Exact hull reformulation for Quadratically Constrained Generalized Disjunctive Programs

Sergey Gusev, David E. Bernal Neira

Davidson School of Chemical Engineering, Purdue University

Publication available here

Background

Generalized Disjunctive Program (GDP) [1-4]

$$
\begin{aligned}
\min_{\mathbf{x},\mathbf{Y}} \quad & f(\mathbf{x}) \\
\text{s.t.} \quad & \mathbf{g}(\mathbf{x}) \le 0, \\
& \bigvee_{i\in D_k} \begin{bmatrix} Y_{ik} \\ h_{ik}(\mathbf{x}) \le 0 \end{bmatrix}, \quad \forall k \in K, \\
& \underline{\vee}_{i\in D_k} Y_{ik}, \quad \forall k \in K, \\
& \Omega(\mathbf{Y}) = \text{True}, \\
& \mathbf{x}^l \le \mathbf{x} \le \mathbf{x}^u, \\
& \mathbf{x} \in \mathbb{R}^n, \\
& Y_{ik} \in \{\text{False}, \text{True}\}, \quad \forall k \in K,\ i \in D_k.
\end{aligned}
$$

Hull reformulation of GDP

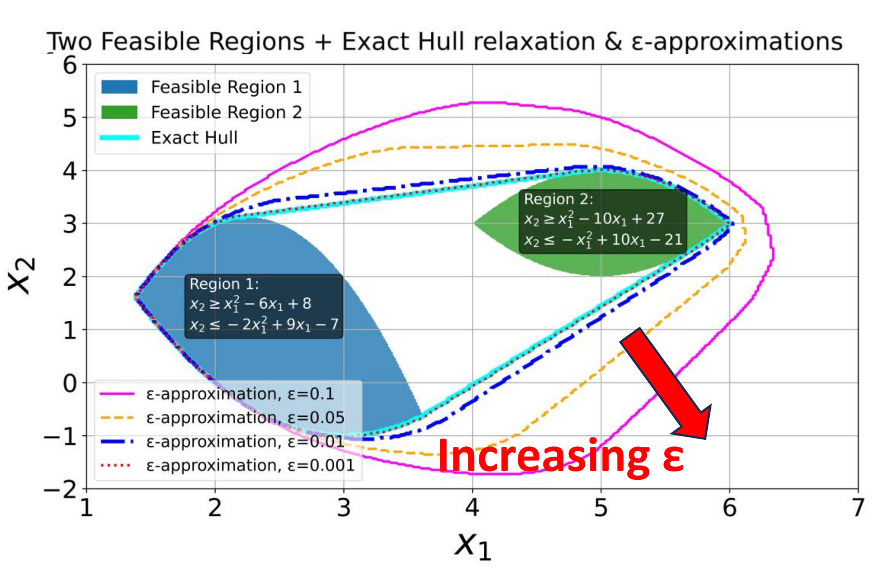
$$
\begin{aligned}
\min_{\mathbf{x},\mathbf{y}} \quad & f(\mathbf{x}) \\
\text{s.t.} \quad & \mathbf{g}(\mathbf{x}) \le 0, \\
& \sum_{i\in D_k} y_{ik} = 1 \quad \forall k \in K, \\
& \mathbf{E}\mathbf{y} \ge \mathbf{e}, \\
& \mathbf{x} = \sum_{i\in D_k} \mathbf{v}_{ik}, \quad k \in K \\
& \left(\text{cl}\,\widetilde{h}_{ik}\right)(\mathbf{v}_{ik}, y_{ik}) \le 0, \quad k \in K,\ i \in D_k \\
& \mathbf{v}_{ik} \in \mathbb{R}^n, \quad k \in K,\ i \in D_k \\
& \mathbf{x}^l \le \mathbf{x} \le \mathbf{x}^u, \\
& \mathbf{x} \in \mathbb{R}^n, \\
& y_{ik} \in \{0,1\} \quad \forall k \in K,\ i \in D_k.
\end{aligned}
$$

