

# A Framework for Globally Optimizing Mixed-Integer Signomial Programs

Ruth Misener · Christodoulos A. Floudas

Received: date / Accepted: date

**Abstract** Mixed-integer signomial optimization problems have broad applicability in engineering. Extending the Global Mixed-Integer Quadratic Optimizer, GloMIQO [1], this manuscript documents a computational framework for deterministically addressing mixed-integer signomial optimization problems to  $\varepsilon$ -global optimality. This framework generalizes the GloMIQO strategies of (1) reformulating user input, (2) detecting special mathematical structure, and (3) globally optimizing the mixed-integer nonconvex program. Novel contributions of this paper include: flattening an expression tree towards term-based data structures; introducing additional nonconvex terms to interlink expressions; integrating a dynamic implementation of the reformulation-linearization technique into the branch-and-cut tree; designing term-based underestimators that specialize relaxation strategies according to variable bounds in the current tree node. Computational results are presented along with comparison of the computational framework to several state-of-the-art solvers.

**Keywords** mixed-integer signomial optimization · global optimization · numerical optimization software

**Mathematics Subject Classification (2000)** MSC 90C26 · MSC 90C57

---

R. Misener · C. A. Floudas (✉)  
Department of Chemical and Biological Engineering  
Princeton University; Princeton, NJ, USA 08544-5263  
E-mail: floudas@titan.princeton.edu  
*Present address:* R.M.; Centre for Process Systems Engineering; Imperial College London

## 1 Introduction

Recent work developed a generic computational framework for addressing mixed-integer quadratically-constrained quadratic programs (MIQCQP) to  $\varepsilon$ -global optimality [1–3]. The Global Mixed-Integer Quadratic Optimizer, GloMIQO, a software implementation of the framework, (1) reformulates user input, (2) detects special mathematical structure, and (3) uses a branch-and-cut global optimization algorithm to deterministically solve the MIQCQP.

Expanding the framework, this paper considers mixed-integer signomial optimization problems (MISO). MISO was originally proposed when convex posynomial geometric programs could not model important engineering applications [4–7]. Signomial functions are nonconvex in general; the seminal paper of Maranas and Floudas [8] introduced the first global signomial optimization algorithm. Interesting domains of MISO include: synthesizing heat exchanger networks [9, 10]; planning in a petrochemical production network [11]; finding the global minimum energy configuration of microclusters [12, 13]; designing heat-integrated nonsharp distillation sequences [14]; synthesizing complex nonisothermal reactor networks [15]; optimizing product portfolio selection [16, 17]; designing metabolic and signal transduction networks [18, 19]; designing a thermochemical-based process superstructure converting biomass, coal, and natural gas to liquid fuels [20].

This manuscript contextualizes a computational framework for MISO with respect to previous work in Section 2, describes the algorithmic components in Section 3, introduces a test suite highly relevant to engineering in Section 4, validates the framework in Section 5, and concludes in Section 6. The primary contribution is to integrate existing ideas into an algorithmic framework for globally optimizing MISO. The novel components: flattening an expression tree; introducing additional nonconvex terms to interlink expressions; specializing relaxation strategies to the current tree node are strategies for meaningfully interconnecting the various algorithms. An implementation is publicly available within the MINLP solver ANTIGONE [21].

## 2 Problem Definition and Literature Review

We define MISO as:

$$\begin{aligned}
 \min \quad & \sum_{s=1}^{S_0} c_{0,s} \cdot f_{0,s}(x) + x^T \cdot Q_0 \cdot x + a_0 \cdot x \\
 \text{s.t.} \quad & b_m^L \leq \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x \leq b_m^U \quad \forall m \in \{1, \dots, M\} \\
 & x \in \mathbb{R}^C \times \{0, 1\}^B \times \mathbb{Z}^I
 \end{aligned} \tag{MISO}$$

where  $C$ ,  $B$ ,  $I$ , and  $M$  represent the number of continuous variables, binary variables, integer variables, and constraints, respectively; recall that setting  $b_m^L = b_m^U$  yields an equality equation. We assume that it is possible to infer finite bounds  $[x_i^L, x_i^U]$  on the variables participating in nonconvex terms and that a relaxation of MISO can be formulated as a bounded MILP. The quadratic and linear terms are defined on continuous, binary, and/or integer variables. The signomial terms  $f_{m,s}(x)$  are defined *only* on the continuous variables:

$$f_{m,s}(x) = \prod_{c=1}^C x_c^{p_{sm,c}} : x \in \mathbb{R}^C \quad (1)$$

where the powers  $p_{sm,c}$  are constant real numbers. Simplifying interval analysis, we assume that each variable  $x_i$  raised to a noninteger power is nonnegative. Finally, we explicitly admit absolute values  $|x_i|$  into the formulation because addressing MISO implicitly encompasses absolute values (*i.e.*, we reformulate  $\sqrt{x_i^2}$  into the equivalent convex term  $|x_i|$  [22]).

The computational framework for MISO global optimization represents a selection of algorithms. For a broader perspective on the range of global optimization methods, we refer the reader to the reviews of Floudas and co-workers [23–25] and to a variety of texts [26–31]. The software implementation of the MISO global optimization framework addresses a class of mixed-integer nonlinear programs (MINLP). We briefly mention several code bases; complete treatment of numerical optimization software is in the review of Bussieck and Vigerske [32].

GloMIQO [1–3] and ANTIGONE [21] are the numerical optimization code bases most similar to the current work; this framework encompasses and extends GloMIQO and is publically available within ANTIGONE (beginning GAMS 24.1). GloMIQO and this framework are equivalent when there are no signomial terms in MISO ( $c_{m,s} = 0 \forall s, m$ ). Other relevant codes include  $\alpha$ BB [27, 33–36]; BARON [31, 37, 38]; Couenne [39, 40]; LINDO [41, 42]; SCIP [43–47]; and SGO [48–51]. Observe that  $\alpha$ BB, ANTIGONE, BARON, Couenne, LINDO, SCIP, and SGO all admit more functional forms than this computational framework; as with GloMIQO, our focus is careful analysis of special mathematical structure for specific problem classes.

Like GloMIQO, the initial components in the algorithmic framework involve reformulating user input. Two complementary reformulation strategies are: constructing an operator-based factorable programming tree [37, 39, 47, 52, 53] and dividing nonconvex expressions into terms that are addressed individually [33–36, 51]. The advantage of the vertical, expression tree data structure is that graph transversal techniques are easily exploited to generate tight convex underestimators and infer vari-

able bounds based on tree relationships [39, 47, 53]. The complementary horizontal, term-based data structures easily admit multivariable relaxations that are specifically designed for particular functional forms. For example, beyond the convex, bilinear, trilinear, fractional, fractional trilinear, univariate concave, and general nonconvex terms as introduced by Adjiman et al. [33, 34], underestimators have been introduced or improved for: fractional terms [31, 36, 54]; trilinear terms [55, 56]; quadrilinear terms [57]; odd degree monomials [58]; signomial terms [8, 48, 50]; low-dimensional edge-concave terms [59–62]; submodular functions [63]; and interesting products [36, 64–67].

Operator- and term-based strategies are mutually reinforcing [68, 69]. Gatzke et al. [69] combine vertical, operator-based data structures with horizontal, term-based data structures; the tree-flattening operations in Section 3.1.1 represent an alternative integration of operator- and term-based strategies. Flattening without integrating term-based convexifications is equivalent to product disaggregation [31, 70]; these reformulations could conceivably tighten convex underestimators even if we were limited to operator-based relaxations.

As in GloMIQO, the framework for MISO detects and exploits special mathematical structure. Beyond identifying individual functional forms, the Reformulation-Linearization Technique (RLT) is a strategy that effectively interlinks variables, nonlinear terms, and equations [30, 71–76]. The GloMIQO RLT implementation identifies all variable/inequality equation and equation/equality products that do not introduce new bilinear terms into the model formulation [1]; GloMIQO 2.0 will even add nonconvex bilinear terms to the problem formulation to increase the number of variable/equality equation products [3]. Depending on the product, GloMIQO may directly add the RLT equation to the model formulation, dynamically introduce the equation as a cutting plane, or use the equation in a bounds-updating strategy. The MISO framework generalizes the GloMIQO RLT strategies; we automatically add nonconvex terms to interlink higher-order terms and dynamically integrate the resulting RLT equations into the branch-and-cut tree.

A major design choice for relaxing nonconvex functions is alternatively using auxiliary variables to represent nonlinear terms [31, 37–39, 47, 53, 69] or exclusively employing underestimators that do not introduce additional variables [33, 34, 51, 52, 77]. Mitsos et al. [77] document advantages to each approach. This computational framework for MISO follows the former tactic and introduces auxiliary variables because of the importance it places on an adaptive RLT implementation. It is easiest to establish RLT interconnections between the variables, nonlinear terms, and equations in MISO with auxiliary variables.

The branch-and-cut global optimization phase of this global optimization framework for MISO is largely analogous to GloMIQO except that the range of functional forms is significantly broader. GloMIQO uses the special structure elucidated in the detection phase to generate tight convex relaxations, dynamically add separating cuts, partition the search space, bound the variables, and find feasible solutions [1, 3].

### 3 Global Optimization Framework for MISO

This section outlines the structure of the global optimization framework for MISO. The major components of the framework are: reformulation (§3.1), special mathematical structure detection (§3.2), and branch-and-cut global optimization (§3.3). The proposed framework expands and extends GloMIQO [1–3]; we therefore limit the discussion to additional algorithmic components.

#### 3.1 Reformulating User Input

The transformation strategies couple vertical, operator-based and horizontal, term-based data structures. The following sections describe the reformulation steps of flattening the expression tree and adding nonconvex terms for advanced RLT equations.

##### 3.1.1 Flattening the expression tree

Smith and Pantelides [53] describe how a nonlinear function:

$$1 / (0.1 + (x_1 - 4)^2 + (x_2 - 4) \times (\sqrt{x_3} - 1)) \quad (2)$$

can be reformulated into a binary expression tree using a diagram similar to Figure 1(a). They further explain how the binary expression tree is, in practice, reformulated into the Figure 1(b) factorable programming tree where an auxiliary variable is introduced for every operator except addition and subtraction. Factorable programming trees are often used in global optimization code bases [31, 37–39, 47, 53, 69, 77].

Adjiman et al. [33, 34] describe an alternative method of addressing nonlinear expressions by splitting each function into a sum of *terms* that are individually underestimated. The  $\alpha$ BB methodology adapts its approach to the special mathematical structure of each nonconvex term; the ideologically similar SGO algorithm underestimates individual terms with a special emphasis on signomial forms [48–51].

The MISO framework transforms a factorable programming tree into a flattened expression tree to capitalize on the development of tight convex underestimators for

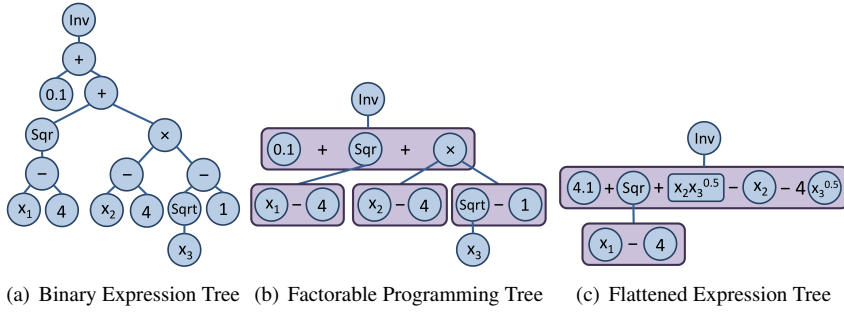


Fig. 1: (a) A binary tree represents algebraic expressions; (b) A factorable programming tree uses operator-based relaxations [53]; (c) Transformation to a flattened expression tree allows access to term-based underestimators; Convex form  $(x_1 - 4)^2$  is not expanded.

specific classes of nonlinear terms (*e.g.*, [8, 31, 36, 48, 50, 54–67]). This hybrid approach reformulates towards multivariable terms with specialized underestimators while maintaining a tree-like representation of powers that cannot be distributed and convex operators that can be exploited by dynamic cut generation. The framework flattens the tree using the following steps:

**1. Expanding Products:**  $(x_1 - x_2^{0.5}) \cdot (x_2 + x_3^3) \implies x_1 \cdot x_2 + x_1 \cdot x_3^3 - x_2^{1.5} - x_2^{0.5} \cdot x_3^3$

As illustrated in Example 3.1.1.1, product expansion exposes additional terms that may be individually underestimated using special structure analysis. The MISO framework also expands products for the purpose of creating a web of interconnections between many individual terms (Section 3.1.2).

