JOTA manuscript No.
(will be inserted by the editor)

Improved convex and concave relaxations of composite bilinear forms*

Matthew E. Wilhelm · **Matthew D. Stuber**

Received: date / Accepted: date

Abstract Deterministic nonconvex optimization solvers generate convex relaxations of nonconvex functions by making use of underlying factorable representations. One approach introduces auxiliary variables assigned to each factor that lifts the problem into a higher-dimensional decision space. In contrast, a generalized McCormick relaxation approach offers the significant advantage of constructing relaxations in the lower dimensionality space of the original problem without introducing auxiliary variables, often referred to as a "reduced-space" approach. Recent contributions illustrated how additional nontrivial inequality constraints may be used in factorable programming to tighten relaxations of the ubiquitous bilinear term. In this work, we exploit an analogous representation of McCormick relaxations and factorable programming to formulate tighter relaxations in the original decision space. We develop the underlying theory to generate necessarily tighter reduced-space McCormick relaxations when *a priori* convex/concave relaxations are known for intermediate bilinear terms. We then show how these rules can be generalized within a McCormick relaxation scheme via three different approaches: the use of a McCormick relaxations coupled to affine arithmetic, the propagation of affine relaxations implied by subgradients, and an enumerative approach that directly uses relaxations of each factor. The developed approaches are benchmarked on a library of optimization problems using the EAGO.jl optimizer. Two case studies are also considered to demonstrate the developments: an application

* Author's final accepted version. Published version: Wilhelm, M.E. and M.D. Stuber. Improved convex and concave relaxations of composite bilinear forms. *Journal of Optimization Theory and Applications.* 197, pg. 174–204. (2023) doi:10.1007/s10957-023-02196-2

Matthew E. Wilhelm
Process Systems and Operations Research Laboratory, Department of Chemical & Biomolecular Engineering, University of Connecticut, 191 Auditorium Road, Unit 3222, Storrs, CT 06269, USA.

Matthew D. Stuber, Corresponding author
Process Systems and Operations Research Laboratory, Department of Chemical & Biomolecular Engineering, University of Connecticut, 191 Auditorium Road, Unit 3222, Storrs, CT 06269, USA.
E-mail: stuber@alum.mit.edu

in advanced manufacturing to optimize supply chain quality metrics and a global dynamic optimization application for rigorous model validation of a kinetic mechanism. The presented subgradient method leads to an improvement in CPU time required to solve the considered problems to ϵ-global optimality.

Keywords Deterministic Global Optimization; Nonconvex Programming; McCormick Relaxations; Branch-and-Bound; Multilinear Products

Mathematics Subject Classification (2000) 65K05 · 90C26 · 90C30

1 Introduction

Deterministic global optimization is required by many routine process systems engineering (PSE) tasks due to the nonconvexity of underlying process models. Two main approaches exist to solve global optimization problems deterministically. The approach that is predominant in state-of-the-art commercial solvers is that of the auxiliary variable method, which exploits a factorable representation of the underlying problem by subsequently lifting the problem into a higher-dimensional space [22]. This higher-dimensional representation simplifies the construction of convex/concave relaxations required to form subproblems, facilitates potentially tighter relaxations of mixed-integer expressions [30], and simplifies a number of important heuristics. For some problems, the introduction of these additional variables may be detrimental as the aforementioned advantages are counterbalanced by increasing complexity. An alternative to this is relaxations that may be computed in the original problem dimensionality space.

The eponymous McCormick relaxations of the bilinear function were first introduced in [27]. These relaxations bound the bilinear term using a series of affine inequalities through the introduction of auxiliary variables; an approach used by many commercially available optimizers, such as ANTIGONE [30] and BARON [54], and the nonconvex solver options of CPLEX [23] and Gurobi [18]. In the past decade, a significant effort has been made to further generalize this approach to arbitrary nonlinear functions. An operator-overloading scheme was detailed by Mitsos et al. [32] for constructing McCormick-based relaxations of functions described by a class of direct algorithms (i.e., algorithms with the number of steps/iterations known *a priori*). Variations on this manner of constructing relaxations in the original problem space through the application of composition rules have been termed *McCormick relaxations*; a convention we adopt herein to maintain consistency with the existing body of literature.

The use of a McCormick relaxation framework [32] potentially offers a significant advantage by allowing for relaxations to be constructed in the original problem space without the introduction of auxiliary variables. Recent developments have dramatically broadened the scope and performance of this approach. Scott et al. [62] developed a generalized McCormick relaxation theory for constructing convex and concave composite relaxations using arbitrary convex and concave functions. Tighter composition rules for multiplication and maximum operators were presented in [67,40]. Methods of generating relaxations of implicit functions were developed by Stuber et al. [65]. Wechsung et al. [71] developed a method of propagating McCormick relaxations backwards on a directed acyclic graph (DAG) representation of a problem. A method for tightening interval bounds was described in [44]. Alternative differentiable relaxations were developed and introduced in [25,26]. Moreover, the theoretical underpinnings of McCormick relaxation performance has been recently explored. These works have illustrated that under mild assumptions, McCormick relaxations exhibit quadratic point-wise convergence [7,41,43]; which may mitigate clustering in branch-and-bound algorithms [24]. Each of these aforementioned advances has been demonstrated to lead to improved performance of global optimizers for specialized classes of simulation-inspired problems.

The benefits of using these reduced-space McCormick-based relaxation methods have been found to span several application areas. These include the deterministic global optimization of process flowsheets [8,9,10], nonconvex optimization problems with embedded surrogate models (such as artificial neural networks and Gaussian process models) [60,59,58,68,75,57], dynamic optimization [73,61], and reachability analysis [56]. Recently, McCormick relaxations have been implemented in two open-source global optimizers: the EAGO [74] toolkit in Julia [5], and the MAiNGO [11] software written in C++. In each implementation, the classic *dependency problem* inherent to set-valued arithmetics naturally arises; wherein, the progressive application of bounding rules leads to expansive departures from convex/concave envelopes of complicated expressions.

To ameliorate the dependency problem, several efforts have been made to expand the typical library of intrinsic functions to include envelopes for common functional forms. These efforts include the development of relaxations of componentwise-convex functions by [38], the construction of novel relaxations of cost and thermodynamic functions in [39], relaxations of activation functions appearing on artificial neural networks [68] and Gaussian processes [57], as well as special relaxations for logarithmic mean temperature difference (LMTD) and its reciprocal [31,42].

One expression of particular interest is that of the bilinear term. This term has been examined extensively within the deterministic nonconvex optimization community. Specialized approaches to treating these problem classes have led to numerous optimizers that initially focused on quadratic (and polynomial) problem formulations and were often subsequently extended to a number of preeminent optimizers including BARON [66], ANTIGONE [30], Gurobi [18], GLOMIQO [29], MOSEK [34], and ALPINE [36,37].

Within the McCormick relaxations literature, the treatment of composite bilinear terms (i.e., $w(\mathbf{z}) = f(\mathbf{z})g(\mathbf{z}),\ \forall \mathbf{z} \in Z$) has been limited to three key theoretical contributions. The first is provided in the work of [32] that details composite relaxations derived from McCormick's original inequalities [27]. The second contribution lies in the analysis of multivariate composite relaxations [67,40] that yield potentially tighter relaxations of the bilinear term under mild assumptions. Lastly, a differentiable relaxation of the bilinear term was detailed in [25,26].

In the larger context of full-space (i.e., lifted-dimensionality space) factorable programming, numerous approaches exist to address the bilinear relaxation, which do not yet directly have an analog in reduced-space factorable programming. One such notable work is that of He and Tawarmalani [19], which details how bilinear relaxations of composite factors can be improved when *a priori* under/overestimators (as well as associated bounds of said under/overestimators) of the bilinear factors, are available. This approach necessarily formulates the relaxations in a higher-dimensional space when applied within a factorable programming context. The authors then propose using a fast combinatorial algorithm to solve a simpler *separation problem* rather than the original formulation [19].

In this paper, we build upon the recent work of He and Tawarmalani [19], detailing how reduced-space McCormick relaxations may be improved when *a priori* knowledge of intermediate convex/concave relaxations exists. We do this by generalizing the results of [19] for factorable programming to composite relaxations. We subsequently discuss three algorithms used to refine convex/concave relaxations of functions for a broad class of nonlinear functions in the original problem space. In Section 2, we detail the mathematical conventions used in the paper. In Section 3, we develop composition rules for generating convex/concave relaxations of intermediate bilinear terms when *a priori* relaxations are known along with associated subgradients. Subsequently, in Section 4, we detail three algorithms that employ this novel theoretical contribution to generate tight relaxations of general nonlinear functions containing bilinear expressions. In Section 5, we provide numerical examples detailing the utility of each algorithm developed herein. We then explore two relevant case studies to demonstrate how improved bilinear relaxations may be applied: an advanced manufacturing system for optimizing supply chain quality metrics in Section 6.1, and a global dynamic optimization application for parameter estimation and rigorous model validation of a kinetic mechanism in Section 6.2. Lastly, we conclude in Section 7 by highlighting potential areas for future research.

2 Mathematical Background

In this section, the mathematical notation and conventions utilized in this paper are defined. Scalar quantities are denoted in lower-case letters (e.g., x) whereas vectors are represented by boldface lower-case letters (e.g., $\mathbf{x}$). Let $X = [\mathbf{x}^L, \mathbf{x}^U]$ represent an n-dimensional interval that is a nonempty compact set defined as $X = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U\}$ with $\mathbf{x}^L$ and $\mathbf{x}^U$ the lower and upper bounds of the interval, respectively. Additionally, let $\mathbb{IR}^n$ be the set of all n-dimensional real intervals, and for any $D \subset \mathbb{R}^n$, $\mathbb{I}D = \{X \in \mathbb{IR}^n : X \subset D\}$ is the set of all interval subsets of D. A set X^n is defined as the Cartesian product $X^n = X \times X \times \cdots \times X$. The mapping $F : \mathbb{I}D \to \mathbb{IR}^n$ is said to be *inclusion monotonic* if $X \subset Y$ implies that $F(X) \subset F(Y)$. The image of X under the mapping $\mathbf{f} : D \to \mathbb{R}^n$ will be denoted by $\mathbf{f}(X)$, whereas an inclusion monotonic interval extension of $\mathbf{f}$ on X will be denoted by $F(X) = [\mathbf{f}^L(X), \mathbf{f}^U(X)]$. From the Fundamental Theorem of Interval Analysis [33, p.47] we have $\mathbf{f}(X) \subset F(X)$, $\forall X \in \mathbb{I}D$. The midpoint of an interval $X \in \mathbb{I}D$ is denoted $\text{mid}(X) = (\mathbf{x}^U + \mathbf{x}^L)/2$ and the radius is given by $\text{rad}(X) = (\mathbf{x}^U - \mathbf{x}^L)/2$, each applied componentwise.

Definition 2.1 (Underestimators and Overestimators) Given a function $w : Z \subset \mathbb{R}^n \to \mathbb{R}$, a function $u : Z \to \mathbb{R}$ is called an *underestimator* of w on Z if and only if $u(\mathbf{z}) \leq w(\mathbf{z})$ for every $\mathbf{z} \in Z$. Similarly, $o : Z \subset \mathbb{R}^n \to \mathbb{R}$ is called an *overestimator* of w on Z if and only if $o(\mathbf{z}) \geq w(\mathbf{z})$ for every $\mathbf{z} \in Z$.

Definition 2.2 (Convex and Concave Relaxations [32]) Given a convex set $Z \subset \mathbb{R}^n$ and a function $w : Z \to \mathbb{R}$, $w^{cv} : Z \to \mathbb{R}$ is a *convex relaxation* of w on Z if and only if it is both convex and an underestimator of w on Z. Similarly, $w^{cc} : Z \to \mathbb{R}$ is a *concave relaxation* of w on Z provided it is both concave and an overestimator of w on Z.

In the case of vector-valued and matrix-valued functions, convex and concave relaxations are defined by the respective componentwise and elementwise application of the above inequalities.

Definition 2.3 (Factorable Function [62]) A function $\mathscr{F} : Z \subset \mathbb{R}^n \to \mathbb{R}$ is *factorable* if it can be expressed in terms of a finite number of factors $v_1, \ldots, v_m$, such that given $\mathbf{z} \in Z$, $v_i = z_i$ for $i = 1, \ldots, n$, and v_k is defined for $n < k \leq m$ as either

1. $v_k = v_i + v_j$, with, $i, j < k$, or
2. $v_k = v_i v_j$, with, $i, j < k$, or
3. $v_k = u_k(v_i)$, with, $i < k$, where $u_k : B_k \to R$ is a univariate intrinsic function,

and $\mathscr{F}(\mathbf{z}) = v_m(\mathbf{z})$, for every $\mathbf{z} \in Z$. A vector-valued function is factorable if each of its components are factorable functions.

Definition 2.4 (Cumulative Mapping [62]) Let the *cumulative mapping* v_k be the mapping $v_k : Z \to \mathbb{R}$ defined for each $\mathbf{z} \in Z$ by the value $v_k(\mathbf{z})$ when the factors of $\mathscr{F}$ are computed recursively, as per Definition 2.3, beginning from $\mathbf{z}$.

Proposition 2.1 (McCormick Multiplication Rule [32]) *Let $Z \subset \mathbb{R}^n$ be a nonempty convex set. Let $w, x_1, x_2 : Z \to \mathbb{R}$ such that $w(\mathbf{z}) = x_1(\mathbf{z})x_2(\mathbf{z})$. Let $x_1^{cv} : Z \to \mathbb{R}$ and $x_1^{cc} : Z \to \mathbb{R}$ be convex and concave relaxations of x_1 on Z, respectively. Let $x_2^{cv} : X \to \mathbb{R}$ and $x_2^{cc} : X \to \mathbb{R}$ be convex and concave relaxations of x_2 on Z, respectively. Further, let $x_1^L, x_1^U, x_2^L, x_2^U \in \mathbb{R}$ be bounds on x_1, x_2 such that*

$$x_1^L \leq x_1(\mathbf{z}) \leq x_1^U \qquad \text{and} \qquad x_2^L \leq x_2(\mathbf{z}) \leq x_2^U, \qquad \forall \mathbf{z} \in Z.$$

Let the following intermediate functions $\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1, \delta_2 : Z \to \mathbb{R}$ be defined as

$$\begin{aligned}
\alpha_1(\cdot) &= \min\left\{x_2^L x_1^{cv}(\cdot), x_2^L x_1^{cc}(\cdot)\right\}, & \alpha_2(\cdot) &= \min\left\{x_1^L x_2^{cv}(\cdot), x_1^L x_2^{cc}(\cdot)\right\},\\
\beta_1(\cdot) &= \min\left\{x_2^U x_1^{cv}(\cdot), x_2^U x_1^{cc}(\cdot)\right\}, & \beta_2(\cdot) &= \min\left\{x_1^U x_2^{cv}(\cdot), x_1^U x_2^{cc}(\cdot)\right\},\\
\gamma_1(\cdot) &= \max\left\{x_2^L x_1^{cv}(\cdot), x_2^L x_1^{cc}(\cdot)\right\}, & \gamma_2(\cdot) &= \max\left\{x_1^U x_2^{cv}(\cdot), x_1^U x_2^{cc}(\cdot)\right\},\\
\delta_1(\cdot) &= \max\left\{x_2^U x_1^{cv}(\cdot), x_2^U x_1^{cc}(\cdot)\right\}, & \delta_2(\cdot) &= \max\left\{x_1^L x_2^{cv}(\cdot), x_1^L x_2^{cc}(\cdot)\right\}.
\end{aligned}$$

Then, convex and concave relaxations of w on Z are given by $w_{\times,0}^{cv}$ and $w_{\times,0}^{cc}$,

$$\begin{aligned}
w_{\times,0}^{cv} &: Z \to \mathbb{R} : \mathbf{z} \mapsto \max\left\{\alpha_1(\mathbf{z}) + \alpha_2(\mathbf{z}) - x_1^L x_2^L, \beta_1(\mathbf{z}) + \beta_2(\mathbf{z}) - x_1^U x_2^U\right\},\\
w_{\times,0}^{cc} &: Z \to \mathbb{R} : \mathbf{z} \mapsto \min\left\{\gamma_1(\mathbf{z}) + \gamma_2(\mathbf{z}) - x_1^U x_2^L, \delta_1(\mathbf{z}) + \delta_2(\mathbf{z}) - x_1^L x_2^U\right\},
\end{aligned}$$

respectively.