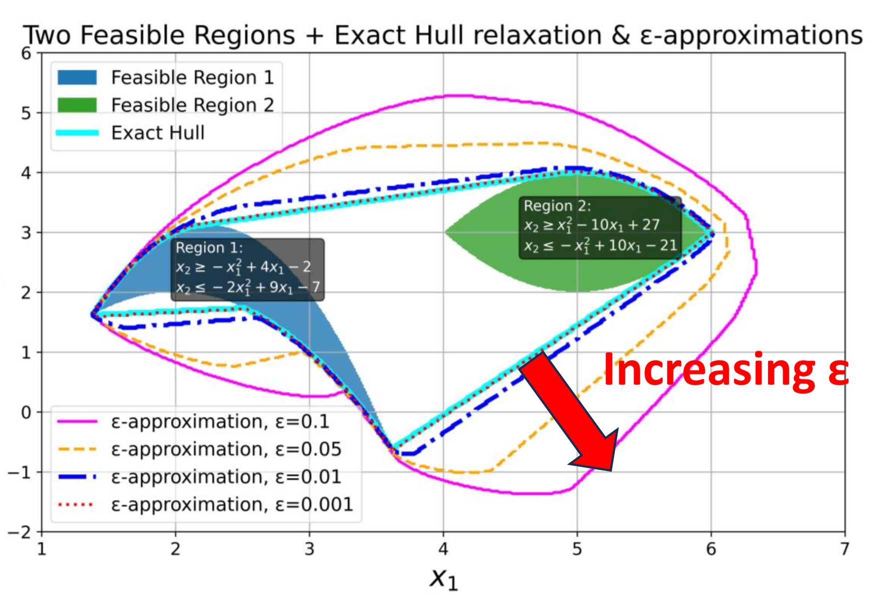
To calculate numerically the closure of a perspective function, an ε-approximation method (where ε is a small number, i.e. 10^{-4}) proposed by Furman, Sawaya, and Grossmann [5] is typically employed:

$$
\left(\text{cl}\,\widetilde{h}\right)(\mathbf{v}, y) \approx ((1-\varepsilon)y + \varepsilon)\, h\left(\frac{\mathbf{v}}{(1-\varepsilon)y+\varepsilon}\right) - \varepsilon\, h(0)(1-y),
$$

For the quadratic function $h(\mathbf{v}) = \mathbf{v}^\top Q\mathbf{v} + \mathbf{c}^\top\mathbf{v} + d$, where $Q \in \mathbb{R}^{n\times n}$, $\mathbf{c} \in \mathbb{R}^n$, and $d \in \mathbb{R}$, after simplification, an ε-approximation results in a general non-linear function:

$$
\left(\text{cl}\ \widetilde{h}\right)(\mathbf{v}, y) \approx \frac{\mathbf{v}^\top Q\mathbf{v}}{(1-\varepsilon)y+\varepsilon} + \mathbf{c}^\top\mathbf{v} + dy
$$





Exact Hull Reformulation of Quadratic GDP

Disjunctions of quadratically constrained GDP can be expressed as follows:

$$
\bigvee_{i\in D_k} \begin{bmatrix} Y_{ik} \\ \mathbf{x}^\top Q_{jik}\mathbf{x} + \mathbf{c}_{jik}^\top\mathbf{x} + d_{jik} \le 0,\ j \in J_{ik} \end{bmatrix}, \quad k \in K
$$

We propose the following reformulations:

Conic Exact Hull Reformulation (CEHR) for convex constraints:

$$
\begin{aligned}
& \mathbf{x} = \sum_{i\in D_k} \mathbf{v}_{ik}, \\
& \mathbf{x}^\ell y_{ik} \le \mathbf{v}_{ik} \le \mathbf{x}^u y_{ik}, \\
& \mathbf{v}_{ik}^\top Q_{jik}\mathbf{v}_{ik} \le t_{jik} y_{ik}, \\
& t_{jik} + \mathbf{c}_{jik}^\top\mathbf{v}_{ik} + d_{jik} y_{ik} \le 0, \\
& \mathbf{v}_{ik} \in \mathbb{R}^n, \quad t_{jik} \in \mathbb{R}_+.
\end{aligned}
$$

General Exact Hull Reformulation (GEHR) for nonconvex constraints:

$$
\begin{aligned}
& \mathbf{x} = \sum_{i\in D_k} \mathbf{v}_{ik}, \\
& \mathbf{v}_{ik}^\top Q_{jik}\mathbf{v}_{ik} + \mathbf{c}_{jik}^\top\mathbf{v}_{ik} y_{ik} + d_{jik} y_{ik}^2 \le 0, \\
& \mathbf{x}^\ell y_{ik} \le \mathbf{v}_{ik} \le \mathbf{x}^u y_{ik}, \\
& \mathbf{v}_{ik} \in \mathbb{R}^n.
\end{aligned}
$$

where $k \in K$ indexes disjunctions, $i \in D_k$ indexes disjuncts in disjunction k, and $j \in J_{ik}$ indexes quadratic constraints in disjunct i. Here, $\mathbf{v}_{ik}$ is the disaggregated variable, y_{ik} is the disjunct indicator, and Q_{jik}, $\mathbf{c}_{jik}$, and d_{jik} are the quadratic, linear, and constant coefficients.