**2. Expanding Integer Powers:**  $(x_1 - 2)^3 \implies x_1^3 - 6 \cdot x_1^2 + 12 \cdot x_1 - 8$

Expansion operations are applied to nonconvex integer powers; the framework preserves convex terms for better dynamic cut generation (§3.3.2). Figure 1(c) illustrates that the framework automatically replaces  $(x_1 - 4)^2$  with an auxiliary variable rather than expanding it to  $x_1^2 - 8 \cdot x_1 + 16$ . A look-ahead function determines the convexity/concavity of a term;  $(x_1 - 4)^2$  would be expanded if, for example, Expression (2) participated in the objective of a maximization problem.

**3. Distributing Noninteger Powers**  $(16 \cdot x_1 \cdot x_2^8)^{0.25} \implies 2 \cdot x_1^{0.25} \cdot x_2^2$

The framework only distributes noninteger powers when the power participates in a nonconvex direction. We do not distribute noninteger powers if more than one component inside the parenthesis can be negative; this is to avoid initializing a tree with multiple root nodes (*e.g.*, distributing  $(x_1 \cdot x_2)^{0.5}$ ;  $x_1, x_2 \in [-1, 1]$  would require an added rule that the variables are either both positive or both negative).

When the framework deconvolutes terms with nested powers, it coincidentally tracks the variable bound rules associated with powers. For example,  $\sqrt{x_i^2}$  is replaced with  $|x_i|$  but  $(\sqrt{x_i})^2$  is equivalent to  $x_i$  with the additional condition  $x_i \geq 0$ . As discussed in the introduction, the framework assumes that only nonnegative numbers can be raised to noninteger powers; therefore  $(x_i^3)^{1/3}$  reformulates to  $x_i$  with the additional condition that  $x_i \geq 0$ .

**Example 3.1.1.1:** Consider the classical Goldstein and Price [78, 79] test problem:

$$\begin{aligned} \min & \left( 1 + (x_1 + x_2 + 1)^2 \cdot (19 - 14 \cdot x_1 + 3 \cdot x_1^2 - 14 \cdot x_2 + 6 \cdot x_1 \cdot x_2 + 3 \cdot x_2^2) \right) \\ & \cdot \left( 30 + (2 \cdot x_1 - 3 \cdot x_2)^2 \cdot (18 - 32 \cdot x_1 + 12 \cdot x_1^2 + 48 \cdot x_2 - 36 \cdot x_1 \cdot x_2 + 27 \cdot x_2^2) \right) \\ \text{s.t. } & x_1; x_2 \in [-2, 2] \end{aligned}$$

A factorable programming tree representing Goldstein and Price [78, 79] is three layers deep; a flattened expression tree is two layers deep. The MISO framework deduces that both  $(x_1 + x_2 + 1)^2$  and  $(2 \cdot x_1 - 3 \cdot x_2)^2$  participate in a convex direction; it therefore introduces two auxiliary variables  $x_3 = x_1 + x_2 + 1$  and  $x_4 = 2 \cdot x_1 - 3 \cdot x_2$ :

$$\begin{aligned} \min & \left( 1 + x_3^2 \cdot (19 - 14 \cdot x_1 + 3 \cdot x_1^2 - 14 \cdot x_2 + 6 \cdot x_1 \cdot x_2 + 3 \cdot x_2^2) \right) \\ & \cdot \left( 30 + x_4^2 \cdot (18 - 32 \cdot x_1 + 12 \cdot x_1^2 + 48 \cdot x_2 - 36 \cdot x_1 \cdot x_2 + 27 \cdot x_2^2) \right) \\ \text{s.t. } & x_3 = x_1 + x_2 + 1 \\ & x_4 = 2 \cdot x_1 - 3 \cdot x_2 \\ & x_1; x_2 \in [-2, 2]; x_3; x_4 \in \mathbb{R} \end{aligned}$$

The framework then distributes  $x_3^2$  and  $x_4^2$  and multiplies the two remaining products:

$$\begin{aligned} \min & -420 \cdot x_1 \cdot x_3^2 + 180 \cdot x_1 \cdot x_2 \cdot x_3^2 - 32 \cdot x_1 \cdot x_4^2 - 800 \cdot x_1 \cdot x_2 \cdot x_3^2 \cdot x_4^2 \\ & + 318 \cdot x_1 \cdot x_2^2 \cdot x_3^2 \cdot x_4^2 - 860 \cdot x_1 \cdot x_3^2 \cdot x_4^2 - 36 \cdot x_1 \cdot x_2 \cdot x_4^2 + 54 \cdot x_1 \cdot x_3^2 \cdot x_3^2 \cdot x_4^2 \\ & + 90 \cdot x_1^2 \cdot x_3^2 + 12 \cdot x_1^2 \cdot x_4^2 + 288 \cdot x_1^2 \cdot x_2 \cdot x_3^2 \cdot x_4^2 - 99 \cdot x_1^2 \cdot x_2^2 \cdot x_3^2 \cdot x_4^2 \\ & + 730 \cdot x_1^2 \cdot x_3^2 \cdot x_4^2 - 264 \cdot x_1^3 \cdot x_3^2 \cdot x_4^2 - 36 \cdot x_1^3 \cdot x_2 \cdot x_3^2 \cdot x_4^2 + 36 \cdot x_1^4 \cdot x_3^2 \cdot x_4^2 \\ & - 420 \cdot x_2 \cdot x_3^2 + 48 \cdot x_2 \cdot x_4^2 + 660 \cdot x_2 \cdot x_3^2 \cdot x_4^2 + 90 \cdot x_2^2 \cdot x_3^2 \\ & + 27 \cdot x_2^2 \cdot x_4^2 - 105 \cdot x_2^2 \cdot x_3^2 \cdot x_4^2 - 234 \cdot x_2^3 \cdot x_3^2 \cdot x_4^2 + 81 \cdot x_2^4 \cdot x_3^2 \cdot x_4^2 \\ & + 342 \cdot x_3^2 \cdot x_4^2 + 570 \cdot x_3^2 + 18 \cdot x_4^2 + 30 \\ \text{s.t. } & x_3 = x_1 + x_2 + 1 \\ & x_4 = 2 \cdot x_1 - 3 \cdot x_2 \\ & x_1; x_2 \in [-2, 2]; x_3; x_4 \in \mathbb{R} \end{aligned}$$

Each of the 25 signomial and 2 convex quadratic terms are underestimated individually using special structure analysis. The terms are also tied together using a dynamic RLT implementation (§3.2.1).

Table 1: Goldstein and Price [78, 79] case study (known global solution: 3)

Solver	Version	Root Node	Objective
Framework		-1.491e+07	3.0000e+00
BARON	12.3.3	-8.397e+08	3.0000e+00
LINDO	8.0	-8.788e+08	3.0000e+00
SCIP	3.0	-8.498e+05	2.9999e+00

Termination Criteria:  $\text{Gap} = 100 \times \left( \frac{UB-LB}{|LB|} \right) = 1 \times 10^{-4}\%$

Goldstein and Price [78, 79] is a simple test case used to demonstrate the effect of flattening the expression tree. Table 1 delineates the root node relaxation for each solver and the objective at the global optimum; Couenne 0.4 is excluded because it has significant numerical trouble. The root node relaxation of the MISO framework is roughly 50 times tighter than either BARON 12.3.3 or LINDO 8.0; this difference is a direct consequence of reformulation.

### 3.1.2 Adding Nonconvex Terms for Advanced RLT

The reformulations described in Section 3.1.1 expose multivariable nonconvex terms that may have specially-designed underestimators, but the transformations introduce two weaknesses that need to be addressed:

- Expanding the terms may discard relationships between expressions; these interconnections are typically used to generate tight underestimators and infer tighter variable bounds [39, 47, 53].
- The convex hull of terms with one or two variables is often known [36, 52, 54, 58], but terms with many variables may be subject to weaker relaxations.

To ameliorate these two weaknesses while preserving the strength of reformulating towards flattened terms, the reformulation stage of the MISO framework introduces nonconvex terms that the special mathematical structure detection phase will later use to tie together the terms; the objective is to create a web of interconnections between the terms and thereby reinforce the relaxation strength.

For each of the nonlinear terms in the reformulated model, the MISO framework checks to see if the convex hull of the term is known. If it is not, then the framework checks to see if the term can be written as the product of two terms or a term and a variable. If not, then the framework introduces two smaller terms. The framework continues introducing terms until either (1) the term is the product of existing terms or variables or (2) the convex hull of the term is known.



For example, in the `minlp.org` problem `GMA_ethanol_model_CH` [80, 81], there are seven signomial terms with nonnegative variables natural to the model formulation; all seven participate in both a convex and concave direction:

$$\begin{aligned} f_1(x) &= x_3^{0.05} \cdot x_4^{0.533} \cdot x_5^{-0.0822} \cdot x_{12}; & f_2(x) &= x_2^{-0.234} \cdot x_6; & f_3(x) &= x_1^{0.746} \cdot x_5^{0.0243} \cdot x_7; \\ f_4(x) &= x_2^{0.732} \cdot x_5^{-0.394} \cdot x_8; & f_5(x) &= x_2^{8.61} \cdot x_9; & f_6(x) &= x_3^{0.616} \cdot x_5^{0.131} \cdot x_{10}; \\ f_7(x) &= x_3^{0.05} \cdot x_4^{0.533} \cdot x_5^{-0.0822} \cdot x_{11}. \end{aligned}$$

The MISO framework detects that  $f_3$  and  $f_6$  are edge-concave (so the convex hull is vertex polyhedral and easy to calculate) and  $f_2$  and  $f_5$  are edge-convex (so the concave hull is vertex polyhedral and easy to calculate). For the remaining under- and over-estimators, the framework uses the exponential transformation [8, 48, 50]. But exponential transformation-based relaxations do not produce the convex hull and may be less effective than other relaxations. To reinforce the exponential transformation, the framework introduces signomial terms until every term is either the product of two other terms or the convex hull is known:

$$\begin{aligned} f_8(x) &= x_5^{-0.0822}; & f_9(x) &= x_3^{0.05} \cdot x_4^{0.533} \cdot x_{12}; & f_{10}(x) &= x_2^{-0.234}; & f_{11}(x) &= x_5^{0.0243}; \\ f_{12}(x) &= x_1^{0.746} \cdot x_7; & f_{13}(x) &= x_5^{-0.394}; & f_{14}(x) &= x_2^{0.732} \cdot x_8; & f_{15}(x) &= x_2^{8.61}; \\ f_{16}(x) &= x_5^{0.131}; & f_{17}(x) &= x_3^{0.616} \cdot x_{10}; & f_{18}(x) &= x_3^{0.05} \cdot x_4^{0.533} \cdot x_{11}; & f_{19}(x) &= x_4^{0.533}; \\ f_{20}(x) &= x_3^{0.05} \cdot x_{12}; & f_{21}(x) &= x_1^{0.746}; & f_{22}(x) &= x_2^{0.732}; & f_{23}(x) &= x_3^{0.616}; \\ f_{24}(x) &= x_3^{0.05} \cdot x_{11}; & f_{25}(x) &= x_3^{0.05}. \end{aligned}$$

The 18 new signomial terms and the seven original terms create a web of interconnections with a total of 54 possible RLT equations. The terms  $f_9, f_{12}, f_{14}, f_{17}, f_{18}, f_{20}, f_{24}$  are edge-convex but need exponential transformation overestimators. The remaining terms are convex in one direction and concave in the other. The reinforcing exponential transformation, convex hulls, and RLT equations allow the MISO framework to solve `GMA_ethanol_model_CH` [80, 81] to a relative optimality tolerance of  $1 \times 10^{-4}\%$  in 1.1 CPU s.

