.

Proof.

Compare x_i and $\gamma(x)_i$, starting at $i = 1$, for increasing i up to n.
9 possible situations of $(x_i, \gamma(x)_i)$:

- $(0, \_)$ or $(\_, 1)$;
- $(0, 0)$ or $(1, 1)$;
- $(1, 0)$;
- $(0, 1)$;
- $(1, \_)$, $(\_, 0)$ or $(\_, \_)$. Can make $(1, 0)$. Test for $(0, 0)$ and $(1, 1)$. $\square$

$$[x|\gamma(x)] = \begin{bmatrix} \vdots & \vdots \\ \multicolumn{2}{c}{\text{Equal}} \\ \vdots & \vdots \\ \hline \underline{0} & \underline{0} \\ \hline \vdots & \vdots \\ \multicolumn{2}{c}{???} \\ \vdots & \vdots \end{bmatrix} \leftarrow$$

TU/e

Propagation for a set of permutations

Lemma (Previous result)

For γ permuting $\{1, \ldots, n\}$, the complete set of fixings of constraint $x \succeq \gamma(x)$ can be found in $\mathcal{O}(n)$ time.

Input: A set S of permutations on $\{1, \ldots, n\}$.

Corollary (Naive propagation loop)

The complete set of fixings for each (individual) constraint $x \succeq \gamma(x)$ for all $\gamma \in S$ can be found in $\mathcal{O}(n^2|S|)$ time.

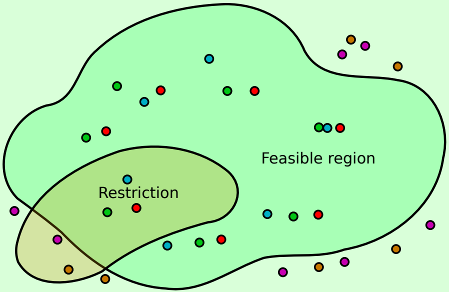
Can do better. $\rightsquigarrow$ $\mathcal{O}(n|S|)$

Solution approach

- Given: A problem symmetry group Γ of the program;
- Restrict region: Symmetrical solutions have (at least) a single representative:
  - Constraint for lexicographic leaders in Γ-orbit:

$$x \succeq \gamma(x) \text{ for all } \gamma \in \Gamma.$$

- Enforce: by propagation.



Meta-algorithm: Find the complete set of fixings

Algorithm: Complete propagation algorithm for lexicographic leaders of Γ-orbit.
Input: Symmetry group Γ, and an initial set of fixings.

if *There are no lexicographic leaders in the Γ-orbit respecting the fixings* **then**
    return INFEASIBLE;
foreach *Entry i with x_i unfixed* **do**
    if *There are no lexicographic leaders y in the Γ-orbit respecting the fixing and* $y_i = 1$ **then**
        Fix x_i to 0.
    if *There are no lexicographic leaders y in the Γ-orbit respecting the fixing and* $y_i = 0$ **then**
        Fix x_i to 1.
return FEASIBLE, and complete set of fixings;

Meta-algorithm: Find the complete set of fixings

Algorithm: Complete propagation algorithm for lexicographic leaders of Γ-orbit.
Input: Symmetry group Γ, and an initial set of fixings.

```
1 if There are no lexicographic leaders in the Γ-orbit respecting the fixings then
2     return INFEASIBLE;
3 foreach Entry i with x_i unfixed do
4     if There are no lexicographic leaders y in the Γ-orbit respecting the fixing and y_i = 1 then
5         Fix x_i to 0.
6     if There are no lexicographic leaders y in the Γ-orbit respecting the fixing and y_i = 0 then
7         Fix x_i to 1.
8 return FEASIBLE, and complete set of fixings;
```

- Oracle for lines 1, 4, 6: **"Does a lexicographic leader exists respecting fixings?"**
- Meta-algorithm takes $\mathcal{O}(n \cdot f(\Gamma, n))$ time, where $f(\Gamma, n)$ is the oracle complexity.

Meta-algorithm: Find the complete set of fixings

Meta-algorithm takes $\mathcal{O}(n \cdot f(\Gamma, n))$ time, where $f(\Gamma, n)$ is the oracle complexity.

“Does a lexicographic leader in the Γ-orbit exist that respects initial fixings?”

- Special case: If all variables are fixed at 0 or 1, testing this is coNP-complete:

Theorem (Babai and Luks (1983), Luks and Roy (2002))

“The problem of testing whether a 0/1 string X is the lexicographic leader in its Γ-orbit is coNP-complete.”

- Restriction to classes of groups Γ.

Special case: A cyclic group with monotone representation

“Does a lexicographic leader in the Γ-orbit exist that respects initial fixings?”

