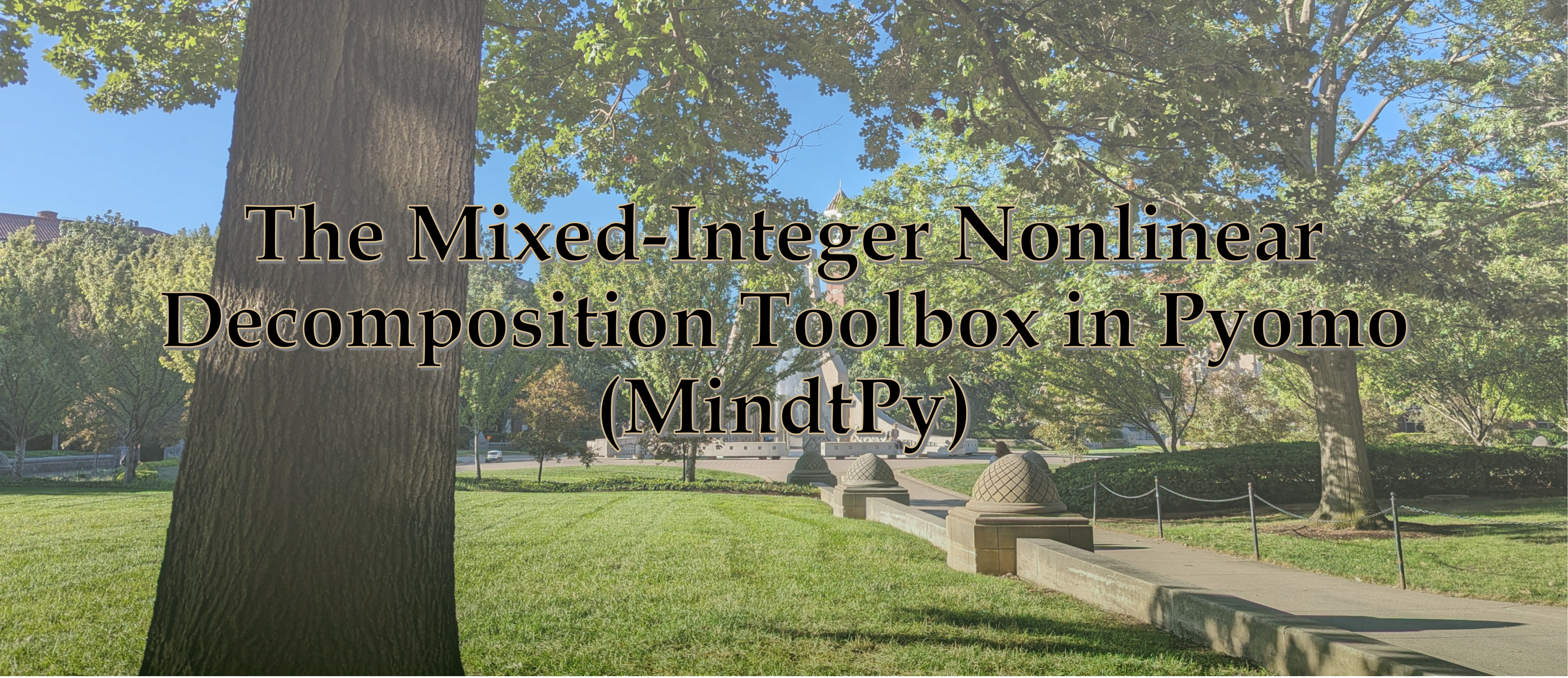


The Mixed-Integer Nonlinear Decomposition Toolbox in Pyomo (MindtPy)

Zedong Peng[1], Ignacio E. Grossmann[2], David E. Bernal Neira[1]

[1]Davidson School of Chemical Engineering, Purdue University
[2]Department of Chemical Engineering, Carnegie Mellon University

PURDUE UNIVERSITY

PSE Seminar October 20th, 2023

Introduction

MindtPy

- **MindtPy** (Mixed-Integer Nonlinear Decomposition Toolbox in Pyomo) is an **open-source meta solver** that allows users to solve both **convex and nonconvex** Mixed-Integer Nonlinear Programs (MINLP) using **decomposition algorithms**.
- These decomposition algorithms usually rely on the solution of Mixed-Integer Linear Programs (MILP) and Nonlinear Programs (NLP).

Supported Algorithms

Convex MINLP

- Extended Cutting Plane
- Outer-Approximation
- LP/NLP based Branch-and-Bound
- Regularized Outer-Approximation
- Regularized LP/NLP based Branch-and-Bound
- Feasibility Pump

Nonconvex MINLP

- Outer Approximation
  - Equality Relaxation
  - Augmented Penalty
- McCormick-relaxation-based Outer-Approximation

Notation

Formulation

$$
\begin{aligned}
\min_{\mathbf{x},\mathbf{y}} \quad & f(\mathbf{x},\mathbf{y}) \\
\text{s.t.} \quad & g_j(\mathbf{x},\mathbf{y}) \leq 0 \\
& \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} \leq \mathbf{b}, \\
& \mathbf{x} \in \mathbb{R}^n, \ \mathbf{y} \in \mathbb{Z}^m.
\end{aligned}
$$

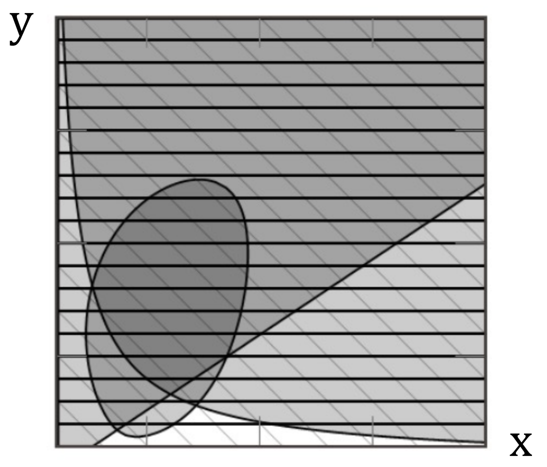
Objective function

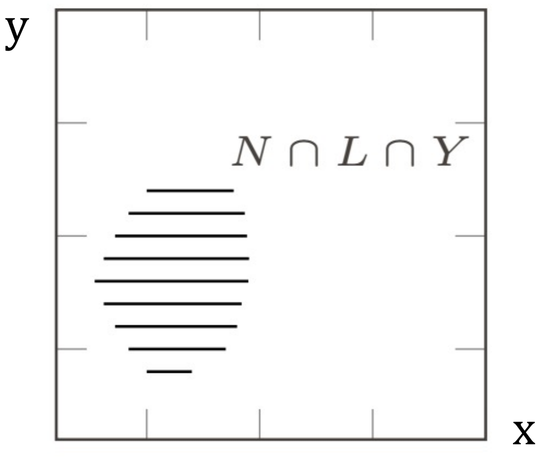
Nonlinear Constraints

Linear constraints

Continuous variables

Discrete variables





Feasible region of the MINLP problem

Compact Notation

$$\min_{(\mathbf{x},\mathbf{y}) \in N \cap L \cap Y} f(\mathbf{x},\mathbf{y}) = \mathbf{c}_1^T \mathbf{x} + \mathbf{c}_2^T \mathbf{y},$$

Objective function

where sets N, L and Y are given by:

$$
\begin{aligned}
N &= \{\mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^m \mid g_j(\mathbf{x},\mathbf{y}) \leq 0 \quad \forall j = 1, \ldots l\}, \\
L &= \{\mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^m \mid \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} \leq \mathbf{b}\}, \\
Y &= \{\mathbf{y} \in \mathbb{Z}^m \mid \underline{y}_i \leq y_i \leq \overline{y}_i \quad \forall\, i = 1, 2, \ldots, m\}.
\end{aligned}
$$

Nonlinear Constraints

Linear constraints

Discrete variables

- An MINLP problem is often considered as convex when its continuous relaxation yields a convex NLP problem.