### 3.2 Elucidating Special Mathematical Structure

After reformulation, the MISO framework detects special mathematical structure in the transformed problem. This section describes the components of finding RLT equations (§3.2.1), recognizing second-order structure (§3.2.2), and establishing underestimators for the branch-and-cut global optimization phase (§3.2.2).

### 3.2.1 Detecting Reformulation Linearization Technique (RLT) Equations

To generate RLT equations [30, 71–76], the framework considers all possible products of variables, nonlinear terms, and equations as shown in Appendix Appendix A. Nonconvex terms may have been added to the model in the reformulation phase (§3.1.2; [3]). No additional nonlinear terms are added in this step; the framework only identifies RLT equations with redundant nonlinear terms in the special structure detection phase.

Equivalently to GloMIQO [1, 3], the MISO framework adds equality equation/variable and equality equation/nonlinear term products directly to the model formulation. The other RLT equations are stored separately, updated at each node and used for dynamic cut generation (§3.3.2) and variable bounding (§3.3.3). Returning to the Goldstein and Price [78, 79] example developed in Section 3.1.1, the MISO framework finds 252 nonlinear term/variable and nonlinear term/nonlinear term products (*e.g.*, nonlinear terms  $x_1^2 \cdot x_4^2$  and  $x_3^2$  have product  $x_1^2 \cdot x_3^2 \cdot x_4^2$  that is a third nonlinear term in the model).

The strategy of finding and using all possible RLT equations is useful even for the very large MISO in GlobalLib, MINLPLib, and `minlp.org`. The C++ software implementation of the MISO framework uses an array of effective programming strategies to expedite the detection and initialization of the RLT equations. For example, the framework partitions the nonlinearly-participating terms into disjoint sets before looking for the RLT equations; there is no reason to try multiplying a variable  $x_1$  by a nonlinear term  $x_2 \cdot x_3$  if we have previously established that  $x_1$  never interacts with  $x_2$  or  $x_3$  in the model. In the case of instance `johnall` in MINLPLib (194 variables; 193 equations; 296 nonlinear terms after reformulation), the MISO framework finishes its preprocessing in approximately 9 CPU seconds; this includes the time to find and initialize the 15,859 possible RLT equations.

### 3.2.2 Recognizing Second-Order Structure: Convexity/Concavity and Edge-Convexity/Edge-Concavity

Two important second-order properties are convexity/concavity and edge-concavity/edge-convexity; the MISO framework looks for each of these four properties in each signomial term of the transformed model. Observe that the sign of the coefficients multiplying the signomial terms and the type of equation where the term participates (*i.e.*, objective function, inequality, and equality) govern whether the static relaxations (§3.2.3) and dynamic cutting planes (§3.3.2) query the convex and edge-concave underestimators or the concave and edge-convex overestimators (or both).

#### 3.2.2.1 Recognizing Convexity and Concavity

Convexity and concavity permits easy generation of a cutting plane at any point  $\hat{x}$ :

$$\begin{aligned} f_s(x) &\geq f_s(\hat{x}) + f'_s(x) \cdot (x - \hat{x}) \text{ (convex)} \\ f_s(x) &\leq f_s(\hat{x}) + f'_s(x) \cdot (x - \hat{x}) \text{ (concave)} \end{aligned}$$

For univariate terms, the framework detects the following:

1.  $x_1^a$ ;  $a \in \mathbb{Z}^+$ ;  $a \bmod 2 = 0 \implies$  convex term
2.  $x_1^a$ ;  $a \in \mathbb{Z}^-$ ;  $a \bmod 2 = 0 \implies$  convex when  $x_1 \geq 0$ ; concave when  $x_1 \leq 0$
3.  $x_1^a$ ;  $a \in \mathbb{Z}$ ;  $a \bmod 2 = 1 \implies$  convex when  $x_1 \geq 0$ ; concave when  $x_1 \leq 0$
4.  $x_1^a$ ;  $a \in (0, 1) \implies$  concave term ( $x_1 \geq 0$  by Section 1 assumptions)
5.  $x_1^a$ ;  $a \notin \mathbb{Z}$ ;  $a > 1$  or  $a < 0 \implies$  convex term ( $x_1 \geq 0$  by Section 1 assumptions)

The MISO framework detects convexity in product  $x_1^a \cdot x_2^b$  using the conditions of Maranas and Floudas [36] as documented in Appendix Appendix B. Using the analysis of univariate terms and Theorems Appendix B.1-3, the MISO framework uses initial variable bounds to label each term according to whether it is always convex, sometimes convex, or never convex; if the term is sometimes convex, the MISO framework records the convexity domain (symmetric analysis for concavity). This information is used in the branch-and-cut phase (§3.3.1, §3.3.2).

### 3.2.2.2 Recognizing Edge-Convexity and Edge-Concavity

While elucidating convexity/concavity regions allows the MISO framework to exploit gradient-based relaxations, detecting edge-concavity/edge-convexity allows the framework to develop vertex polyhedral underestimators. The following discussion of edge-concavity/edge-convexity is based on results of Tardella [60–62] and Meyer and Floudas [59]; our previous work specialized edge-concavity for both low- and high-dimensional aggregations of quadratic expressions [2, 3].

**Definition 3.2.2.1 [61]:** Let  $D = \{d_1, \dots, d_k\}$  be a set of vectors such that for each edge  $E$  of a polyhedron  $P$ ,  $D$  contains a vector parallel to  $E$ . Function  $f(x_1, \dots, x_n)$  is *edge-concave* on  $P$  if and only if it is concave on all segments in  $P$  that are parallel to an edge of  $P$ .

Tardella [60] proved that edge-concave functions admit a vertex polyhedral envelope and observed that for the special case of twice-continuously differentiable function  $f : \mathbb{R}^n \mapsto \mathbb{R}$  defined on a box  $P \equiv [x^L, x^U]$ , the edge-concave definition is equivalent to [61]:

$$\frac{\partial^2 f}{\partial x_i^2} \leq 0 \quad \forall i = 1, \dots, n. \quad (3)$$

For each signomial term, the MISO framework analyzes diagonal elements of the interval Hessian ( $[h_{ii}] \in [H]$ ) that bound all Hessian values of  $f$  on the domain  $[x_1^L, x_1^U] \times \dots \times [x_n^L, x_n^U]$  with respect to Equation (3) and finds whether each element on the diagonal of the interval Hessian is: nonpositive; spanning 0; positive. The signomial term is: *always* edge-concave if  $[h_{ii}] \leq 0 \forall i$ ; *sometimes* edge-concave if  $[h_{ii}]$  is either nonpositive or spanning 0 for all  $i$ ; *never* edge-concave if  $\exists i$  such that  $[h_{ii}] > 0$ .

These calculations are performed using the Boost interval arithmetic library [82, 83]. Although interval arithmetic typically overestimates the range of a function, there is no range overestimation in finding the edge-concavity properties of a single signomial term because (1) variables are repeated at most once and (2) the interval arithmetic library contains special functions including: powers; squares; roots; inverses. If the signomial term is sometimes edge-concave, the framework records the relevant domain and applies locally-valid cuts at relevant nodes of the branch-and-cut tree (Section 3.3.2). Symmetric analysis is performed for edge-convexity.

### 3.2.3 Establishing underestimators

Beyond the domains of convexity/concavity and edge-convexity/edge-concavity described in Section 3.2.2, the current framework implementation recognizes absolute values, odd monomials [58] and products of two convex functions [36]. The underestimating exponential transformation is applied when the variables participating in the signomial term are on a positive domain [8, 48, 50]; the current implementation does not shift the domain for negative variable values as in Maranas and Floudas [8].

Specialized underestimators have been designed for a variety of functional forms (e.g., [8, 31, 36, 48, 50, 54–67]). One advantage of the flattening transformations is that it exposes a variety of multivariable terms; flexible software design allows easy integration of new relaxations corresponding to further research advances. The C++ implementation of the MISO framework has an object-oriented structure with specialized term types inheriting from more general terms; polymorphism automatically applies the most specific relaxation to each term based on current variable bounds.

Consider the term  $|x_i|$  as an example of the way the framework specializes itself. In the most general case, the framework introduces variables  $x_i^+$ ,  $x_i^-$ , and  $y_i$  to model the MILP representation of  $|x_i|$ :

$$|x_i| \implies \begin{cases} |x_i| = x_i^+ + x_i^-; & x_i^- \leq |x_i|^U \cdot (1 - y_i); \\ x_i = x_i^+ - x_i^-; & x_i^+ \leq |x_i|^U \cdot y_i; \\ x_i^+, x_i^- \geq 0; y_i \in \{0, 1\}. \end{cases} \quad (4)$$

The MISO framework uses the MILP representation of  $|x_i|$  (*i.e.*, Equation 4) rather than the convex relaxation (*i.e.*,  $|x_i| \geq x_i$ ;  $|x_i| \geq -x_i$ ;  $|x_i| \leq \frac{|x_i^U| - |x_i^L|}{x_i^U - x_i^L} \cdot (x_i - x_i^L) + |x_i^L|$ ); this moves the complexity of addressing nonconvex absolute values to the MILP solver. The auxiliary binary variable  $y_i$  is only needed when  $x_i$  can be either positive or negative and the  $|x_i|$  term participates in a concave direction (*e.g.*, if there is a term  $-2 \cdot |x_i|$  in the objective function of a minimization problem). In cases where  $|x_i|$  only participates in a convex direction (*e.g.*, only with a positive coefficient in the objective function of a minimization problem), the binary variable  $y_i$  is discarded and Equation (4) reduces to the LP reformulation of  $|x_i|$ . In regions of the branch-and-cut tree where  $x_i \geq 0$ ,  $|x_i|$  reduces to  $x_i^+$ . The special mathematical structure detection phase identifies which underestimators will be useful during the branch-and-cut phase; the least complex underestimator is applied at each node (§3.3.1).

### 3.3 Branch-and-cut Global Optimization

After the reformulation and special structure detection phases, the MISO framework initiates a branch-and-cut global optimization algorithm that generates tight convex underestimators (§3.3.1), dynamically generates separating hyperplanes (§3.3.2), bounds the variables, branches on the search space, and finds feasible solutions (§3.3.3).

#### 3.3.1 Generating tight underestimators

As in GloMIQO, the framework generates an MILP relaxation of the MISO problem at each node of the branch-and-cut tree; Smith and Pantelides [53] discuss trade-offs associated with this strategy. The special mathematical structure phase has already detected (using the sign of the coefficients and the sense of the equations) which terms require overestimators, underestimators, or both. When an underestimator (overestimator) is nonlinear and convex (concave), the framework linearizes the relaxation using gradient-based cuts at the vertices and center of the term domain; the linearized relaxation will later be updated with cutting planes (§3.3.2).

When a term of four or fewer dimensions has an edge-concave (edge-convex) underestimator (overestimator), all facets are directly added to the relaxation using an extension of the Meyer and Floudas [59] algorithm developed for GloMIQO 2.0 [3]. Like GloMIQO, the MISO framework generalizes the diagonal-to-diagonal and diagonal-to-corner dominance tests of Meyer and Floudas [59] to many dimensions [3]. These dominance tests weed out candidate facets from entering the expensive Rikun [84] validation test. The edge-concave facet generation differs from

the GloMIQO implementation only in deducing dominance relationships that discard possible facets. For general terms, the edge-concave facet generator has to consider dominance relationships on every face of the cube while GloMIQO (for the special case of bilinear terms) reduces the number of dominance tests by inferring that one projection sufficiently captures the others [3].