Special case: $\Gamma \leq \langle(1, 2, \dots, n)\rangle$. (*monotone* cycle)

Special case: A cyclic group with monotone representation

"Does a lexicographic leader in the Γ-orbit exist that respects initial fixings?"

Special case: $\Gamma \leq \langle(1, 2, \ldots, n)\rangle$. (*monotone* cycle)

Proposition (Feasibility statement for monotone cycles)

Let $\Gamma \leq \langle(1, \ldots, n)\rangle$, a set of variable fixings, and $F \subseteq \{0, 1\}^n$ the set of all binary vectors respecting the variable fixings.

If the set of fixings is complete for $x \succeq \gamma(x)$ for each $\gamma \in \Gamma$, then:

For all $\gamma \in \Gamma$ there exists an $x \in F$ with $x \succeq \gamma(x)$	*(Feasible vector exists for all $x \succeq \gamma(x)$)*
$\iff$	$\iff$
There exists an $x \in F$ with $x \succeq \gamma(x)$ for all $\gamma \in \Gamma$.	*(Lex-leader exists in Γ-orbit.)*

Collecting results

- Oracle: "Does a lex. leader in Γ-orbit exist, respecting initial fixings?": $\mathcal{O}(f(n,\Gamma))$.
- Finding the complete set of fixings for lex. leaders in the Γ-orbit: $\mathcal{O}(n \cdot f(n,\Gamma))$.

Oracle for $\Gamma \leq \langle(1,\dots,n)\rangle$:

1. Proposition: If the set of fixings is complete for $x \succeq \gamma(x)$ for each individual $\gamma \in \Gamma$, then:

   Feasible vector exists for all $x \succeq \gamma(x)$

   $\iff$

   Lex-leader exists in Γ-orbit.

2. Compute complete set of fixings for $x \succeq \gamma(x)$ for each individual $\gamma \in \Gamma$: $\mathcal{O}(n \cdot \mathrm{ord}(\Gamma))$

$$\rightsquigarrow f(n,\Gamma) = \mathcal{O}(n\,\mathrm{ord}(\Gamma)) = \mathcal{O}(n^2)$$

Proof of proposition

Proposition (Feasibility statement for monotone cycles)

Let $\Gamma \leq \langle(1,\dots,n)\rangle$, *and* $F \subseteq \{0,1\}^n$ *a set of binary solution vectors with variable fixings. If the set of fixings is complete for* $x \succeq \gamma(x)$ *for each* $\gamma \in \Gamma$, *then:*

$$\forall \gamma \in \Gamma\ \exists x \in F : x \succeq \gamma(x) \iff \exists x \in F\ \forall \gamma \in \Gamma : x \succeq \gamma(x).$$

Proof idea.

$\Leftarrow$: Trivial.
$\Rightarrow$: If $|F| = 1$, then no unfixed entries. Trivial.

Proof of proposition

Proposition (Feasibility statement for monotone cycles)

Let $\Gamma \leq \langle(1,\dots,n)\rangle$, *and* $F \subseteq \{0,1\}^n$ *a set of binary solution vectors with variable fixings. If the set of fixings is complete for* $x \succeq \gamma(x)$ *for each* $\gamma \in \Gamma$, *then:*

$$\forall \gamma \in \Gamma\ \exists x \in F : x \succeq \gamma(x) \iff \exists x \in F\ \forall \gamma \in \Gamma : x \succeq \gamma(x).$$

Proof idea.

$\Rightarrow$: If $|F| > 1$, then unfixed entries exist.

Construct $\tilde{x} \in F$ with $\tilde{x}_i = \begin{cases} 1, & \text{if } i \text{ is a 1-fixing or the first unfixed entry,} \\ 0, & \text{if } i \text{ is a 0-fixing or not the first unfixed entry.} \end{cases}$

Claim: $\tilde{x} \in F$ is a certificate. $\square$

A stronger result

Proposition (Last slide)

For $\Gamma \leq \langle (1, \dots, n) \rangle$, the complete set of fixings for lexicographic leaders in the Γ-orbit can be determined in $\mathcal{O}(n^3)$ time.

Proposition (Stronger result)

- $\Gamma \leq \langle \zeta_1 \circ \zeta_2 \circ \cdots \circ \zeta_k \rangle$ *with maximal cycle length m,*
- *each subcycle ζ_i ($i \in \{1, \dots, k\}$) has exactly one descend point* (monotone), *and*
- *for $i, j \in \{1, \dots, k\}$ with $i < j$ the entries of ζ_i are smaller than the entries of ζ_j* (ordered).

The complete set of fixings for lexicographic leaders in the Γ-orbit can be determined in $\mathcal{O}(n^2 m)$ time.

