Warm Starting of Mixed Integer Linear Optimization Problems via Parametric Disjunctive Cuts

Shannon Kelley, Aleks Kazachkov, Ted Ralphs

October 11, 2024

Overview

Motivation

Disjunctive cuts can be strong but often expensive. Can we retain their strength while reducing their cost?
For series of similar MILPs, we can accomplish both through parameterization!

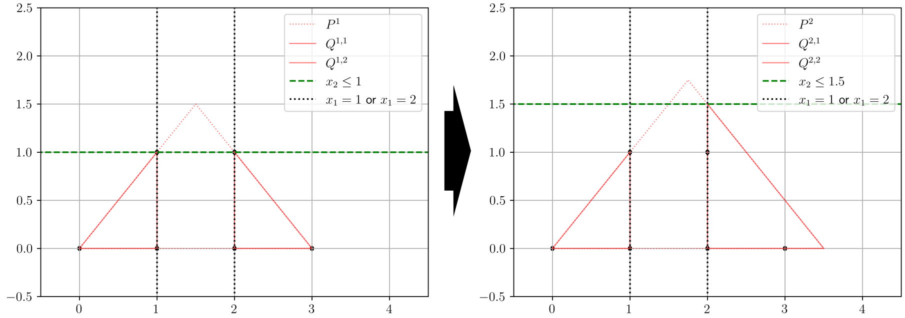

Figure 1: (Left) given $\mathcal{P}^1$ and a disjunction $x_1 \leq 1 \vee x_1 \geq 2$, we parameterize disjunctive cut $x_2 \leq 1$ to generate (Right) $x_2 \leq 1.5$, which is valid for the disjunction applied to $\mathcal{P}^2$.

Further Motivation and Outline

- Disjunctive cuts improve MILP solvers' ability to close optimality gap compared to default cutting planes [1].
- But, they are inconsistent in improving solver run time [1].
- Parameterization significantly reduces disjunctive cut generation time [5].
- For series of MILPs sharing the same variables and number of constraints, this reduction can improve solvers' overall performance [5].
- Applications within MIP are more common than one might think!
  - Branch-and-Price
  - Lagrangian Dual Decomposition
  - Multi-Objective
  - Bilevel

In this presentation, we detail:

- how to parameterize disjunctive cuts and expectations on effectiveness
- empirical impact parameterized disjunctive cuts have on solving MILPs

Input

We define the following:

- $\mathcal{P}^k := \{x \in \mathbb{R}^n : A^k x \geq b^k\}$.
- $\mathcal{S}^k := \{x \in \mathcal{P}^k : x_j \in \mathbb{Z}\ \forall\, j \in I\}$.
- $\mathcal{X}^t := \{x \in \mathbb{R}^n : D^t \geq D_0^t\}$.
- $A^{kt} := \begin{bmatrix} A^k \\ D^t \end{bmatrix}$ and $b^{kt} := \begin{bmatrix} b^k \\ D_0^t \end{bmatrix}$.
- $\mathcal{Q}^{kt} := \mathcal{P}^k \cap \mathcal{X}^t$
  $= \{x \in \mathbb{R}^n : A^{kt} x \geq b^{kt}\}$.
- $\bar{x}^{kt} := \arg\min_{x \in \mathcal{Q}^{kt}} c^k x$.
- $\bar{N}^{kt}$ indexes constraints tight at $\bar{x}^{kt}$.
- $\bar{\mathcal{C}}^{kt} := \{x \in \mathbb{R}^n : A^{kt}_{\bar{N}^{kt},*} x \geq b^{kt}\}$

$$\begin{aligned} \max_{x \in \mathbb{R}^n} \quad & c^k x \\ & A^k x \geq b^k \\ & x_j \in \mathbb{Z} \qquad \forall j \in \mathcal{I} \end{aligned} \qquad (IP_k)$$

- $A^k \in \mathbb{R}^{q \times n}$ and $b^k \in \mathbb{R}^q$ for all $k \in K$.
- $\{\mathcal{X}^t\}_{t \in T}$ is a **disjunction**.
- $\{\mathcal{X}^t\}_{t \in T}$ is **valid** for a set $\mathcal{S}^k \in \mathbb{R}^n$ if $\mathcal{S}^k \subseteq \bigcup_{t \in T} \mathcal{X}^t$.
- $\bar{\mathcal{C}}^{kt}$ is an optimal **basis cone**.

