

# A CONVEXIFICATION-BASED OUTER-APPROXIMATION METHOD FOR CONVEX AND NONCONVEX MINLP

Zedong Peng<sup>a</sup>, Kaiyu Cao<sup>a</sup>, Kevin C. Furman<sup>b</sup>, Can Li<sup>a</sup>, Ignacio E. Grossmann<sup>c</sup>, David E. Bernal Neira<sup>a,d,e</sup>

<sup>a</sup>*Davidson School of Chemical Engineering, Purdue University, 480 Stadium Mall Drive, West Lafayette, IN, 47907, USA*

<sup>b</sup>*ExxonMobil Upstream Research Co, 22777 Springwoods Village Parkway, Spring, TX, 77389, USA*

<sup>c</sup>*Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA*

<sup>d</sup>*Research Institute for Advanced Computer Science, Universities Space Research Association, Mountain View, CA, USA*

<sup>e</sup>*Quantum Artificial Intelligence Laboratory (QuAIL), NASA Ames Research Center, Mountain View, CA, USA*  
*dbernaln@purdue.edu*

## Abstract

The advancement of domain reduction techniques has significantly enhanced the performance of solvers in mathematical programming. This paper delves into the impact of integrating convexification and domain reduction techniques within the Outer-Approximation method. We propose a refined convexification-based Outer-Approximation method alongside a Branch-and-Bound method for both convex and nonconvex Mixed-Integer Nonlinear Programming problems. These methods have been developed and incorporated into the open-source Mixed-Integer Nonlinear Decomposition Toolbox for Pyomo-MindtPy. Comprehensive benchmark tests were conducted, validating the effectiveness and reliability of our proposed algorithms. These tests highlight the improvements achieved by incorporating convexification and domain reduction techniques into the Outer-Approximation and Branch-and-Bound methods.

**Keywords:** Mixed-Integer Nonlinear Programming, Outer-Approximation, LP/NLP Branch and Bound, Domain Reduction.

## 1. Introduction

Mixed-integer nonlinear programming (MINLP) has broad applications in process systems engineering (PSE), including planning, scheduling, and control. It offers a powerful modeling framework that optimizes discrete and continuous variables involved in linear and nonlinear constraints. However, the combinatorial complexity, nonlinearity, and even nonconvexity lead to substantial challenges in optimizing such problems.

Generally, MINLP can be classified as convex and nonconvex, depending on the convexity of its continuous relaxation. The algorithms for MINLP are primarily categorized into Branch-and-Bound (B&B) methods and decomposition methods (Kronqvist et al., 2019). The main idea of MINLP decomposition algorithms is to generate linear inequalities to approximate nonlinear constraints and iteratively solve the

relaxed Mixed-Integer Linear Programming (MILP) main problem and Nonlinear Programming (NLP) subproblems. Decomposition methods for convex MINLP problems include the Outer-Approximation (OA) method (Duran and Grossmann, 1986). This method involves solving an MILP defined by linear inequalities that relax the nonlinear constraints, known as OA cuts, and an NLP with the main problem's integer solution fixed. The first-order Taylor approximation of the nonlinear constraints defines these OA cuts. To reduce the MILP problem solution time, the Linear Programming and Nonlinear Programming-based B&B (LP/NLP-B&B) method (Quesada and Grossmann, 1992) maintains the same B&B for the MILP, solves NLPs at the tree's integer nodes, and uses OA cuts to improve the searching bounds. Consequently, these decomposition methods are often called multi-tree and single-tree, based on their management of the MILP problem. A significant limitation of these methods is their initialization, usually given by relaxing the nonlinearity of the problem and relying on the cuts generated at each iteration to provide a better linear approximation as iterations progress.

While the OA method is effective for convex MINLPs, its limitations become apparent for nonconvex MINLP, where OA cuts do not guarantee validity in relaxations of nonlinear functions, precluding global optimality guarantees. Other relaxation techniques have been developed for nonconvex MINLPs addressing this challenge. Among these, the Auxiliary Variable Method (AVM) and McCormick relaxations are successful strategies for generating relaxations of nonconvex factorable functions (Tawarmalani and Sahinidis, 2013). AVM achieves this by introducing an auxiliary variable and a corresponding equality constraint for each intermediate nonlinear factor in a function, leading to computational efficiency by decomposing the function into simpler, lower-dimensional components. However, this method entails incorporating many auxiliary variables and constraints. In contrast, McCormick relaxations maintain the dimension of the original function and use a recursive approach to produce the required convex and concave relaxations effectively (Chachuat, 2013).

