Systematic Design of Complex Processes using Generalized Disjunctive Programming: A Comparison of Reformulations

Emma S. Johnson, Michael L. Bynum, John D. Siirola

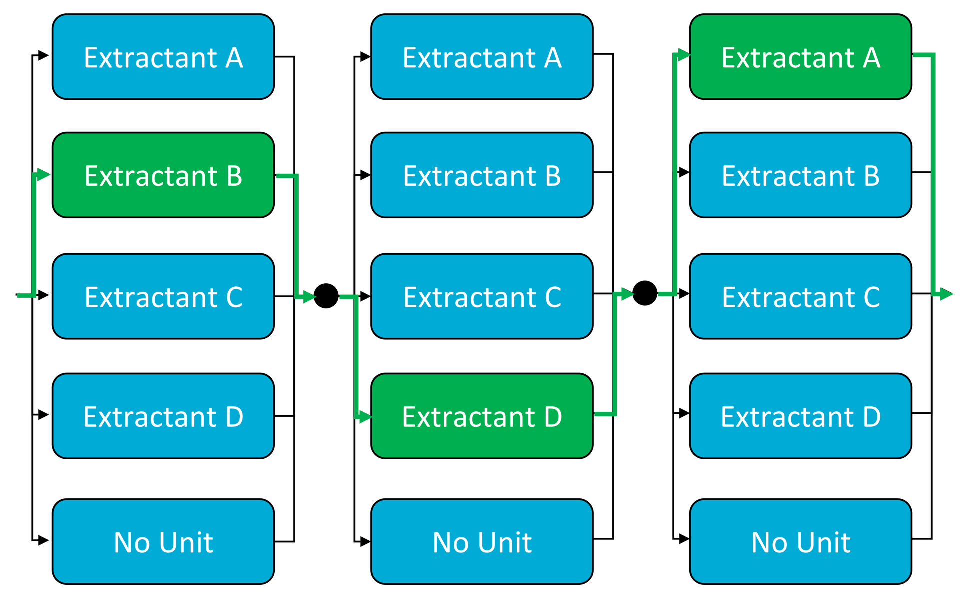
Sandia National Laboratories

PROMMIS
Process Optimization and Modeling for Minerals Sustainability

What is Generalized Disjunctive Programming (GDP)?

- Modeling framework for expressing discrete decisions and logical constraints (e.g., Solvent extraction sequencing)
- Naturally supports nested decisions, for example:

$$
\begin{aligned}
\max \quad & x \\
\text{s.t.} \quad & \begin{bmatrix} Y_1 \\ 1.15 \le x \le 8 \\ \begin{bmatrix} Z_1 \\ x \ge 1.1 \end{bmatrix} \vee \begin{bmatrix} Z_2 \\ x \ge 1.2 \end{bmatrix} \\ Z_1 \veebar Z_2 \end{bmatrix} \vee \begin{bmatrix} Y_2 \\ x = 9 \end{bmatrix} \\
& Y_1 \veebar Y_2 \\
& Y_2 \implies Z_1 \wedge Z_2 \\
& 1 \le x \le 10
\end{aligned}
$$



- Enables systematic design for complex processes (CM processing)

GDP Provides Solution Flexibility

- GDPs can be solved directly with specialized algorithms: via outer approximation, logic-based branch and bound, or, for few enough disjunctions, enumeration.
- Alternatively, there are myriad transformations from GDP to MINLP in the literature, most of which are implemented in Pyomo[1,2].

PYOMO GDP

	Transformation	Advantages	Disadvantages
	Big-M	Simple, requires few variables/constraints, familiar structure for solvers	Potentially weak continuous relaxation
	Hull	Tighter continuous relaxation	Large model: Requires many variables/constraints
	Multiple Big-M	Tighter continuous relaxation with smaller model	Requires calculating quadratically many M values
	Binary Multiplication	Additional structure for solver to exploit	Introduces nonlinearity
Hybrids of Big-M and Hull	Between Steps	Tighter continuous relaxation with smaller model in cases where there are many more variables than constraints in each disjunct	Relaxation quality is sensitive to choice of variable partition and variable bounds
Hybrids of Big-M and Hull	Cutting planes	Targeted tightening of Big-M relaxation in direction of improved objective values	Can cause numerical instability

Theory vs. Practice

The Question: What solution techniques are best for what problems?

- For linear GDPs, we know from experience that theoretically good formulations (e.g., hull) are rarely good in practice using commercial solvers (e.g. Gurobi) and are in fact outperformed by Big-M usually.
- Given advances in MIQCP and MINLP solvers, binary multiplication may be becoming tractable
- We have limited computational experience with hybrid formulations between Big-M and hull: What structures are amenable for these?

Acknowledgements

This effort was funded by the U.S. Department of Energy's Process Optimization and Modeling for Minerals Sustainability (PrOMMiS) Initiative, supported by the Office of Fossil Energy and Carbon Management's Office of Resource Sustainability.