How to Generate Disjunctive Cuts

We can generate a $\mathcal{V}$-*polyhedral* disjunctive cut (VPC) by solving the Point-Ray LP (PRLP):

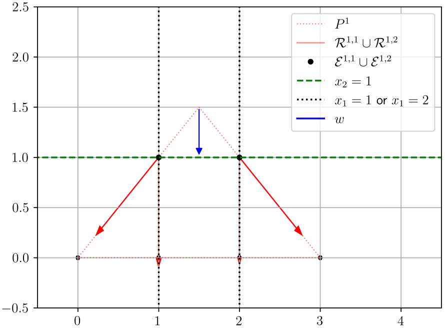

Figure 2: Solving PRLP for IP_1 with $w = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$ yields the VPC $x_2 \leq 1$.

$$
\begin{aligned}
\max_{(\alpha,\beta)\in\mathbb{R}^{n+1}} \quad & \alpha^\mathsf{T} w \\
& \alpha^\mathsf{T} p \geq \beta \quad \forall p \in \cup_{t\in T} \mathcal{E}^{kt} \\
& \alpha^\mathsf{T} r \geq 0 \quad \forall r \in \cup_{t\in T} \mathcal{R}^{kt}
\end{aligned}
\tag{PRLP}
$$

For all $k \in K$ and $t \in T$:

- $\mathcal{E}^{kt}$:= extreme point of $\bar{\mathcal{C}}^{kt}$.
- $\mathcal{R}^{kt}$:= extreme rays of $\bar{\mathcal{C}}^{kt}$.

How to Find Farkas Multipliers

Lemma 1

Let (α, β) be a valid cut for IP_k and $\{\mathcal{X}^t\}_{t \in T}$ be a disjunction. Then there exists v^t such that

$$\left.\begin{aligned} \alpha^T &= v^t A^{kt} \\ \beta &\leq v^t b^{kt} \\ v^t &\geq 0 \end{aligned}\right\} \text{ for all } t \in T. \text{ [3]}$$

We refer to $\{v^t\}_{t \in T}$ as **Farkas multipliers**.

Lemma 2

Let $k \in K$ and $\{\mathcal{X}^t\}_{t \in T}$ be a disjunction. Let $t \in T$ and $\alpha^T x \geq \beta$ be valid for all $x \in \bar{\mathcal{C}}^{kt}$. Then v_i^t, the Farkas multiplier on constraint i, is calculated:

- $\alpha^\intercal (A^{kt}_{\bar{N}^{kt},*})^{-1}_{*,h}$, *for $i \in \bar{N}^{kt}$ and h such that $\bar{N}^{kt}_h = i$.*
- 0, *otherwise. [2]*

How to Parameterize Disjunctive Cuts

Theorem 3

Let $\{v^t\}_{t \in T}$ be a set of nonnegative Farkas multipliers for a disjunction $\{\mathcal{X}^t\}_{t \in T}$ valid for $\mathbb{Z}^n$. For $\ell \in K$ and for all $j \in [n]$, let $\alpha_j := \max_{t \in T}\{v^t A^{\ell t}_{.j}\}$ and $\beta := \min_{t \in T}\{v^t b^{\ell t}\}$. Then $\alpha^T x \geq \beta$ is valid for all $x \in \mathcal{S}^\ell$. [5]

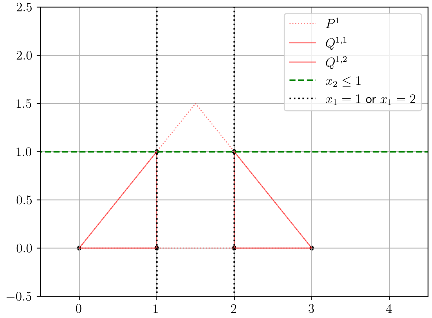

Figure 3: Generate VPC $x_2 \leq 1$ for IP_1 and calculate Farkas multipliers $\{v^1, v^2\}$.