In addition to advancements in reformulations and optimization algorithms, the performance of optimization solvers has significantly improved through domain reduction techniques. These techniques encompass bound tightening, eliminating redundant variables and constraints, and convexification (Zhang et al., 2020). The bound tightening techniques include Feasibility-based Bound Tightening (FBBT), Optimality-based Bound Tightening (OBBT), and Marginals-based Bound Tightening (Zhang et al., 2020). Domain reduction methods, including convexification cuts and bound tightening techniques, have been successful for spatial B&B methods. These tighter relaxations provide stronger dual bounds, accelerating the B&B process by facilitating node pruning and efficiently identifying optimal solutions. However, decomposition-based MINLP solvers have not yet fully harnessed the potential of domain reduction techniques.

This work investigates the efficacy of domain reduction techniques within OA methods, applicable to both convex and nonconvex MINLP problems. It is understood that domain reduction techniques can be used during the presolve stage and at each node within the B&B tree, a strategy known as the branch-and-reduce method. Similarly, in the OA method, these techniques can be employed both in the presolve phase and during solving integer-fixed NLP subproblems. However, this work focuses on the impact of domain reduction methods at the method's initialization stage.

## **2. Solution algorithm**

The general form of a MINLP problem is as follows.

*A CONVEXIFICATION-BASED OUTER-APPROXIMATION METHOD FOR CONVEX AND NONCONVEX MINLP*

$$\begin{aligned}
 & \min_{x,y} f(x,y) \\
 & \text{s.t. } g_j(x,y) \leq 0, \forall j = 1, \dots, l \\
 & x \in [\underline{x}, \bar{x}] \subseteq \mathbb{R}^n, y \in \{\underline{y}, \dots, \bar{y}\} \subseteq \mathbb{Z}^m,
 \end{aligned} \tag{MINLP}$$

Where  $x$  and  $y$  represent continuous variables and discrete variables, respectively. Upper and lower variable bounds are determined by over- and underbars, respectively. Both the objective  $f(x,y)$  and constraints  $g_j(x,y)$  are potentially nonlinear functions.

The OA solution method for MINLP involves an iterative two-step procedure. The first step in iteration  $k$  determines the integer variables' values  $y^{k+1}$  by solving problem (OA-MILP), defined by the OA cuts. Its optimal objective function value, encoded in variable  $\mu$ , provides a dual lower bound (LB) to the original MINLP problem's optimal objective.

$$\begin{aligned}
 & \min_{x,y,\mu} \mu \\
 & \text{s.t. } f(x^i, y^i) + \nabla f(x^i, y^i)^T \begin{bmatrix} x - x^i \\ y - y^i \end{bmatrix} \leq \mu \quad \forall i = 1, \dots, k \\
 & g_j(x^i, y^i) + \nabla g_j(x^i, y^i)^T \begin{bmatrix} x - x^i \\ y - y^i \end{bmatrix} \leq 0 \quad \forall i = 1, \dots, k, \forall j \in I \\
 & x \in [\underline{x}, \bar{x}] \subseteq \mathbb{R}^n, y \in \{\underline{y}, \dots, \bar{y}\} \subseteq \mathbb{Z}^m, \mu \in \mathbb{R}^1.
 \end{aligned} \tag{OA - MILP}$$

The second step is determining the continuous variables' values  $x^{k+1}$  by solving problem (NLP-I) whose optimal solution yields a primal upper bound (UB) to problem (MINLP).

$$\begin{aligned}
 & \min_{x,y} f(x, y^{k+1}) \\
 & \text{s.t. } g_j(x, y^{k+1}) \leq 0, \forall j = 1, \dots, l \\
 & x \in [\underline{x}, \bar{x}] \subseteq \mathbb{R}^n.
 \end{aligned} \tag{NLP - I}$$

If problem (NLP-I) is infeasible, the following feasibility subproblem is solved to minimize a norm  $p$  of the constraint violations  $s$ , as a result updating  $x^{k+1}$ .