Computational experiments

Computational experiments were conducted on various convex and non-convex quadratic GDPs, such as random convex and non-convex quadratic GDPs, convex k-means clustering, a non-convex CSTR network benchmark, and constrained layout instances.

Solver outcomes across reformulation strategies for **convex random** quadratic GDPs

Strategy	Optimal	Timeout	Infeasible	Objective mismatch	Total
Solver: `Gurobi`					
BigM	240	0	0	0	240
Hull ε-approx. ($\varepsilon = 10^{-4}$)	2	0	200	38	240
GEHR	240	0	0	0	240
CEHR	240	0	0	0	240
Binary Mult.	167	73	0	0	240
Solver: `BARON`					
BigM	233	7	0	0	240
Hull ε-approx. ($\varepsilon = 10^{-4}$)	5	233	0	2	240
GEHR	29	183	0	28	240
CEHR	20	190	0	30	240
Binary Mult.	180	60	0	0	240
Solver: `SCIP`					
BigM	240	0	0	0	240
Hull ε-approx. ($\varepsilon = 10^{-4}$)	47	182	0	11	240
GEHR	104	136	0	0	240
CEHR	240	0	0	0	240
Binary Mult.	94	146	0	0	240
Solver: `SCIP` **(convex flag)**					
BigM	240	0	0	0	240
Hull ε-approx. ($\varepsilon = 10^{-4}$)	238	0	0	2	240
GEHR	4	0	217	19	240
CEHR	240	0	0	0	240
Binary Mult.	0	0	219	21	240

Solver outcomes across reformulation strategies for non-**convex random** quadratic GDPs

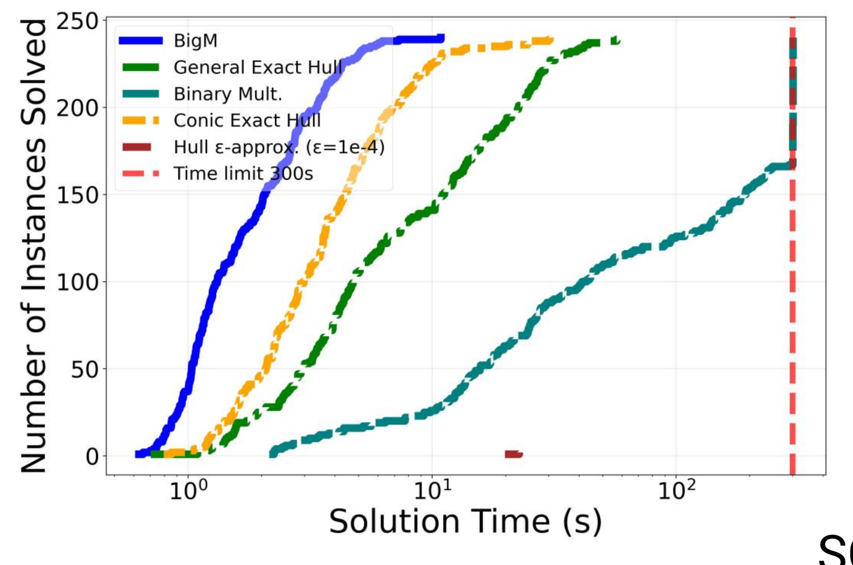
Strategy	Optimal	Timeout	Infeasible	Objective mismatch	Total
Solver: `Gurobi`					
BigM	91	9	0	0	100
Hull ε-approx. ($\varepsilon = 10^{-4}$)	2	86	0	12	100
GEHR	59	41	0	0	100
Binary Mult.	83	17	0	0	100
Solver: `BARON`					
BigM	95	4	0	1	100
Hull ε-approx. ($\varepsilon = 10^{-4}$)	0	99	1	0	100
GEHR	7	86	0	7	100
Binary Mult.	88	12	0	0	100
Solver: `SCIP`					
BigM	91	9	0	0	100
Hull ε-approx. ($\varepsilon = 10^{-4}$)	11	89	0	0	100
GEHR	41	59	0	0	100
Binary Mult.	39	61	0	0	100