Cafferi et al. [57] and Belotti et al. [85] recently showed the relative advantages of quadrilinear term relaxations based on varying applications of associativity (e.g.,  $(x_1 \cdot x_2 \cdot x_3) \cdot x_4$  and  $((x_1 \cdot x_2) \cdot x_3) \cdot x_4$ ). This work diverges from theirs by directly adding all (up to 28) facets of the convex hull to the relaxation for terms of dimension four or fewer; Example 3.3.1.1 shows the result of this design choice [86].

**Example 3.3.1.1:** The Hartree-Fock Problem instance `beryllium` [86]:

$$\begin{aligned}
\min & -15.73426 \cdot c_{12}^2 - 15.73426 \cdot c_{11}^2 + 0.5721648 \cdot c_{12} \cdot c_{22} \cdot c_{21}^2 + 1.56814504 \cdot c_{12}^2 \cdot c_{11} \cdot c_{21} \\
& + 1.56814504 \cdot c_{11}^2 \cdot c_{12} \cdot c_{22} - 7.7290488 \cdot c_{11} \cdot c_{21} - 7.7290488 \cdot c_{12} \cdot c_{22} - 4.204318 \cdot c_{21}^2 \\
& - 4.204318 \cdot c_{22}^2 + 2.2988306 \cdot c_{11}^4 + 4.5976612 \cdot c_{11}^2 \cdot c_{12}^2 - 1.329488452 \cdot c_{11} \cdot c_{21} \cdot c_{12} \cdot c_{22} \\
& + 0.8353663 \cdot c_{21}^2 \cdot c_{22}^2 + 0.41768315 \cdot c_{21}^4 + 0.41768315 \cdot c_{22}^4 + 2.124875442 \cdot c_{11}^2 \cdot c_{22}^2 \\
& + 2.124875442 \cdot c_{12}^2 \cdot c_{21}^2 + 1.460131216 \cdot c_{12}^2 \cdot c_{22}^2 + 0.5721648 \cdot c_{11} \cdot c_{21}^3 + 0.5721648 \cdot c_{12} \cdot c_{22}^3 \\
& + 0.5721648 \cdot c_{11} \cdot c_{21} \cdot c_{22}^2 + 1.56814504 \cdot c_{12}^3 \cdot c_{22} + 1.460131216 \cdot c_{11}^2 \cdot c_{21}^2 + 1.56814504 \cdot c_{11}^3 \cdot c_{21} \\
& + 2.2988306 \cdot c_{12}^4 \\
\text{s.t. } & c_{11}^2 + c_{21}^2 + 2 \cdot 0.259517 \cdot c_{11} \cdot c_{21} = 1; \quad c_{21}^2 + c_{22}^2 + 2 \cdot 0.259517 \cdot c_{12} \cdot c_{22} = 1 \\
& c_{11} \cdot c_{12} + c_{21} \cdot c_{22} + 0.259517 \cdot (c_{11} \cdot c_{22} + c_{21} \cdot c_{12}) = 0 \\
& c_{11}; c_{12}; c_{21}; c_{22} \in [-2, 1]
\end{aligned}$$

has 19 signomial terms and a root node relaxation of  $-1.141 \times 10^2$  when all facets of the convex envelopes for quadrilinear terms are introduced and the interlinking RLT equations are excluded; when we switch to directly adding facets only for dimensions three or fewer, the root node relaxation drops to  $-1.164 \times 10^2$ . In this example there is roughly a 2% difference in relaxation strength between using the convex hull of the quadrilinear term and recursively applying relaxations; the MISO framework takes the more computationally expensive option in return for the stronger relaxation.

### 3.3.2 Dynamically generating separating hyperplanes

The MISO framework convexifies the MILP relaxation of the MISO to an LP and solves the LP. Cutting planes then tighten the LP relaxation; cuts are added until the objective of the LP no longer improves. The cuts added at this phase:

- tighten individual terms; these cuts are for gradient-based nonlinear convex (concave) underestimators (overestimators) or vertex-based edge-concave (edge-convex) facets from terms with five or more variables;
- improve relaxation of bilinear term aggregates; equivalent to GloMIQO 2.0 [3];

- integrate the violated RLT equations identified in Section 3.2.1
- introduce previously-generated cuts that are violated at the current relaxation solution; each node can access the locally-valid cuts generated by its ancestors and any of the globally-valid cuts.

After cuts are no longer improving the LP objective, the framework activates the integrality constraints and solves the MILP relaxation. Dynamic cut generation is used for the first few levels of the branch-and-cut tree; in later levels of the tree, parent nodes may deactivate dynamic cut generation in their children. The MISO framework distinguishes between *globally* and *locally* valid cuts. A globally valid cut generated at node  $n$  can be applied to any node of the branch-and-cut tree; a locally valid cut generated at node  $n$  can be applied to any descendent node.

### 3.3.3 Other Global Optimization Considerations

The branching and bounding strategies are nearly identical to GloMIQO [1–3]. The reliability branching implementation extends the error analysis rule in GloMIQO [1, 39, 87] from the discrepancy between the auxiliary variable representing a quadratic term ( $\hat{w}_{i,j}^{xx}$ ) and the true term value at the current feasible point ( $\hat{x}_i \cdot \hat{x}_j$ ) to the discrepancy corresponding to both the quadratic and signomial terms:

$$\arg \max_i \left[ \sum_j |\hat{w}_{i,j}^{xx} - \hat{x}_i \cdot \hat{x}_j| + \sum_{s_i \in S} |\hat{f}_{s_i}(x) - f_{s_i}(\hat{x})| \right]$$

where  $s_i$  represents the signomial terms incorporating the variable  $x_i$ ,  $\hat{f}_{s_i}(x)$  is the value of the auxiliary variable representing the signomial term, and  $f_{s_i}(\hat{x})$  is the non-linear term substituted with the variable values at the current feasible point.

The variable bounding scheme addresses more functional forms but it is otherwise identical to GloMIQO [1]. The code base uses the Boost interval arithmetic library [82, 83] for validated interval arithmetic; this increases the reliability of the software. Although GloMIQO became somewhat slower on problem classes including computational geometry after integrating the Boost interval arithmetic library, the reliability on problem classes such as the periodic scheduling of multi-purpose batch plants and water distribution network (Tables S8 and S11 in the online supplementary material; <http://helios.princeton.edu/ANTIGONE/MISOSupp.pdf>) improved dramatically. At this writing, there is no way for a user to manually switch off the validated interval arithmetic.

Table 2: MISO Test Suite of 257 Problems

Problem Class		# Cases	Discrete	Source
minlp.org	Belgian Chocolate Problem	4	✓	[88, 89]
	Cascading Tanks	8	✓	[90]
	Cyclic Scheduling and Control	1	✓	[91]
	Distillation Sep. Sequences	1	✓	[92–95]
	Heat Exchanger Networks	3	✓	[96, 97]
	Metabolic Networks	2	✓	[80, 81]
	Multi-Product Batch Plants	2	✓	[98–100]
	Periodic Scheduling	13	✓	[101, 102]
	Supply Chain Design	2	✓	[103, 104]
	Three-Echelon Supply Chain	8	✓	[105–107]
	Unit Commitment	2	✓	[108, 109]
	Water Distribution Network	8	✓	[110, 111]
	Water Treatment Network	6	✓	[112–115]
	Test Libraries	GLOBALLib	82	
MINLPLib		90	✓	[79, 117]
MacMOOPLib		4		
AMPL Book Lib		7		[118]
Bonmin Test Set		10	✓	[119, 120]
Lit Problems		4		[121]

Like GloMIQO, the MISO framework uses every MILP relaxation solution to multistart a local NLP solver (default CONOPT; alternative SNOPT). The only differences are three strategies to get better quality solutions; the MISO framework:

- disallows starting points at regions of discontinuity (*e.g.*, the framework will perturb  $\hat{x}_1 = -b/a$  away from its relaxation solution if there is a term  $1/(a \cdot x_1 + b)$ );
- freezes variables if their current value is feasible but at a point with an infinite derivative (*e.g.*, if the MILP relaxation sets  $\hat{x}_1 = 0$  and there is a term  $x_1^{0.7}$ , the framework locks the variable value);
- removes redundant RLT equations from the model formulation before starting the NLP solver; redundant equality equations are unnecessarily difficult for local NLP codes [22, pp 56].

#### 4 Definition of the MISO Test Suite

Table 2 outlines the MISO test suite; which are available on the ANTIGONE website (<http://helios.princeton.edu/ANTIGONE/MISOSupp.pdf>). It includes



257 problems with signomial components; test cases with only quadratic and bilinear terms are excluded from the analysis because we want to demonstrate the new MISO framework. Tables in the supplementary material analyze the size of each of the problems.

#### 4.1 Test Suite Definition: 60 `minlp.org` Problems

The `minlp.org` test set is challenging; all 60 problems are less than 5 years old and industrially relevant. The 60 problems are outlined in Table 2 & S1 – S13.

#### 4.2 Test Suite Definition: 82 GLOBAL Lib Problems

The 82 GLOBALLib [79, 116] test cases represent all the sufficiently bounded MISO problems with a non-quadratic component (*i.e.*, at least one solver was able to deterministically infer variable bounds). The models and their sizes are documented in Table S14 (online supplementary material). The MISO framework is additionally validated for the following eight unbounded problems not in the 257 set: `ex4_1_5`; `ex8_1_3`; `ex8_1_4`; `ex8_1_5`; `ex8_3_6`; `ex8_3_10`; `ex8_6_1`; `ship`.

#### 4.3 Test Suite Definition: 100 MINLP Library Problems

The 90 MINLPLib [79, 117] and 10 Bonmin Test Set [119, 120] problems are all bounded signomial test cases with a non-quadratic component; excluded is the unbounded `minlphix`. Tables S15 and S18 in the supplementary material document these problems.

#### 4.4 Test Suite Definition: 19 Other Problems

The remaining set of problems is from a variety of sources. There are 4 test cases from MacMOOPLib (<http://wiki.mcs.anl.gov/leyffer/index.php/MacMOOP>; <http://www.gamsworld.org/performance/macmoop/macmooplib.htm>) documented in Table S16. The 7 AMPL Book Lib Problems (<http://www.ampl.com/EXAMPLES/index.html>; <http://www.gamsworld.org/performance/amplbook/amplbooklib.htm>) are documented in Table S17.

YanJun et al. [121] solve the final 4 test cases analytically and find globally optimal points at singularities (*e.g.*,  $\hat{x} = 0$  when  $1/x$  is one term in a constraint). To eliminate the numerically-unmanageable points, the test problems have been reformulated

so that all variables are bounded below by  $10^{-3}$  rather than by 0 as in their original version. The modeling files are documented in Table S19. The MISO framework is also validated with `transpl2` in the AMPL Book Library and `ytz_1` from Yanjun et al. [121], but these are unbounded problems where time profiles are irrelevant.

## 5 Testing the Framework

The C++ implementation interfaces with CPLEX 12.5 for the MILP relaxations, CONOPT 3 [22] for the local NLP solves, LAPACK [122] for solving systems of equations and Boost [82, 83] for validated interval arithmetic. The code is linked to GAMS via an adaptation of GAMSLinks [40, 123] and is publicly available as a subset of the MINLP solver ANTIGONE [21].

To test the performance of the MISO framework, we compared the 257 test problems outlined in Table 2 against several state-of-the-art global optimization solvers: BARON 12.3.3 [38], Couenne 0.4 [39], LINDO 8.0 [41, 42], and SCIP 3.0 [47]. All of the solvers are subject to active development; we have used the latest version of each solver available to us as of July 2013 (GAMS 24.1.2).