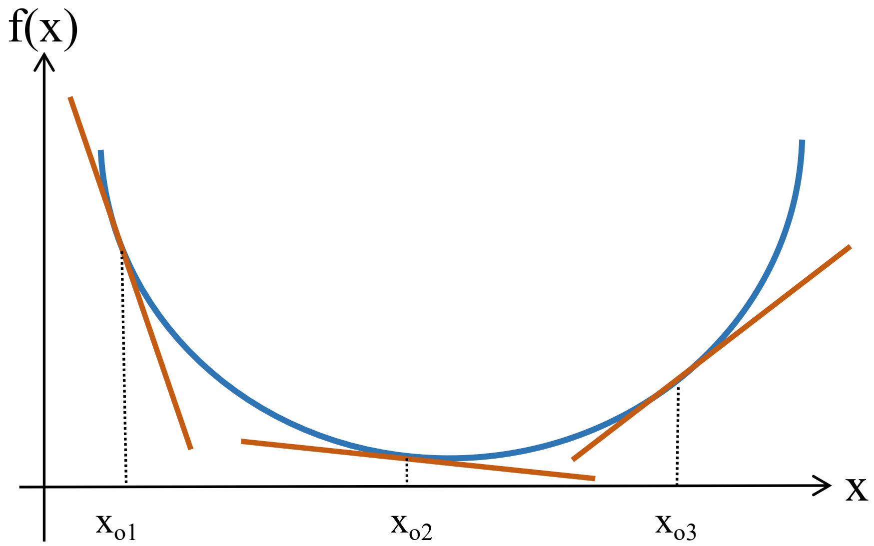
Decomposition methods for convex MINLP

Fixing a subset of variables makes the problem in the rest variables considerably more tractable

- **Complicating** variables are the **discrete variables** (MINLP→NLP)
- **Decompose** MINLP
  - MILP master problem
  - Continuous subproblem

Several methods have been proposed

- Outer-Approximation (**OA**)[1]
- Partial Surrogate Cuts (**PSC**)[2]
- Extended Cutting plane (**ECP**)[3]
- Generalized Benders Decomposition (**GBD**)[4]
- Extended Supporting Hyperplanes (**ESH**)[5]



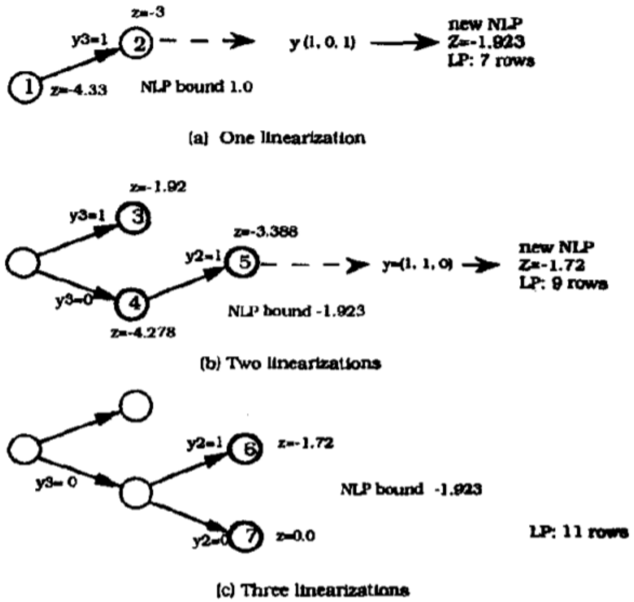
Linearized nonlinear function at 3 points

1. Duran M., Grossmann, I.E. "An outer-approximation algorithm for a class of mixed-integer nonlinear programs." 1986.
2. Quesada, I., Grossmann, I.E., "An LP/NLP based branch and bound algorithm for convex MINLP optimization problems." 1992
3. Westerlund, T., Pettersson, F., "An extended cutting plane method for solving convex MINLP problems." 1995
4. Geoffrion, A.M., "Generalized Benders decomposition." 1972
5. Kronqvist, J., Lundell, A., Westerlund, T., "The extended supporting hyperplane algorithm for convex mixed-integer nonlinear programming." 2016

Decomposition methods for convex MINLP

- When iteratively solving the MIP master problems:
  - Practically the same MILP BB tree close to the root.
  - Expensive to set up new MILP at each iteration
  - Why not a single MILP tree and then add cuts?

- LP/(NLP)-based BB[1,2]
  - Have a single MILP problem (single-tree approach[3])
  - Whenever an integer solution is found, fix it and solve continuous problem
  - Add cuts.



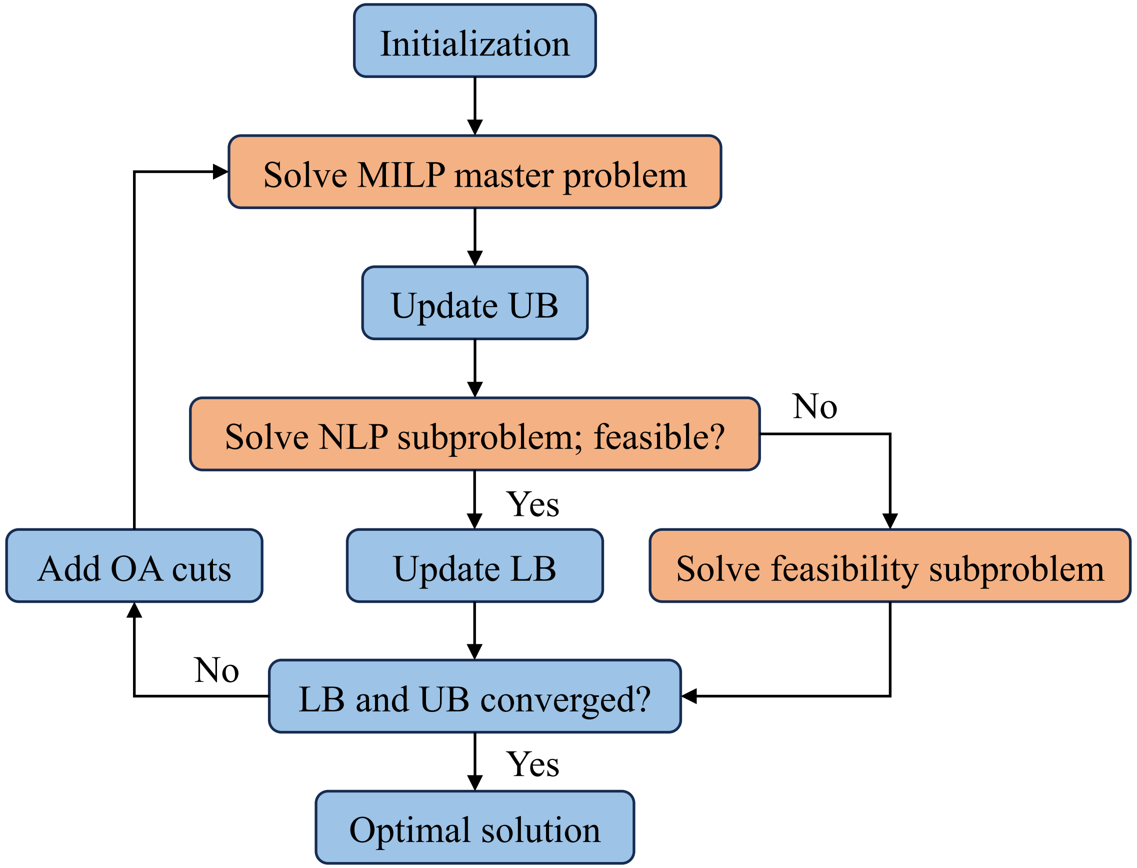
LP/NLP BB method from Quesada and Grossmann[1]

- Multi-tree Solvers: DICOPT (OA), α-ECP (ECP), BONMIN (OA), Muriqui
- Single-tree solvers: SHOT (ESH), AIMMS OA, BONMIN, MINOTAUR

1. Quesada, I., Grossmann, I.E., "An LP/NLP based branch and bound algorithm for convex MINLP optimization problems." 1992
2. Kronqvist, J., Lundell, A., Westerlund, T., "The extended supporting hyperplane algorithm for convex mixed-integer nonlinear programming." 2016
3. Abhishek K, Leyffer S, Linderoth J "FilMINT: an outer approximation-based solver for convex mixed-integer nonlinear programs." 2010