Note that the definitions of $w^{cv}_{\times,0}$ and $w^{cc}_{\times,0}$ in Proposition 2.1 arise from standard McCormick relaxations of the bilinear term. In Section 3, we define two additional sets of composite relaxations, denoted $w^{cv}_{\times,1}$, $w^{cc}_{\times,1}$, and $w^{cv}_{\times,2}$, $w^{cc}_{\times,2}$, respectively, which may be combined with $w^{cv}_{\times,0}$ and $w^{cc}_{\times,0}$ to yield tighter relaxations.

The principal contributions of this paper lie in an extension of the work of [19] to relaxations necessarily as tight as those presented in Proposition 2.1.

Theorem 2.1 (Underestimators of the bilinear term implied by *a priori* underestimators (Theorem 1 in [19])) *Let $f_1^L \leq a_1 \leq f_1^U$ and $f^L \leq a_2 \leq f^U$. Then, consider the set:*

$$P = \Big\{(u_1, f_1, u_2, f_2) \,:\, f_1^L \leq u_1 \leq \min\{f_1, a_1\}, f_1 \leq f_1^U, \\ f_2^L \leq u_2 \leq \min\{f_2, a_2\},\ f_2 \leq f_2^U\Big\}.$$

The following linear inequalities are valid for the epigraph of $f_1 f_2$ over P:

$$f_1 f_2 \geq \max \left\{ \begin{array}{l} e_1 := f_1 f2^U + f_2 f_1^U - f_1^U f_2^U \\ e_2 := (f_2^U - a_2)u_1 + (f_1^U - a_1)u_2 + a_2 f_1 + a_1 f_2 \\ \quad + a_1 a_2 - a_1 f_2^U - f_1^U a_2 \\ e_3 := (f_2^U - f_2^L)u_1 + f_2^L f_1 + a_1 f_2 - a_1 f_2^U \\ e_4 := (f_1^U - f_1^L)u_2 + a_2 f_1 + f_1^L f_2 - f_1^U a_2 \\ e_5 := (a_2 - f_2^L)u_1 + (a_1 - f_1^L)u_2 + f_2^L f_1 + f_1^L f_2 - a_1 a_2 \\ e_6 := f_1^L f_2 + f_1 f_2^L - f_1^L f_2^L \end{array} \right\}. \tag{1}$$

Theorem 2.2 (Overestimators of the bilinear term implied by *a priori* underestimators (Theorem 5 in [19])) *Let $f_1^L \leq a_1 \leq f_1^U$ and $f^L \leq a_2 \leq f^U$. Then, consider the set:*

$$P = \Big\{(u_1, f_1, u_2, f_2) \,:\, f_1^L \leq u_1 \leq \min\{f_1, a_1\}, f_1 \leq f_1^U, \\ f_2^L \leq u_2 \leq \min\{f_2, a_2\},\ f_2 \leq f_2^U\Big\}.$$

The following linear inequalities are valid for the epigraph of $f_1 f_2$ over P:

$$f_1 f_2 \leq \min \left\{ \begin{array}{l} r_1 := f_2^L f_1 + f_1^U f_2 - f_1^U f_2^L \\ r_2 := (f_2^L - a_2)u_1 + (a_1 - f_1^U)u_2 + a_2 f_1 + f_1^U f_2 - a_1 f_2^L \\ r_3 := (f_2^L - f_2^U)u_1 + a_1 f_2 + f_2^U f_1 - a_1 f_2^L \\ r_4 := (f_1^L - f_1^U)u_2 + a_2 f_1 + f_1^U f_2 - f_1^L a_2 \\ r_5 := (a_2 - f_2^U)u_1 + (f_1^L - a_1)u_2 + f_2^U f_1 + a_1 f_2 - f_1^L a_2 \\ r_6 := f_1 f_2^U + f_1^L f_2 - f_1^L f_2^U \end{array} \right\}. \tag{2}$$

We note that e_1, e_6, r_1, and r_6 in Theorem 2.1 and Theorem 2.2 are simply restatements of the inequalities presented by McCormick [27]. The other inequalities are derived from simple algebraic arguments. For a full review of these derivations, the reader is directed to [19]. We will recreate one such derivation from [19] here as an example:

$$\begin{aligned} f_1 f_2 &= (f_1 - f_1^L)(f_2 - f_2^L) + f_1^L f_2 + f_2^L f_1 - f_1^L f_2^L \\ &\geq (u_1 - f_1^L)(f_2 - f_2^L) + f_1^L f_2 + f_2^L f_1 - f_1^L f_2^L \\ &\geq (u_1 - f_1^L)(f_2 - f_2^L) + (a_1 - f_1^L)(f_2 - f_2^L) \\ &- (a_1 - f_1^L)(f_2^U - f_2^L) + f_2^L f_1 + f_1^L f_2 - f_1^L f_2^L \\ &= e_3. \end{aligned}$$

Definition 2.5 (Subgradients [65]) Let $Z \subset \mathbb{R}^n$ be a nonempty convex set, $w^{cv} : Z \to \mathbb{R}$ be convex, and $w^{cc} : Z \to \mathbb{R}$ be concave. A function $\mathbf{s}_w^{cv} : Z \to \mathbb{R}^n$ is a *subgradient* of w^{cv} on Z if for each $\bar{\mathbf{z}} \in Z$, $w^{cv}(\mathbf{z}) \geq w^{cv}(\bar{\mathbf{z}}) + \mathbf{s}_w^{cv}(\bar{\mathbf{z}})^{\mathrm{T}}(\mathbf{z} - \bar{\mathbf{z}}), \forall \mathbf{z} \in Z$. Similarly, a function $\mathbf{s}_w^{cc} : Z \to \mathbb{R}^n$ is a subgradient of w^{cc} on Z if for each $\bar{\mathbf{z}} \in Z$, $w^{cc}(\mathbf{z}) \leq w^{cc}(\bar{\mathbf{z}}) + \mathbf{s}_w^{cc}(\bar{\mathbf{z}})^{\mathrm{T}}(\mathbf{z} - \bar{\mathbf{z}}), \forall \mathbf{z} \in Z$.

Note that subgradients of vector-valued functions and subgradients of matrix-valued functions will be defined analogously and will be matrix-valued functions and third-order tensor-valued functions, respectively.

Theorem 2.3 (Multiplication Rule for Subgradients [32]) *Suppose that $Z \subset \mathbb{R}^n$ is a nonempty convex set, and $\mathbf{z} \in Z$. Let $w, x_1, x_2 : Z \to \mathbb{R}$ such that $w(\mathbf{z}) = x_1(\mathbf{z})x_2(\mathbf{z})$. Let $x_1^{cv} : Z \to \mathbb{R}$ and $x_1^{cc} : Z \to \mathbb{R}$ be convex and concave relaxations of x_1 on Z, respectively. Let $x_2^{cv} : X \to \mathbb{R}$ and $x_2^{cc} : X \to \mathbb{R}$ be convex and concave relaxations of x_2 on Z, respectively. Further, let $x_1^L, x_1^U, x_2^L, x_2^U \in \mathbb{R}$ be bounds on x_1, x_2 such that*

$$x_1^L \leq x_1(\mathbf{z}) \leq x_1^U \qquad \textit{and} \qquad x_2^L \leq x_2(\mathbf{z}) \leq x_2^U, \qquad \forall \mathbf{z} \in Z.$$

Let the intermediate functions $\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1, \delta_2 : Z \to \mathbb{R}$ and convex/concave relaxations $w_{\times,0}^{cv}, w_{\times,0}^{cc} : Z \to \mathbb{R}$ be defined as in Proposition 2.1. Then the subgradients of $\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \delta_1, \delta_2$ at $\bar{\mathbf{z}} \in Z$ are respectively given by:

$$\mathbf{s}_{\alpha_1}(\bar{\mathbf{z}}) = \begin{cases} x_2^L \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}) & \textit{if } x_2^L \geq 0, \\ x_2^L \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}}) & \textit{otherwise}, \end{cases} \qquad \mathbf{s}_{\alpha_2}(\bar{\mathbf{z}}) = \begin{cases} x_1^L \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}) & \textit{if } x_1^L \geq 0, \\ x_1^L \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}}) & \textit{otherwise}, \end{cases}$$

$$\mathbf{s}_{\beta_1}(\bar{\mathbf{z}}) = \begin{cases} x_2^U \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}) & \textit{if } x_2^U \geq 0, \\ x_2^U \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}}) & \textit{otherwise}, \end{cases} \qquad \mathbf{s}_{\beta_2}(\bar{\mathbf{z}}) = \begin{cases} x_1^U \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}) & \textit{if } x_1^U \geq 0, \\ x_1^U \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}}) & \textit{otherwise}, \end{cases}$$

$$\mathbf{s}_{\gamma_1}(\bar{\mathbf{z}}) = \begin{cases} x_2^L \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}}) & \textit{if } x_2^L \geq 0, \\ x_2^L \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}) & \textit{otherwise}, \end{cases} \qquad \mathbf{s}_{\gamma_2}(\bar{\mathbf{z}}) = \begin{cases} x_1^U \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}}) & \textit{if } x_1^U \geq 0, \\ x_1^U \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}) & \textit{otherwise}, \end{cases}$$

$$\mathbf{s}_{\delta_1}(\bar{\mathbf{z}}) = \begin{cases} x_2^U \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}}) & \textit{if } x_2^U \geq 0, \\ x_2^U \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}) & \textit{otherwise}, \end{cases} \qquad \mathbf{s}_{\delta_2}(\bar{\mathbf{z}}) = \begin{cases} x_1^L \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}}) & \textit{if } x_1^L \geq 0, \\ x_1^L \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}) & \textit{otherwise}, \end{cases}$$

where $\mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}})$, $\mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})$, $\mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}})$, $\mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})$, are, respectively, subgradients of x_1^{cv}, x_1^{cc}, x_2^{cv}, x_2^{cc} on Z at $\bar{\mathbf{z}} \in Z$. Finally, the subgradients $\mathbf{s}_{w_{\times,0}}^{cv}(\bar{\mathbf{z}})$, $\mathbf{s}_{w_{\times,0}}^{cc}(\bar{\mathbf{z}})$ of $w_{\times,0}^{cv}$, $w_{\times,0}^{cc}$ on Z at $\bar{\mathbf{z}} \in Z$

are, respectively, given by:

$$\mathbf{s}^{cv}_{w_\times,0}(\bar{\mathbf{z}}) = \begin{cases} \mathbf{s}_{\alpha_2}(\bar{\mathbf{z}}) + \mathbf{s}_{\alpha_2}(\bar{\mathbf{z}}), & \text{if } \alpha_1(\bar{\mathbf{z}}) + \alpha_2(\bar{\mathbf{z}}) - x_1^L x_2^L \geq \beta_1(\bar{\mathbf{z}}) + \beta_2(\bar{\mathbf{z}}) - x_1^U x_2^U, \\ \mathbf{s}_{\beta_1}(\bar{\mathbf{z}}) + \mathbf{s}_{\beta_2}(\bar{\mathbf{z}}), & \text{otherwise,} \end{cases}$$

$$\mathbf{s}^{cc}_{w_\times,0}(\bar{\mathbf{z}}) = \begin{cases} \mathbf{s}_{\gamma_1}(\bar{\mathbf{z}}) + \mathbf{s}_{\gamma_2}(\bar{\mathbf{z}}), & \text{if } \gamma_1(\bar{\mathbf{z}}) + \gamma_2(\bar{\mathbf{z}}) - x_1^U x_2^L \geq \delta_1(\bar{\mathbf{z}}) + \delta_2(\bar{\mathbf{z}}) - x_1^L x_2^U \\ \mathbf{s}_{\delta_1}(\bar{\mathbf{z}}) + \mathbf{s}_{\delta_2}(\bar{\mathbf{z}}), & \text{otherwise.} \end{cases}$$

For details relating to computing relaxations and associated subgradients of factors using other functional forms, the reader is referred to [32]. For convex/-concave relaxations $v_k^{cv}, v_k^{cc} : Z \to \mathbb{R}$ computed through the recursive application of these rules to each factor v_k, the factorable function $\mathscr{F}$ also constitutes a cumulative mapping.

3 Tight Composite Relaxations of Bilinear Terms

We now describe two major theoretical contributions. First, we develop Theorem 3.1 as an extension of Theorem 2.1 and Theorem 2.2 (Theorem 1 and Theorem 5 from [19], respectively), to compute convex and concave relaxations of the bilinear term using convex and concave relaxations of its arguments and *a priori* convex underestimators. This new result differs from the preliminary work of [19] in that the introduction of auxiliary variables for intermediate bilinear terms into the optimization formulation, is not required. Secondly, a corresponding approach that uses *a priori* concave relaxations is then detailed in Theorem 3.2. We then combine these results in Theorem 3.3 to obtain tight relaxations of bilinear terms exploiting both *a priori* convex and concave relaxations, simultaneously. Finally, we develop subgradients of these relaxations in Theorem 3.4.