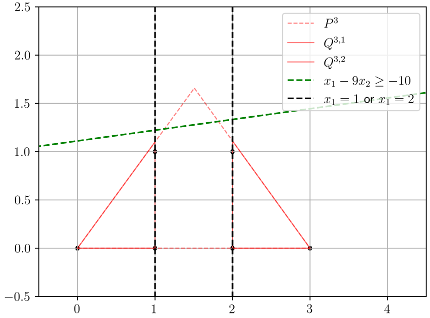

Figure 4: Apply Theorem 3 to IP_3 and $\{v^1, v^2\}$, generating $x_1 - 9x_2 \geq -10$.

Expectation Setting

Definition 4

The pair $(x, \{\mathcal{X}^t\}_{t \in T})$ is a *certificate* for IP_k when $\min_{t \in T} c^k \bar{x}^{kt} = c^k x$.

Definition 5

The *Warm-Started MILP* takes as inputs IP_k, IP_ℓ, and a certificate $(x, \{\mathcal{X}^t\}_{t \in T})$ for IP_k such that exactly one of the following statements is true:

- $c^k \neq c^\ell$, or
- there exists exactly one $i \in [q]$ such that $A^\ell_{i\cdot} \neq A^k_{i\cdot}$ or $b^\ell_i \neq b^k_i$.

It returns a certificate $(\bar{x}, \{\bar{\mathcal{X}}\}_{t \in \bar{T}})$ for IP_ℓ if $\mathcal{S}^\ell \neq \emptyset$ and null otherwise.

Theorem 6

The Warm-Started MILP is NP-Hard. [5]

Translation: No warm-start can improve the complexity class of solving MILPs.

Expectation Setting

Theorem 7

Let $\{\mathcal{X}^t\}_{t\in T}$ *be a disjunction and* $\epsilon > 0$. *Let* IP_k *and* IP_ℓ *be such that*

- $\min_{t\in T} \min_{x\in \mathcal{Q}^{kt}}\{c^k \bar{x}^{kt}\} > c^\ell \bar{x}^k$
- $A^\ell = A^k + e_{i_A, j_A}\epsilon$, $b^\ell = b^k + e_{i_b}\epsilon$ *or* $b^\ell = c^k + e_{j_c}\epsilon$.

Then there exists $A \in \mathbb{R}^{q\times n}$, $b \in \mathbb{R}^q$, $c \in \mathbb{R}^n$, $(i_A, j_A) \in [q]\times[n]$, $i_b \in [q]$, *and* $j_c \in [n]$ *such that* $\min_{t\in T} \min_{x\in \mathcal{Q}^{\ell t}}\{c^k \bar{x}^{\ell t}\} = c^\ell \bar{x}^\ell$. *[5]*

Translation: Parameterized disjunctive cuts are not guaranteed to improve disjunctive dual bound.

Experimental Setup

We run an experiment as follows:

- Create a base test set from 104 presolved MIPLIB 2017 instances with at most 5000 variables and 5000 constraints.
- Create an experimental test set of 5 random perturbations of objective, RHS, and/or matrix for each instance in the base set.
- Use VPCs [1] as the disjunctive cut.
- Replications vary by the following parameters:
    - 4, 16, or 64 term disjunctions to generate VPCs
    - 0.5 or 2 degrees of random perturbation
    - run with no VPCs, VPCs via [1], or parameterized VPCs.
- Solve the experiment set for each combination of parameters using Gurobi 10.

Experimental Results (Root Node)

We compare the ability to close the optimality gap at the root node:

degree	terms	Average Root Optimality Gap Closed		
		No VPCs	VPCs via [1]	Param. VPCs
0.5	4	61.87%	62.35%	62.30%
	16	61.87%	62.96%	62.82%
	64	61.87%	63.55%	63.35%
2.0	4	63.46%	63.45%	63.36%
	16	63.46%	63.76%	63.53%
	64	63.46%	64.73%	63.91%

Parameterization maintains some of [1]'s ability to close additional root optimality gap with disjunctive cuts as compared to Gurobi's default cuts.