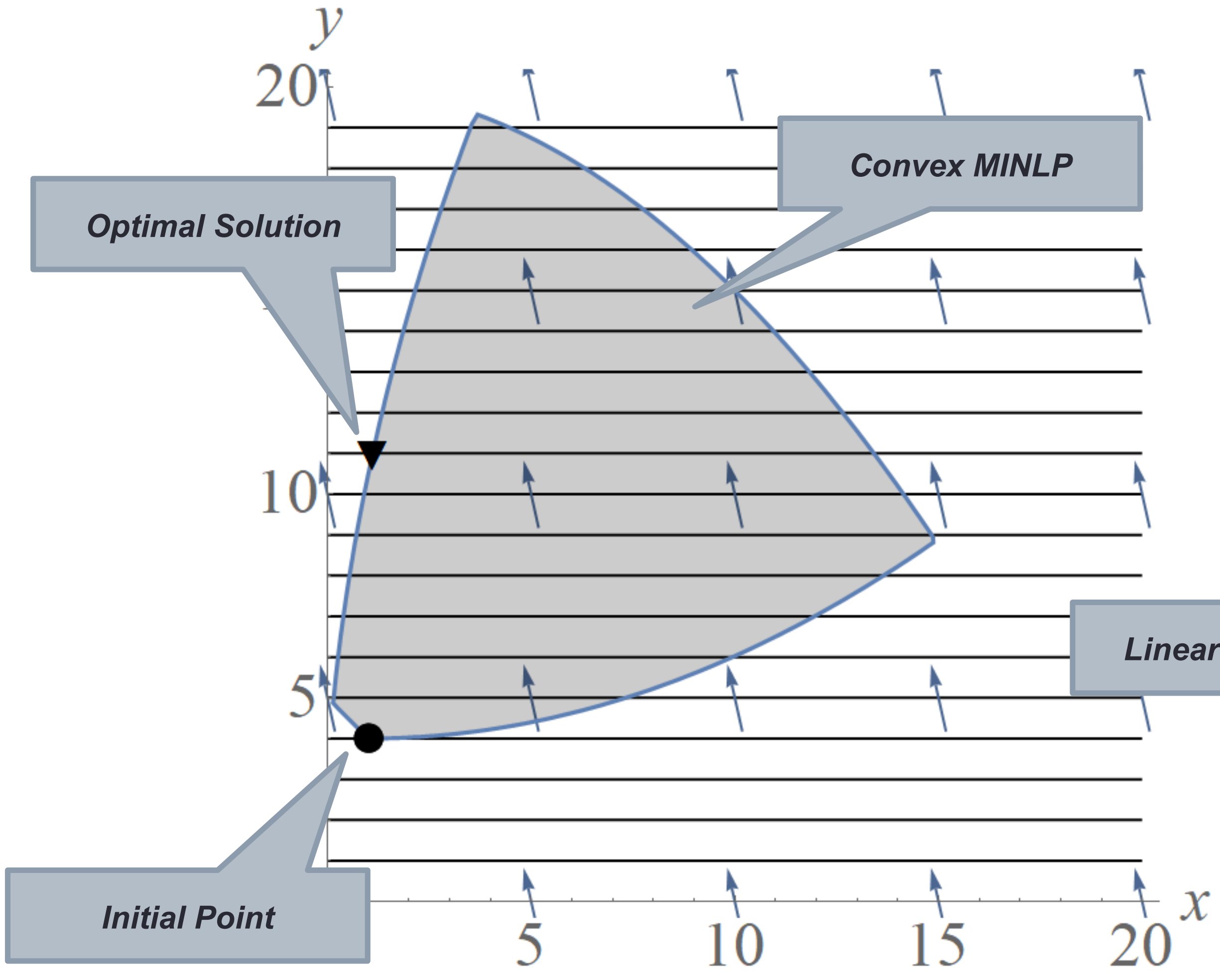
Outer-Approximation (OA) method

- Iterates between **master MILP problem** (**LB**) constructed with the 1st **order Taylor approximations** and the **NLP subproblem** with fixed discrete variables (**UB**).
- **LB** predicted by MILP master problem is **at least as good** as with GBD and PSC.
- Converges to the **global optimal solution** of convex MINLP.
- MINLP solvers as **DICOPT** and **BONMIN**.



1. Duran M., Grossmann, I.E. "An outer-approximation algorithm for a class of mixed-integer nonlinear programs." 1986.

Outer-Approximation (OA) method - Example



Feasible region of example problem

Linear objective

$$\min_{x,y} \quad f(x,y) = x - \frac{y}{4.5} + 2$$

$$\text{s.t.:} \frac{x^2}{20} + y \le 20$$

$$\frac{(x-1)^2}{40} - y \le -4$$

$$y - 10\sqrt{x + 0.1} \le 0$$

$$-x - y \le -5$$

$$0 \le x \le 20$$

$$0 \le y \le 20$$

$$x \in \mathbb{R}, y \in \mathbb{Z}$$

Nonlinear constraints

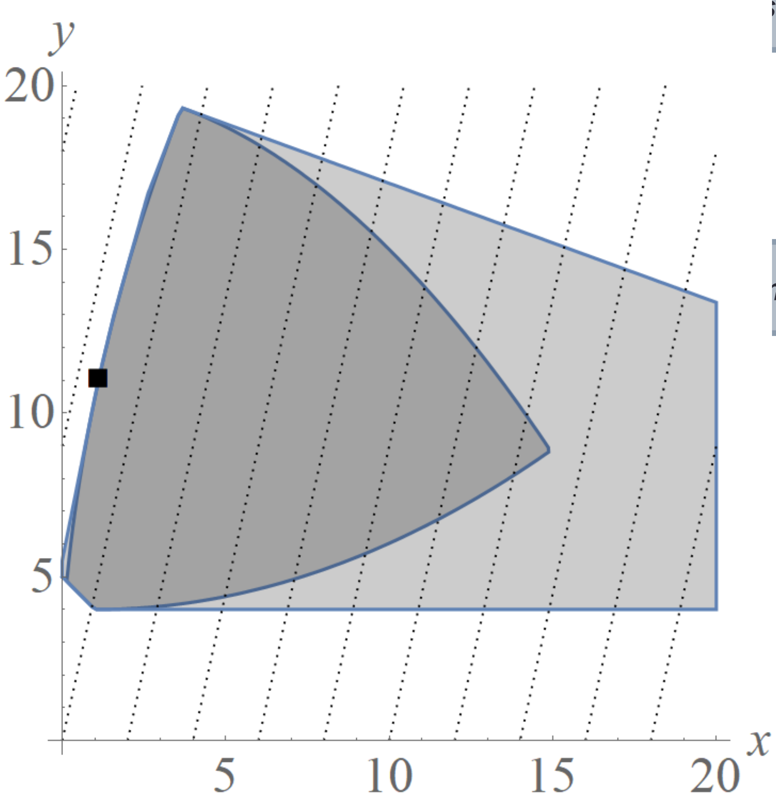
Linear constraint

Discrete variable

Outer-Approximation (OA) method - Example

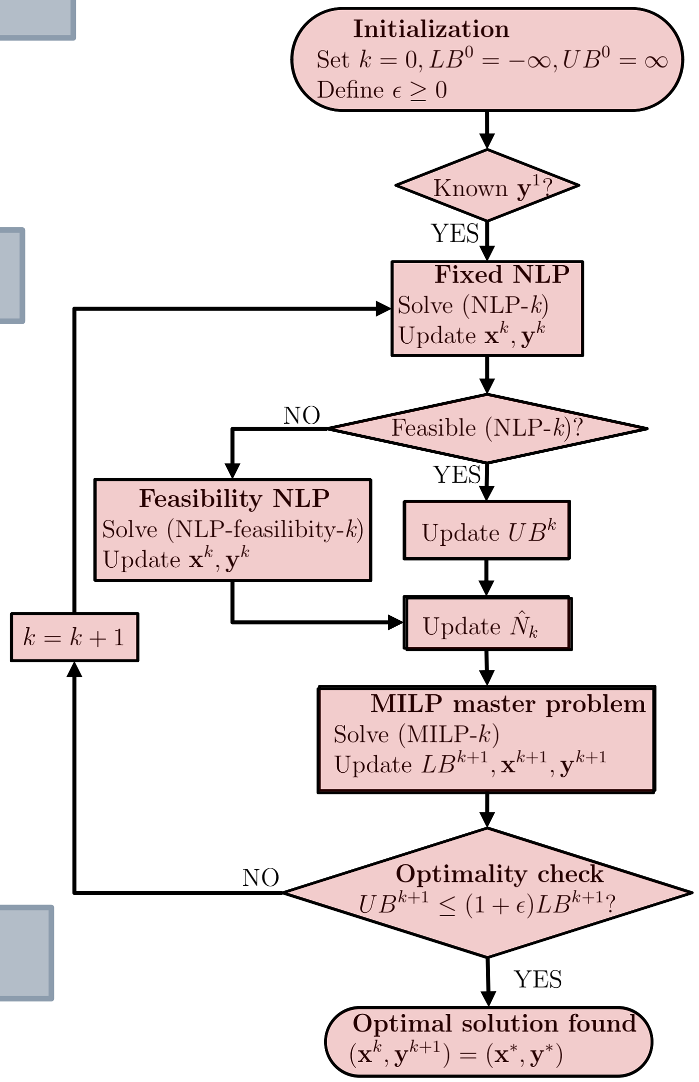
ible NLP

mp”