Theorem 3.1 *Define $x_1 : Z \subset \mathbb{R}^n \to X_1 \subset \mathbb{R}$ and $x_2 : Z \subset \mathbb{R}^n \to X_2 \subset \mathbb{R}$ with corresponding convex/convex relaxations $x_1^{cv}, x_1^{cc}, x_2^{cv}, x_2^{cc} : Z \to \mathbb{R}$ on Z. Let $u_1, u_2 : Z \subset \mathbb{R}^n \to \mathbb{R}$ be underestimators of x_1, x_2 on Z, respectively, with associated $(a_1, a_2) \in X_1 \times X_2 \in \mathbb{IR}^2$ such that $x_1^L \leq u_1(\cdot) \leq \min\{x_1(\cdot), a_1\}$ and $x_2^L \leq u_2(\cdot) \leq \min\{x_2(\cdot), a_2\}$. Further, suppose that convex relaxations of u_1 and u_2 on Z are available. Let the following intermediate factors be defined as:*

$$\begin{aligned} \alpha_1(\cdot) &= \min\{a_2 x_1^{cv}(\cdot), a_2 x_1^{cc}(\cdot)\}, & \beta_1(\cdot) &= \max\{x_1^U x_2^{cv}(\cdot), x_1^U x_2^{cc}(\cdot)\}, \\ \alpha_2(\cdot) &= \min\{a_1 x_2^{cv}(\cdot), a_1 x_2^{cc}(\cdot)\}, & \beta_2(\cdot) &= \max\{a_2 x_2^{cv}(\cdot), a_2 x_2^{cc}(\cdot)\}, \\ \alpha_3(\cdot) &= \min\{x_2^L x_1^{cv}(\cdot), x_2^L x_1^{cc}(\cdot)\}, & \beta_3(\cdot) &= \max\{a_1 x_2^{cv}(\cdot), a_1 x_2^{cc}(\cdot)\}, \\ \alpha_4(\cdot) &= \min\{x_1^L x_2^{cv}(\cdot), x_1^L x_2^{cc}(\cdot)\}, & \beta_4(\cdot) &= \max\{x_2^U x_2^{cv}(\cdot), x_2^U x_2^{cc}(\cdot)\}, \\ \rho_1 &= a_1 a_2 - a_1 x_2^U - a_2 x_1^U. \end{aligned}$$

Then, the following expressions:

$$w_1^{cv}(\cdot) = (x_2^U - a_2)u_1^{cv}(\cdot) + (x_1^U - a_1)u_2^{cv}(\cdot) + \alpha_1(\cdot) + \alpha_2(\cdot) + \rho_1, \tag{3}$$
$$w_2^{cv}(\cdot) = (x_2^U - x_2^L)u_1^{cv}(\cdot) + \alpha_2(\cdot) + \alpha_3(\cdot) - a_1 x_2^U, \tag{4}$$
$$w_3^{cv}(\cdot) = (x_1^U - x_1^L)u_2^{cv}(\cdot) + \alpha_1(\cdot) + \alpha_4(\cdot) - a_2 x_1^U, \tag{5}$$
$$w_4^{cv}(\cdot) = (a_2 - x_2^L)u_1^{cv}(\cdot) + (a_1 - x_1^L)u_2^{cv}(\cdot) + \alpha_3(\cdot) + \alpha_4(\cdot) - a_1 a_2, \tag{6}$$

are convex relaxations of $w(\cdot) = x_1(\cdot)x_2(\cdot)$ *on* Z*. Moreover, the following expressions:*

$$w_1^{cc}(\cdot) = (x_2^L - a_2)u_1^{cv}(\cdot) + (a_1 - x_1^U)u_2^{cv}(\cdot) + \beta_1(\cdot) + \beta_2(\cdot) - a_1 x_2^L, \tag{7}$$
$$w_2^{cc}(\cdot) = (x_2^L - x_2^U)u_1^{cv}(\cdot) + \beta_3(\cdot) + \beta_4(\cdot) - a_1 x_2^L, \tag{8}$$
$$w_3^{cc}(\cdot) = (x_1^L - x_1^U)u_2^{cv}(\cdot) + \beta_1(\cdot) + \beta_2(\cdot) - a_2 x_1^L, \tag{9}$$
$$w_4^{cc}(\cdot) = (a_2 - x_2^U)u_1^{cv}(\cdot) + (x_1^L - a_1)u_2^{cv}(\cdot) + \beta_3(\cdot) + \beta_4(\cdot) - a_2 x_1^L, \tag{10}$$

are concave relaxations of $w(\cdot) = x_1(\cdot)x_2(\cdot)$ *on* Z*. Lastly, the expressions*

$$w_{\times,1}^{cv}(\cdot) = \max\left\{w_1^{cv}(\cdot), w_2^{cv}(\cdot), w_3^{cv}(\cdot), w_4^{cv}(\cdot)\right\}, \tag{11}$$
$$w_{\times,1}^{cc}(\cdot) = \min\left\{w_1^{cc}(\cdot), w_2^{cc}(\cdot), w_3^{cc}(\cdot), w_4^{cc}(\cdot)\right\}, \tag{12}$$

are convex and concave relaxations of $w(\cdot) = x_1(\cdot)x_2(\cdot)$ *on* Z*, respectively.*

Proof He and Tawarmalani [19] define nontrivial underestimators and overestimators of $w = w^*(\xi_1, \xi_2) = \xi_1\xi_2$ on $X_1 \times X_2$ by $e_i(\xi_1, \xi_2) \leq w$ and $r_i(\xi_1, \xi_2) \geq w$ for $i = 2, \ldots 5$, respectively, as detailed in Theorem 2.1 and Theorem 2.2. As we have $x_i : Z \to X_i$ for $i \in 1, 2$, we may write $\mathscr{E}_i(\cdot) = e_i(x_1(\cdot), x_2(\cdot))$ and $\mathscr{R}_i(\cdot) = r_i(x_1(\cdot), x_2(\cdot))$ for $i = 2, \ldots 5$, respectively, which are under/overestimators of $w(\cdot)$ on Z. Namely,

$$\mathscr{E}_2(\cdot) = (x_2^U - a_2)u_1(\cdot) + (x_1^U - a_1)u_2(\cdot) + a_2 x_1(\cdot) + a_1 x_2(\cdot) + \rho_1, \tag{13}$$
$$\mathscr{E}_3(\cdot) = (x_2^U - x_2^L)u_1(\cdot) + x_2^L x_1(\cdot) + a_1 x_2(\cdot) - a_1 x_2^U, \tag{14}$$
$$\mathscr{E}_4(\cdot) = (x_1^U - x_1^L)u_2(\cdot) + a_2 x_1(\cdot) + x_1^L x_2(\cdot) - a_2 x_1^U, \tag{15}$$
$$\mathscr{E}_5(\cdot) = (a_2 - x_2^L)u_1(\cdot) + (a_1 - x_1^L)u_2(\cdot) + x_2^L x_1(\cdot) + x_1^L x_2(\cdot) - a_1 a_2, \tag{16}$$
$$\mathscr{R}_2(\cdot) = (x_2^L - a_2)u_1(\cdot) + (a_1 - x_1^U)u_2(\cdot) + a_2 x_1(\cdot) + x_1^U x_2(\cdot) - a_1 x_2^L, \tag{17}$$
$$\mathscr{R}_3(\cdot) = (x_2^L - x_2^U)u_1(\cdot) + a_1 x_2(\cdot) + x_2^U x_1(\cdot) - a_1 x_2^L, \tag{18}$$
$$\mathscr{R}_4(\cdot) = (x_1^L - x_1^U)u_2(\cdot) + a_2 x_1(\cdot) + x_1^U x_2(\cdot) - a_2 x_1^L, \tag{19}$$
$$\mathscr{R}_5(\cdot) = (a_2 - x_2^U)u_1(\cdot) + (x_1^L - a_1)u_2(\cdot) + x_2^U x_1(\cdot) + a_1 x_2(\cdot) - a_2 x_1^L. \tag{20}$$

First, we note that the terms $(x_i^U - a_i)$, $(x_i^U - x_i^L)$, and $(a_i - x_i^L)$ are positive for $i \in \{1, 2\}$. As such, convex relaxations of the $\alpha u_i(\cdot)$ terms in (13)-(16) on Z are given by $\alpha u_i^{cv}(\cdot)$, for $i \in \{1, 2\}$. Next, we note that $(a_i - x_i^U)$, $(x_i^L - x_i^U)$, and $(x_i^L - a_i)$ are negative by construction for $i \in \{1, 2\}$. As such, concave relaxations of the $\alpha u_i(\cdot)$ terms in (17)-(20) on Z are given by $\alpha u_i^{cv}(\cdot)$, for $i \in \{1, 2\}$. The remaining coefficients of $x_1(\cdot)$ and $x_2(\cdot)$ may be either positive or negative, and as such, a convex relaxation of $\alpha x_i(\cdot)$ on Z is given by $\min\left\{\alpha x_i^{cv}(\cdot), \alpha x_i^{cc}(\cdot)\right\}$, whereas a concave relaxation of

$\alpha x_i(\cdot)$ on Z is given by $\max\left\{\alpha x_i^{cv}(\cdot), \alpha x_i^{cc}(\cdot)\right\}$ for $i \in \{1,2\}$. The sum of convex functions is convex, and the sum of concave functions is concave. The pointwise maximum of convex underestimators is convex while the pointwise minimum of concave overestimators is concave. Thus, the expressions (3)-(12) hold. □

The above relaxations are derived from Theorem 2.1 and Theorem 2.2. This requires knowledge of valid underestimating functions u_1 and u_2. When propagating relaxations through a composite function, it is quite common to have *a priori* knowledge of valid overestimating functions as well. The following Theorem 3.2 is a new result that adapts the results of Theorem 3.1 to improve relaxations using valid overestimating functions known *a priori.*

Theorem 3.2 *Define* $x_1 : Z \subset \mathbb{R}^n \to X_1 \subset \mathbb{R}$ *and* $x_2 : Z \subset \mathbb{R}^n \to X_2 \subset \mathbb{R}$ *with corresponding convex/convex relaxations* $x_1^{cv}, x_1^{cc}, x_2^{cv}, x_2^{cc} : Z \to \mathbb{R}$ *on* Z*. Let* $o_1, o_2 : Z \to \mathbb{R}$ *be overestimators of* x_1, x_2 *on* Z*, respectively, with associated* $(b_1, b_2) \in X_1 \times X_2 \in \mathbb{IR}^2$ *such that* $x_1^U \geq o_1(\cdot) \geq \max\{x_1(\cdot), b_1\}$ *and* $x_2^U \geq o_2(\cdot) \geq \max\{x_2(\cdot), b_2\}$*. Furthermore, suppose that concave relaxations of* o_1 *and* o_2 *on* Z *are available. Let the following intermediate factors be defined as*

$$
\begin{aligned}
\delta_1(\cdot) &= \min\{b_2 x_1^{cv}(\cdot), b_2 x_1^{cc}(\cdot)\}, & \gamma_1(\cdot) &= \max\{b_2 x_1^{cv}(\cdot), b_2 x_1^{cc}(\cdot)\},\\
\delta_2(\cdot) &= \min\{b_1 x_2^{cv}(\cdot), b_1 x_2^{cc}(\cdot)\}, & \gamma_2(\cdot) &= \max\{x_1^L x_2^{cv}(\cdot), x_1^L x_2^{cc}(\cdot)\},\\
\delta_3(\cdot) &= \min\{x_2^U x_1^{cv}(\cdot), x_2^U x_1^{cc}(\cdot)\}, & \gamma_3(\cdot) &= \max\{b_1 x_2^{cv}(\cdot), b_1 x_2^{cc}(\cdot)\},\\
\delta_4(\cdot) &= \min\{x_1^U x_2^{cv}(\cdot), x_1^U x_2^{cc}(\cdot)\}, & \gamma_4(\cdot) &= \max\{x_2^L x_1^{cv}(\cdot), x_2^L x_1^{cc}(\cdot)\},\\
\rho_2 &= b_1 b_2 - b_1 x_2^L - b_2 x_1^L.
\end{aligned}
$$

Then, the following expressions:

$$w_5^{cv}(\cdot) = (x_2^L - b_2) o_1^{cc}(\cdot) + (x_1^L - b_1) o_2^{cc}(\cdot) + \delta_1(\cdot) + \delta_2(\cdot) + \rho_2, \tag{21}$$
$$w_6^{cv}(\cdot) = (x_2^L - x_2^U) o_1^{cc}(\cdot) + \delta_2(\cdot) + \delta_3(\cdot) - b_1 x_2^L, \tag{22}$$
$$w_7^{cv}(\cdot) = (x_1^L - x_1^U) o_2^{cc}(\cdot) + \delta_1(\cdot) + \delta_4(\cdot) - b_2 x_1^L, \tag{23}$$
$$w_8^{cv}(\cdot) = (b_2 - x_2^U) o_1^{cc}(\cdot) + (b_1 - x_1^U) o_2^{cc}(\cdot) + \delta_3(\cdot) + \delta_4(\cdot) - b_1 b_2, \tag{24}$$

are convex relaxations of $w(\cdot) = x_1(\cdot) x_2(\cdot)$ *on* Z*. Moreover, the following expressions:*

$$w_5^{cc}(\cdot) = (x_2^U - b_2) o_1^{cc}(\cdot) + (b_1 - x_1^L) o_2^{cc}(\cdot) + \gamma_1(\cdot) + \gamma_2(\cdot) - b_1 x_2^U, \tag{25}$$
$$w_6^{cc}(\cdot) = (x_2^U - x_2^L) o_1^{cc}(\cdot) + \gamma_3(\cdot) + \gamma_4(\cdot) - x_2^U b_1, \tag{26}$$
$$w_7^{cc}(\cdot) = (x_1^U - x_1^L) o_2^{cc}(\cdot) + \gamma_1(\cdot) + \gamma_2(\cdot) - x_1^U b_2, \tag{27}$$
$$w_8^{cc}(\cdot) = (b_2 - x_2^L) o_1^{cc}(\cdot) + (x_1^U - b_1) o_2^{cc}(\cdot) + \gamma_3(\cdot) + \gamma_4(\cdot) - x_1^U b_2, \tag{28}$$

are concave relaxations of $w(\cdot) = x_1(\cdot) x_2(\cdot)$ *on* Z*. Lastly, the expressions*

$$
\begin{aligned}
w_{\times,2}^{cv}(\cdot) &= \max\left\{w_5^{cv}(\cdot), w_6^{cv}(\cdot), w_7^{cv}(\cdot), w_8^{cv}(\cdot)\right\},\\
w_{\times,2}^{cc}(\cdot) &= \min\left\{w_5^{cc}(\cdot), w_6^{cc}(\cdot), w_7^{cc}(\cdot), w_8^{cc}(\cdot)\right\},
\end{aligned}
$$

are convex and concave relaxations of $w(\cdot) = x_1(\cdot) x_2(\cdot)$ *on* Z*, respectively.*