$$\begin{aligned}
 & \min_{x,y,s} \|s\|_p \\
 & \text{s.t. } g_j(x, y^{k+1}) \leq s_j \quad \forall j = 1, \dots, l \\
 & x \in [\underline{x}, \bar{x}] \subseteq \mathbb{R}^n, s \in [0, \infty) \subseteq \mathbb{R}_+^l.
 \end{aligned} \tag{NLP - f}$$

As shown in Figure 1.a, the OA method begins by solving the relaxed NLP problem and then iteratively solves the (OA-MILP), (NLP-I), and (NLP-f) problems. The key to this process is the progressive accumulation of OA cuts, which incrementally narrows the gap between the LB and the UB. The iterations continue until LB and UB converge, culminating in the OA method reaching the optimal solution. This method is guaranteed to find the global optimal solution of convex MINLPs (Duran and Grossmann, 1986).

Maintaining a single MILP tree for the LP/NLP-B&B method can be implemented using the *LazyConstraint* callback function through callback functions in current MILP solvers, as shown in Figure 1.b. This method initializes by solving the relaxed NLP problem as well. Then, a B&B method is used to solve problem (OA-MILP). Whenever an incumbent solution is found in the search tree, (NLP-I) is solved, and OA cuts are added as lazy constraints to the MILP tree. This B&B process is guaranteed to terminate at the global optimal solution of convex MINLP problems (Quesada and Grossmann, 1992).

The OA and LP/NLP-B&B methods generate tight cuts at the boundary of the nonlinear feasible region defined by the original problem constraints by incurring the cost of solving NLP subproblems. If the (NLP-I) subproblem is infeasible, (NLP-f) is solved to find the point closest to the feasible region to generate the tightest possible cuts. Since the linear inequality constraints are accumulated iteratively, the main problem in the early iterations

is that a poor approximation of the original MINLP model is obtained. Consequently, the integer combination provided by the main problem tends to yield an infeasible (NLP-I) subproblem, and no primal bound can be obtained.

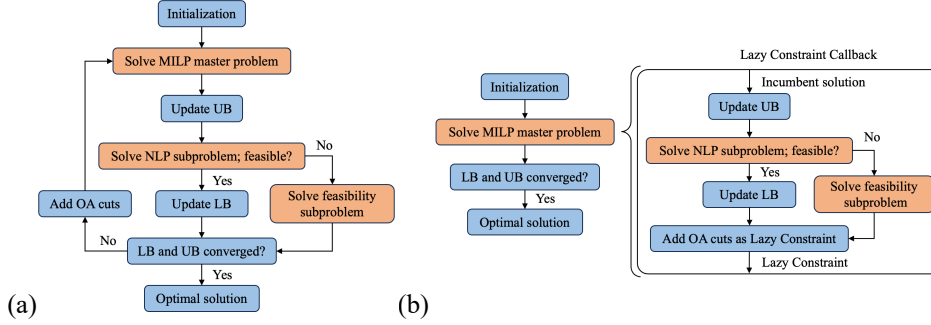


Figure 1. (a) Outer-Approximation method and (b) LP/NLP-based B&B method

In this work, we apply the domain reduction methods to the initialization stage of the OA method to resolve this issue. Eq. (1) shows the tightened bounds of the discrete and continuous variables,  $[\underline{x}', \bar{x}']$  and  $\{\underline{y}', \dots, \bar{y}'\}$ , respectively. Eq. (2) corresponds to the convexification linear cuts generated by the auxiliary variable method or reformulations, where  $z$  are auxiliary variables. Since both Eq. (1) and (2) are applied at the initialization stage, they are valid for (OA-MILP), (NLP-I), and (NLP-f).

$$x \in [\underline{x}', \bar{x}'] \in [\underline{x}, \bar{x}] \subseteq \mathbb{R}^n; y \in \{\underline{y}', \dots, \bar{y}'\} \in \{\underline{y}, \dots, \bar{y}\} \subseteq \mathbb{Z}^m \quad (1)$$

$$Ax + By + Cz \leq b \quad (2)$$

Considering that the convexification cuts are the relaxation of nonlinear constraints, they are redundant in the NLP subproblems where the original nonlinear constraints are included. Therefore, we denote the (NLP-I) and (NLP-f) problems with convexification cuts and tightened bounds as complete-scale NLP problems. The (NLP-I) and (NLP-f) problems with only tightened bounds are denoted reduced-scale NLP problems.