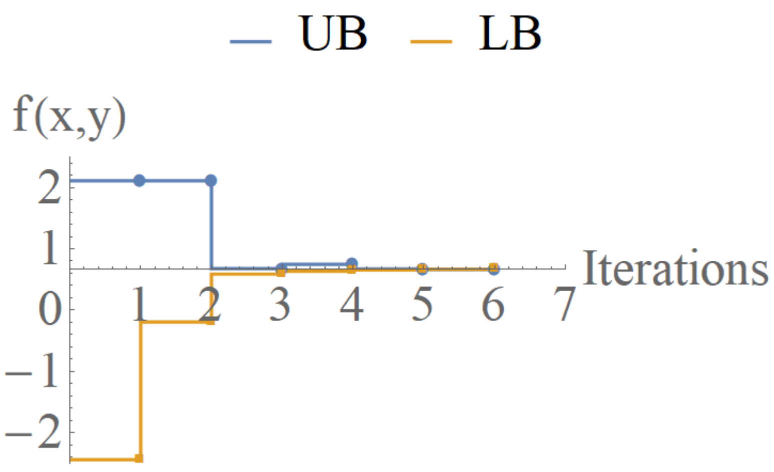


Feasible region and solution of example using OA

4 iterations later…



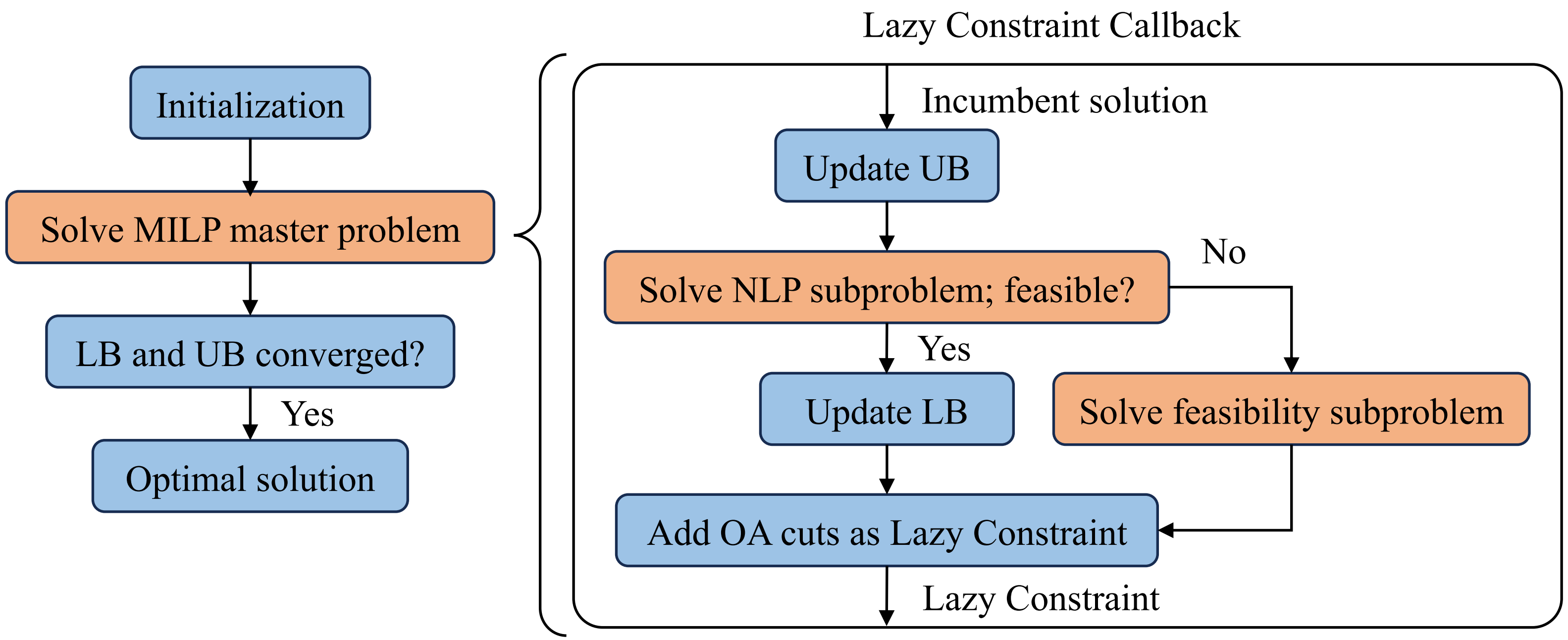
No UB improvement



Objective function bounds at each iteration

LP/NLP-based Branch and Bound

- Proposed by I.Quesada and I.E.Grossmann in 1992.
- Only need to solve the MIP master problem once.
- Usually solves more fixed-NLP subproblem.
- Also called single-tree implementation.



1. Quesada, Ignacio, and Ignacio E. Grossmann. "An LP/NLP based branch and bound algorithm for convex MINLP optimization problems." Computers & chemical engineering 16.10-11 (1992): 937-947.

Outer approximation (OA) method - Method limitations

Inherits limitations cutting plane method NLP

- Performance as good as linearization

  → **Poor performance** if highly nonlinear[1]

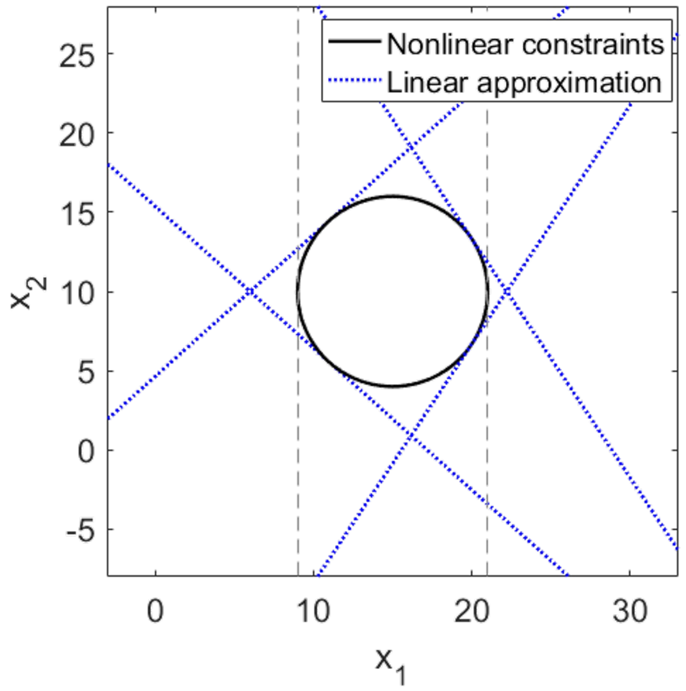
- Solutions of MILP master **more likely to lie outside** nonlinear constraints

  - **Infeasible** NLP subproblems
  - No new **UB**

- Integer combinations may "jump" in search space

  - Has been shown to be unstable[2]

- The OA cut is only valid for convex MINLP.



Poor overestimation of feasible region

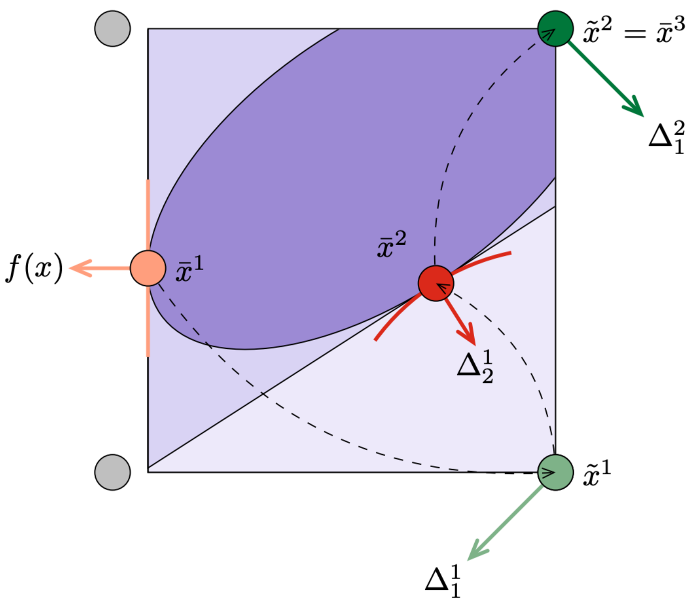
Feasibility pump algorithm for convex MINLP

- Highly nonlinear constraints ⟶ Poor linear approximations ⟶ **Solution outside of the feasible region**
- Iterations of MILP-NLP solely focused on feasibility ⟶ **Feasibility Pump**
- In MindtPy, feasibility pump can be used as
  - An initialization method
  - A standalone method to find a δ-optimal solution.
- Distance calculation (L1, L2, L infinity norm)

Feaspump-OA

$$(\mathbf{x}^{k+1}, \mathbf{y}^{k+1}) \in \underset{\mathbf{x},\mathbf{y} \in \hat{N}_k \cap L \cap Y}{\arg\min} \ ||\mathbf{y} - \mathbf{y}^k||_1 \qquad \text{(FP-MILP-}k\text{)}$$

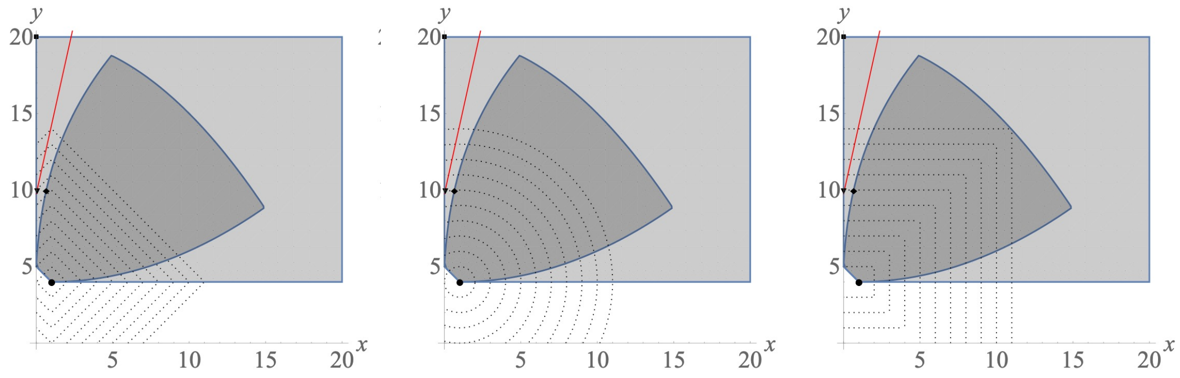
$$\begin{aligned} (\mathbf{x}^{k}, \mathbf{y}^{k}) \in & \underset{(\mathbf{x},\mathbf{y}) \in N \cap L}{\arg\min} \ ||\mathbf{y} - \mathbf{y}^k||_2 \\ \text{s.t.} & \quad \mathbf{y} = \mathbf{y}^{k+1} \end{aligned} \qquad \text{(FP-NLP-}k\text{)}$$