Proof First, we note that $x_1(\cdot)x_2(\cdot) = y_1(\cdot)y_2(\cdot)$, with $y_1(\cdot) = -x_1(\cdot)$ and $y_2(\cdot) = -x_2(\cdot)$. Define $(a_1, a_2) \in Y_1 \times Y_2$, where $Y_i = -X_i$ and $a_i = -b_i$. Let $u_1, u_2 : Z \to \mathbb{R}$ be underestimators of y_1, y_2 on Z, respectively, defined as $u_i(\cdot) = -o_i(\cdot)$. Then, we see that $y_1^L \le u_1(\cdot) \le \min\{y_1(\cdot), a_1\}$, $y_1(\cdot) \le y_1^U$ and $y_2^L \le u_2(\cdot) \le \min\{y_2(\cdot), a_2\}$, $y_2(\cdot) \le y_2^U$. Similarly, we have $y_i^L \le u_i(\cdot) \le \min\{y_i(\cdot), a_i\}$ and $-y_i^L \ge -u_i(\cdot) \ge -\min\{y_i(\cdot), a_i\}$, and therefore $x_i^U \ge -u_i(\cdot) \ge -\min\{-x_i(\cdot), a_i\} = \max\{x_i(\cdot), -a_i\} = \max\{x_i(\cdot), b_i\}$. It remains to show that the facets defined by (21)-(28) are valid. Again inspecting the underestimators and overestimators of Theorem 2.1 and Theorem 2.2, we have

$$
\begin{aligned}
\mathscr{E}_2(\cdot) &= (y_2^U - a_2)u_1(\cdot) + (y_1^U - a_1)u_2(\cdot) + a_2 y_1(\cdot) + a_1 y_2(\cdot) + a_1 a_2 - a_1 y_2^U - y_1^U a_2,\\
\mathscr{E}_3(\cdot) &= (y_2^U - y_2^L)u_1(\cdot) + y_2^L y_1(\cdot) + a_1 y_2(\cdot) - a_1 y_2^U,\\
\mathscr{E}_4(\cdot) &= (y_1^U - y_1^L)u_2(\cdot) + a_2 y_1(\cdot) + y_1^L y_2(\cdot) - a_2 y_1^U,\\
\mathscr{E}_5(\cdot) &= (a_2 - y_2^L)u_1(\cdot) + (a_1 - y_1^L)u_2(\cdot) + y_2^L y_1(\cdot) + y_1^L y_2(\cdot) - a_1 a_2,\\
\mathscr{R}_2(\cdot) &= (y_2^L - a_2)u_1(\cdot) + (a_1 - y_1^U)u_2(\cdot) + a_2 y_1(\cdot) + y_1^U y_2(\cdot) - a_1 y_2^L,\\
\mathscr{R}_3(\cdot) &= (y_2^L - y_2^U)u_1(\cdot) + a_1 y_2(\cdot) + y_2^U y_1(\cdot) - a_1 y_2^L,\\
\mathscr{R}_4(\cdot) &= (y_1^L - y_1^U)u_2(\cdot) + a_2 y_1(\cdot) + y_1^U y_2(\cdot) - a_2 y_1^L,\\
\mathscr{R}_5(\cdot) &= (a_2 - y_2^U)u_1(\cdot) + (y_1^L - a_1)u_2(\cdot) + y_2^U y_1(\cdot) + a_1 y_2(\cdot) - a_2 y_1^L.
\end{aligned}
$$

Substituting in b_i for a_i, $x_i(\cdot)$ for $y_i(\cdot)$, and $o_i(\cdot)$ for $u_i(\cdot)$, we get

$$
\begin{aligned}
\mathscr{E}_2(\cdot) &= (x_2^L - b_2)o_1(\cdot) + (x_1^L - b_1)o_2(\cdot) + b_2 x_1(\cdot) + b_1 x_2(\cdot) + b_1 b_2 - b_1 x_2^L - b_2 x_1^L,\\
\mathscr{E}_3(\cdot) &= (x_2^L - x_2^U)o_1(\cdot) + x_2^U x_1(\cdot) + b_1 x_2(\cdot) - b_1 x_2^L,\\
\mathscr{E}_4(\cdot) &= (x_1^L - x_1^U)o_2(\cdot) + b_2 x_1(\cdot) + x_1^U x_2(\cdot) - b_2 x_1^L,\\
\mathscr{E}_5(\cdot) &= (b_2 - x_2^U)o_1(\cdot) + (b_1 - x_1^U)o_2(\cdot) + x_2^U x_1(\cdot) + x_1^U x_2(\cdot) - b_1 b_2,\\
\mathscr{R}_2(\cdot) &= (x_2^U - b_2)o_1(\cdot) + (b_1 - x_1^L)o_2(\cdot) + b_2 x_1(\cdot) + x_1^L x_2(\cdot) - b_1 x_2^U,\\
\mathscr{R}_3(\cdot) &= (x_2^U - x_2^L)o_1(\cdot) + b_1 x_2(\cdot) + x_2^L x_1(\cdot) - b_1 x_2^U,\\
\mathscr{R}_4(\cdot) &= (x_1^U - x_1^L)o_2(\cdot) + b_2 x_1(\cdot) + x_1^L x_2(\cdot) - b_2 x_1^U,\\
\mathscr{R}_5(\cdot) &= (b_2 - x_2^L)o_1(\cdot) + (x_1^U - b_1)o_2(\cdot) + x_2^L x_1(\cdot) + b_1 x_2(\cdot) - b_2 x_1^U.
\end{aligned}
$$

We then construct convex and concave relaxations of these expressions in a manner similar to that for Theorem 3.1. □

Theorem 3.3 *Define $x_1 : Z \subset \mathbb{R}^n \to X_1 \subset \mathbb{R}$ and $x_2 : Z \subset \mathbb{R}^n \to X_2 \subset \mathbb{R}$ with corresponding convex/convex relaxations $x_1^{cv}, x_1^{cc}, x_2^{cv}, x_2^{cc} : Z \to \mathbb{R}$ on Z. Let $u_1, u_2 : Z \subset \mathbb{R}^n \to \mathbb{R}$ be underestimators and $o_1, o_2 : Z \to \mathbb{R}$ be overestimators of x_1, x_2 on Z, respectively. Let $(a_1, a_2), (b_1, b_2) \in X_1 \times X_2 \in \mathbb{IR}^2$ such that $x_1^L \le u_1(\cdot) \le \min\{x_1(\cdot), a_1\}$, $\max\{x_1(\cdot), b_1\} \le o_1(\cdot) \le x_1^U$, $x_2^L \le u_2(\cdot) \le \min\{x_2(\cdot), a_2\}$, and $\max\{x_2(\cdot), b_2\} \le o_2(\cdot) \le x_2^U$. Then, convex and concave relaxations of $w(\cdot) = x_1(\cdot)x_2(\cdot)$ are, respectively, given by*

$$
\begin{aligned}
w^{cv}(\cdot) &= \max\left\{w^{cv}_{\times,0}(\cdot), w^{cv}_{\times,1}(\cdot), w^{cv}_{\times,2}(\cdot), w^L\right\},\\
w^{cc}(\cdot) &= \min\left\{w^{cc}_{\times,0}(\cdot), w^{cc}_{\times,1}(\cdot), w^{cc}_{\times,2}(\cdot), w^U\right\}.
\end{aligned}
$$

Proof This result follows directly from the application of Proposition 2.1, Theorem 3.1, and Theorem 3.2, and basic convexity/concavity properties. □

Similar to the discussion in [19], the relaxations from Theorem 3.1 and Theorem 3.2 reduce to the form given by Proposition 2.1 if $a_i, b_i \in \{x_i^L, x_i^U\}$ for $i \in \{1,2\}$.

Proposition 3.1 *Define $x_1 : Z \subset \mathbb{R}^n \to X_1 \subset \mathbb{R}$ and $x_2 : Z \subset \mathbb{R}^n \to X_2 \subset \mathbb{R}$ with corresponding convex/convex relaxations $x_1^{cv}, x_1^{cc}, x_2^{cv}, x_2^{cc} : Z \to \mathbb{R}$ on Z. Let $u_1, u_2 : Z \subset \mathbb{R}^n \to \mathbb{R}$ be underestimators of x_1 and x_2 on Z, respectively. Let $(a_1, a_2) \in X_1 \times X_2 \in \mathbb{IR}^2$ such that $x_1^L \leq u_1^{cv}(\cdot) \leq u_1(\cdot) \leq \min\{x_1(\cdot), a_1\}$, $x_2^L \leq u_2^{cv}(\cdot) \leq u_2(\cdot) \leq \min\{x_2(\cdot), a_2\}$. Further, suppose that $a_i \in \{x_i^L, x_i^U\}$ for $i \in \{1,2\}$ then the convex relaxations presented in Theorem 3.1 and Theorem 3.2 reduce to $w_{\times,0}^{cv}(\cdot)$.*

Proof We proceed by enumeration of the possible cases to illustrate this result. For brevity, we observe that the relaxations obtained from $w(\cdot) = x_1(\cdot)x_2(\cdot)$ and $w(\cdot) = x_2(\cdot)x_1(\cdot)$ are equivalent. Moreover, $w(\cdot) = x_1(\cdot)x_2(\cdot)$ can be written as $w(\cdot) = (-x_1(\cdot))(-x_2(\cdot))$ and in doing so negates and swaps the positions of the upper and lower interval bounds. Then without loss of generality, we restrict our consideration to two cases: $(a_1, a_2) = (x_1^L, x_2^L)$ and $(a_1, a_2) = (x_1^L, x_2^U)$.

For $(a_1, a_2) = (x_1^L, x_2^L)$, we have $u_i^{cv} = u_i = x_i^L$ which is identical to the $(a_1, a_2) = (x_1^L, x_2^L)$ case of He and Tawarmalani [19, Thm. 1] (Theorem 2.1). For $(a_1, a_2) = (x_1^L, x_2^U)$, we have

$$\mathscr{E}_2(\cdot) = (x_1^U - x_1^L)u_2(\cdot) + x_2^U x_1(\cdot) + x_1^L x_2(\cdot) - x_1^U x_2^U,$$
$$\mathscr{E}_3(\cdot) = x_2^L x_1(\cdot) + x_1^L x_2(\cdot) - x_2^L x_1^L,$$

with $\mathscr{E}_4(\cdot) = \mathscr{E}_2(\cdot)$ and $\mathscr{E}_5(\cdot) = \mathscr{E}_3(\cdot) = \mathscr{E}_1(\cdot)$. Moreover, $(x_1^U - x_1^L)u_2(\cdot) \leq (x_1^U - x_1^L)x_2(\cdot)$ and $\mathscr{E}_2(\cdot) \leq x_1^U x_2(\cdot) + x_2^U x_1(\cdot) - x_1^U x_2^U = \mathscr{E}_6(\cdot)$. □

Clearly, nontrivial lower and upper bounds must be available if relaxations of this form are expected to improve on the McCormick composition approach. In the next section, we propose three computational approaches to achieving this. In the remainder of this section, we detail associated rules for propagating valid subgradients of the convex/concave relaxations that are necessary when forming affine relaxations or as an input to nonsmooth convex NLP solvers.

Definition 3.1 **($\boldsymbol{\omega}, \boldsymbol{\Omega}$)** Let $a \in \mathbb{R}$, $\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2 \in \mathbb{R}^n$. The functions $\boldsymbol{\omega}, \boldsymbol{\Omega} : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ are defined as

$$\boldsymbol{\omega}(a, \boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \equiv \begin{cases} a\boldsymbol{\sigma}_1 & a \geq 0, \\ a\boldsymbol{\sigma}_2 & \text{otherwise,} \end{cases}$$

$$\boldsymbol{\Omega}(a, \boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) \equiv \begin{cases} a\boldsymbol{\sigma}_1 & a \leq 0, \\ a\boldsymbol{\sigma}_2 & \text{otherwise.} \end{cases}$$

Theorem 3.4 *Let $Z \subset \mathbb{R}^n$ be a nonempty convex set and $w, x_1, x_2 : Z \to \mathbb{R}$ such that $w(\cdot) = x_1(\cdot)x_2(\cdot)$ with corresponding convex/convex relaxations $x_1^{cv}, x_1^{cc}, x_2^{cv}, x_2^{cc} : Z \to \mathbb{R}$ on Z. Let $u_1, u_2 : Z \subset \mathbb{R}^n \to \mathbb{R}$ be underestimators and $o_1, o_2 : Z \to \mathbb{R}$ be overestimators of x_1, x_2 on Z, respectively. Let $(a_1, a_2), (b_1, b_2) \in X_1 \times X_2 \in \mathbb{IR}^2$ such that*

$x_1^L \le u_1(\cdot) \le \min\{x_1(\cdot), a_1\}$, $\max\{x_1(\cdot), b_1\} \le o_1(\cdot) \le x_1^U$, $x_2^L \le u_2(\cdot) \le \min\{x_2(\cdot), a_2\}$, *and* $\max\{x_2(\cdot), b_2\} \le o_2(\cdot) \le x_2^U$. *Let* $w_i^{cv}(\cdot)$ *and* $w_i^{cc}(\cdot)$ *on* Z *be defined as in Theorem 3.1 and Theorem 3.2. Then, subgradients* $\mathbf{s}_{w_i}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{w_i}^{cc}(\bar{\mathbf{z}})$ *of* w_i^{cv} *and* w_i^{cc} *on* Z*, evaluated at* $\bar{\mathbf{z}} \in Z$*, for* $i = 1, \ldots, 8$*, are given by*

$$\begin{aligned}
\mathbf{s}_{w_1}^{cv}(\bar{\mathbf{z}}) &= (x_2^U - a_2)\mathbf{s}_{u_1}^{cv}(\bar{\mathbf{z}}) + (x_1^U - a_1)\mathbf{s}_{u_2}^{cv}(\bar{\mathbf{z}}) + \boldsymbol{\omega}\left(a_2, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) \\
&\quad + \boldsymbol{\omega}\left(a_1, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_2}^{cv}(\bar{\mathbf{z}}) &= (x_2^U - x_2^L)\mathbf{s}_{u_1}^{cv}(\bar{\mathbf{z}}) + \boldsymbol{\omega}\left(x_2^L, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) + \boldsymbol{\omega}\left(a_1, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_3}^{cv}(\bar{\mathbf{z}}) &= (x_1^U - x_1^L)\mathbf{s}_{u_2}^{cv}(\bar{\mathbf{z}}) + \boldsymbol{\omega}\left(a_2, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}})\right) + \boldsymbol{\omega}\left(x_1^L, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_4}^{cv}(\bar{\mathbf{z}}) &= (a_2 - x_2^L)\mathbf{s}_{u_1}^{cv}(\bar{\mathbf{z}}) + (a_1 - x_1^L)\mathbf{s}_{u_2}^{cv}(\bar{\mathbf{z}}) + \boldsymbol{\omega}\left(x_2^L, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) \\
&\quad + \boldsymbol{\omega}\left(x_1^L, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_5}^{cv}(\bar{\mathbf{z}}) &= (x_2^L - b_2)\mathbf{s}_{o_1}^{cc}(\bar{\mathbf{z}}) + (x_1^L - b_1)\mathbf{s}_{o_2}^{cc}(\bar{\mathbf{z}}) + \boldsymbol{\omega}\left(b_2, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) \\
&\quad + \boldsymbol{\omega}\left(x_1^U, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_6}^{cv}(\bar{\mathbf{z}}) &= (x_2^L - x_2^U)\mathbf{s}_{o_1}^{cc}(\bar{\mathbf{z}}) + \boldsymbol{\omega}\left(x_2^U, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) + \boldsymbol{\omega}\left(b_1, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_7}^{cv}(\bar{\mathbf{z}}) &= (x_1^L - x_1^U)\mathbf{s}_{o_2}^{cc}(\bar{\mathbf{z}}) + \boldsymbol{\omega}\left(b_2, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) + \boldsymbol{\omega}\left(x_1^U, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_8}^{cv}(\bar{\mathbf{z}}) &= (b_2 - x_2^U)\mathbf{s}_{o_1}^{cc}(\bar{\mathbf{z}}) + (b_1 - x_1^U)\mathbf{s}_{o_2}^{cc}(\bar{\mathbf{z}}) + \boldsymbol{\omega}\left(x_2^U, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) \\
&\quad + \boldsymbol{\omega}\left(x_1^U, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right),
\end{aligned}$$