This work also considers modified alternatives of the OA and LP/NLP-B&B methods to guarantee global optimality for nonconvex MINLP problems. Several modifications have been introduced to provide such global optimality guarantees, denoted as global OA (GOA) and global LP/NLP-B&B (GLP/NLP-B&B). First, instead of adding OA cuts, the affine underestimators and overestimators are generated based on the convex and concave McCormick relaxations using subgradient propagation (Chachuat, 2013). Second, to guarantee the algorithm's convergence, no-good cuts are generated to cut off the explored integer combinations and prevent the algorithm from repeatedly cycling through the same combinations. These enhancements enable the global algorithms to converge to the global optimum of nonconvex MINLP problems if the NLP subproblems are solved to global optimality (Kesavan et al., 2004). Furthermore, we integrate the domain reduction techniques in GOA and GLP/NLP-B&B and investigate their effect on its performance.

### 3. Benchmarking and Results

To evaluate the impact of domain reduction techniques, we use test instances from the problem library MINLPLib (Vigerske, 2014). 434 convex instances and 181 nonconvex instances are selected, adhering to the criteria that each instance must have at least one discrete variable and at least one continuous variable. For clarity in our analysis, we use (r) and (c) to distinguish between the reduced-scale and complete-scale NLP subproblems

*A CONVEXIFICATION-BASED OUTER-APPROXIMATION METHOD FOR CONVEX AND NONCONVEX MINLP*

used in the convexification-based OA and LP/NLP-B&B methods. Moreover, we indicate the results with convexification with the prefix (C-) in the following results.

The benchmark implementation is based on the **Mixed-integer nonlinear decomposition toolbox for Pyomo-MindtPy** (Bernal et al., 2018). We use both the multi-tree and single-tree implementation of OA and GOA strategy, maintaining their default configurations as a baseline. Moreover, a special version of BARON 19.4.4 is used to tighten the bounds and generate convexification cuts. Option *dolocal* is set to 0, and *numloc* is set to 0 to turn off local search during upper bounding and preprocessing in BARON. All range reduction and relaxation options are retained at their default settings. Nonlinear FBBT, OBBT, marginals-based, and linear-feasibility-based bound tightening are applied. Outer approximations of convex univariate functions and cutting planes are also applied. For the termination criteria of the algorithm, we set the absolute tolerances  $\epsilon = 10^{-5}$  and  $\epsilon_{rel} = 10^{-3}$ , along with a time limit of 900s. We use GUROBI 10.0.0 as the MILP solver, IPOPTH 3.14 as the NLP solver for convex instances, and BARON 23.6.22 as the NLP solver for nonconvex instances. All tests ran on a Linux cluster with 48 AMD EPYC 7643 2.3GHz CPUs and 1 TB RAM, with each test restricted to using only a single thread.

The time and iteration performance profiles of the convex instances are presented in Figure 2. For the LP/NLP-B&B method, the number of iterations refers to the number of (NLP-I) subproblems solved. Overall, the convexification-based OA and LP/NLP-B&B methods utilizing reduced-scale NLP subproblems outperform the other solver alternatives regarding solution time. Regarding the number of iterations, both the OA and LP/NLP-B&B methods benefit from the convexification cuts and the bound tightening techniques. Interestingly, the choice between complete-scale and reduced-scale NLP subproblems does not significantly impact iteration performance.

However, it is noteworthy that the convexification-based LP/NLP-B&B method with complete-scale NLP subproblems underperforms in time performance compared to the standard LP/NLP-B&B method. This coincides with our previous statement that the convexification cuts are redundant and increase the computational complexity of (NLP-I) and (NLP-f) subproblems. For instances that can be solved within one second, both the standard OA method and the LP/NLP B&B method are more efficient than their convexified counterparts, as both bound tightening and convexification cuts entail additional processing time. Nonetheless, the bound tightening and convexification cuts generally enhance the performance of both the OA and LP/NLP B&B methods.

The benchmark results of nonconvex instances are presented in Figure 3. Similar to the convex cases, the convexification-based OA and LP/NLP B&B methods outperform the others in solution time and number of iterations. This consistent performance across convex and nonconvex MINLP problems demonstrates the effectiveness of the proposed convexification-based OA and LP/NLP B&B methods.