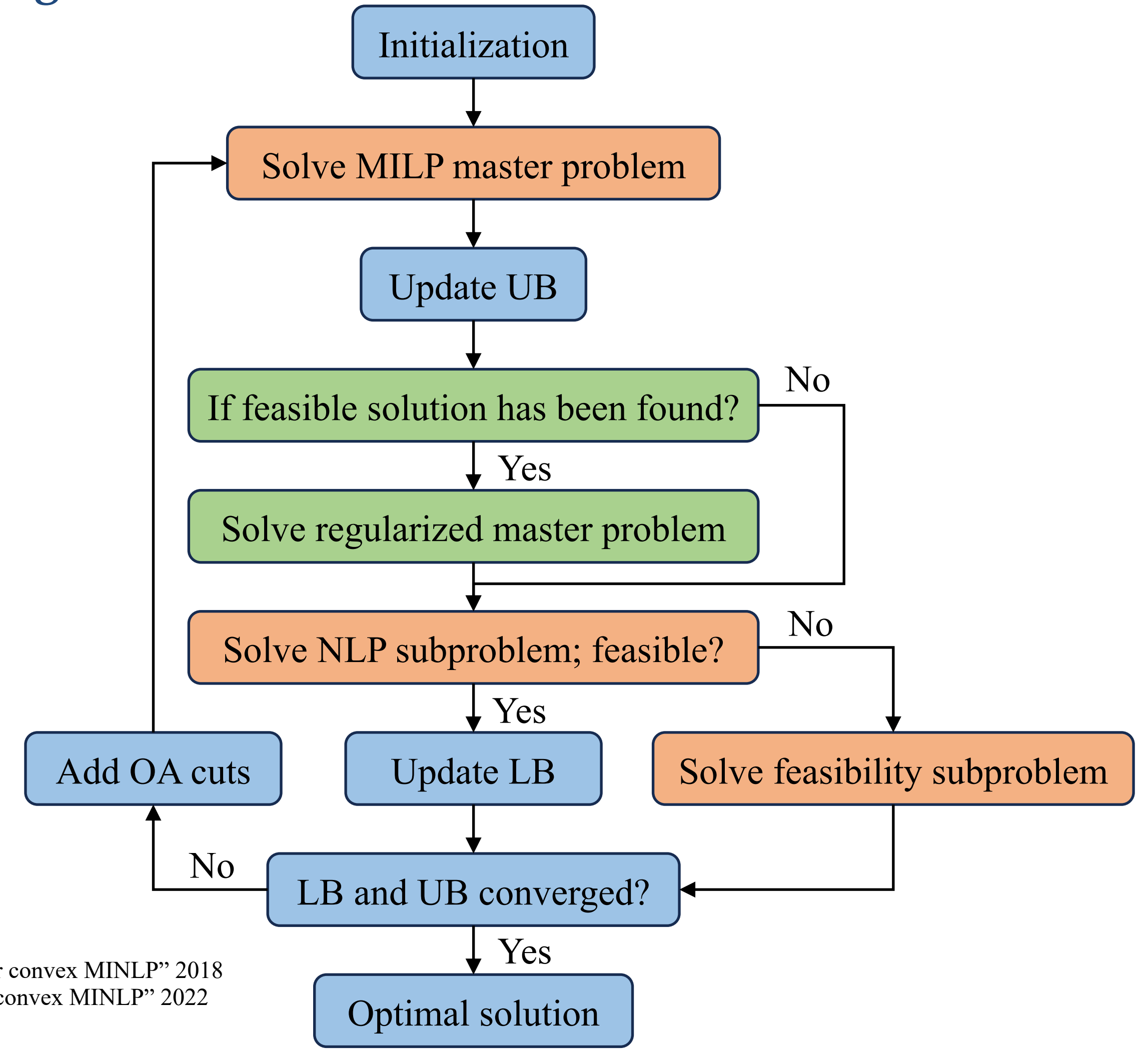
1. B., Vigerske, Trespalacios, Grossmann, (2019). "Improving the performance of DICOPT in convex MINLP problems using a feasibility pump."

Regularized OA and Regularized LP/NLP B&B

- Cutting plane approach may be unstable (Big jump) → **Regularization term**
- Solution of Regulization problem in every iteration
  - Norm based regularization
  - Lagrangean-based regularization
- Equivalent to **trust region** approach for MINLP
- Efficient for highly nonlinear MINLP models.



L1 norm, L2 norm, L-inf norm



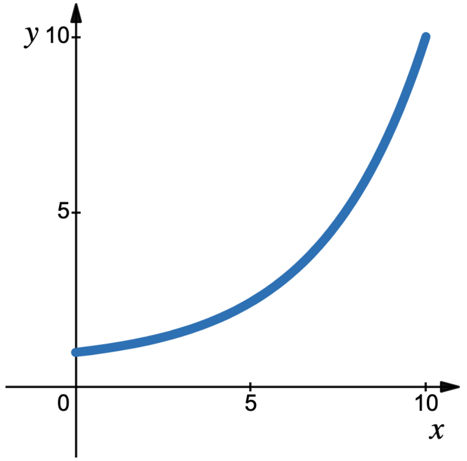
1. Kronqvist, J., Bernal, D. E. and Grossmann, I. E. "Using regularization and second order information in outer approximation for convex MINLP" 2018
2. Bernal, D. E., Peng, Z., Kronqvist, J., & Grossmann, I. E. "Alternative regularizations for Outer-Approximation algorithms for convex MINLP" 2022

Outer-Approximation for Nonconvex MINLP

Convergence guarantee

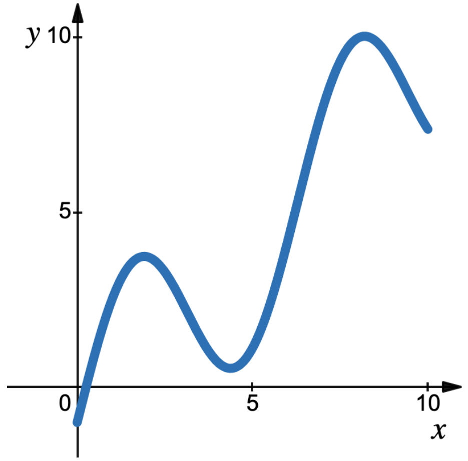
- **Assumption 1.** The nonlinear functions $g_1, \ldots, g_l \colon \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}$ are convex and continuously differentiable.
- **Assumption 2.** The intersection $L \cap Y$ defines a compact nonempty set, i.e., all variables must be bounded.
- **Assumption 3.** For each feasible integer combination **y**, an integer combination such that there exist **x** variables for which the problem is feasible, a constraint qualification holds.

Therefore, Outer Approximation cuts does not apply to Nonconvex MINLP.



(a) $y = \frac{1}{3}(e^{\frac{x}{3}} + 2)$

Equality constraint with convex function

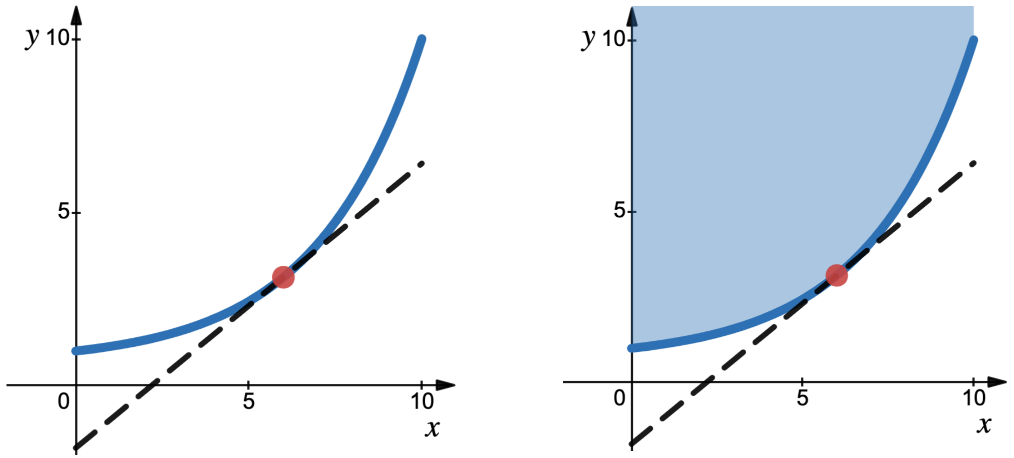


(b) $y = 3sin(x) + x - 1$

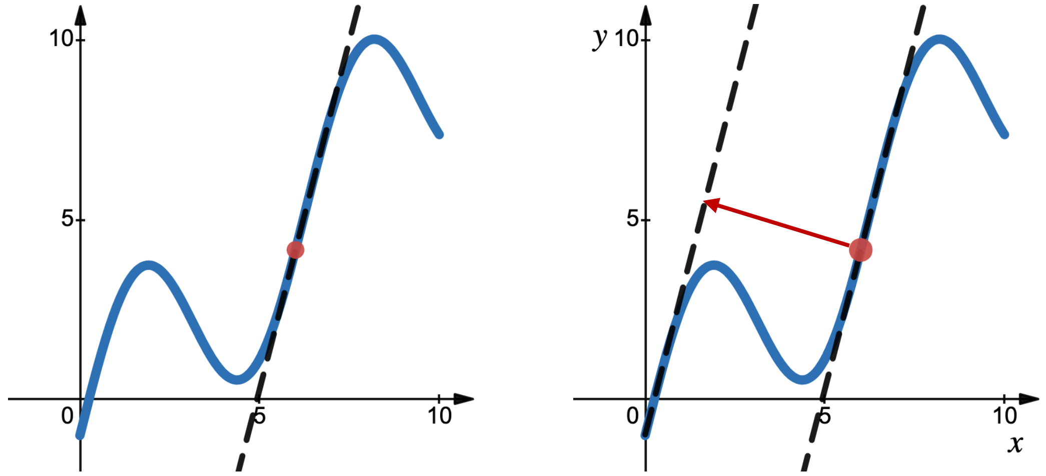
Equality constraint with nonconvex function

Outer Approximation for Nonconvex MINLP

- **Equality relaxation**
  - Convexity of the nonlinear functions
  - The equality constraint can be relaxed to be an inequality constraint.