We ran each of the  $5 \cdot 257 = 1285$  computational experiments under 4 termination criteria: (1) a relative optimality gap  $\varepsilon = \frac{UB-LB}{|LB|} \leq 1 \times 10^{-6} = 1 \times 10^{-4}\%$ ; (2) an absolute optimality gap  $\varepsilon = UB - LB = 1 \times 10^{-6}$ ; (3) a 7200 s time limit; (4) an iteration limit of  $9 \cdot 10^9$ . No other parameters were altered from default. Following the recommendation of Dolan and Moré [124], each of the 5 solvers addressed a test problem successively so as minimize the effect of fluctuating machine load. We performed our computational studies running 64 bit Linux on an Intel Xeon X5650 2.67 GHz processor with 24 GB RAM.

We considered the 257 test cases outlined in Table 2. After solving each of the 257 test cases using GAMS, a Perl script asserted the feasibility of the solution returned by each solver. Instances where solvers returned infeasible points but claimed they were  $\varepsilon$ -global optimum points, local optimum points, or integer feasible points were automatically relabeled as failures (*fl*). Dashes (–) in columns LB and UB of Tables S20 – S27 in the online supplementary material indicates a benign failure (the solver chose to terminate based on numerical instability or other internal issue and did not return a solution to the user).

Comparisons between the solvers are based on performance profiles [124]:

$$t_{p,s} \equiv \text{Performance metric (e.g., time in seconds) for problem } p \text{ by technique } s \in S$$

$$r_{p,s} \equiv \frac{t_{p,s}}{\min \{t_{p,s'} : s' \in S\}}; s \in S$$

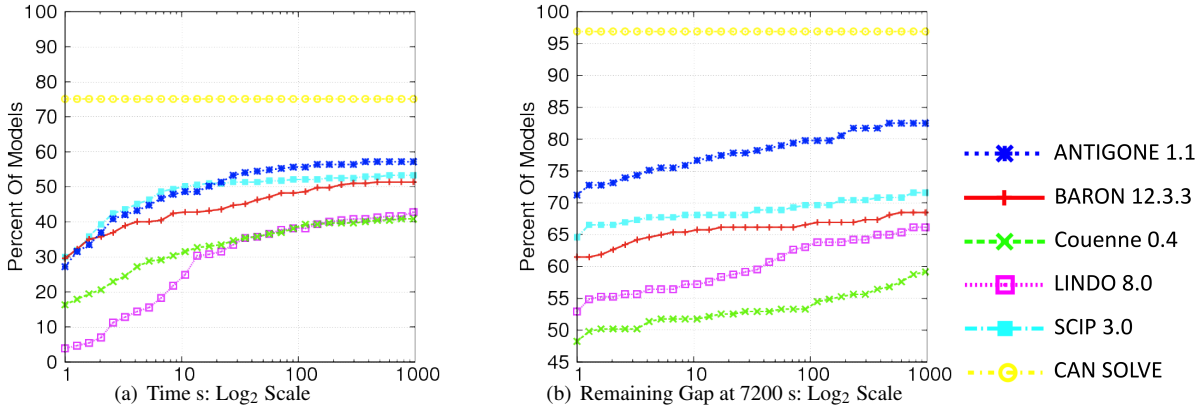


Fig. 2: **257 Complete Test Suite**: Performance Profile (a) compares the CPU time for all 257 test cases; Profile (b) diagrams the percent gap at 7200 s

$$\rho_s(\tau) = \frac{1}{n_p} \text{size} \{p \in P : r_{p,s} \leq \tau\}$$

The logarithmic plots in this paper, which diagram  $\rho_s(\log_{10}(\tau))$  as a function of  $\log_{10}(\tau)$ , were generated using the GAMS Performance Tools with options: `colselect`; `useobjest`; `bnd=1e-4`; `gaptol=1e-4` (PTOOLS 1.1; [www.gamsworld.org/performance/paver/](http://www.gamsworld.org/performance/paver/); accessed 11 March 2013). Each of the Figures 2 – 5 has a subfigure (a) and (b). Subfigures (a) are the output of PTOOLS using **solver reported time** as a resource measurement and the `objest` option to enforce an objective gap of  $10^{-4}$ . Subfigures (b) use the `colselect` command (and switch off `useobjest`); this allows us to compare the solvers with respect to their optimality gap at termination ( $100 \cdot \frac{UB-LB}{|LB|}$ ); this column in the GAMS trace file was created with the same Perl script that asserts feasibility of the solution. Note that the performance profiles in Figures 2 – 5 are *only* representative of problems with well-defined bounds; the MISO framework is further validated with unbounded problems.

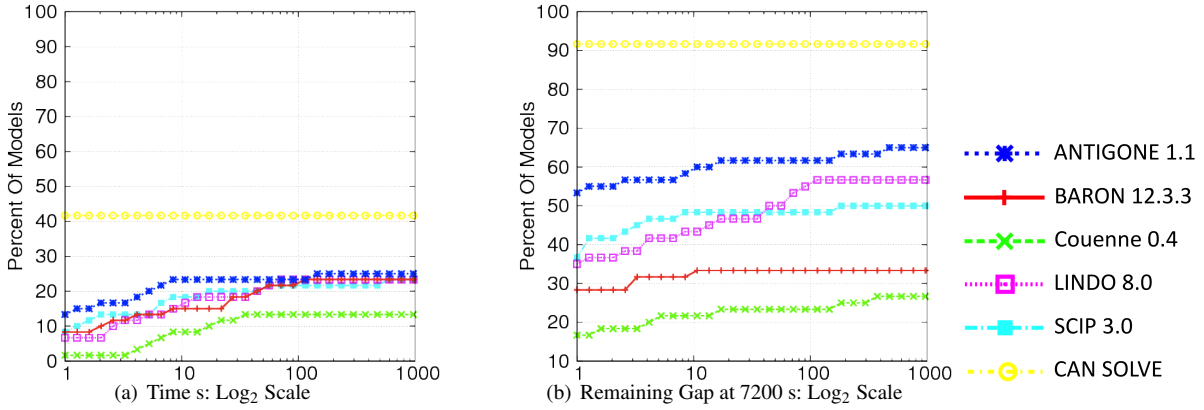
### 5.1 Results of the Computational Study

Tables S20 – S27 give the results of the computational study; Figure 2 summarizes the computational results with a performance profile of the entire test suite. Sections 5.2 – 5.5 detail the individual test sets.

Table 3: `minlp.org`: Results of `ruiz_concbased` [115]

	Time (s)	Gap	LB	UB
MISO Framework	7	1.000e-04	-3.483e+05	-3.483e+05
BARON 12.3.3	168	1.000e-04	-3.483e+05	-3.483e+05
LINDO 8.0	762	1.000e-04	-3.483e+05	-3.483e+05
SCIP 3.0	9	1.000e-04	-3.483e+05	-3.483e+05

Gap  $\equiv 100 \cdot \left( \frac{UB-LB}{|LB|} \right)$ ; Termination Criteria: Gap =  $1 \times 10^{-4}\%$  or Time = 7200 s

Fig. 3: **60 `minlp.org` Test Cases**: Performance Profile (a) compares the CPU time for 60 problems from `minlp.org`; Profile (b) diagrams the percent gap at 7200 s

## 5.2 Computational Results: 60 `minlp.org` Problems

The MISO framework is particularly strong on the 60 `minlp.org` test cases; results of the computational study are given in Tables S20 – S21 and Figure 3. Table 3 uses `ruiz_concbased` as a representative example from this test set; this is a water network problem with a concave cost function that the MISO framework solves in approximately 7 seconds. After the reformulation stage, `ruiz_concbased` has 115 bilinear and 5 univariate concave terms. The strong performance on this test problem is largely an outcome of the MISO framework reducing to GloMIQO in the case of quadratic problems; the complexity of this problem is in the 115 bilinear terms. Couenne 0.4 is not included in Table 3; it fails the post-processing feasibility tests. Another set of examples where the MISO framework does very well is the MTG and STG periodic scheduling problems [101, 102]; these are large, poorly-scaled test cases (size up to: 310 continuous variables; 384 discrete variables; 388 constraints;

Table 4: MINLPLib: Results of `cecil_13` [79, 117]

	Time (s)	% Gap	LB	UB
MISO Framework	209	1.000e-04	-1.157e+05	-1.157e+05
BARON 12.3.3	–	9.219e-02	-1.158e+05	-1.157e+05
Couenne 0.4	–	5.254e+00	-1.216e+05	-1.156e+05
LINDO 8.0	–	1.475e+00	-1.174e+05	-1.157e+05
SCIP 3.0	–	1.625e+00	-1.175e+05	-1.156e+05

Gap  $\equiv 100 \cdot \left( \frac{UB-LB}{|LB|} \right)$ ; Termination Criteria: Gap =  $1 \times 10^{-4}\%$  or Time = 7200 s

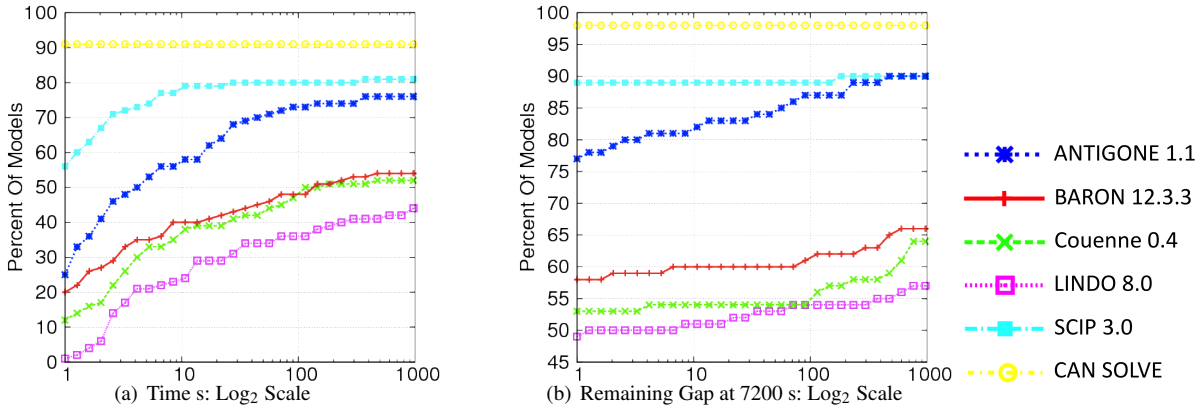


Fig. 4: **100 MINLP Library Cases:** Performance Profile (a) compares the CPU time for 100 problems from MINLPLib and the Bonmin Test Set; Profile (b) diagrams the percent gap at 7200 s

166 nonlinear terms; variables ranging from order 1 to order  $10^6$  in many equations). Of the solvers, only the MISO framework is able to reliably generate high-quality solutions for MTG and STG.

### 5.3 Computational Results: 100 MINLP Library Problems

Computational results for the 100 MINLP library problems from MINLPLib and the Bonmin test set are in Tables S22 – S23 and Figure 4. Table 4 presents `cecil_13` as an example from MINLPLib; the MISO framework is able to close the gap in approximately 3 minutes while the remaining solvers do not converge in 2 hours; this is a result of the MISO framework finding 34 nonlinear term / equality equation RLT products that it directly adds to the model formulation.