Computational experiments

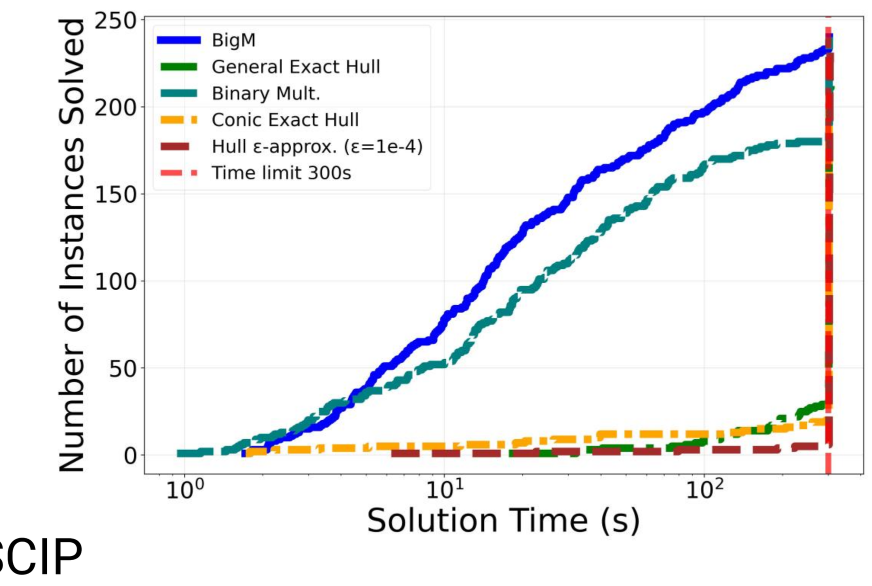
240 **Convex random instances** ($Q \succeq 0$) **:**

$3 \le |K| \le 10, \quad 10 \le |D_k| \le 15, \quad |J_{ik}| = 10, \quad 3 \le n \le 7$