and

$$\begin{aligned}
\mathbf{s}_{w_1}^{cc}(\bar{\mathbf{z}}) &= (x_2^L - a_2)\mathbf{s}_{u_1}^{cv}(\bar{\mathbf{z}}) + (a_1 - x_1^U)\mathbf{s}_{u_2}^{cv}(\bar{\mathbf{z}}) + \boldsymbol{\Omega}\left(a_2, x_1^{cv}(\bar{\mathbf{z}}), x_1^{cc}(\bar{\mathbf{z}})\right) \\
&\quad + \boldsymbol{\Omega}\left(x_1^U, x_2^{cv}(\bar{\mathbf{z}}), x_2^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_2}^{cc}(\bar{\mathbf{z}}) &= (x_2^L - x_2^U)\mathbf{s}_{u_1}^{cv}(\bar{\mathbf{z}}) + \boldsymbol{\Omega}\left(a_1, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right) + \boldsymbol{\Omega}\left(x_2^U, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_3}^{cc}(\bar{\mathbf{z}}) &= (x_1^L - x_1^U)\mathbf{s}_{u_2}^{cv}(\bar{\mathbf{z}}) + \boldsymbol{\Omega}\left(a_2, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) + \boldsymbol{\Omega}\left(x_1^U, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_4}^{cc}(\bar{\mathbf{z}}) &= (a_2 - x_2^U)\mathbf{s}_{u_1}^{cv}(\bar{\mathbf{z}}) + (x_1^L - a_1)\mathbf{s}_{u_2}^{cv}(\bar{\mathbf{z}}) + \boldsymbol{\Omega}\left(x_2^U, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) \\
&\quad + \boldsymbol{\Omega}\left(a_1, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_5}^{cc}(\bar{\mathbf{z}}) &= (x_2^U - b_2)\mathbf{s}_{o_1}^{cc}(\bar{\mathbf{z}}) + (b_1 - x_1^L)\mathbf{s}_{o_2}^{cc}(\bar{\mathbf{z}}) + \boldsymbol{\Omega}\left(b_2, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) \\
&\quad + \boldsymbol{\Omega}\left(x_1^L, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_6}^{cc}(\bar{\mathbf{z}}) &= (x_2^U - x_2^L)\mathbf{s}_{o_1}^{cc}(\bar{\mathbf{z}}) + \boldsymbol{\Omega}\left(b_1, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right) + \boldsymbol{\Omega}\left(x_2^L, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_7}^{cc}(\bar{\mathbf{z}}) &= (x_1^U - x_1^L)\mathbf{s}_{o_2}^{cc}(\bar{\mathbf{z}}) + \boldsymbol{\Omega}\left(b_2, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) + \boldsymbol{\Omega}\left(x_1^L, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right), \\
\mathbf{s}_{w_8}^{cc}(\bar{\mathbf{z}}) &= (b_2 - x_2^L)\mathbf{s}_{o_1}^{cc}(\bar{\mathbf{z}}) + (x_1^U - b_1)\mathbf{s}_{o_2}^{cc}(\bar{\mathbf{z}}) + \boldsymbol{\Omega}\left(x_2^L, \mathbf{s}_{x_1}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_1}^{cc}(\bar{\mathbf{z}})\right) \\
&\quad + \boldsymbol{\Omega}\left(b_1, \mathbf{s}_{x_2}^{cv}(\bar{\mathbf{z}}), \mathbf{s}_{x_2}^{cc}(\bar{\mathbf{z}})\right),
\end{aligned}$$

where $\mathbf{s}_{x_1}^{cv}$, $\mathbf{s}_{x_1}^{cc}$, $\mathbf{s}_{x_2}^{cv}$, $\mathbf{s}_{x_2}^{cc}$, $\mathbf{s}_{u_1}^{cv}$, $\mathbf{s}_{u_1}^{cc}$, $\mathbf{s}_{u_2}^{cv}$, $\mathbf{s}_{u_2}^{cc}$, $\mathbf{s}_{o_1}^{cv}$, $\mathbf{s}_{o_1}^{cc}$, $\mathbf{s}_{o_2}^{cv}$, $\mathbf{s}_{o_2}^{cc}$ *are, respectively, subgradients of* x_1^{cv}, x_1^{cc}, x_2^{cv}, x_2^{cc}, u_1^{cv}, u_1^{cc}, u_2^{cv}, u_2^{cc}, o_1^{cv}, o_1^{cc}, o_2^{cv}, o_2^{cc} *on* Z*. Further, let* $q_{max} \in$

$\arg\max\{w_1^{cv}(\bar{\mathbf{z}}),\ldots,w_8^{cv}(\bar{\mathbf{z}}),w^U\}$ *and* $q_{min} \in \arg\min\{w_1^{cc}(\bar{\mathbf{z}}),\ldots,w_8^{cc}(\bar{\mathbf{z}}),w^L\}$, *then*

$$\mathbf{s}_w^{cv}(\bar{\mathbf{z}}) = \begin{cases} \mathbf{s}_{w_{q_{min}}}^{cv}(\bar{\mathbf{z}}), & \textit{if } 1 \le q_{min} \le 8, \\ \mathbf{0}, & \textit{otherwise}, \end{cases}$$

and

$$\mathbf{s}_w^{cc}(\bar{\mathbf{z}}) = \begin{cases} \mathbf{s}_{w_{q_{max}}}^{cc}(\bar{\mathbf{z}}), & \textit{if } 1 \le q_{max} \le 8, \\ \mathbf{0}, & \textit{otherwise}. \end{cases}$$

Proof The proof follows from the construction of the relaxations in Theorem 3.3. The functions defined in Definition 3.1 select subgradients that respect the rules for scalar multiplication of relaxations [32]. For each equation (21)-(28), the subgradients are then summed using the standard additive relationship [32,21]. □

4 Computability of Tight Composite Relaxations of Bilinear Terms

In this section, we describe three approaches to compute the requisite *a priori* relaxations in a reduced-space McCormick relaxation context.

4.1 Composite Convex/Concave Relaxations based on Under/Overestimators

For low-dimensional expressions (i.e., $n \sim 1$), we may exploit the properties of convex/concave functions. Convex/concave relaxations of the arguments of the bilinear expression may be used as the valid *a priori* relaxations. As illustrated in a subsequent example, if the a_1, a_2, b_1, b_2 values are selected judiciously, then this may lead to nontrivial affine relaxations and, in turn, improved relaxations of the bilinear term in reduced-space. The constants a_1 and a_2 can be selected by maximizing convex functions $x_1^{cv}(\cdot)$ and $x_2^{cv}(\cdot)$ on a convex polyhedron $\mathscr{P} = \text{conv}(v_1, v_2, \ldots, v_k)$, with v_i vertices. It is well-known that the extremal value will be achieved at a vertex, i.e., $\max_{\mathbf{z}\in\mathscr{P}} f(\mathbf{z}) = \max_i f(v_i)$. Similarly, we may compute b_1, b_2 by minimizing the concave functions $x_1^{cc}(\cdot)$ and $x_2^{cc}(\cdot)$. This is in itself a series of nonconvex optimization problems. While specialized algorithms may exist to address this class of problems (e.g., [15,4]), it is reasonable to conclude that this approach is too computationally expensive to be practical, as two nonsmooth concave optimization problems must be solved for each intermediate bilinear term in order to evaluate relaxations of the nonlinear function.

In the case of low-dimensional expressions, we may simply compute convex/-concave relaxations at each vertex and solve each optimization problem via enumeration. In many cases, the evaluation of relaxations is often much less time intensive than other routines, such as solving a series of linear programs in optimization-based bounds tightening [50], or solving a nonlinear program in order to furnish valid upper bounds within a branch-and-bound routine for deterministic global optimization [22]. One may intuit that this method will yield a tighter relaxation of the bilinear term than using a weaker *a priori* relaxation. As it turns out, this is in fact false, and the use of a weaker *a priori* relaxation may lead to tighter relaxations of the bilinear term owing to the dependence of the relaxations of Theorem 3.3 on a_1, a_2, b_1, and b_2. A counterexample is provided in Example 4.1 and illustrated in Figure 1. In the subsequent section, a less computationally expensive method is developed that may provide comparably tight bounds.

4.2 Improved Relaxations Using Subgradient-Based Under/Overestimators

In addition to the composite bilinear relaxation theory outlined herein, the use of *a priori* relaxations to refine the relaxations of a univariate factor can be accomplished via Proposition 4.1.

Proposition 4.1 *Let $v_k : Z \subset \mathbb{R}^n \to V$ be a cumulative mapping. Let $v^{cv}_{k,a}/v^{cc}_{k,a}$ be convex/concave relaxations of v_k on Z, and suppose we have additional convex/-concave relaxations $v^{cv}_{k,b}/v^{cc}_{k,b}$ of v_k on Z. Then,*

$$v^{cv}_k(\cdot) := \max\left\{v^{cv}_{k,a}(\cdot), v^{cv}_{k,b}(\cdot)\right\},$$
$$v^{cc}_k(\cdot) := \min\left\{v^{cc}_{k,a}(\cdot), v^{cc}_{k,b}(\cdot)\right\},$$

are convex and concave relaxations of v_k on Z, respectively.

The subgradients associated with Proposition 4.1 are then simply the subgradients of the argument returned. In Proposition 4.2, we note that, provided with convex/-concave relaxations of a factor at a particular point $\bar{\mathbf{z}} \in Z$ along with associated subgradients, then new affine relaxations may be derived.

Proposition 4.2 *Let $v_k : Z \subset \mathbb{R}^n \to V$ be a cumulative mapping. Let v^{cv}_k/v^{cc}_k be convex/concave relaxations of v_k on Z and their respective subgradients $\mathbf{s}^{cv}_{v_k}(\bar{\mathbf{z}})$, $\mathbf{s}^{cc}_{v_k}(\bar{\mathbf{z}})$ computed at $\mathbf{z} = \bar{\mathbf{z}} \in Z$. The functions $\xi, \zeta : Z \to \mathbb{R}$ are the affine relaxations of the convex and concave relaxations of v_k on Z, respectively:*

$$\xi(\mathbf{z}) \equiv v^{cv}_k(\bar{\mathbf{z}}) + \mathbf{s}^{cv}_{v_k}(\bar{\mathbf{z}})^{\mathrm{T}}(\mathbf{z} - \bar{\mathbf{z}}), \tag{29}$$
$$\zeta(\mathbf{z}) \equiv v^{cc}_k(\bar{\mathbf{z}}) + \mathbf{s}^{cc}_{v_k}(\bar{\mathbf{z}})^{\mathrm{T}}(\mathbf{z} - \bar{\mathbf{z}}). \tag{30}$$

As noted in [44], interval extensions of (29) and (30) can be used to derive valid upper bounds of the factor and subsequently refine the associated interval bounds. We propose using this result as follows. Using *a priori* relaxations, refined relaxations of composite factors, and their subgradients, are calculated via Proposition 4.1. These are then used to construct affine relaxations via Proposition 4.2. Natural interval extensions of (29) and (30) are then used to define valid a_1, a_2, b_1, b_2 terms for these affine relaxations, which are then used for the direct application of Theorem 3.3 for improved relaxations of composite bilinear terms.

The following numerical example is provided to illustrate the results of applying Theorem 3.3 and Theorem 3.4 using the methods of Section 4.1 and Section 4.2 versus the previously established McCormick-based approaches.

Example 4.1 Consider the function $f : Z \to \mathbb{R}$, with $Z = [-0.5, 1]$, defined as

$$f(z) = \left(z - z^2\right)\left(z^3 - \exp(z)\right).$$

As illustrated in Figure 1, *a priori* affine relaxations constructed at a single reference point $\bar{z} = 0.25$ yield similar relaxations to the direct enumeration approach. Note that in this example, neither approach yields relaxations that are a strict improvement over the other for the entire domain.

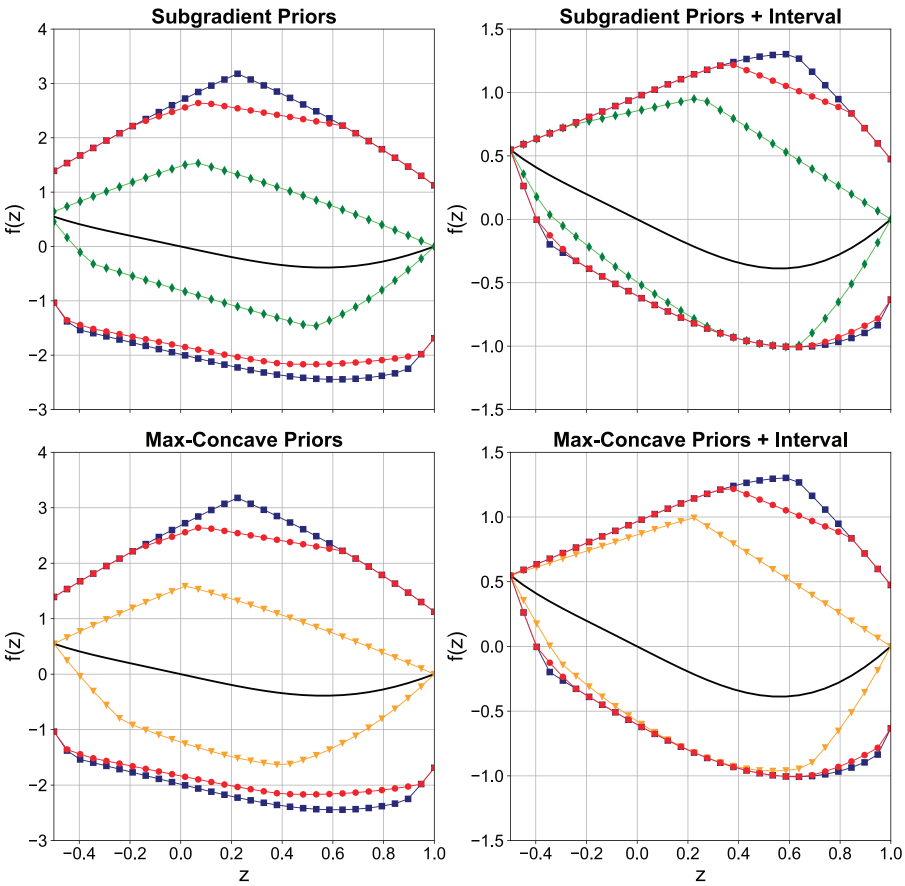

Fig. 1 Relaxations of $f(z) = (z - z^2)(z^3 - \exp(z))$ (——) on $Z = [-0.5, 1]$, are constructed using existing approaches and compared with approaches developed in this paper. Relaxations computed using *a priori* subgradients at $\bar{z} = 0.25$ (◆) lead to tighter relaxations than the use of (■) standard and (●) multivariate McCormick relaxation strategies. This occurs when (**top-left**) subgradients are only used as *a priori* relaxations and (**top-right**) when the subgradients are used to refine the interval bounds of each factor [44]. (**bottom-left**) The *a priori* relaxations constructed by computing the maxima and minima of the operands' relaxations (▼) also lead to an improvement over standard and multivariate McCormick relaxations. (**bottom-right**) Marginal additional improvement over standard and multivariate McCormick relaxations may be observed when using the subgradient method to refine interval bounds of each factor [44].