Table 5: GLOBALLib: Results of  $\text{ex7\_2\_3}$  [79, 116]

	Time (s)	% Gap	LB	UB
MISO Framework	2906	1.000e-04	7.049e+03	7.049e+03
BARON 12.3.3	–	2.357e+02	2.100e+03	7.049e+03
Couenne 0.4	–	2.354e+02	2.102e+03	7.049e+03
LINDO 8.0	–	2.357e+02	2.100e+03	7.049e+03
SCIP 3.0	–	2.357e+02	2.100e+03	7.049e+03

Gap  $\equiv 100 \cdot \left( \frac{UB-LB}{|LB|} \right)$ ; Termination Criteria: Gap =  $1 \times 10^{-4}\%$  or Time = 7200 s

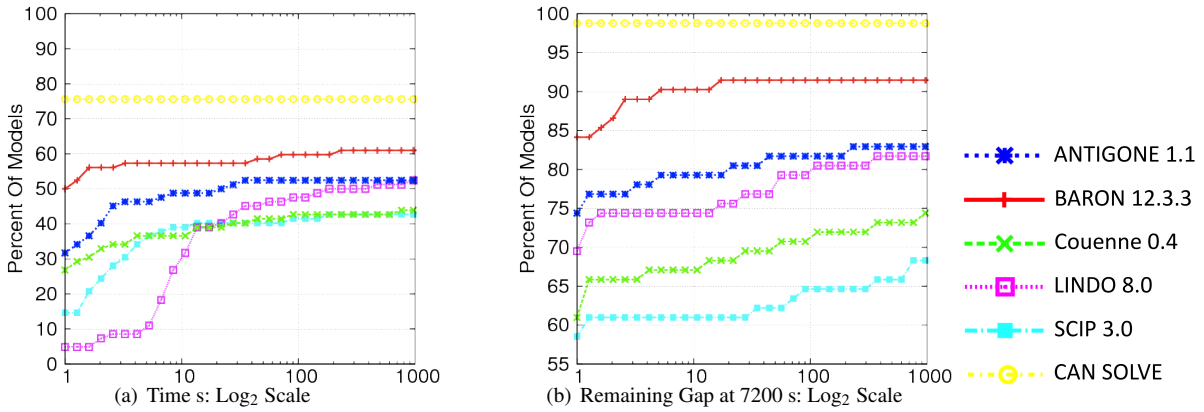


Fig. 5: **82 GLOBALLib Cases:** Performance Profile (a) compares the CPU time for 82 problems from GLOBALLib; Profile (b) diagrams the percent gap at 7200 s

#### 5.4 Computational Results: 82 GLOBAL Lib Problems

Tables S24 and S25 and Figure 5 give results for the 82 GLOBAL Lib Problems. For test case  $\text{ex7\_2\_3}$ , Table 5 gives the solution bounds for each of the five solvers after two hours; observe that the MISO framework has closed the gap using the 40 RLT equations interlinking the terms. All other solvers terminate at a gap which is 6 orders of magnitude larger.

#### 5.5 Computational Results: 19 Other Problems

Computational results for the final 19 problems are in Tables S26 – S27. The strong performance of the 5 `steelpl` cases in the AMPL Library is a result of automatically reformulating  $\sqrt{x_i^2}$  to  $|x_i|$ .

## 6 Conclusions

This manuscript introduces a framework for globally optimizing MISO that builds on the core GloMIQO strategies of (1) reformulating user input, (2) detecting special mathematical structure, and (3) globally optimizing via branch-and-cut. New advances include: reformulating towards special functional forms; adding nonconvex terms to interlink expressions; integrating a dynamic implementation of RLT into the branch-and-cut tree; designing underestimators that specialize themselves according to the spot in the branch-and-bound tree. Computational results show that the framework is a powerful tool for globally optimizing MISO; the framework compares favorably to state-of-the-art solvers. The strong performance is especially noticeable in the `minlp.org` test set.

## Acknowledgements

C.A.F. is thankful for support from the National Science Foundation (CBET – 0827907). This material is based upon work supported by the National Science Foundation Graduate Research Fellowship to R.M. under Grant No. DGE-0646086 and a Royal Academy of Engineering Research Fellowship to R.M.

## References

1. Misener, R., Floudas, C.A.: GloMIQO: Global Mixed-Integer Quadratic Optimizer. *J. Glob. Optim.* **57**, 3–50 (2013)
2. Misener, R., Floudas, C.A.: Global optimization of mixed-integer quadratically-constrained quadratic programs (MIQCQP) through piecewise-linear and edge-concave relaxations. *Math. Program. B* **136**, 155–182 (2012)
3. Misener, R., Smadbeck, J.B., Floudas, C.A.: Dynamically-generated cutting planes for mixed-integer quadratically-constrained quadratic programs and their incorporation into GloMIQO 2.0. *Optim. Method. Softw.* (2014). In Press; DOI: 10.1080/10556788.2014.916287
4. Duffin, R.J., Peterson, E.L.: Duality theory for geometric programming. *SIAM J. Appl. Math.* **14**(6), 1307–1349 (1966)
5. Duffin, R.J., Peterson, E.L.: Geometric programming with signomials. *J. Optim. Theory Appl.* **11**(1), 3–35 (1973)
6. Blau, G.E., Wilde, D.J.: Generalized polynomial programming. *The Canadian Journal of Chemical Engineering* **47**(4), 317–326 (1969)
7. Passy, U., Wilde, D.J.: Generalized polynomial optimization. *SIAM Journal on Applied Mathematics* **15**(5), 1344–1356 (1967)
8. Maranas, C.D., Floudas, C.A.: Global optimization in generalized geometric programming. *Comput. Chem. Eng.* **21**(4), 351 – 369 (1997)
9. Floudas, C.A., Ciric, A.R.: Strategies for overcoming uncertainties in heat exchanger network synthesis. *Comput. Chem. Eng.* **13**(10), 1133 – 1152 (1989)

10. Quesada, I., Grossmann, I.E.: Global optimization algorithm for heat exchanger networks. *Ind. Eng. Chem. Res.* **32**(3), 487–499 (1993)
11. Kallrath, J.: Mixed-integer nonlinear programming applications. In: T.A. Ciri-ani (ed.) *Operational Research in Industry*, Ichor Business Books. Ichor Business Books (1999)
12. Maranas, C.D., Floudas, C.A.: A global optimization approach for Lennard-Jones microclusters. *J. Chem. Phys.* **97**(10), 7667–7678 (1992)
13. Maranas, C.D., Floudas, C.A.: Global optimization for molecular conformation problems. *Annals of Oper. Res.* **42**, 85–117 (1993)
14. Aggarwal, A., Floudas, C.A.: Synthesis of heat integrated nonsharp distillation sequences. *Comput. Chem. Eng.* **16**(2), 89 – 108 (1992)
15. Kokossis, A.C., Floudas, C.A.: Optimization of complex reactor networks–II. Nonisothermal operation. *Chem. Eng. Sci.* **49**(7), 1037 – 1051 (1994)
16. Kallrath, J.: Exact computation of global minima of a nonconvex portfolio optimization problem. In: C.A. Floudas, P.M. Pardalos (eds.) *Frontiers in Global Optimization*. Kluwer Academic Publishers (2003)
17. Lin, X., Floudas, C.A., Kallrath, J.: Global solution approach for a nonconvex MINLP problem in product portfolio optimization. *J. Glob. Optim.* **32**, 417–431 (2005)
18. Hatzimanikatis, V., Floudas, C.A., Bailey, J.E.: Analysis and design of metabolic reaction networks via mixed-integer linear optimization. *AIChE J.* **42**(5), 1277–1292 (1996)
19. Hatzimanikatis, V., Floudas, C.A., Bailey, J.E.: Optimization of regulatory architectures in metabolic reaction networks. *Biotechnology and Bioengineering* **52**(4), 485–500 (1996)
20. Baliban, R.C., Elia, J.A., Floudas, C.A.: Optimization framework for the simultaneous process synthesis, heat and power integration of a thermochemical hybrid biomass, coal, and natural gas facility. *Comput. Chem. Eng.* **35**(9), 1647 – 1690 (2011)
21. Misener, R., Floudas, C.A.: ANTIGONE: Algorithms for coNTinuous Integer Global Optimization of Nonlinear Expressions. *J. Glob. Optim.* **59**, 503–526 (2014)
22. Drud, A.: CONOPT. <http://www.gams.com/dd/docs/solvers/conopt.pdf> (2012). Version 3
23. Floudas, C.A., Gounaris, C.E.: A review of recent advances in global optimization. *J. Glob. Optim.* **45**(1), 3 – 38 (2009)
24. Floudas, C.A., Akrotirianakis, I.G., Caratzoulas, S., Meyer, C.A., Kallrath, J.: Global optimization in the 21st century: Advances and challenges. *Comput. Chem. Eng.* **29**, 1185 – 1202 (2005)
25. Misener, R., Floudas, C.A.: Global optimization of mixed-integer models with quadratic and signomial functions: A review. *Applied and Computational Mathematics* **11**(3), 317–336 (2012)
26. Floudas, C.A.: *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. Oxford University Press, New York, NY (1995)
27. Floudas, C.A.: *Deterministic Global Optimization : Theory, Methods and Applications*. *Nonconvex Optimization and Its Applications*. Kluwer Academic



- Publishers, Dordrecht, Netherlands (2000)
28. Horst, R., Hoang, T.: *Global optimization: Deterministic approaches*. Springer (1996)
  29. Nowak, I.: *Relaxation and decomposition methods for mixed integer nonlinear programming*. International series of numerical mathematics. Birkhäuser (2005)
  30. Serali, H.D., Adams, W.P.: *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems. Nonconvex Optimization and Its Applications*. Kluwer Academic Publishers, Dordrecht, Netherlands (1999)
  31. Tawarmalani, M., Sahinidis, N.V.: *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Applications, Software, and Applications. Nonconvex Optimization and Its Applications*. Kluwer Academic Publishers, Norwell, MA, USA (2002)
  32. Bussieck, M.R., Vigerske, S.: MINLP solver software. In: J.J. Cochran, L.A. Cox, P. Keskinocak, J.P. Kharoufeh, J.C. Smith (eds.) *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc. (2010)
  33. Adjiman, C.S., Dallwig, S., Floudas, C.A., Neumaier, A.: A global optimization method,  $\alpha$ BB, for general twice differentiable NLPs-I. Theoretical advances. *Comput. Chem. Eng.* **22**, 1137 – 1158 (1998a)
  34. Adjiman, C.S., Androulakis, I.P., Floudas, C.A.: A global optimization method,  $\alpha$ BB, for general twice differentiable NLPs-II. Implementation and computational results. *Comput. Chem. Eng.* **22**, 1159 – 1179 (1998b)
  35. Androulakis, I.P., Maranas, C.D., Floudas, C.A.:  $\alpha$ BB: A global optimization method for general constrained nonconvex problems. *J. Glob. Optim.* **7**, 337 – 363 (1995)
  36. Maranas, C.D., Floudas, C.A.: Finding all solutions of nonlinearly constrained systems of equations. *J. Glob. Optim.* **7**(2), 143–182 (1995)
  37. Tawarmalani, M., Sahinidis, N.V.: Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Math. Program.* **99**, 563–591 (2004)
  38. Tawarmalani, M., Sahinidis, N.V.: A polyhedral branch-and-cut approach to global optimization. *Math. Program.* **103**, 225–249 (2005)
  39. Belotti, P., Lee, J., Liberti, L., Margot, F., Wächter, A.: Branching and bounds tightening techniques for non-convex MINLP. *Optim. Method. Softw.* **24**(4-5), 597–634 (2009)
  40. Lougee-Heimer, R.: The Common Optimization INterface for Operations Res.: Promoting open-source software in the operations research community. *IBM Journal of Research and Development* **47**(1), 57 –66 (2003)
  41. Gau, C.Y., Schrage, L.E.: Implementation and testing of a branch-and-bound based method for deterministic global optimization: Operations research applications. In: C.A. Floudas, P.M. Pardalos (eds.) *Frontiers in Global Optimization*, pp. 145–164. Kluwer Academic Publishers (2003)
  42. Lin, Y., Schrage, L.: The global solver in the LINDO API. *Optim. Methods Software* **24**(4-5), 657–668 (2009)