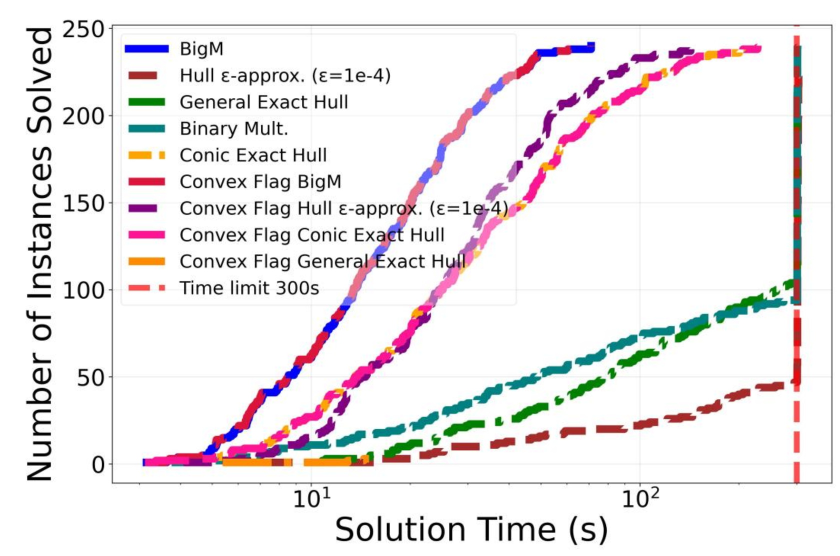
Gurobi



Baron



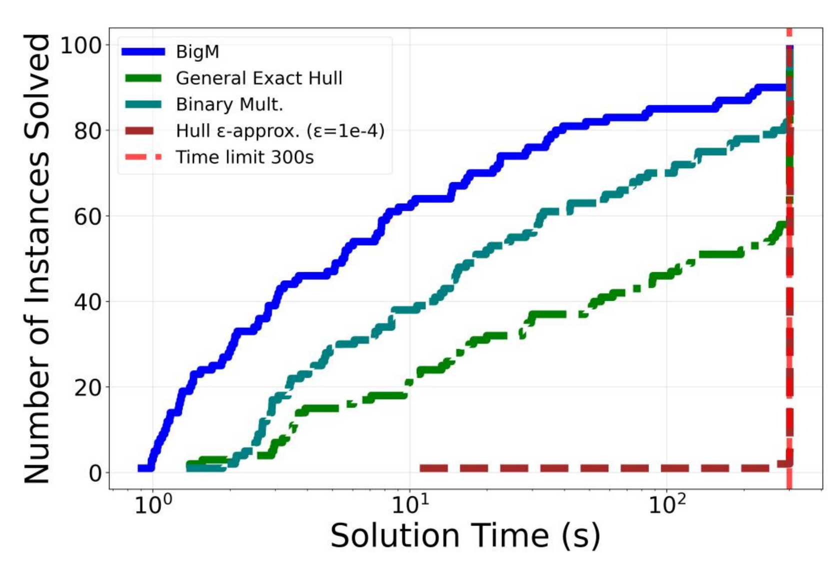
SCIP



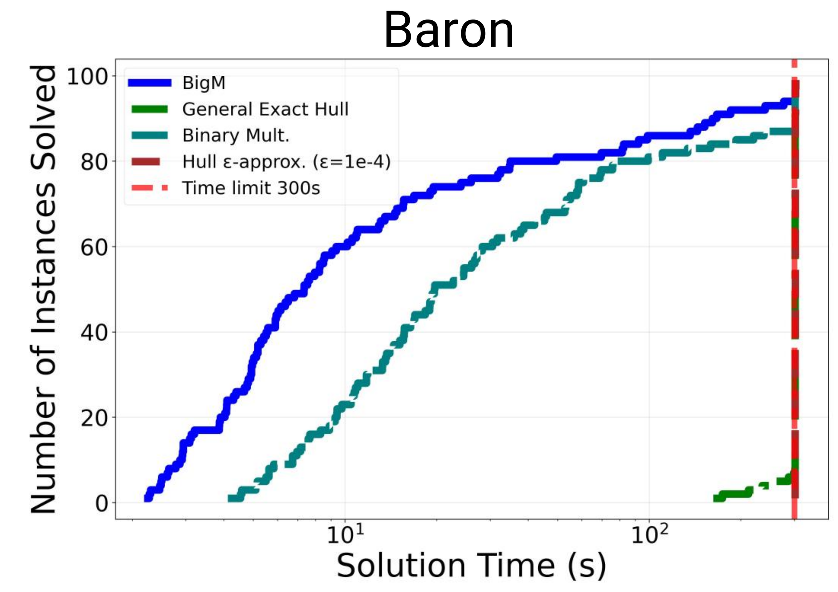
100 **Non-convex random instances:**

$3 \le |K| \le 10, \quad 10 \le |D_k| \le 15, \quad |J_{ik}| = 10, \quad 3 \le n \le 9$

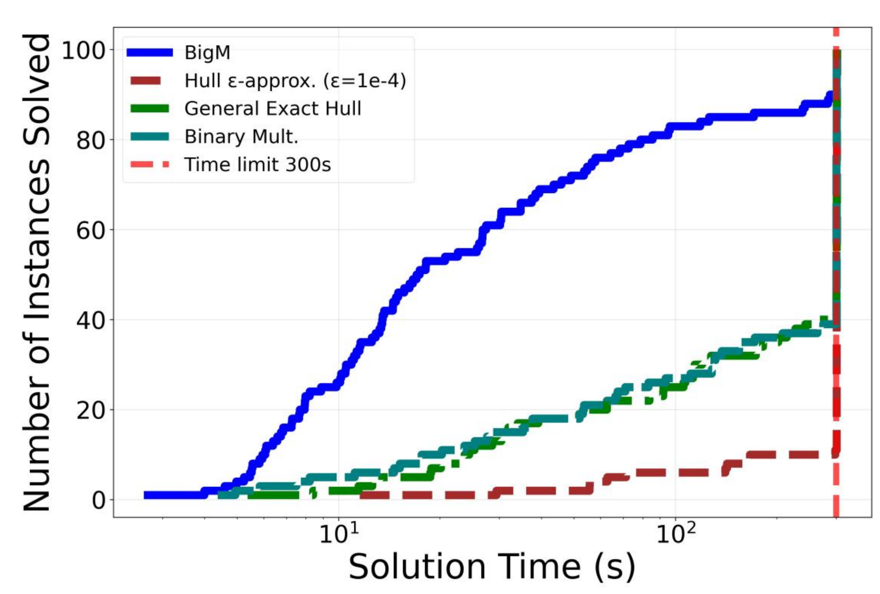
Gurobi



Baron



SCIP



Conclusion

- **Exact Hull Reformulation:** The proposed exact hull reformulation offers practical benefits by preserving the original quadratic problem structure and eliminating approximations in GDPs
- **Improved Solver Performance:** Numerical experiments show that state-of-the-art Mixed-Integer Nonlinear Programming (MINLP) solvers solve the proposed reformulations faster and more reliably than ε-approximation-based approaches.
- **Value of Customization:** Results highlight the importance of developing tailored, exact reformulations, especially when constraints exhibit simpler structural forms.