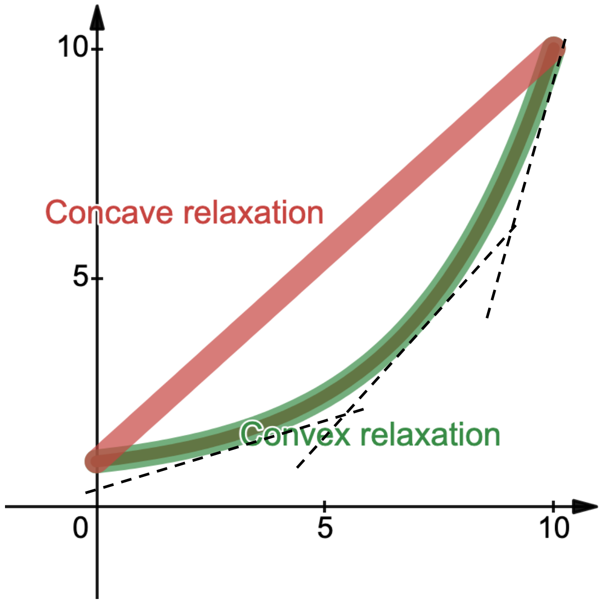
- **Augmented Penalty**
  - Add slack variable on the right hand side.



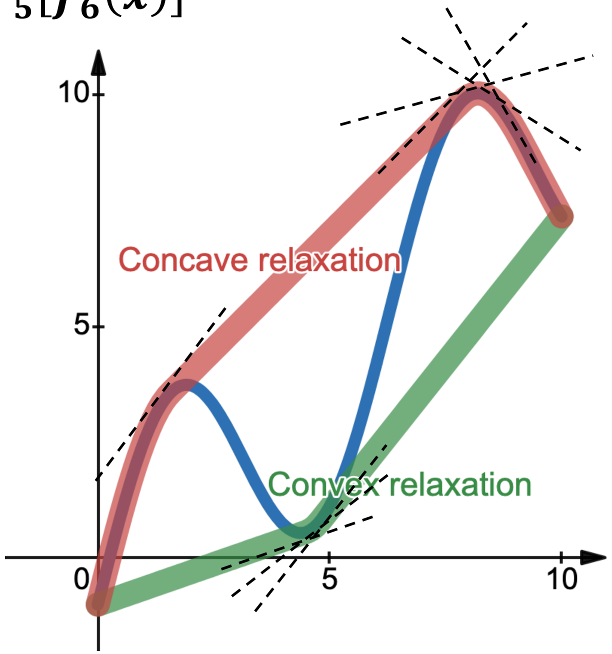
McCormick relaxation-based Outer Approximation

- Proposed by G P. McCormick in 1976 for NLP.
- Generate the convex and concave relaxation for nonconvex factorable functions.
- Factorable functions is a a collection of elementary operations (e.g., sum, product). The general form is

$$\boldsymbol{g(x) = f_1[f_2(x)] + f_3[f_4(x)] \cdot f_5[f_6(x)]}$$



(a) $y = \frac{1}{3}(e^{\frac{x}{3}} + 2)$



(b) $y = 3sin(x) + x - 1$

1. McCormick G P. Computability of global solutions to factorable nonconvex programs: Part I—Convex underestimating problems[J]. Mathematical programming, 1976, 10(1): 147-175.

McCormick relaxation-based Outer Approximation

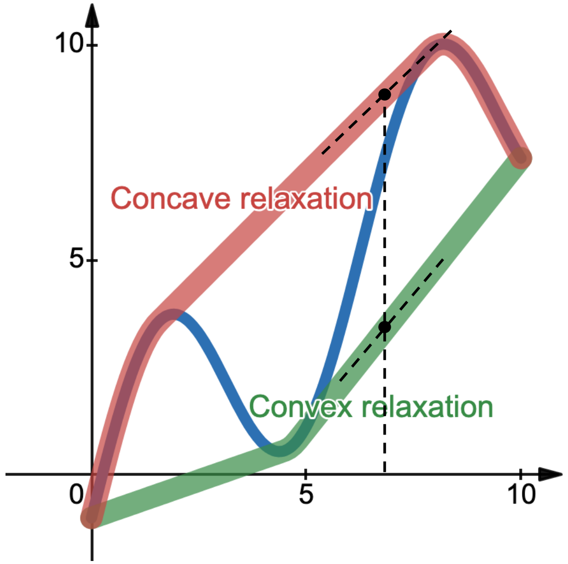
Convergence guarantee

- With the McCormick relaxation-based cuts, cycling might happen and bounds might not meet.

  ⟶ **No-good (Integer) cuts**

  ⟶ **Tabu list**

$$\sum_{j\in V_1} y_j - \sum_{j\in V_0} y_j \le |V_1| - 1, where\ V_1 = \{j|\ y_j = 1\}\ and\ \ V_0 = \{j|y_j = 0\}$$



Fix Bound

- The dual bound is not valid due to no-good cuts and tabu list.

  ⟶ **Solve an extra relaxed problem.**

Initialization

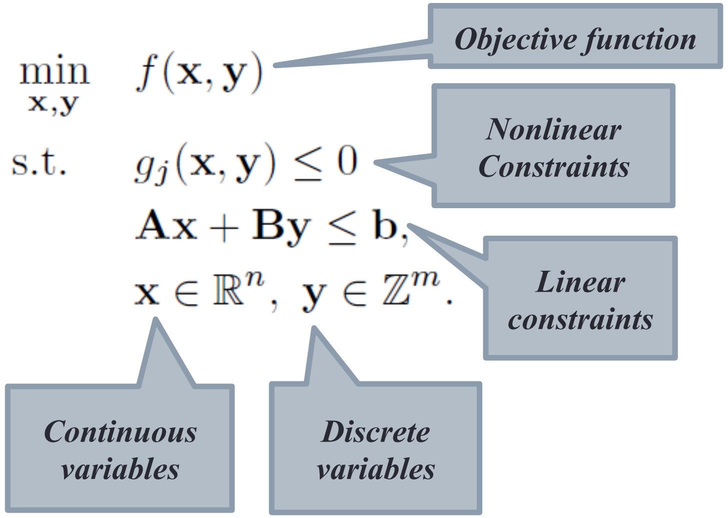
- The relaxed NLP problem might be hard to solve.

  ⟶ **Solve an MILP to maximize the sum of binary variables.**

$$\max_{\mathbf{x},\mathbf{y}} \sum_{i\le m} \mathbf{y}_i$$
$$\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} \le \mathbf{b},$$
$$\mathbf{x} \in \mathbb{R}^n,\ \mathbf{y} \in \mathbb{Z}^m.$$

GreyBox

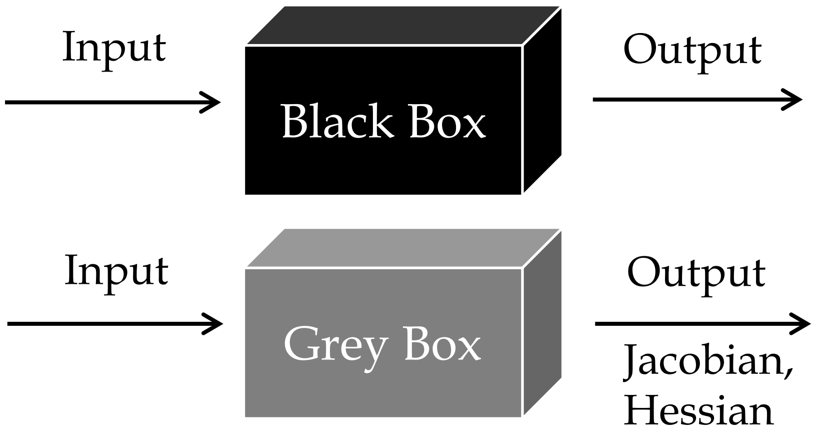
Formulation



$$
\begin{aligned}
\min_{\mathbf{x},\mathbf{y}} \quad & f(\mathbf{x},\mathbf{y}) \\
\text{s.t.} \quad & g_j(\mathbf{x},\mathbf{y}) \leq 0 \\
& \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} \leq \mathbf{b}, \\
& \mathbf{x} \in \mathbb{R}^n, \ \mathbf{y} \in \mathbb{Z}^m.
\end{aligned}
$$

Grey Box

Wide application in complex systems engineering, materials design, drug discovery, chemical process synthesis, computational biology.