43. Achterberg, T.: SCIP: Solving constraint integer programs. *Math. Program. Comput.* **1**(1), 1–41 (2009)
44. Achterberg, T., Berthold, T., Koch, T., Wolter, K.: Constraint integer programming: a new approach to integrate CP and MIP. In: *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*. CPAIOR (2008)
45. Berthold, T., Heinz, S., Vigerske, S.: Extending a CIP framework to solve MIQCPs. In: J. Lee, S. Leyffer (eds.) *Mixed Integer Nonlinear Programming, The IMA Volumes in Mathematics and its Applications*, vol. 154, pp. 427–444. Springer New York (2012)
46. Berthold, T., Gleixner, A.M., Heinz, S., Vigerske, S.: Analyzing the computational impact of MIQCP solver components. *Numerical Algebra, Control and Optimization* **2**(4), 739 – 748 (2012)
47. Vigerske, S.: Decomposition in multistage stochastic programming and a constraint integer programming approach to mixed-integer nonlinear programming. PhD in Mathematics, Humboldt-University Berlin (2012)
48. Lundell, A., Westerlund, T.: Convex underestimation strategies for signomial functions. *Optim. Methods Software* **24**(4-5), 505–522 (2009)
49. Lundell, A., Westerlund, T.: Global optimization of mixed-integer signomial programming problems. In: J. Lee, S. Leyffer (eds.) *Mixed Integer Nonlinear Programming, The IMA Volumes in Mathematics and its Applications*, vol. 154, pp. 349–369. Springer New York (2012)
50. Lundell, A., Westerlund, J., Westerlund, T.: Some transformation techniques with applications in global optimization. *J. Glob. Optim.* **43**, 391–405 (2009)
51. Lundell, A., Skjäl, A., Westerlund, T.: A reformulation framework for global optimization. *J. Glob. Optim.* (2012). 10.1007/s10898-012-9877-4
52. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs: Part 1-convex underestimating problems. *Math. Program.* **10**(1), 147 – 175 (1976)
53. Smith, E.M.B., Pantelides, C.C.: A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex MINLPs. *Comput. Chem. Eng.* **23**(4 - 5), 457 – 478 (1999)
54. Tawarmalani, M., Sahinidis, N.V.: Semidefinite relaxations of fractional programs via novel convexification techniques. *J. Glob. Optim.* **20**, 133–154 (2001)
55. Meyer, C.A., Floudas, C.A.: Trilinear monomials with positive or negative domains: Facets of the convex and concave envelopes. In: C.A. Floudas, P.M. Pardalos (eds.) *Frontiers in Global Optimization*, pp. 327–352. Kluwer Academic Publishers (2003)
56. Meyer, C.A., Floudas, C.A.: Trilinear monomials with mixed sign domains: Facets of the convex and concave envelopes. *J. Glob. Optim.* **29**(2), 125–155 (2004)
57. Cafieri, S., Lee, J., Liberti, L.: On convex relaxations of quadrilinear terms. *J. Glob. Optim.* **47**, 661–685 (2010)
58. Liberti, L., Pantelides, C.C.: Convex envelopes of monomials of odd degree. *J. Glob. Optim.* **25**, 157–168 (2003)

59. Meyer, C.A., Floudas, C.A.: Convex envelopes for edge-concave functions. *Math. Program.* **103**(2), 207–224 (2005)
60. Tardella, F.: On a class of functions attaining their maximum at the vertices of a polyhedron. *Discret. Appl. Math.* **22**, 191–195 (1988/89)
61. Tardella, F.: On the existence of polyhedral convex envelopes. In: C.A. Floudas, P.M. Pardalos (eds.) *Frontiers in Global Optimization*, pp. 563–573. Kluwer Academic Publishers (2003)
62. Tardella, F.: Existence and sum decomposition of vertex polyhedral convex envelopes. *Optim. Lett.* **2**, 363–375 (2008)
63. Tawarmalani, M., Richard, J.P.P., Xiong, C.: Explicit convex and concave envelopes through polyhedral subdivisions. *Math. Program.* **138**(1-2), 531–577 (2013)
64. Gounaris, C.E., Floudas, C.A.: Convexity of products of univariate functions and convexification transformations for geometric programming. *J. Optim. Theory Appl.* **138**, 407–427 (2008)
65. Khajavirad, A., Sahinidis, N.V.: Convex envelopes generated from finitely many compact convex sets. *Math. Program.* **137**(1-2), 371–408 (2013)
66. Khajavirad, A., Sahinidis, N.V.: Convex envelopes of products of convex and component-wise concave functions. *J. Glob. Optim.* **52**, 391–409 (2012)
67. Shikhman, V., Stein, O.: On jet-convex functions and their tensor products. *Optim.* **61**(6), 717–731 (2012)
68. Bompadre, A., Mitsos, A.: Convergence rate of McCormick relaxations. *J. Glob. Optim.* **52**(1), 1–28 (2011)
69. Gatzke, E.P., Tolsma, J.E., Barton, P.I.: Construction of convex relaxations using automated code generation techniques. *Optim. Eng.* **3**, 305–326 (2002)
70. Tawarmalani, M., Ahmed, S., Sahinidis, N.V.: Product disaggregation in global optimization and relaxations of rational programs. *Optim. Eng.* **3**, 281–303 (2002)
71. Audet, C., Hansen, P., Jaumard, B., Savard, G.: A branch and cut algorithm for nonconvex quadratically constrained quadratic programming. *Math. Program.* **87**(1), 131 – 152 (2000)
72. Liberti, L., Pantelides, C.C.: An exact reformulation algorithm for large nonconvex NLPs involving bilinear terms. *J. Glob. Optim.* **36**(2), 161–189 (2006)
73. Sherali, H., Dalkiran, E., Liberti, L.: Reduced RLT representations for nonconvex polynomial programming problems. *J. Glob. Optim.* **52**(3), 447–469 (2012)
74. Sherali, H.D., Alameddine, A.: A new reformulation-linearization technique for bilinear programming problems. *J. Glob. Optim.* **2**, 379 – 410 (1992)
75. Sherali, H.D., Tuncbilek, C.H.: A reformulation-convexification approach for solving nonconvex quadratic-programming problems. *J. Glob. Optim.* **1995**(7), 1 (1-31)
76. Sherali, H.D., Tuncbilek, C.H.: New reformulation linearization/convexification relaxations for univariate and multivariate polynomial programming problems. *Oper. Res. Letters* **21**(1), 1 – 9 (1997)
77. Mitsos, A., Chachuat, B., Barton, P.I.: McCormick-based relaxations of algorithms. *SIAM J. Optim.* **20**(2), 573–601 (2009)

78. Goldstein, A.A., Price, J.F.: On descent from local minima. *Mathematics of Computation* **25**(115), 569–574 (1971)
79. Floudas, C.A., Pardalos, P.M., Adjiman, C.S., Esposito, W.R., Günius, Z.H., Harding, S.T., Klepeis, J.L., Meyer, C.A., Schweiger, C.A.: *Handbook of Test Problems in Local and Global Optimization*. Kluwer Academic Publishers (1999)
80. Guillén-Gosálbez, G., Pozo, C.: Optimization of metabolic networks in biotechnology (2010). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=81](http://www.minlp.org/library/problem/index.php?i=81)
81. Guillén-Gosálbez, G., Sorribas, A.: Identifying quantitative operation principles in metabolic pathways: a systematic method for searching feasible enzyme activity patterns leading to cellular adaptive responses. *BMC Bioinformatics* **10**(386) (2009)
82. Brönnimann, H., Melquiond, G., Pion, S.: The Boost interval arithmetic library. In: *Proceedings of the 5th Conference on Real Numbers and Computers*, pp. 65–80. Lyon, France (2003)
83. Brönnimann, H., Melquiond, G., Pion, S.: The design of the Boost interval arithmetic library. *Theoretical Computer Science* **351**, 111–118 (2006)
84. Rikun, A.D.: A convex envelope formula for multilinear functions. *J. Glob. Optim.* **10**, 425 – 437 (1997)
85. Belotti, P., Cafieri, S., Lee, J., Liberti, L., Miller, A.: On the composition of convex envelopes for quadrilinear terms. In: A. Chinchuluun, P.M. Pardalos, R. Enkhbat, E.N. Pistikopoulos (eds.) *Optimization, Simulation, and Control, Springer Optimization and Its Applications*, vol. 76, pp. 1–16. Springer New York (2013)
86. Liberti, L., Lavor, C., Maculan, N., Nascimento, M.: Reformulation in mathematical programming: an application to quantum chemistry. *Discrete Applied Mathematics* **157**(6), 1309–1318 (2009)
87. Achterberg, T., Koch, T., Martin, A.: Branching rules revisited. *Oper. Res. Lett.* **33**(1), 42–54 (2005)
88. Chang, Y.J., Sahinidis, N.V.: Global optimization in stabilizing controller design. *J. Glob. Optim.* **38**, 509–526 (2007)
89. Chang, Y.J., Sahinidis, N.V.: Stabilizing controller design and the belgian chocolate problem (2009). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=57](http://www.minlp.org/library/problem/index.php?i=57)
90. Gopalakrishnan, A., Biegler, L.: MINLP and MPCC formulations for the cascading tanks problem (2011). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=140](http://www.minlp.org/library/problem/index.php?i=140)
91. Flores-Tlacuahuac, A., Grossmann, I.E.: Simultaneous cyclic scheduling and control of a multiproduct cstr (2009). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=71](http://www.minlp.org/library/problem/index.php?i=71)
92. Caballero, J.A., Grossmann, I.E.: Generalized disjunctive programming model for the optimal synthesis of thermally linked distillation columns. *Ind. Eng. Chem. Res.* **40**(10), 2260–2274 (2001)
93. Caballero, J.A., Grossmann, I.E.: Design of distillation sequences: From conventional to fully thermally coupled distillation systems. *Comput. Chem. Eng.*

- 28**(11), 2307 – 2329 (2004)
94. Caballero, J.A., Grossmann, I.E.: Structural considerations and modeling in the synthesis of heat integrated thermally coupled distillation sequences. *Ind. Eng. Chem. Res.* **45**(25), 8454–8474 (2006)
  95. Caballero, J.A., Grossmann, I.E.: Optimal separation sequences based on distillation: From conventional to fully thermally coupled systems (2009). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=69](http://www.minlp.org/library/problem/index.php?i=69)
  96. Escobar, M., Grossmann, I.E.: Mixed-integer nonlinear programming models for optimal simultaneous synthesis of heat exchangers network (2010). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=93](http://www.minlp.org/library/problem/index.php?i=93)
  97. Yee, T.F., Grossmann, I.E.: Simultaneous optimization models for heat integrationII. Heat exchanger network synthesis. *Comput. Chem. Eng.* **14**(10), 1165 – 1184 (1990)
  98. Grossmann, I.E., Sargent, R.W.H.: Optimum design of multipurpose chemical plants. *Ind. Eng. Chem. Process Des. Dev.* **18**(2), 343–348 (1979)
  99. Kocis, G.R., Grossmann, I.E.: Global optimization of nonconvex mixed-integer nonlinear programming (MINLP) problems in process synthesis. *Ind. Eng. Chem. Res.* **27**(8), 1407–1421 (1988)
  100. You, F., Grossmann, I.E.: Mixed-integer nonlinear programming models for the optimal design of multi-product batch plant (2009). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=48](http://www.minlp.org/library/problem/index.php?i=48)
  101. Castro, P., Novais, A.: Optimal periodic scheduling of multistage continuous plants with single and multiple time grid formulations. *Ind. Eng. Chem. Res.* **46**(11), 3669–3683 (2007)
  102. Castro, P., Novais, A.: Periodic scheduling of continuous multiproduct plants (2009). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=34](http://www.minlp.org/library/problem/index.php?i=34)
  103. You, F., Grossmann, I.E.: Mixed-integer nonlinear programming models and algorithms for large-scale supply chain design with stochastic inventory management. *Ind. Eng. Chem. Res.* **47**(20), 7802–7817 (2008)
  104. You, F., Grossmann, I.E.: Mixed-integer nonlinear programming models and algorithms for supply chain design with stochastic inventory management (2009). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=30](http://www.minlp.org/library/problem/index.php?i=30)
  105. Nyberg, A., Grossmann, I.E., Westerlund, T.: The optimal design of a three-echelon supply chain with inventories under uncertainty (2012). Available from CyberInfrastructure for MINLP [[www.minlp.org](http://www.minlp.org), a collaboration of Carnegie Mellon University and IBM Research] at: [www.minlp.org/library/problem/index.php?i=157](http://www.minlp.org/library/problem/index.php?i=157)
  106. Nyberg, A., Grossmann, I.E., Westerlund, T.: An efficient reformulation of the multiechelon stochastic inventory system with uncertain demands. *AIChE J.* **59**(1), 23–28 (2013)