Experimental Results (Root Node)

We compare the time (in seconds) to process the root node:

degree	terms	Average Root Node Processing Time		
		No VPCs	VPCs via [1]	Param. VPCs
0.5	4	0.929	10.480	0.999
	16	0.936	29.483	1.394
	64	0.921	56.614	2.185
2.0	4	0.892	4.293	0.927
	16	0.870	17.576	1.394
	64	0.861	48.773	2.295

Parameterization significantly reduces the time to generate disjunctive cuts as compared to [1].

Experimental Results (Total Solve)

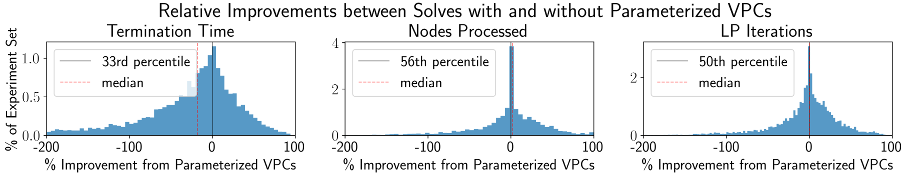

Figure 5: A significant portion of our experiment set sees performance improvements when parameterized VPCs are added to Gurobi's cut generators.

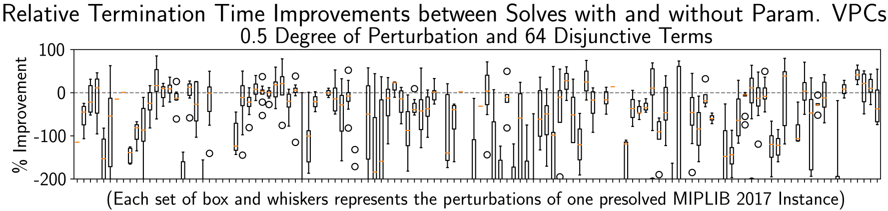

Figure 6: Time improvements appear to be random when fixing degree of perturbation, size of disjunction, and instance.

Future Directions and Conclusion

Parameterization:

- Reduces the cost of generating disjunctive cuts vs. [1].
- Often increases strength of root default cuts.
- Improves solver performance overall for many instances.

Next Steps:

- Improve efficiency of parameterization.
- Understand when parameterized disjunctive cuts help solver performance.

Parameterizing disjunctive cuts can improve a MILP solver's performance, but under what conditions remains an open question.

How to Tighten Parametric Disjunctive Cuts

Lemma 8

Let $k, \ell \in [K]$ such that $A^k = A^\ell$. Let $\{\mathcal{X}^t\}_{t \in T}$ be a disjunction and $\{v^t\}_{t \in T}$ be nonnegative Farkas multipliers derived while $\{\mathcal{X}^t\}_{t \in T}$ applied to IP_k. Let (α, β) be the result of applying Theorem 3 to $\{\mathcal{X}^t\}_{t \in T}$, $\{v^t\}_{t \in T}$, and IP_ℓ. Then (α, β) is tight for $\operatorname{cl}\operatorname{conv}(\cup_{t \in T} \mathcal{Q}^{\ell t})$. *[5]*

Theorem 9

Let $k, \ell \in [K]$. Let $\{\mathcal{X}^t\}_{t \in T}$ be a disjunction and $\{v^t\}_{t \in T}$ be nonnegative Farkas multipliers derived while $\{\mathcal{X}^t\}_{t \in T}$ applied to IP_k. Let $(\bar{\alpha}, \bar{\beta})$ be the result of applying Theorem 3 to $\{\mathcal{X}^t\}_{t \in T}$, $\{v^t\}_{t \in T}$, and IP_ℓ. If $A^k \neq A^\ell$, let:

- *$\{\bar{v}^t\}_{t \in T}$ be the Farkas multipliers derived for $(\bar{\alpha}, \bar{\beta})$ [2]*
- *(α, β) be the result of applying Theorem 3 to $\{\mathcal{X}^t\}_{t \in T}$, $\{\bar{v}^t\}_{t \in T}$, and IP_ℓ*

Else, let:

- $(\alpha, \beta) = (\bar{\alpha}, \bar{\beta})$

Then (α, β) is tight for $\operatorname{cl}\operatorname{conv}(\cup_{t \in T} \mathcal{Q}^{\ell t})$. *[5]*

How to Tighten Parametric Disjunctive Cuts

Visually, the application of Theorem 9 looks like the following:

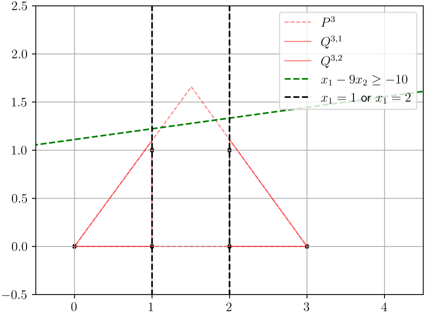

Figure 7: Apply Theorem 3 to IP_3 and $\{v^1, v^2\}$, generating $x_1 - 9x_2 \geq -10$.

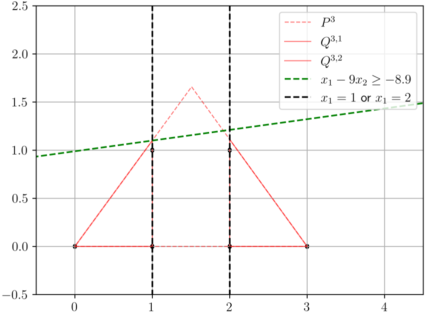

Figure 8: Calculate $\{\bar{v}^t\}_{t \in T}$ and reapply Theorem 3, generating $x_1 - 9x_2 \geq -8.9$.

Parameterizing $x_2 \leq 1$ with Theorem 9 yields $x_1 - 9x_2 \geq -8.9$, a **tight** cut for the convex hull of the disjunction applied to IP_3.

Finding a Basis for Infeasible Disjunctive Terms

		Average Root Optimality Gap Closed			Average Root Node Processing Time			Average % Perturbed
Degree	Terms	No VPCs	VPCs via [1]	Param. VPCs	No VPCs	VPCs via [1]	Param. VPCs	Terms Becoming Feasible
0.5	4	61.87%	62.35%	62.30%	0.929	10.480	0.999	0.000%
	16	61.87%	62.96%	62.82%	0.936	29.483	1.394	0.102%
	64	61.87%	63.55%	63.35%	0.921	56.614	2.185	0.201%
2	4	63.46%	63.45%	63.36%	0.892	4.293	0.927	0.000%
	16	63.46%	63.76%	63.53%	0.870	17.576	1.394	0.558%
	64	63.46%	64.73%	63.91%	0.861	48.773	2.295	0.596%

Figure 9: As disjunctions and degree of perturbation increase, so does the number of originally infeasible terms that become feasible.

Problem:

- Calculating v^t relies on $\mathcal{Q}^{kt} \neq \emptyset$.
- When $\mathcal{Q}^{kt} = \emptyset$, we currently set $v^t = 0$.
- Weakens parameterization when $\mathcal{Q}^{\ell t} \neq \emptyset$.

Possible solutions include using the basis from:

- Pivoting the last branching constraint into a feasible basis from the parent node.
- The solver's Farkas proof of infeasibility.
- $\bar{x}^{\ell t}$ for $\ell \in K$ such that $\mathcal{Q}^{\ell t} \neq \emptyset$.