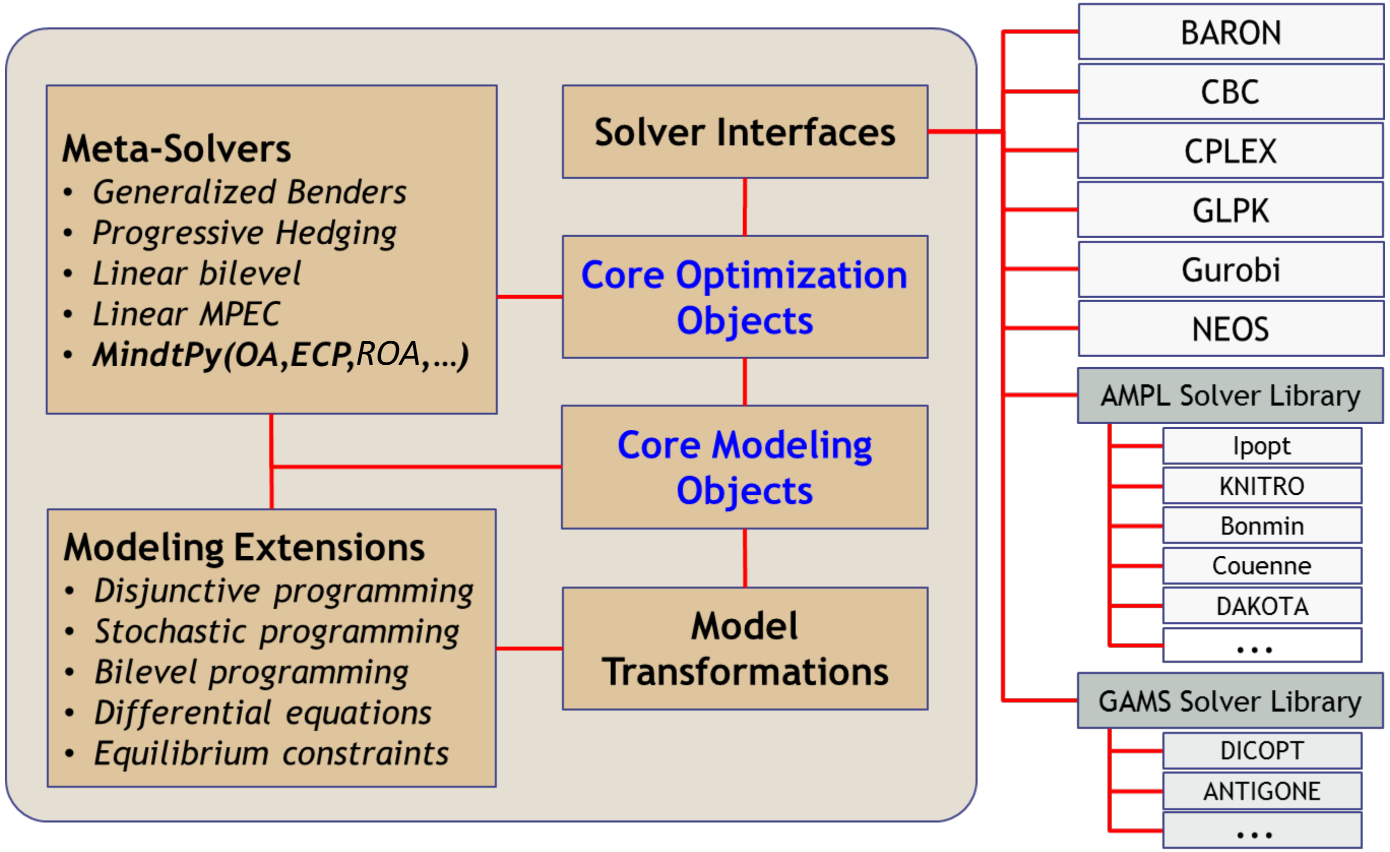


- In an MINLP model, we can replace the equality constraint with a GreyBox.
- MindtPy is able to solve MINLPs with GreyBox.
  - Use CYIPOPT as the NLP solver.

Key features of MindtPy

- **MIP Solver**
  - CPELX, Gurobi, Highs, CBC, GLPK, GAMS
- **NLP Solver**
  - IPOPT, BARON, CYIPOPT, GAMS
- **Master problem**
  - MILP, MIQP, MIQCP
- **Cuts**
  - OA cuts, No-good (Integer Cuts), McCormick-relaxation-based cuts
- **Initialization Method**
  - Relaxed NLP, Max binary, Initial binary, Feasibility pump
- **Distance Calculation**
  - L1, L2, L infinity norm
- **Other enhancement**
  - Solution Pool, Tabu list, Greybox

Implementations



- **Pyomo: Py**thon **o**ptimization **m**odeling **o**bjects[1]
- Use of **python expands** skilled **user base**
- **Open repository** at **Github**

- **Example**

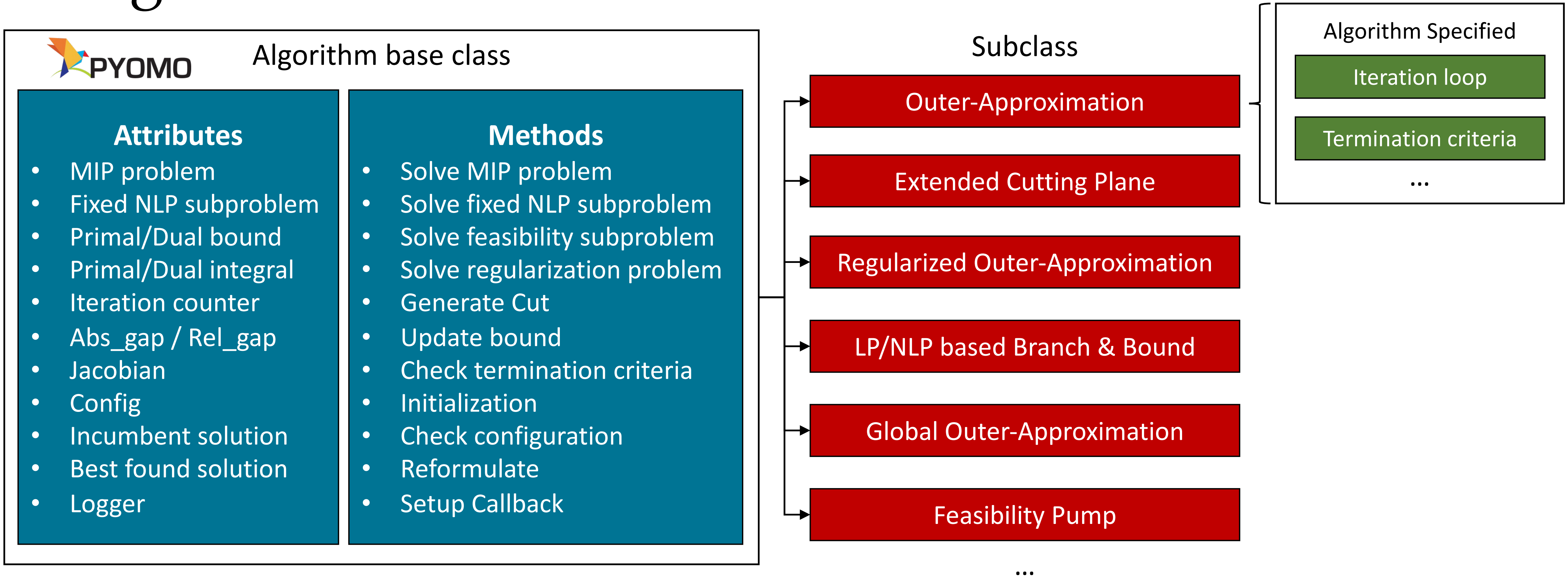
```python
from pyomo.environ import *

model = ConcreteModel()
model.x = Var(bounds=(1.0,10.0),initialize=5.0)
model.y = Var(within=Binary)
model.c1 = Constraint(expr=(model.x-3.0)**2 <= 50.0*(1-model.y))
model.c2 = Constraint(expr=model.x*log(model.x)+5.0 <= 50.0*(model.y))
model.objective = Objective(expr=model.x, sense=minimize)

SolverFactory('mindtpy').solve(model, mip_solver='glpk', nlp_solver='ipopt')
```

PYOMO python™ GitHub COIN|OR

Design and Architecture



- Object-oriented.
- Easy extension and modification of the core algorithm.
- Easy to integrate with other modules in / based on Pyomo, eg. SUSPECT.

Benchmark

Matched benchmark repository

- https://github.com/SECQUOIA/pyomo-MINLP-benchmarking

Convex instance

- 438 instances that have at least one discrete variable and at least one continuous variable in MINLPLib.