4.3 Affine Arithmetic

Affine arithmetic has been proposed as an alternative set-valued arithmetic to interval arithmetic. In this approach, an affine representation of a function is constructed. In the case of affine functions, this representation is exact. For nonlinear terms, the enclosure is linearized and some overestimation necessarily occurs. Two common choices of linearization techniques include minimizing the range of the enclosure or minimizing the maximum width of enclosure, which is the Chebyshev approximation [53]. In the original description of affine arithmetic [12] and the later introductory papers [64, 17], computations began with an affine representation of each term and an additional noise term was added for every nonlinear term introduced. This approach introduced significant computational complexity. In this work, we will instead address the use of two simplified forms of affine arithmetic that were proposed by [28]. In each of these forms, the intermediate term $v_k(\mathbf{z}) \in \tilde{v}_0 + \sum_{i=1}^{n} \tilde{v}_i \mathbf{z} + R_k$ is represented by a linear function with a small remainder R_k that encloses truncation error [45]. The first affine form **AF1** uses a single component to represent the R_k truncation error.

Definition 4.1 **(AF1)** The affine form AF1 is defined as

$$\hat{v} = v_0 + \sum_{i=1}^{n} v_i \epsilon_i + v_{n+1} \epsilon_{\pm},$$

where $\epsilon_i \in [-1,1]$, $v_i \in \mathbb{R}$ for $i = 1, \ldots n+1$, $\epsilon_{\pm} \in [-1,1]$, and $v_0 \in \mathbb{R}$

In addition to this form, [28] discussed the use of a second affine form **AF2** that uses separate components to represent positive truncation error, negative truncation error, and mixed-sign truncation error. This distinction leads to tighter enclosures of certain operators.

Definition 4.2 **(AF2)** The affine form AF2 is defined as

$$\hat{v} = v_0 + \sum_{i=1}^{n} v_i \epsilon_i + v_{n+1} \epsilon_{\pm} + v_{n+2} \epsilon_{+} + v_{n+3} \epsilon_{-},$$

where $v_i \in \mathbb{R}$ for $i = 1, \ldots n+3$, $\epsilon_i \in [-1,1]$ for $i = 1, \ldots n$, $\epsilon_{\pm} \in [-1,1]$, $\epsilon_{+} \in [0,1]$, $\epsilon_{-} \in [-1,0]$, and $v_0 \in \mathbb{R}$

Each affine form definition implies the existence of affine relaxations as described by [46]. We proceed to state the corresponding relaxations in Proposition 4.3 (adapted from [46, Prop. 2]) and Proposition 4.4 (adapted from [46, Prop. 3]). Next, note that components ϵ_i may be expressed as simple nondimensionalized forms of the decision variables as $\epsilon_i(z_i)$, and can be converted to the dimensional form using the following equation:

$$\epsilon_i(z_i) = (z_i - \text{mid}(Z_i))/\text{rad}(Z_i).$$

Proposition 4.3 (Affine Relaxation from AF1) *Let $v : Z \to V$ be a factor with an affine form as in Definition 4.1. Then $v^{cv}_{AF1}, v^{cc}_{AF1} : Z \to V$ defined as*

$$v^{cv}_{AF1}(\mathbf{z}) = v_0 + \sum_{i=1}^{n} v_i \epsilon_i(z_i) - v_{n+1}, \tag{31}$$

$$v^{cc}_{AF1}(\mathbf{z}) = v_0 + \sum_{i=1}^{n} v_i \epsilon_i(z_i) + v_{n+1}, \tag{32}$$

are affine relaxations of v at $\mathbf{z} \in Z$.

Proposition 4.4 (Affine Relaxation from AF2) *Let $v : Z \to V$ be a factor with an affine form as in Definition 4.2. Then $v^{cv}_{AF2}, v^{cc}_{AF2} : Z \to V$ defined as*

$$v^{cv}_{AF2}(\mathbf{z}) = v_0 + \sum_{i=1}^{n} v_i \epsilon_i(z_i) - v_{n+1} - v_{n+2} - v_{n+3}, \tag{33}$$

$$v^{cc}_{AF2}(\mathbf{z}) = v_0 + \sum_{i=1}^{n} v_i \epsilon_i(z_i) + v_{n+1} + v_{n+2} + v_{n+3}, \tag{34}$$

are affine relaxations of v at $\mathbf{z} \in Z$.

The extrema of (31)-(34) on Z can be readily computed via natural interval extensions. We note that this operation is no more complicated than converting the affine representation to an interval form. As illustrated in Example 4.2, the use of *a priori* information propagated through affine forms may yield tighter relaxations than simply using interval bounds calculated via affine arithmetic.

Example 4.2 Consider the function $f : X \times Y \to \mathbb{R}$ defined as

$$z = f(x, y) = (x^2 - x)(y^2 - y),$$

on the domain $X \times Y = [0.1, 1.9]^2$. We compute convex and concave relaxations of this function using four distinct approaches: 1) standard McCormick arithmetic, 2) affine arithmetic of style AF1, 3) the tightest relaxations available using AF1 or standard multiplication rules, and 4) relaxations generated using approach 3), where the relaxations of the bilinear operator are computed using Theorem 3.1 through Theorem 3.3. The results are illustrated in Figure 2. It is clear that the relaxations computed using approach 3) outperform both approaches 1) and 2), and 4) further tightens the relaxations obtained by approach 3).

5 Benchmark Results

All numerical experiments in this work were run on a single thread of an Intel Xeon E3-1270 v5 3.60/4.00GHz (base/turbo) processor with 16GB ECC RAM allocated to a virtual machine running the Ubuntu 18.04LTS operating system and Julia v1.6 [5]. Absolute and relative convergence tolerances for the B&B algorithm of 10^{-4} were specified for all example problems, unless otherwise noted. EAGO.jl v0.7.0

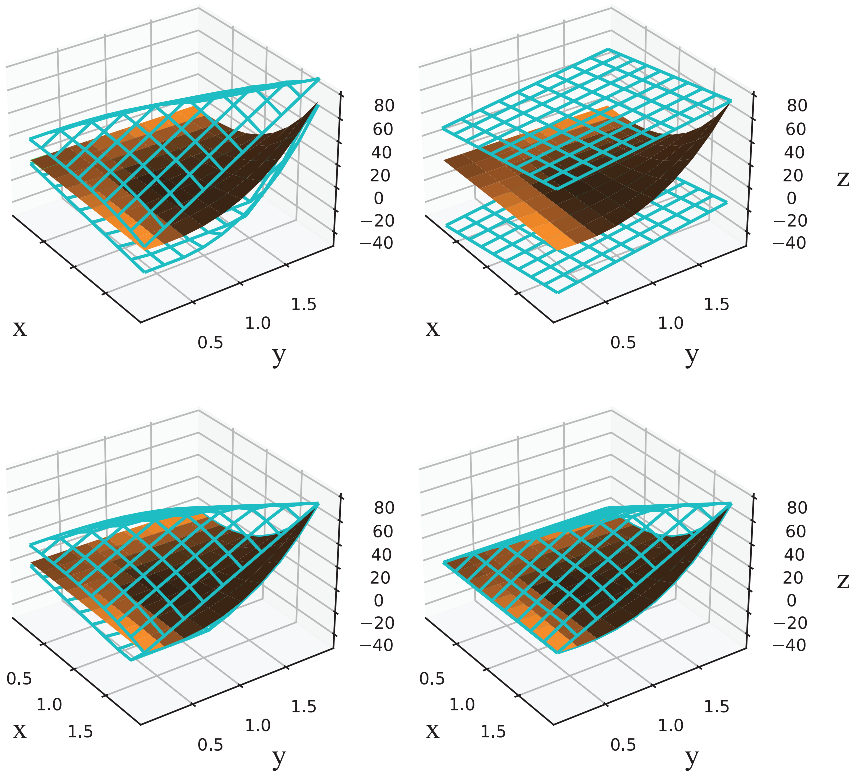

Fig. 2 The function $z = (x^2 - x)(y^2 - y)$ (■) is plotted along with its corresponding convex and concave relaxations (——) on $X \times Y = [0.1, 1.9]$. (**top-left**) A standard McCormick relaxation approach is contrasted with (**top-right**) relaxations constructed by affine arithmetic using AF1. (**bottom-left**) The use of composite relaxations formed by intersecting affine enclosures with standard McCormick relaxations is compared to (**bottom-right**) the relaxations constructed by the use of *a priori* bounds per Theorem 3.1 through Theorem 3.3 to tighten composite relaxations. We note that AF2 did not yield appreciably different results from AF1, so it was omitted from the plot.

[74] was used to solve each optimization problem. Relaxations of intrinsic functions have been implemented in the openly-available McCormick.jl [72] subpackage of EAGO.jl. EAGO's approach to solve lower-bounding problems is to construct linear programming relaxations of the corresponding nonsmooth relaxed problems, which are then solved using the GLPK solver, by default. BARON v21.1.13 [66, 55] was used for performance comparisons. The Intel MKL (2019 Update 2) [16] was used to perform all LAPACK [1,69] and BLAS [6] routines. The data used with and generated from the following numerical examples are openly available in the following Git repository `https://github.com/PSORLab/RSBilinear` along with the corresponding problem formulations. For comparison purposes, three configurations were explored: **EAGO** the standard configuration for the EAGO optimizer; **EAGO Sub** denotes the use of subgradient-based *a priori* relaxations; and **EAGO**

Aff denotes the use of AF1 affine-arithmetic and associated *a priori* relaxations. Interval bounds implied by interval extensions using the subgradient as well as interval bounds computed using affine arithmetic are used to refine interval bounds of each factor computed. The enumerative approach detailed in Section 4.1 is explicitly excluded from consideration as it is not a practical approach for even moderately sized problems due to the high computational complexity. A randomly generated benchmark library adapted from He and Tawarmalani [19, Ex. 5] was used to assess the performance of each approach. Each instance takes the form given by

$$\begin{aligned} \min_{\mathbf{x}} \ & \mathbf{c}^{\mathrm{T}}\mathbf{x} + \sum_i \sum_j q_{ij} y_{ij} \\ \text{s.t.} \ & \mathbf{x} \in [-1,1]^n \\ & \mathbf{u} = (x_1^2 - x_1, x_1^3 - x_1, x_1^4 - x_1, \ldots, x_n^2 - x_n, x_n^3 - x_n, x_n^4 - x_n) \\ & \mathbf{Y} = \mathbf{u}\mathbf{u}^{\mathrm{T}}, \end{aligned}$$

where q_{ij} are elements of $\mathbf{Q} \in \mathbb{R}^{n\times n}$; a strictly upper-triangular matrix with density of nonzero elements given by χ. A set of 200 instances was randomly generated with $\chi \in [0.3, 0.5, 0.7]$, $n \in [10, 15, 20]$ and $\mathbf{c} \in [-512, -2]^n$ randomly selected from a uniform distribution. Each instance was then solved for each solver configuration. A 5-minute (300 s) CPU time limit was enforced for each instance. Solver performance was assessed using the shifted geometric mean time. A performance profile was generated for comparison using the methodology of Dolan and Moré [14]. The *performance* of a solver configuration s is set to the solution time $t_{p,s}$ in CPU seconds (single-threaded) for problem p. The *performance ratio* of problem p by solver s is then the ratio of solver s performance to the best solver performance in the set:

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : s \in S\}}.$$

This *performance profile* of solver s on a benchmark set depicts the cumulative distribution function of the performance metric $\rho_s(\tau)$; which is the probability that a performance ratio $r_{p,s}$ is within $\tau \in \mathbb{R}$ of the best possible ratio

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

where $\mathscr{I}$ is the set of problems with $n_p = \text{card}(\mathscr{I})$. A plot comparing r_s for each configuration $s \in S$ then illustrates the relative performance of each solver.

First, we note that based on the performance data available in Table 1 and Table 2 and the profiles in Figure 3, all configurations of EAGO significantly underperform BARON in this benchmark. This result is to be expected as the benchmark set is a polynomial optimization problem that may be reformulated as a higher-dimensional nonconvex quadratic program. Provided that this reformulation occurs, BARON may implement any number of specialized approaches [2, 3, 77, 48, 47] that currently have no analog for reduced-space optimization approaches. However, as discussed previously, these approaches cannot be readily applied to reduced-

Solver Configuration	Solved	Unsolved
BARON	200 (100.0%)	0 (0.0%)
EAGO	136 (68.0%)	64 (32.0%)
EAGO Aff	127 (63.5%)	73 (36.5%)
EAGO Sub	183 (91.5%)	17 (8.5%)

Table 1 The number of benchmark problems solved within 5 minutes by solver configuration are tabulated. BARON readily solves all problems likely due to it efficiency in decomposing the problem to a higher-dimensional quadratic form and then applying specialized methods that EAGO currently lacks. However, **EAGO Sub** substantially increases the number of problems solved within the time limit relative to **EAGO** while the use of affine arithmetic **EAGO Aff** is associated with decreased performance.

Solver Configuration	τ	δ_{rel}
BARON	0.69	N/A
EAGO	40.7	3.8×10^{-2}
EAGO Aff	52.4	4.6×10^{-2}
EAGO Sub	13.7	1.2×10^{-2}

Table 2 The shifted geometric mean τ of solve times $t_1, t_2, \ldots, t_n$ defined by $\tau = (\prod_{i=1}^{n}(t_i + s))^{1/n} - s$ are given by solver configuration with $s = 1$ along with the average relative gap remaining for any problems not solved within the 5-minute time limit. For problems that terminate due to the specified time limit, the relative gap remaining can be compared to assess solver performance. The relative gap remaining is given by $\delta_{rel} = (|U| - |L|)/\max(|U|, |L|)$ where $|U|$ is the upper bound (best feasible objective value) and $|L|$ is the lower bound. **EAGO Sub** substantially reduces τ and δ_{rel} relative to **EAGO**, whereas **EAGO Aff** shows an increase in these metrics over **EAGO**.

space applications for which the problem does not have a factorable representation.

Next, we observe that the configuration **EAGO Sub** reduces the shifted geometric mean run time relative to **EAGO** by a factor of 3, increasing the number of problems solved within 5 minutes by 23.5%, as indicated by the data in Table 1 and Table 2. Interestingly, configuration **EAGO Aff** apparently increases the mean solve time relative to the standard configuration **EAGO** as the increased time spent performing affine arithmetic calculations offsets any potential benefit from reducing overestimation that occurs in the relaxed problem. As subgradients are already calculated when computing relaxations in configuration **EAGO**, the use of subgradients to tighten the composite relaxation of the bilinear term in configuration **EAGO Sub** does not substantially increase calculation time.