Computational study

Practical concerns

- Generators do not need to be monotone and ordered:
  For example: $(1, 10)(2, 5, 4)(3, 11, 8, 7, 6, 9)$.

- Symmetry groups could have more than one generator:

Practical concerns

- Generators do not need to be monotone and ordered:
  For example: $(1, 10)(2, 5, 4)(3, 11, 8, 7, 6, 9)$.
  - Preprocessing step *(Relabeling)*: Relabel variable indices

$$(1, 10)(2, 5, 4)(3, 11, 8, 7, 6, 9) \rightsquigarrow (1, 2)(3, 4, 5)(6, 7, 8, 9, 10, 11)$$

- Symmetry groups could have more than one generator:

Practical concerns

- Generators do not need to be monotone and ordered:
  For example: $(1, 10)(2, 5, 4)(3, 11, 8, 7, 6, 9)$.
  - Preprocessing step *(Relabeling)*: Relabel variable indices

$$(1, 10)(2, 5, 4)(3, 11, 8, 7, 6, 9) \rightsquigarrow (1, 2)(3, 4, 5)(6, 7, 8, 9, 10, 11)$$

- Symmetry groups could have more than one generator:
  - Add more symmetry handling constraints for various subgroups.

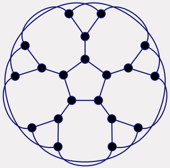
Computational setup

Implemented as SCIP plugin.

- SCIP: “Solving Constraint Integer Programs” (Academic solver);
- Compatible symmetry handling techniques;
- For monotone and ordered generators:
  - Strong variant: Complete: $\mathcal{O}(n^2 m)$;
  - Weak variant: Complete for all individual permutations: $\mathcal{O}(nm)$.
- For arbitrary cyclic groups Γ (No guarantees):
  - Strong variant: $\mathcal{O}(n^2 \operatorname{ord}(\Gamma))$;
  - Weak variant: $\mathcal{O}(n \operatorname{ord}(\Gamma))$.

Computational results: Test instances

- Verify that flower snark graphs are not 3-edge-colorable;
- MIPLIB2010 + MIPLIB2017 instances.



Flower snark J_5

Computational results: Configuration

Options:

- `nosym`: No symmetry handling;
- `gen`: Propagate $x \succeq \tilde{\gamma}(x)$ for a generator $\tilde{\gamma}$ (SCIP default choice), $\mathcal{O}(n)$;
- `group`: Propagate $x \succeq \gamma(x)$ for all cyclic subgroup members $\gamma \in \langle\tilde{\gamma}\rangle$, $\mathcal{O}(n^2 \operatorname{ord}(\langle\tilde{\gamma}\rangle))$;
- `nopeek`: Propagate $x \succeq \gamma(x)$ for all cyclic subgroup members $\gamma \in \langle\tilde{\gamma}\rangle$, $\mathcal{O}(nm)$ †;
- `peek`: Strong variant, $\mathcal{O}(n^2 m)$ †.

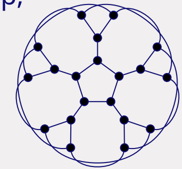
Relabeling:

- `original`: Respect original variable relabeling;
- `max`, `min`: Largest/Smallest cycles go first;
- `respect`: Sort by first entry of cycle in original relabeling.

†: m is maximal cycle length of generator $\tilde{\gamma}$. Assuming $\tilde{\gamma}$ is monotone and ordered generator.

Edge 3-coloring flower snark instances

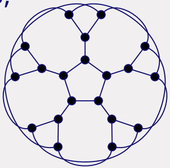
- Family of graphs J_n with graph automorphism having cyclic subgroup;
- 5 runs per instance and setting, with different random seed;
- Instances are infeasible.



Flower snark J_5

Edge 3-coloring flower snark instances

- Family of graphs J_n with graph automorphism having cyclic subgroup;
- 5 runs per instance and setting, with different random seed;
- Instances are infeasible.



Flower snark J_5

Conclusions:

- Results are very sensitive to chosen relabeling;
- On average, the weak and strong method are at least 5% faster than `group`;
- Strong method (peek) is costs significant time, but is effective.