For more information, please reach out to Thomas Tarka, PrOMMiS Technical Director (Thomas.Tarka@netl.doe.gov)

Preliminary Computational Results

Experimental setup:

- Baron Version 24.1.5
- Time limit of 500 seconds (denoted by 'T')
- Solved NLP relaxation with ipopt. 'NC' denotes that it did not converge.
- Every GDP-to-MINLP transformation in pyomo.gdp
- Two transformation of logical constraints: a sparse one using factorable programming (FP), and one to conjunctive normal form (CNF)

GDP Reactor (CSTR) Model[3]

Logical Transformation	GDP-to-MINLP Transformation	Baron Solve Time (s)	Nodes Explored	LB	UB	NLP Relaxation	LB After Presolve	UB After Presolve
FP	Big-M	16	6297	3.06	3.06	0.00	0.09	3.06
FP	Hull	51	21461	3.06	3.06	0.00	0.09	3.13
FP	Cutting planes	T	330747	2.91	3.06	0.00	0.09	3.06
FP	Multiple Big-M	NA						
FP	Binary Multiplication	111	11516	3.06	3.06	NC	0.09	3.06
FP	Between Steps: p=2	136	4189	3.06	3.06	0.00	0.05	4.06
FP	Between Steps: p=3	T	140371	2.92	3.06	0.00	0.05	4.06
FP	Between Steps: p=4	137	7558	3.06	3.06	0.00	0.05	9.89
CNF	Big-M	**11**	4675	3.06	3.06	0.00	0.09	3.06
CNF	Hull	T	538389	3.04	3.06	0.00	0.09	3.06
CNF	Cutting planes	T	1249456	3.02	3.06	0.00	0.09	3.06
CNF	Multiple Big-M	T	1110055	2.91	3.06	0.00	0.09	3.06
CNF	Binary Multiplication	T	301025	3.00	3.06	NC	0.09	3.06
CNF	Between Steps: p=2	131	**4189**	3.06	3.06	0.00	0.05	4.06
CNF	Between Steps: p=3	T	141129	2.92	3.06	0.00	0.05	4.06
CNF	Between Steps: p=4	138	7558	3.06	3.06	0.00	0.05	9.89

Batch Processing Model[4]

GDP-to-MINLP Transformation	Baron Solve Time (s)	Nodes Explored	Lower Bound	Upper Bound	NLP Relaxation	LB After Presolve	UB After Presolve
Big-M	T	79	607575	684827	546850	579099	712688
Hull	T	27	612034	684455	547333	577482	12206917
Cutting planes	**426**	**563**	679365	679365	546850	668226	680249
Multiple Big-M	T	168	670523	680249	547333	574856	680249
Binary Multiplication	T	5	592717	680249	715226	592717	694108
Between Steps: p=2	T	7	574481	702026	546850	574481	714243
Between Steps: p=3	T	4	581970	694605	546850	581970	694605
Between Steps: p=4	T	6	585961	680249	546850	571656	680249

Conclusions:

- There is no conclusion: That's why we have a toolbox!
- It is model-dependent what technique will result in the shortest solve time, and in some cases, even in a tractable model.
- While there is correlation between faster solve time and fewer nodes explored in the tree, some formulations explore many more nodes faster and can still be superior if solve time is the metric of success.
- Even when it is the preferable transformation, cutting planes does not always improve the relaxation from the Big-M transformation!

References

[1] Bynum, Michael L., Gabriel A. Hackebeil, William E. Hart, Carl D. Laird, Bethany L. Nicholson, John D. Siirola, Jean-Paul Watson, and David L. Woodruff. Pyomo - Optimization Modeling in Python. Third Edition Vol. 67. Springer, 2021.

[2] Q. Chen, E.S. Johnson, D.E. Bernal, R. Valentin, S. Kale, J. Bates, J.D. Siirola, & I.E. Grossmann, (2021) Pyomo.GDP: an ecosystem for logic based modeling and optimization development. Optimization and Engineering, Volume 23, Pages 607-642

[2] Linan, D. A., Bernal, D. E., Gomez, J. M., & Ricardez-Sandoval, L. A. (2021). Optimal synthesis and design of catalytic distillation columns: A rate-based modeling approach. Chemical Engineering Science, Volume 231.

[3] F. Trespalacios, & I. E. Grossmann (2015). Improved Big-M reformulation for generalized disjunctive programs. Computers & Chemical Engineering, Volume 76, Pages 98-103.

NATIONAL ENERGY TECHNOLOGY LABORATORY · BERKELEY LAB · Sandia National Laboratories · Carnegie Mellon · West Virginia University · UNIVERSITY OF NOTRE DAME · Georgia Tech · U.S. DEPARTMENT OF ENERGY