Nonconvex instance

- 129 non-convex MINLP in MINLPLib which are non-convex and have < 100 binary variables.

Computation environment

- Linux cluster with 48 AMD EPYC 7643 2.3GHz CPUs and 1 TB RAM.
- The thread is limited to 1 for each run.
- Time limit of 15 minutes per instance.

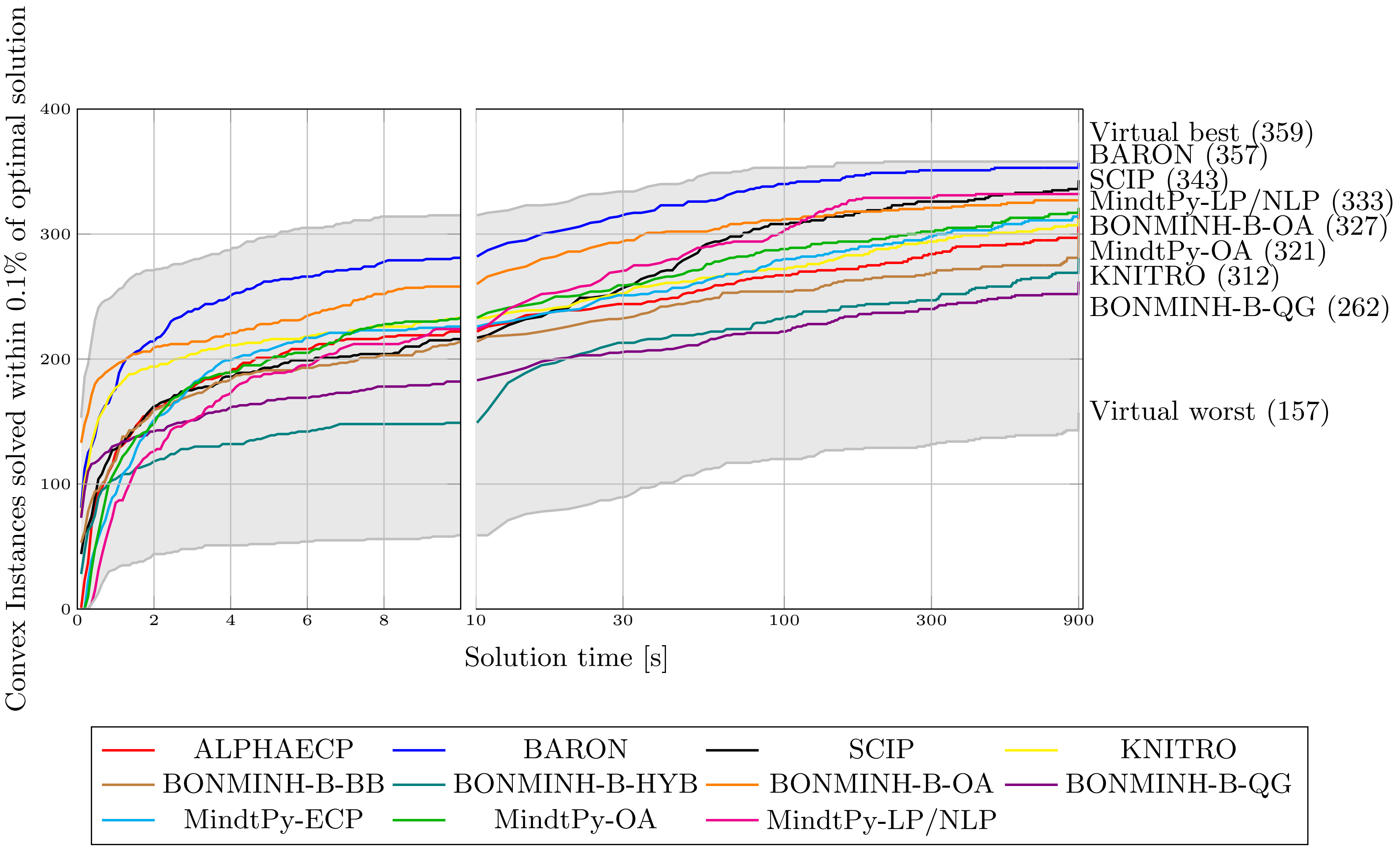
Solver version

- CPLEX 22.1.0.0
- GUROBI 10.0.0
- GAMS 44.3
- SCIP 8.0
- KNITRO 13.2
- BONMINH 1.8
- IPOPTH 3.14
- BARON 15.6.5
- CONOPT 4.02

Computational results - Convex

MindtPy

- MIP solver
  - GUROBI
- NLP solver
  - IPOPTH

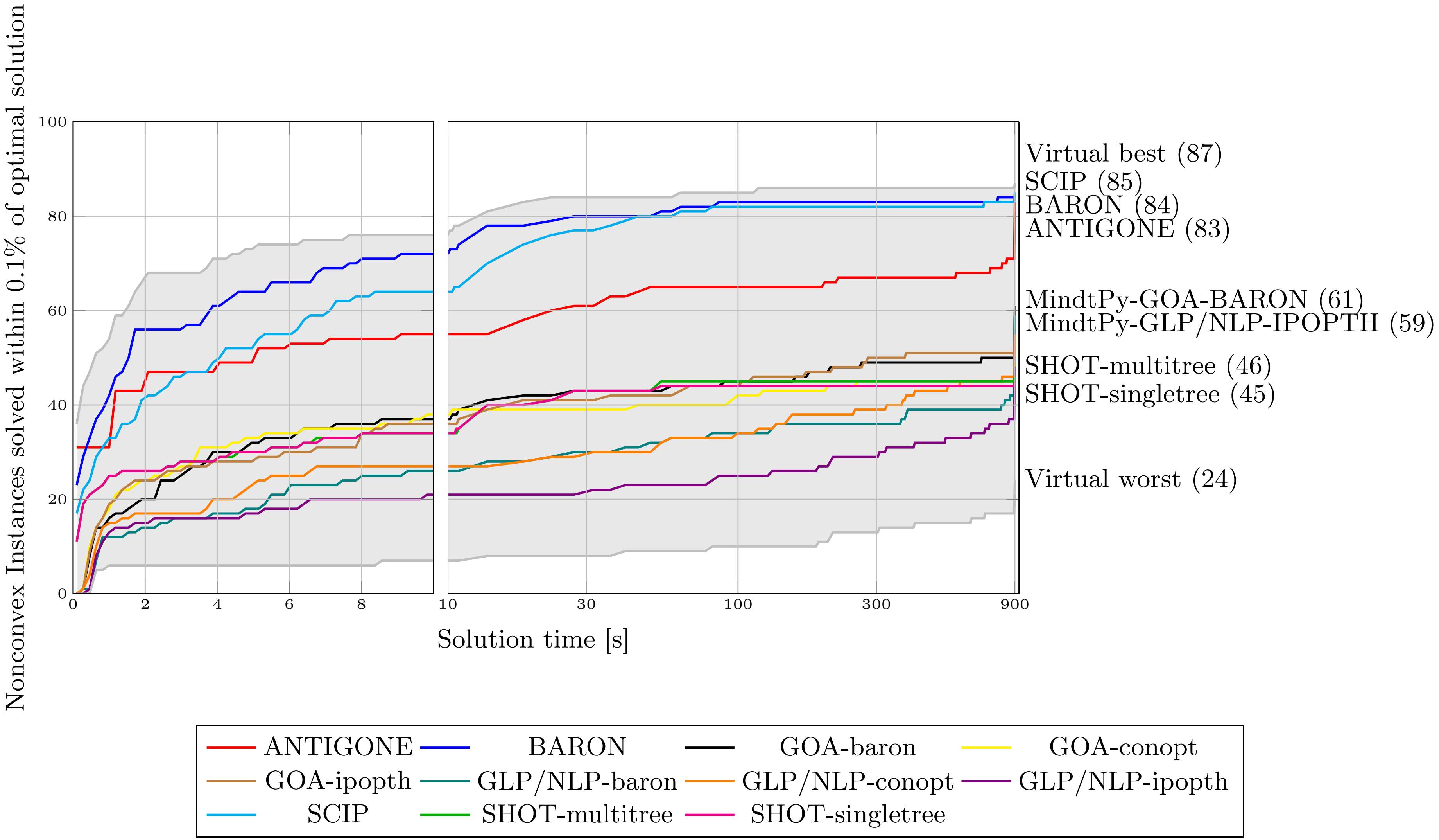


MindtPy holds the 3rd place among BARON, SCIP, BONMINH, KNITRO.

Computational results - Nonconvex

MindtPy

- MIP solver
  - GUROBI
- NLP solver
  - IPOPTH
  - CONOPT
  - BARON



There is a huge space for MindtPy to improve.

Thanks for your attention.

The Mixed-Integer Nonlinear Decomposition Toolbox in Pyomo (MindtPy)

Zedong Peng[1], Ignacio E. Grossmann[2], David E. Bernal Neira[1]

[1]Davidson School of Chemical Engineering, Purdue University
[2]Department of Chemical Engineering, Carnegie Mellon University

PURDUE UNIVERSITY

PSE Seminar October 20th, 2023