For parameters $n \in \{3, 5, 7, \ldots, 49\}$

	nosym		gen		group		nopeek		peek	
relabeling	time(s)	S	time(s)	S	time(s)	S	time(s)	S	time(s)	S
original	730.78	54	172.35	88	187.56	87	169.79	93	153.23	97
max	–		407.02	65	312.93	70	278.00	77	270.19	78
min	–		97.99	102	131.58	95	127.48	92	119.67	95
respect	–		184.44	88	173.41	86	174.32	88	178.46	87
aggregated	730.78	54	189.90	343	191.75	338	180.33	350	172.84	357
relative to group	+281.1%		-1.0%		–		-5.6%		-9.9%	

For parameters $n \in \{27, 29, \ldots, 49\}$ **(Not solvable by** nosym**)**

	nosym		gen		group		nopeek		peek	
relabeling	time(s)	S	time(s)	S	time(s)	S	time(s)	S	time(s)	S
original	7200.00	0	1266.33	33	1526.99	32	1269.40	38	1056.42	42
max	–		4752.76	10	3501.50	15	2884.53	22	2734.31	23
min	–		531.09	47	912.18	40	889.73	37	797.14	40
respect	–		1480.93	33	1444.56	31	1452.88	33	1509.23	32
aggregated	7200.00	0	1478.12	123	1630.32	118	1475.85	130	1366.37	137
relative to group	+341.6%		-9.4%		–		-9.5%		-16.2%	

Time limit 7200s; Using shifted geometric mean (+10s)

Benchmark instances MIPLIB2010, MIPLIB2017

MIPLIB2010 + MIPLIB2017:

- 1427 instances;
- 35 nontrivial instances with cyclic symmetry structure;
- 11 solvable by some method in 7200s.

Benchmark instances MIPLIB2010, MIPLIB2017

MIPLIB2010 + MIPLIB2017:

- 1427 instances;
- 35 nontrivial instances with cyclic symmetry structure;
- 11 solvable by some method in 7200s.

Conclusions:

- Results are very sensitive to chosen relabeling;
- Variation between different instances is huge;
- nopeek and peek are slower than `group`.
- Without instances that are solved fastest by `nosym`, the weak and strong methods are at least 4% faster than `group`.

TU/e

All relevant instances

	nosym		gen		group		nopeek		peek	
relabeling	time(s)	S	time(s)	S	time(s)	S	time(s)	S	time(s)	S
original	1853.78	42	491.69	50	407.76	50	449.34	48	506.23	48
max	–		1061.29	47	812.31	48	771.56	48	796.69	49
min	–		901.65	48	612.10	50	837.48	47	657.29	49
respect	–		970.19	47	743.01	49	639.20	50	723.93	49
aggregated	1853.78	42	822.47	192	623.37	197	656.66	193	662.01	195
relative to group	+197.4%		+31.9%		–		+5.3%		+6.2%	

Without instances solved fastest by nosym (*neos-3004026-krka*, *neos-920392*, and *supportcase29*)

	nosym		gen		group		nopeek		peek	
relabeling	time(s)	S	time(s)	S	time(s)	S	time(s)	S	time(s)	S
original	3917.80	27	501.57	35	393.39	35	337.55	35	338.57	35
max	–		1193.75	33	668.57	35	694.06	35	698.20	35
min	–		1062.99	33	719.87	35	710.16	35	706.24	35
respect	–		1226.05	33	761.70	35	678.57	35	715.09	35
aggregated	3917.80	27	940.64	134	616.62	140	580.20	140	588.38	140
relative to group	+535.4%		+52.5%		–		-5.9%		-4.6%	

Time limit 7200s; Using shifted geometric mean (+10s)

TU/e

Overview of the results

Handling symmetries in (binary) integer linear programs by propagation:

- Complete set of fixings of $x \succeq \gamma(x)$ for a permutation γ on $\{1, \ldots, n\}$; $\mathcal{O}(n)$
- Complete set of fixings of $x \succeq \gamma(x)$ for all γ on $\{1, \ldots, n\}$ in a set S; $\mathcal{O}(n|S|)$
- Complete set of fixings of lexicographic leaders in cyclic Γ-orbit, with Γ generated by a monotone and ordered permutation with maximal subcycle length m. $\mathcal{O}(n^2 m)$
- Arbitrary cyclic groups Γ: No guarantee of completeness:
  - Stronger version: $\mathcal{O}(n^2 \operatorname{ord}(\Gamma))$; Weaker version: $\mathcal{O}(n \operatorname{ord}(\Gamma))$.

Computational results:

- Effectiveness is measurable in various instances:
  - Flower snark 3-edge-coloring;
  - MIPLIB instances.

Thank you!

References

László Babai and Eugene M Luks.
Canonical labeling of graphs.
In *Proceedings of the fifteenth annual ACM symposium on Theory of computing*, pages 171–183, 1983.

Eugene Luks and Amitabha Roy.
The complexity of symmetry-breaking formulas.
Annals of Mathematics and Artificial Intelligence, 41, 08 2002.

François Margot.
Symmetric ILP: Coloring and small integers.
Discrete Optimization, 4(1):40–62, 2007.