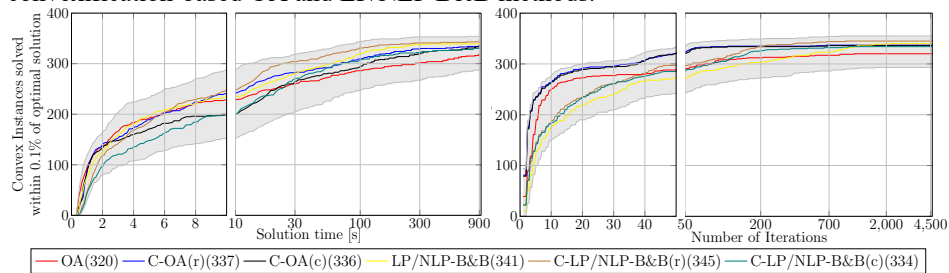


Figure 2. Time and iteration performance profile of 434 convex MINLP instances

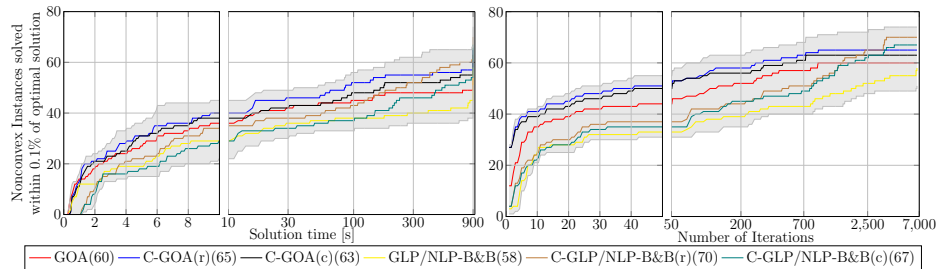


Figure 3. Time and iteration performance profile of 181 nonconvex MINLP instances

#### 4. Conclusions

This work explores the impact of domain reduction techniques implemented in B&B-based solvers on the OA and the LP/NLP-B&B methods. These effects were investigated for variations of these methods to solve both convex and nonconvex MINLPs to global optimality. The proposed convexification-based OA and LP/NLP-B&B methods are implemented within the open-source solver MindtPy. Our benchmarking results highlight the significant improvements by domain reduction techniques in enhancing the efficiency of the OA and LP/NLP-B&B methods, observed by reducing the computational times and the number of iterations required for solving convex and nonconvex MINLP problems to global optimality. These results highlight the value of implementing domain reduction techniques, which are successful for B&B methods, in MINLP decomposition algorithms.

#### References

1. Kronqvist, J., Bernal, D. E., Lundell, A., Grossmann, I. E., 2019. A review and comparison of solvers for convex MINLP. *Optimization and Engineering*, 20, 397-455.
2. Duran, M. A., Grossmann, I. E., 1986. An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical programming*, 36, 307-339.
3. Quesada, I., Grossmann, I. E., 1992. An LP/NLP based branch and bound algorithm for convex MINLP optimization problems. *Computers & chemical engineering*, 16(10-11), 937-947.
4. Tawarmalani, M., Sahinidis, N. V., 2013. *Convexification and global optimization in continuous and mixed-integer nonlinear programming: theory, algorithms, software, and applications*, 65.
5. Zhang, Y., Sahinidis, N. V., Nohra, C., Rong, G., 2020. Optimality-based domain reduction for inequality-constrained NLP and MINLP problems. *Journal of Global Optimization*, 77, 425-454.
6. Chachuat, B.: MC++, 2013: a versatile library for bounding and relaxation of factorable functions. <http://www3.imperial.ac.uk/environmentenergyoptimisation/software>.
7. Kesavan, P., Allgor, R. J., Gatzke, E. P., Barton, P. I. 2004. Outer approximation algorithms for separable nonconvex mixed-integer nonlinear programs. *Mathematical Programming*, 100, 517-535.
8. Vigerske, S., 2014. Towards MINLPLib 2.0 Model instance collections.
9. Bernal, D. E., Chen, Q., Gong, F., Grossmann, I. E., 2018. Mixed-integer nonlinear decomposition toolbox for Pyomo (MindtPy). In *Computer Aided Chemical Engineering*, 44, 895-900.