Shrinking PRLP

```
CglVPC: Finishing with exit reason: PRLP_TIME_LIMIT: 0.22032726434662364
CglVPC: Finishing with exit reason: TIME_LIMIT: 0.04378889144964278
CglVPC: Finishing with exit reason: NO_CUTS_LIKELY: 0.46877160636091264
CglVPC: Finishing with exit reason: PRLP_INFEASIBLE: 0.0884996542982254
CglVPC: Finishing with exit reason: SUCCESS: 0.01866789582853192
CglVPC: Finishing with exit reason: OPTIMAL_SOLUTION_FOUND: 0.07605439041253745
CglVPC: Finishing with exit reason: FAIL_LIMIT: 0.011292924637013136
CglVPC: Finishing with exit reason: NO_DISJUNCTION: 0.07259737266651302
```

Figure 10: 22% of PRLPs fail due to hitting their time limits.

Problem:

- Currently, PRLP solves with all $\bar{\mathcal{C}}^{kt} \neq \emptyset$.
- Strong branching fixes create disjunctions larger than specified.
- This can make PRLP intractible.

Proposed solution:

- Solve PRLP with only $\bar{\mathcal{C}}^{kt}$ representing unprocessed nodes in solver.
- Apply Lemma 2 to remaining $\bar{\mathcal{C}}^{kt} \neq \emptyset$ to calculate their farkas multipliers.
- Risks weakening parameterization.

Pruning the Disjunction before Parameterization

Definition 10

An *optimality inequality* for MILP instance IP_k is a pair $(\alpha, \beta) \in \mathbb{R}^n \times \mathbb{R}$ such that $\alpha^\mathsf{T} x^* \geq \beta$ for all $x^* \in \arg IP_k$.

Theorem 11

Let $k, \ell \in [K]$. Let $\{\mathcal{X}^t\}_{t \in T}$ be a valid disjunction and $\{v^t\}_{t \in T}$ be farkas multipliers for an inequality $(\bar{\alpha}, \bar{\beta})$ valid for IP_k. Let $T' = \{t \in T : c^\ell \bar{x}^{\ell t} \leq c^\ell x^*\}$ such that $x^* \in \arg IP_\ell$. Then (α, β) output from Theorem 9 applied to IP_ℓ, $\{\mathcal{X}^t\}_{t \in T'}$, and $\{v^t\}_{t \in T'}$ is an optimality inequality for IP_ℓ. [5]

Translation: Parameterized disjunctive cuts may be tightened by ignoring disjunctive terms that are proven to not contain an optimal solution.

Enabling Cutting Planes While Generating Disjunctions

		Average Root Optimality Gap Closed			Average Root Node Processing Time			Average % Perturbed Terms Becoming Feasible
Degree	Terms	No VPCs	VPCs via [1]	Param. VPCs	No VPCs	VPCs via [1]	Param. VPCs	
0.5	4	61.87%	62.35%	62.30%	0.929	10.480	0.999	0.000%
	16	61.87%	62.96%	62.82%	0.936	29.483	1.394	0.102%
	64	61.87%	63.55%	63.35%	0.921	56.614	2.185	0.201%
2	4	63.46%	63.45%	63.36%	0.892	4.293	0.927	0.000%
	16	63.46%	63.76%	63.53%	0.870	17.576	1.394	0.558%
	64	63.46%	64.73%	63.91%	0.861	48.773	2.295	0.596%

Figure 11: Could we close more root optimality gap with a "better" disjunction?

Problem:

- Currently, default and disjunctive cuts refine the same root relaxation.
- Perhaps both have large overlaps in contributions.
- Maybe we could close more root optimality gap by reducing this overlap.

Proposed solution:

- Turn on cutting planes while generating a disjunction.
- Such disjunctions refine relaxation accounting for cuts at root.
- Go back and remove cuts from disjunction when generation complete.