107. You, F., Grossmann, I.E.: Integrated multi-echelon supply chain design with inventories under uncertainty: Minlp models, computational strategies. *AICHE Journal* **56**(2), 419–440 (2010)
108. Niknam, T., Khodaei, A., Fallahi, F.: A new decomposition approach for the thermal unit commitment problem. *Applied Energy* **86**(9), 1667 – 1674 (2009)
109. Zondervan, E., Grossmann, I.E.: A deterministic security constrained unit commitment model (2009). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=41](http://www.minlp.org/library/problem/index.php?i=41)
110. Bragalli, C., D'Ambrosio, C., Lee, J., Lodi, A., Toth, P.: On the optimal design of water distribution networks: a practical MINLP approach. *Optim. Eng.* **13**, 219–246 (2012)
111. D'Ambrosio, C., Bragalli, C., Lee, J., Lodi, A., Toth, P.: Optimal design of water distribution networks (2011). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=134](http://www.minlp.org/library/problem/index.php?i=134)
112. Ahmetović, E., Grossmann, I.E.: Integrated process water networks design problem (2010). Available from CyberInfrastructure for MINLP at: [www.minlp.org/library/problem/index.php?i=101](http://www.minlp.org/library/problem/index.php?i=101)
113. Ahmetović, E., Grossmann, I.E.: Global superstructure optimization for the design of integrated process water networks. *AICHE J.* **57**(2), 434–457 (2011)
114. Karuppiah, R., Grossmann, I.E.: Global optimization for the synthesis of integrated water systems in chemical processes. *Comput. Chem. Eng.* **30**, 650 – 673 (2006)
115. Ruiz, J.P., Grossmann, I.E.: Water treatment network design (2009). Available from CyberInfrastructure for MINLP [[www.minlp.org](http://www.minlp.org), a collaboration of Carnegie Mellon University and IBM Research] at: [www.minlp.org/library/problem/index.php?i=24](http://www.minlp.org/library/problem/index.php?i=24)
116. Meeraus, A.: GLOBALLib. <http://www.gamsworld.org/global/globallib.htm>
117. Bussieck, M.R., Drud, A.S., Meeraus, A.: MINLPLib - a collection of test models for mixed-integer nonlinear programming. *INFORMS J. Comput* **15**(1) (2003)
118. Fourer, R., Gay, D.M., Kernighan, B.W.: The AMPL Book. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press, Brooks/Cole Publishing Company (2002)
119. Bonami, P., Biegler, L.T., Conn, A.R., Cornuéjols, G., Grossmann, I.E., Laird, C.D., Lee, J., Lodi, A., Margot, F., Sawaya, N., Wächter, A.: An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization* **5**(2), 186 – 204 (2008)
120. Sawaya, N.W.: Reformulations, relaxations and cutting planes for generalized disjunctive programming. PhD in Chemical Engineering, Carnegie Mellon University (2006)
121. Yanjun, W., Tao, L., Zhian, L.: A general algorithm for solving generalized geometric programming with nonpositive degree of difficulty. *Computational Optimization and Applications* **44**, 139–158 (2009)
122. Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Croz, J.D., Greenbaum, A., Hammarling, S., McKenney, A., Sorensen, D.: LA-

PACK Users' Guide, third edn. Society for Industrial and Applied Mathematics (1999)

123. Vigerske, S.: COIN-OR/GAMSLinks. <https://projects.coin-or.org/GAMSLinks/> (2011)

124. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. *Math. Program.* **91**, 201–213 (2002)

## Appendix A Reformulation Linearization Technique (RLT) Equations

The equations in this section are presented in factored form for clarity, but the MISO framework addresses them in expanded form. The RLT equations are updated throughout the branch-and-cut tree; any upper bound on a variable or a nonlinear term represents the upper bound at the current tree node.

**Equation/Variable** products may generate as many as four RLT equations (an equation with only one inequality bound produces two RLT equations; equality equations produce exactly one RLT equation):

$$\begin{aligned} & \left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^U \right) \cdot (x_i - x_i^U) \geq 0 \\ -1 \cdot & \left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^U \right) \cdot (x_i - x_i^L) \geq 0 \\ -1 \cdot & \left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^L \right) \cdot (x_i - x_i^U) \geq 0 \\ & \left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^L \right) \cdot (x_i - x_i^L) \geq 0 \\ & \forall m \in \{1, \dots, M\}; i \in C \times B \times I \end{aligned}$$

**Nonlinear Term/Variable** products introduce four RLT equations;  $s \in \{1, \dots, S\}$  denotes a list of the quadratic, bilinear, and signomial terms:

$$\begin{aligned} & (f_s(x) - f_s(x)^{UP}) \cdot (x_i - x_i^U) \geq 0 \quad \forall s \in \{1, \dots, S\}; i \in C \times B \times I \\ -1 \cdot & (f_s(x) - f_s(x)^{UP}) \cdot (x_i - x_i^L) \geq 0 \quad \forall s \in \{1, \dots, S\}; i \in C \times B \times I \\ -1 \cdot & (f_s(x) - f_s(x)^{LO}) \cdot (x_i - x_i^U) \geq 0 \quad \forall s \in \{1, \dots, S\}; i \in C \times B \times I \\ & (f_s(x) - f_s(x)^{LO}) \cdot (x_i - x_i^L) \geq 0 \quad \forall s \in \{1, \dots, S\}; i \in C \times B \times I \end{aligned}$$

**Equation/Nonlinear Term** products may generate up to four RLT equations;  $s \in \{1, \dots, S\}$  represents a list of the quadratic, bilinear, and signomial terms:

$$\begin{aligned} & \left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^U \right) \cdot (f_s(x) - f_s(x)^{UP}) \geq 0 \\ -1 \cdot & \left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^U \right) \cdot (f_s(x) - f_s(x)^{LO}) \geq 0 \\ -1 \cdot & \left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^L \right) \cdot (f_s(x) - f_s(x)^{UP}) \geq 0 \\ & \left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^L \right) \cdot (f_s(x) - f_s(x)^{LO}) \geq 0 \\ & \forall m \in \{1, \dots, M\}; s \in \{1, \dots, S\} \end{aligned}$$

**Nonlinear Term/Nonlinear Term** products result in up to four RLT equations:

$$\begin{aligned} (f_s(x) - f_s(x)^{\text{UP}}) \cdot (f_{s'}(x) - f_{s'}(x)^{\text{UP}}) &\geq 0 \quad \forall s, s' \in \{1, \dots, S\} \\ -1 \cdot (f_s(x) - f_s(x)^{\text{UP}}) \cdot (f_{s'}(x) - f_{s'}(x)^{\text{LO}}) &\geq 0 \quad \forall s, s' \in \{1, \dots, S\} \\ -1 \cdot (f_s(x) - f_s(x)^{\text{LO}}) \cdot (f_{s'}(x) - f_{s'}(x)^{\text{UP}}) &\geq 0 \quad \forall s, s' \in \{1, \dots, S\} \\ (f_s(x) - f_s(x)^{\text{LO}}) \cdot (f_{s'}(x) - f_{s'}(x)^{\text{LO}}) &\geq 0 \quad \forall s, s' \in \{1, \dots, S\} \end{aligned}$$

**Equation/Equation** products may add up to four RLT equations:

$$\begin{aligned} &\left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^{\text{U}} \right) \cdot \left( \sum_{s=1}^{S_{m'}} c_{s_{m'}} \cdot f_{s_{m'}}(x) + x^T \cdot Q_{m'} \cdot x + a_{m'} \cdot x - b_{m'}^{\text{UP}} \right) \geq 0 \\ -1 \cdot &\left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^{\text{U}} \right) \cdot \left( \sum_{s=1}^{S_{m'}} c_{s_{m'}} \cdot f_{s_{m'}}(x) + x^T \cdot Q_{m'} \cdot x + a_{m'} \cdot x - b_{m'}^{\text{LO}} \right) \geq 0 \\ -1 \cdot &\left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^{\text{L}} \right) \cdot \left( \sum_{s=1}^{S_{m'}} c_{s_{m'}} \cdot f_{s_{m'}}(x) + x^T \cdot Q_{m'} \cdot x + a_{m'} \cdot x - b_{m'}^{\text{UP}} \right) \geq 0 \\ &\left( \sum_{s=1}^{S_m} c_{m,s} \cdot f_{m,s}(x) + x^T \cdot Q_m \cdot x + a_m \cdot x - b_m^{\text{L}} \right) \cdot \left( \sum_{s=1}^{S_{m'}} c_{s_{m'}} \cdot f_{s_{m'}}(x) + x^T \cdot Q_{m'} \cdot x + a_{m'} \cdot x - b_{m'}^{\text{LO}} \right) \geq 0 \\ &\quad \forall m, m' \in \{1, \dots, M\} \end{aligned}$$

## Appendix B Recognizing Convexity and Concavity

The MISO framework detects convexity in product  $x_1^a \cdot x_2^b$  using the conditions of Maranas and Floudas [36]. Repeating their results without proof:

**Theorem Appendix B.1 [36]:** If one of the following conditions holds for the product  $x_1^a \cdot x_2^b$ :

1.  $x_1, x_2 \geq 0$
2.  $a, b \in \mathbb{Z}; a \bmod 2 = 0; b \bmod 2 = 0$
3.  $a, b \in \mathbb{Z}; a \bmod 2 = 1; b \bmod 2 = 1; x_1 \cdot x_2 \geq 0$
4.  $a, b \in \mathbb{Z}; a \bmod 2 = 1; b \bmod 2 = 0; x_1 \geq 0$  (symmetrically  $a \bmod 2 = 0; b \bmod 2 = 1; x_2 \geq 0$ )

Then  $x_1^a \cdot x_2^b$  is convex if at least two of the following are true:  $a \leq 0; b \leq 0; 1 - a - b \leq 0$ . The term is concave if  $a, b \geq 0$  and  $a + b \leq 1$ .  $\square$

**Theorem Appendix B.2 [36]:** If one of the following conditions holds for the product  $x_1^a \cdot x_2^b$ :

1.  $a, b \in \mathbb{Z}; a \bmod 2 = 1; b \bmod 2 = 1; x_1 \cdot x_2 \leq 0$
2.  $a, b \in \mathbb{Z}; a \bmod 2 = 1; b \bmod 2 = 0; x_1 \leq 0$  (symmetrically  $a \bmod 2 = 0; b \bmod 2 = 1; x_2 \leq 0$ )

Then  $x_1^a \cdot x_2^b$  is concave if at least two of the following are true:  $a \leq 0; b \leq 0; 1 - a - b \leq 0$ .  $\square$

For general signomial terms, the MISO framework recognizes convexity/concavity on positive domains:

**Theorem Appendix B.3 [36]:** Signomial term  $\prod_{c=1}^C x_c^{p_{s,c}}$  :  $x \in \mathbb{R}_+^C$  is convex if:

1.  $p_{s,c} \leq 0 \forall c = \{1, \dots, C\}$
- or
2.  $\exists c'$  such that  $p_{s,c'} \geq 1 - \sum_{c \neq c'} p_{s,c}$  and  $p_{s,c} \leq 0 \forall c \neq c'; c = \{1, \dots, C\}$

and concave if  $p_{s,c} \geq 0 \forall c = \{1, \dots, C\}; \sum_c p_{s,c} \leq 1$ .  $\square$