6 Case Studies

We now examine how the proposed relaxations may be applied to two separate case studies. In the first, bilinear compositions are introduced through the sequential use of response surface models. In the second case, bilinear terms appear in

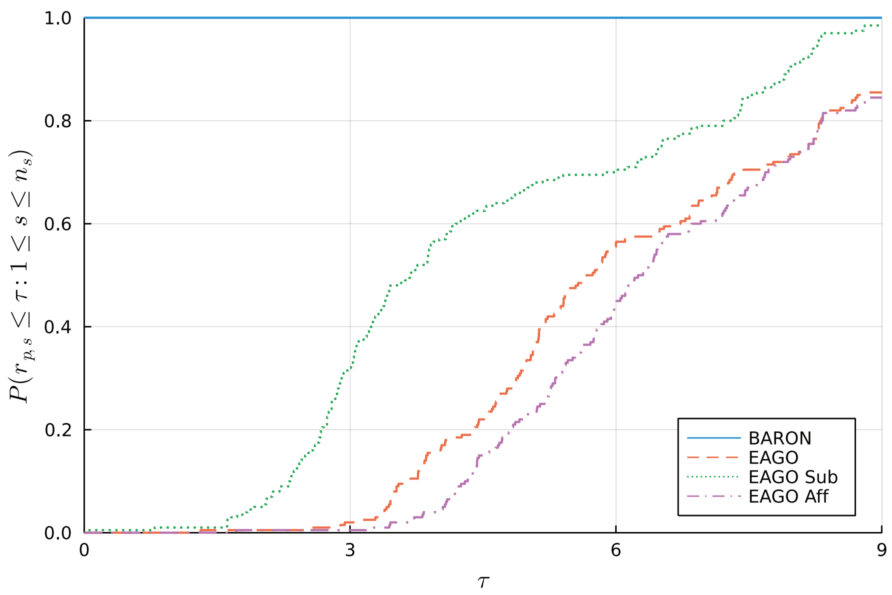

Fig. 3 As illustrated by the performance profiles shown for each solver configuration, computing relaxations using the *a priori* relaxation based on subgradients leads to a substantial decrease in CPU solution time for a typical problem within the benchmark set when compared the naïve McCormick approach implemented in EAGO. The horizontal line plot for BARON indicates that it uniformly provides substantially faster run times due to it decomposing the benchmark problems into higher-dimensional quadratic forms and applying specialized methods.

the governing equations of a dynamic optimization problem, which are repeatedly composed at each time step.

6.1 Process Optimization Through Sequential Response Surface Methodology

Response surface methodology (RSM) is one of the most common statistical approaches used in industrial applications for creating predictive models of complicated processes [13]. In RSM, quadratic models of the form:

$$y = a + \mathbf{c}^{\mathrm{T}}\mathbf{x} + \tfrac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x},$$

are used to approximate the behavior of a system (i.e., the response variable) as a function of the input or dependent variables. These models are particularly appealing when first-principles process models are highly complex but variation in process outputs with respect to control variables behave in a predictable manner. Applications of RSMs span a wide range with examples including machining via abrasive waterjet turning [76], Nd:YAG laser drilling [52], electron beam welding [35] plasma spray coating [49], and diffusion bonding of alloys [51]. While these models are ubiquitous, the use of these models in optimal design problems readily leads to nonconvex problem formulations due to the presence of bilinear terms.

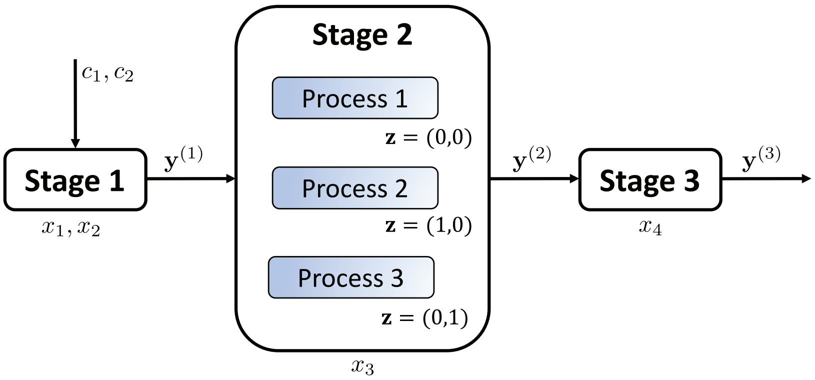

Fig. 4 An illustration of the three-stage machining process considered in Section 6.1, consisting of (Stage 1) initial shaft machining process, (Stage 2) followed by one of three different roughing processes, and then (Stage 3) processing in a final surface finishing step. The parameters c_1, c_2 represent the initial diameter and roundness, respectively. The feed rate x_1, cut depth x_2, rougher rate x_3, and finish rate x_4 comprise the continuous decision variable vector $\mathbf{x}$, whereas the binary decision variable vector $\mathbf{z}$ specifies the machine type (i.e., Process 1 through 3). The specification of these variables determines the output diameter $y_1^{(1)}$, output roundness $y_2^{(1)}$, and process time $y_3^{(1)}$ of Stage 1; the output diameter $y_1^{(2)}$ and process time $y_2^{(2)}$ of Stage 2; and the final diameter $y_1^{(3)}$ and process time $y_2^{(3)}$ of Stage 3.

One particularly interesting area of application for RSMs lies in the quality chain design of multistage manufacturing systems [20]. Here we revisit the numerical example that was previously addressed in [20] using local and stochastic methods. We will show that this model form may be readily addressed using reduced-space global optimization and solved to a certificate of ϵ-global optimality using a simplified set of RSM models generated from the original data provided in [20].

The manufacturing process is illustrated in Figure 4 and consists of an initial shaft machining (Stage 1) in which the output diameter $y_1^{(1)}$, output roundness $y_2^{(1)}$, and process time $y_3^{(1)}$ are determined by feed rate x_1, cut depth x_2, initial diameter c_1, and roundness c_2 of the shaft bought from a supplier. This step is followed by a rough machining process (Stage 2) wherein rougher rate x_3, the part diameter, and machine type specified by binary decision variables $\mathbf{z} = (z_1, z_2) \in Z = \{0,1\}^2$ determine the output diameter $y_1^{(2)}$ and process time $y_2^{(2)}$. The process concludes with finish machining (Stage 3) in which finish rate x_4 is adjusted to determine final diameter $y_1^{(3)}$ and process time $y_2^{(3)}$. Machine operating parameters and a nominal roundness specification may then be varied to specify the process, forming the continuous decision variable vector $\mathbf{x} = (x_1, x_2, x_3, x_4)$. Each process step relates inputs and operating parameters to outputs by means of the RSM, given by

$$\begin{aligned}\hat{y}_1^{(1)}(\mathbf{x}) &= 3.55 + 0.27c_1 + 0.58c_2 + 60.6x_2 - 2.8c_1x_2 - 2.3c_2x_2\\ \hat{y}_1^{(2)}(\mathbf{x},\mathbf{z}) &= 126586.5 - 21466.8\hat{y}_1^{(1)}(\mathbf{x}) + 520.43x_3 + 56.29z_1 + 315.95z_2,\\ &\quad - 43.72x_3\hat{y}_1^{(1)}(\mathbf{x}) + 3.74x_3^2 + 910.1\hat{y}_1^{(1)}(\mathbf{x})^2\\ &\quad - 30.6\hat{y}_1^{(1)}(\mathbf{x})z_1 - 173.17\hat{y}_1^{(1)}(\mathbf{x})z_2,\\ \hat{y}_1^{(3)}(\mathbf{x},\mathbf{z}) &= 9.16 + 0.092x_3 + 0.73x_4 + 0.64x_3x_4 - 0.49x_4^2 - 0.13x_4\hat{y}_1^{(2)}(\mathbf{x},\mathbf{z})\\ &\quad + 0.0019\hat{y}_1^{(2)}(\mathbf{x},\mathbf{z})^2 + 0.018\hat{y}_1^{(2)}(\mathbf{x},\mathbf{z}),\end{aligned}$$

where the hat on the y variables signify the RSM surrogate that predicts the value of the corresponding process output (i.e., $\hat{y}_1^{(1)}$ is the surrogate for predicting $y_1^{(1)}$).

The original work [20] was principally concerned with minimizing the variation in $\mathbf{y}^{(3)}$. We instead consider the problem of identifying a nominal process in which the value of $y_1^{(3)}$ is close to the desired value of $\eta = 5$. This is motivated by balancing a maximal process capability index (CpK) with respect to the final diameter setpoint and the desire for just-in-time completion of the process. As such, the design decision may be posed as the following reduced-space optimization problem:

$$\begin{aligned} f^* = \min_{\mathbf{x}\in X, \mathbf{z}\in Z} & \left(\hat{y}_1^{(3)}(\mathbf{x},\mathbf{z}) - \eta\right)^2\\ \text{s.t.}\ \ & z_1 + z_2 \leq 1\\ & X = [7,10] \times [0.1, 0.3] \times [-1.078, 1.078] \times [-1.078, 1.078]\\ & c_1 = 0.001,\ \ c_2 = 0.03\end{aligned}$$

We then proceed to solve this problem in EAGO as well as using relaxation-based *a priori* relaxations for nonlinear terms **EAGO Relax**, subgradient-based *a priori* relaxations **EAGO Sub**, and an affine arithmetic *a priori* relaxation **EAGO Aff**. A solve time of 26.3 seconds is required for **EAGO** to furnish a solution, while a longer solve time of 47.6 seconds is required for the **EAGO Aff** method. The relaxation-based *a priori* relaxations for nonlinear terms **EAGO Relax** leads to a slightly faster run time of 25.5 seconds, while **EAGO Sub** yields a significantly faster run time of 7.1 seconds.

6.2 Kinetic Parameter Estimation

The *a priori* relaxation methods presented here can readily be applied to dynamic optimization problems as well. We demonstrate this using an adaption of a kinetic parameter estimation problem [63,32]. The reaction mechanism is described by the ordinary differential equation initial value problem (ODE-IVP):

$$\frac{dx_A}{dt} = k_1 x_Z x_Y - c_{O_2}(k_{2f} + k_{3f})x_A + \frac{k_{2f}}{K_2}x_D + \frac{k_{3f}}{K_3}x_B - k_5 x_A^2,$$
$$\frac{dx_B}{dt} = c_{O_2}k_{3f}x_A - \left(\frac{k_{3f}}{K_3} + k_4\right)x_B, \quad \frac{dx_Z}{dt} = -k_1 x_Z x_Y,$$
$$\frac{dx_D}{dt} = c_{O_2}k_{2f}x_A - \frac{k_{2f}}{K_2}x_D, \quad \frac{dx_Y}{dt} = -k_{1s}x_Z x_Y,$$
$$x_A(0) = 0, x_B(0) = 0, x_D(0) = 0, x_Y(0) = 0.4, x_Z(0) = 140,$$

where x_j is the concentration of species $j \in \{A, B, D, Y, Z\}$ and the constants are given by $T = 273$, $K_2 = 46\exp(6500/T - 18)$, $K_3 = 2K_2$, $k_1 = 53$, $k_{1s} = k_1 \times 10^{-6}$, $k_5 = 1.2 \times 10^{-3}$, and $c_{O_2} = 2 \times 10^{-3}$. A least-squares fit is sought to fit available intensity and time data that exhibit a known dependency on the concentration, that is, $I = x_A + \frac{2}{21}x_B + \frac{2}{21}x_D$ [63]. The reaction rate constants $k_{2f} \in [10, 1200]$, $k_{3f} \in [10, 1200]$, and $k_4 \in [0.001, 40]$ are the decision variables $\mathbf{p} = (k_{2f}, k_{3f}, k_4)$.

We consider an explicit Euler discretization of the problem [32] for simplicity. A semi-explicit approach is used in which the relaxations of state variables $\mathbf{x}$ in the ODE-IVP are computed. A discretization consisting of 50 steps is sufficient for a high degree of accuracy for this problem on the time domain $t \in [0, 0.5]$. We note that a smaller time domain than the original $t \in [0, 2.0]$ is considered to ensure convergence to the 10^{-4} tolerance for each method investigated. The discretized model becomes:

$$x_A^{i+1} = x_A^i + \Delta t\left(k_1 x_Y^i x_Z^i - c_{O_2}(k_{2f} + k_{3f})x_A^i + \frac{k_{2f}}{K_2}x_D^i + \frac{k_{3f}}{K_3}x_B^i - k_5 (x_A^i)^2\right),$$
$$x_B^{i+1} = x_B^i + \Delta t\left(k_{3f}c_{O_2}x_A^i - \left(\frac{k_{3f}}{K_3} + k_4\right)x_B^i\right),$$
$$x_D^{i+1} = x_D^i + \Delta t\left(k_{2f}c_{O_2}x_A^i - \frac{k_{2f}}{K_2}x_D^i\right),$$
$$x_Y^{i+1} = x_Y^i + \Delta t\left(-k_{1s}x_Y^i x_Z^i\right),$$
$$x_Z^{i+1} = x_Z^i + \Delta t\left(-k_{1s}x_Y^i x_Z^i\right),$$

where $i = 0, \ldots, 49$ and $\Delta t = 0.01$. The objective function is then given by

$$f(\mathbf{p}) = \sum_{i=1}^{n}\left(I_i^c(\mathbf{p}) - I_i^d\right)^2,$$

where I_i^c is the calculated intensity value at time step i from the model and I_i^d is the experimental measurement. We solve this problem with the standard configuration **EAGO** as well as using subgradient-based *a priori* relaxations with configuration **EAGO Sub**. Note that configuration **EAGO Relax** was not considered for this example due to the problem complexity, similarly to Section 5, and configuration **EAGO Aff** was not used as it consistently underperformed **EAGO** in all previously presented examples. Configuration **EAGO** solves this problem in 450.0

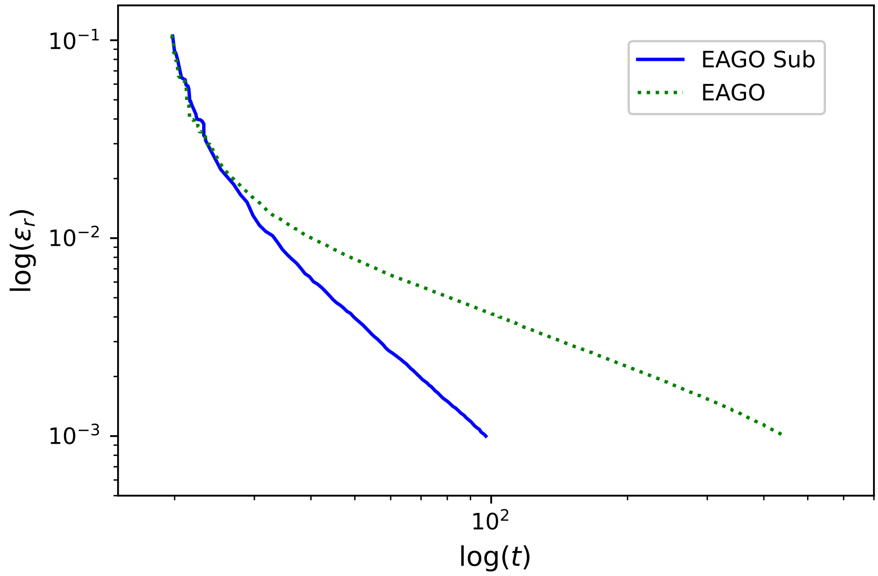

Fig. 5 A log-log plot of relative gap remaining $\epsilon_r = (UBD - LBD)/\max(UBD, LBD)$ by solver configuration at time t for the kinetic parameter estimation problem in Section 6.2. Solver configuration **EAGO Sub** accelerates the rate of convergence to an optimal solution by a factor of 4.76.

seconds and 121,881 iterations whereas configuration **EAGO Sub** solves this problem in only 94.5 seconds after 8,905 iterations, as illustrated by the relative gap convergence plot provided in Figure 5. This illustrates how dynamic optimization problem structures that introduce a large number of composite bilinear terms may benefit substantially from the use of the *a priori* relaxation approach presented herein.