Learning When Parametric Disjunctive Cuts Help

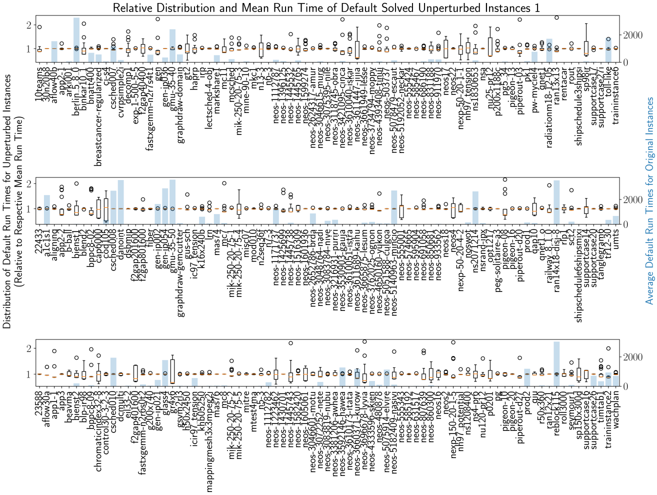

Figure 12: Intra-instance variance does not appear to explain Figure 6's inter-instance variance regarding run time.

Learning When Parametric Disjunctive Cuts Help

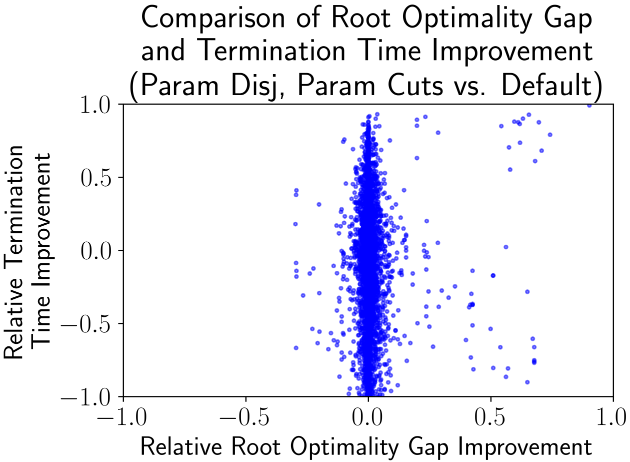

Figure 13: There is no relationship between the improvements to root optimality gap closed and run time.

Learning When Parametric Disjunctive Cuts Help

Goal: Determine when parametric disjunctive cuts improve run time.

Hypothesis: Not all perturbations of the same degree are created equal.

Perhaps we can identify when our parameterization helps by collecting the following:

- Number of pivots away new solutions are from warm start in root LP relaxation
- In disjunctive terms for which cuts are tight:
  - Number of pivots away new solutions are from warm start
  - Whether the optimal basis for each term includes the branching constraint that created it
- Open to suggestions (:

Comparing to Warm-Starting the Node Queue

Goal: Determine when parametric disjunctive cuts improve run time more than warm-starting with a previous disjunction.

- Branch-and-Cut warm-starts include initial primal solutions, pseudo costs, cuts, and disjunction.
- Want to compare warm-starting with:
  - Parameterized disjunctive cuts.
  - Previous terminal disjunctions. [4]
- Latter can process many unnecessary nodes.
- Perhaps our parameterization can be more effective for some problems since we only work with the root node.

Questions

Are there any changes you'd like to see?

Bibliography

[1] Egon Balas and Aleksandr M. Kazachkov. $\mathcal{V}$-polyhedral disjunctive cuts, 2022. URL `https://arxiv.org/abs/2207.13619`.

[2] Egon Balas and Aleksandr M. Kazachkov. Monoidal strengthening of $\mathcal{V}$-polyhedral disjunctive cuts, 2023. URL `https://optimization-online.org/2023/02/monoidal-strengthening-of-simple-v-polyhedral-disjunctive-`

[3] Julius Farkas. Theorie der einfachen Ungleichungen. *J. Reine Angew. Math.*, 124:1–27, 1902. URL `https://doi.org/10.1515/crll.1902.124.1`.

[4] M. Güzelsoy. *Dual Methods in Mixed Integer Linear Programming*. PhD, Lehigh University, 2009. URL `http://coral.ie.lehigh.edu/~ted/files/papers/MenalGuzelsoyDissertation09.pdf`.

[5] Shannon Kelley, Aleksandr Kazachkov, and Ted Ralphs. Warm starting of mixed integer linear optimization problems via parametric disjunctive cuts. 2024.