References

[1] I. E. Grossmann and S. Lee, "Generalized Convex Disjunctive Programming: Nonlinear Convex Hull Relaxation," *Comput. Optim. Appl.*, vol. 26, no. 1, pp. 83–100, Oct. 2003, doi: 10.1023/A:1025154322278.

[2] I. E. Grossmann and J. P. Ruiz, "Generalized Disjunctive Programming: A Framework for Formulation and Alternative Algorithms for MINLP Optimization," in *Mixed Integer Nonlinear Programming*, vol. 154, J. Lee and S. Leyffer, Eds., in The IMA Volumes in Mathematics and its Applications, vol. 154. , New York, NY: Springer New York, 2012, pp. 93–115. doi: 10.1007/978-1-4614-1927-3_4.

[3] P. Bonami, A. Lodi, A. Tramontani, and S. Wiese, "On mathematical programming with indicator constraints," *Math. Program.*, vol. 151, no. 1, pp. 191–223, Jun. 2015, doi: 10.1007/s10107-015-0891-4.

[4] D. E. Bernal Neira and I. E. Grossmann, "Convex Mixed-Integer Nonlinear Programs Derived from Generalized Disjunctive Programming using Cones," Feb. 19, 2024, *arXiv*: arXiv:2109.09657. doi: 10.48550/arXiv.2109.09657.

[5] K. C. Furman, N. W. Sawaya, and I. E. Grossmann, "A computationally useful algebraic representation of nonlinear disjunctive convex sets using the perspective function," *Comput. Optim. Appl.*, vol. 76, no. 2, pp. 589–614, Jun. 2020, doi: 10.1007/s10589-020-00176-0.

[6] O. Günlük and J. Linderoth, "Perspective Reformulation and Applications," in *Mixed Integer Nonlinear Programming*, vol. 154, J. Lee and S. Leyffer, Eds., in The IMA Volumes in Mathematics and its Applications, vol. 154. , New York, NY: Springer New York, 2012, pp. 61–89. doi: 10.1007/978-1-4614-1927-3_3.

[7] M. S. Aktürk, A. Atamtürk, and S. Gürel, "A strong conic quadratic reformulation for machine-job assignment with controllable processing times," *Oper. Res. Lett.*, vol. 37, no. 3, pp. 187–191, May 2009, doi: 10.1016/j.orl.2008.12.009.

[8] D. A. Liñán, D. E. Bernal, L. A. Ricardez-Sandoval, and J. M. Gómez, "Optimal design of superstructures for placing units and streams with multiple and ordered available locations. Part I: A new mathematical framework," *Comput. Chem. Eng.*, vol. 137, p. 106794, Jun. 2020, doi: 10.1016/j.compchemeng.2020.106794.

[9]D. Ovalle *et al.*, "Logic-Based Discrete-Steepest Descent: A solution method for process synthesis Generalized Disjunctive Programs," *Comput. Chem. Eng.*, vol. 195, p. 108993, Apr. 2025, doi: 10.1016/j.compchemeng.2024.108993.

[10] M. L. Bynum *et al.*, *Pyomo — Optimization Modeling in Python, Version 6.9.2*, vol. 67. in Springer Optimization and Its Applications, vol. 67. Cham: Springer International Publishing, 2021. doi: 10.1007/978-3-030-68928-5.

[11] GAMS Development Corporation. General Algebraic Modeling System (GAMS), Version 49.1.0. Available: https://www.gams.com/download/

[12] *Gurobi Optimization, LLC.* Version 12.0. Available: https://www.gurobi.com/

[13] *The Optimization Firm. BARON*. Available: https://minlp.com/baron-solver

Acknowledgments

We acknowledge and express sincere gratitude to the Davidson School of Chemical Engineering at Purdue University for providing a supportive research environment and funding.

We thank the GAMS Development Corporation for supplying a personal license and access to the BARON solver for our group.

We thank Gurobi Optimization for providing access to software through their academic licensing program.

We also thank Can Li for providing access to the hardware.

All experiments were run on a single thread of an AMD EPYC 7643 48-Core Processor (96 logical cores) with 256 MB of L3 cache in a server running Red Hat Enterprise Linux 9.5 (Plow). The server is equipped with 1 TB of RAM and 15 GB of Swap Space. Versions of software used: Pyomo - Version 6.9.2, GAMS - Version 49.1.0, Gurobi - Version 12.0