7 Conclusions

New theory was developed for computing improved relaxations of composite bilinear terms when relaxations of *a priori* underestimators and overestimators are available. A corresponding result for computing subgradient information was also detailed herein. Three distinct methods by which the new results may be used within a generalized McCormick relaxation framework were also described: an enumeration approach with standard McCormick relaxations; the use of intermediate affine relaxations defined by subgradient expansions; and the use of affine relaxations defined by an affine arithmetic. Two case studies were presented that illustrate how each method may lead to significantly improved relaxations despite requiring an approximately 4-fold increase in floating point operations for each bilinear term relaxed. Lastly, each method was incorporated into a version of the EAGO global optimizer and the relative performance of each approach was demonstrated on a small benchmarking test set. In this benchmark set, the subgradient expansion reduced the computational time by a factor of three compared to the standard approach and the other methods presented herein.

Since the improved relaxations developed herein are based upon the McCormick theory, they can be readily incorporated within other McCormick-based relaxation methods. First, they may be incorporated into the reverse McCormick relaxations [71] to tighten relaxations of expressions involving division operators. Direct extensions to the relaxations of multilinear terms may also be made by expanding products encountered in derivations of the envelope; then, deriving relaxations of the resulting under/over estimators in a manner that parallels the derivation detailed in Theorem 3.1 through Theorem 3.3. Alternatively, a recursive method for generating tighter relaxations of the multilinear term could be implemented using the improved composite bilinear relaxation defined herein. Lastly, tighter relaxations of min/max/mid operators derived using under/overestimators may be of interest given the significance of these operators in nonsmooth formulations of refrigeration and heat integration models [70].

Acknowledgements This material is based upon work supported by the National Science Foundation under Grant No. 1932723. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. We also gratefully acknowledge the Air Force Research Laboratory, Materials and Manufacturing Directorate (AFRL/RXMS) for support via Contract No. FA8650-20-C-5206. The views, opinions, and/or findings contained in this paper are those of the authors and should not be interpreted as representing the official views or policies, either expressed or implied, of the Air Force Research Laboratory, the United States Air Force, or the Department of Defense.

Data Availability

The datasets generated during and/or analysed during the current study are available in a GitHub repository: `https://github.com/PSORLab/RSBilinear`.

References

1. Anderson, E., Bai, Z., Bischof, C., Blackford, L.S., Demmel, J., Dongarra, J., Croz, J.D., Greenbaum, A., Hammarling, S., McKenney, A., Sorensen, D.: LAPACK Users' Guide. Society for Industrial and Applied Mathematics, Philadelphia, PA (1999). doi:10.1137/1.9780898719604
2. Bao, X., Sahinidis, N.V., Tawarmalani, M.: Multiterm polyhedral relaxations for nonconvex, quadratically constrained quadratic programs. Optimization Methods & Software **24**(4-5), 485–504 (2009). doi:10.1080/10556780902883184
3. Bao, X., Sahinidis, N.V., Tawarmalani, M.: Semidefinite relaxations for quadratically constrained quadratic programming: A review and comparisons. Mathematical programming **129**(1), 129–157 (2011). doi:10.1007/s10107-011-0462-2
4. Benson, H.P.: Separable concave minimization via partial outer approximation and branch and bound. Operations Research Letters **9**(6), 389–394 (1990). doi:10.1016/0167-6377(90)90059-E
5. Bezanson, J., Edelman, A., Karpinski, S., Shah, V.B.: Julia: A fresh approach to numerical computing. SIAM Review **59**(1), 65–98 (2017). doi:10.1137/141000671
6. Blackford, L.S., Demmel, J., Dongarra, J., Duff, I., Hammarling, S., Henry, G., Heroux, M., Kaufman, L., Lumsdaine, A., , Petitet, A., Pozo, R., Remington, K., Whaley, R.C.: An updated set of basic linear algebra subprograms (BLAS). ACM Transactions on Mathematical Software **28**(2), 135–151 (2002). doi:10.1145/567806.567807

7. Bompadre, A., Mitsos, A.: Convergence rate of McCormick relaxations. Journal of Global Optimization **52**(1), 1–28 (2011). doi:10.1007/s10898-011-9685-2
8. Bongartz, D., Mitsos, A.: Deterministic global optimization of process flowsheets in a reduced space using McCormick relaxations. Journal of Global Optimization **69**(4), 761–796 (2017). doi:10.1007/s10898-017-0547-4
9. Bongartz, D., Mitsos, A.: Deterministic global flowsheet optimization: Between equation-oriented and sequential-modular methods. AIChE Journal **65**(3), 1022–1034 (2019). doi:10.1002/aic.16507
10. Bongartz, D., Najman, J., Mitsos, A.: Deterministic global optimization of steam cycles using the IAPWS-IF97 model. Optimization and Engineering **10** (2020). doi:10.1007/s11081-020-09502-1
11. Bongartz, D., Najman, J., Sass, S., Mitsos, A.: MAiNGO: McCormick based algorithm for mixed integer nonlinear global optimization. Tech. rep., RWTH Aachen (2018). URL `https://git.rwth-aachen.de/avt-svt/public/maingo`
12. Comba, J.L.D., Stolfi, J.: Affine arithmetic and its applications to computer graphics. In: L.H. de Figueiredo, J. de Miranda Gomes (eds.) Proceedings of VI SIBGRAPI (Brazilian Symposium on Computer Graphics and Image Processing), pp. 9–18 (1993). URL `http://urlib.net/ibi/8JMKD3MGPBW34M/3D6UQ68`
13. De Oliveira, L.G., de Paiva, A.P., Balestrassi, P.P., Ferreira, J.R., da Costa, S.C., da Silva Campos, P.H.: Response surface methodology for advanced manufacturing technology optimization: theoretical fundamentals, practical guidelines, and survey literature review. The International Journal of Advanced Manufacturing Technology **104**(5), 1785–1837 (2019). doi:10.1007/s00170-019-03809-9
14. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Mathematical programming **91**(2), 201–213 (2002). doi:https://doi.org/10.1007/s101070100263
15. Falk, J.E., Hoffman, K.L.: Concave minimization via collapsing polytopes. Operations research **34**(6), 919–929 (1986). doi:10.1287/opre.34.6.919
16. Fedorov, G., Nguyen, K.T., Harrison, P., Singh, A.: Intel Math Kernel Library 2019 Update 2 Release Notes (2019). URL `https://software.intel.com/en-us/mkl`
17. de Figueiredo, L.H., Stolfi, J.: Affine arithmetic: Concepts and applications. Numerical Algorithms **37**, 147–158 (2004). doi:10.1023/b:numa.0000049462.70970.b6
18. Gurobi Optimization, LLC: Gurobi optimizer reference manual (2020). URL `http://www.gurobi.com`
19. He, T., Tawarmalani, M.: A new framework to relax composite functions in nonlinear programs. Mathematical Programming pp. 1–40 (2020). doi:10.1007/s10107-020-01541-x
20. Hejazi, T.H., Seyyed-Esfahani, M., Mahootchi, M.: Quality chain design and optimization by multiple response surface methodology. The International Journal of Advanced Manufacturing Technology **68**(1-4), 881–893 (2013). doi:10.1007/s00170-013-4950-9
21. Hiriart-Urruty, J.B., Lemaréchal, C.: Fundamentals of Convex Analysis. Springer Science & Business Media, Heidelberg (2004). doi:10.1007/978-3-642-56468-0
22. Horst, R., Tuy, H.: Global Optimization: Deterministic Approaches, third edn. Springer Science & Business Media, Berlin, Germany (1996). doi:10.1007/978-3-662-03199-5
23. IBM: IBM ILOG CPLEX Optimizer. `https://www.ibm.com/software/integration/optimization/cplex-optimizer/` (2020)
24. Kannan, R., Barton, P.I.: The cluster problem in constrained global optimization. Journal of Global Optimization **69**(3), 629–676 (2017). doi:10.1007/s10898-017-0531-z
25. Khan, K.A., Watson, H.A.J., Barton, P.I.: Differentiable McCormick relaxations. Journal of Global Optimization **67**(4), 687–729 (2016). doi:10.1007/s10898-016-0440-6
26. Khan, K.A., Wilhelm, M., Stuber, M.D., Cao, H., Watson, H.A.J., Barton, P.I.: Corrections to: Differentiable McCormick relaxations. Journal of Global Optimization **70**(3), 705–706 (2018). doi:10.1007/s10898-017-0601-2
27. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs: Part I — convex underestimating problems. Mathematical Programming **10**(1), 147–175 (1976). doi:10.1007/bf01580665
28. Messine, F.: Extensions of affine arithmetic: Application to unconstrained global optimization. Journal of Universal Computer Science **8**(11), 992–1015 (2002). doi:10.3217/jucs-008-11-0992
29. Misener, R., Floudas, C.A.: GloMIQO: Global mixed-integer quadratic optimizer. Journal of Global Optimization **57**(1), 3–50 (2013). doi:10.1007/s10898-012-9874-7
30. Misener, R., Floudas, C.A.: ANTIGONE: Algorithms for continuous/integer global optimization of nonlinear equations. Journal of Global Optimization **59**(2-3), 503–526 (2014). doi:10.1007/s10898-014-0166-2

31. Mistry, M., Misener, R.: Optimising heat exchanger network synthesis using convexity properties of the logarithmic mean temperature difference. Computers & Chemical Engineering **94**, 1–17 (2016). doi:10.1016/j.compchemeng.2016.07.001
32. Mitsos, A., Chachuat, B., Barton, P.I.: McCormick-based relaxations of algorithms. SIAM Journal on Optimization **20**(2), 573–601 (2009). doi:10.1137/080717341
33. Moore, R.E.: Interval Analysis. Prentice-Hall, Englewood Cliffs, NJ (1966)
34. Mosek, A.: The MOSEK optimization software (2010). URL https://www.mosek.com
35. Mostaan, H., Shamanian, M., Safari, M.: Process analysis and optimization for fracture stress of electron beam welded ultra-thin feco-v foils. The International Journal of Advanced Manufacturing Technology **87**(1), 1045–1056 (2016). doi:10.1007/s00170-016-8553-0
36. Nagarajan, H., Lu, M., Wang, S., Bent, R., Sundar, K.: An adaptive, multivariate partitioning algorithm for global optimization of nonconvex programs. Journal of Global Optimization (2019). doi:10.1007/s10898-018-00734-1
37. Nagarajan, H., Lu, M., Yamangil, E., Bent, R.: Tightening McCormick relaxations for nonlinear programs via dynamic multivariate partitioning. In: M. Rueher (ed.) International Conference on Principles and Practice of Constraint Programming, pp. 369–387. Springer (2016). doi:10.1007/978-3-319-44953-1_24
38. Najman, J., Bongartz, D., Mitsos, A.: Convex relaxations of componentwise convex functions. Computers & Chemical Engineering **130**, 106,527 (2019). doi:10.1016/j.compchemeng.2019.106527
39. Najman, J., Bongartz, D., Mitsos, A.: Relaxations of thermodynamic property and costing models in process engineering. Computers & Chemical Engineering **130**, 106,571 (2019). doi:10.1016/j.compchemeng.2019.106571
40. Najman, J., Bongartz, D., Tsoukalas, A., Mitsos, A.: Erratum to: Multivariate McCormick relaxations. Journal of Global Optimization **68**, 219–225 (2017). doi:10.1007/s10898-016-0470-0
41. Najman, J., Mitsos, A.: Convergence analysis of multivariate McCormick relaxations. Journal of Global Optimization **66**(4), 1–32 (2016). doi:10.1007/s10898-016-0408-6
42. Najman, J., Mitsos, A.: Convergence order of McCormick relaxations of LMTD function in heat exchanger networks. Computer Aided Chemical Engineering **38**, 1605–1610 (2016). doi:10.1016/B978-0-444-63428-3.50272-1
43. Najman, J., Mitsos, A.: On tightness and anchoring of McCormick and other relaxations. Journal of Global Optimization (2017). doi:10.1007/s10898-017-0598-6
44. Najman, J., Mitsos, A.: Tighter McCormick relaxations through subgradient propagation. Journal of Global Optimization **75**, 565–593 (2019). doi:10.1007/s10898-019-00791-0
45. Nedialkov, N.S., Kreinovich, V., Starks, S.A.: Interval arithmetic, affine arithmetic, Taylor series methods: why, what next? Numerical Algorithms **37**(1), 325–336 (2004). doi:10.1023/B:NUMA.0000049478.42605.cf
46. Ninin, J., Messine, F., Hansen, P.: A reliable affine relaxation method for global optimization. 4OR **13**(3), 247–277 (2015). doi:10.1007/s10288-014-0269-0
47. Nohra, C.J., Raghunathan, A.U., Sahinidis, N.: Spectral relaxations and branching strategies for global optimization of mixed-integer quadratic programs. SIAM Journal on Optimization **31**(1), 142–171 (2021). doi:10.1137/19M1271762
48. Nohra, C.J., Raghunathan, A.U., Sahinidis, N.V.: SDP-quality bounds via convex quadratic relaxations for global optimization of mixed-integer quadratic programs. Mathematical Programming pp. 1–31 (2021). doi:10.1007/s10107-021-01680-9
49. Pakseresht, A.H., Ghasali, E., Nejati, M., Shirvanimoghaddam, K., Javadi, A.H., Teimouri, R.: Development empirical-intelligent relationship between plasma spray parameters and coating performance of yttria-stabilized zirconia. The International Journal of Advanced Manufacturing Technology **76**(5), 1031–1045 (2015). doi:10.1007/s00170-014-6212-x
50. Puranik, Y., Sahinidis, N.V.: Domain reduction techniques for global NLP and MINLP optimization. Constraints **22**(3), 338–376 (2017). doi:10.1007/s10601-016-9267-5
51. Rajakumar, S., Balasubramanian, V.: Diffusion bonding of titanium and aa 7075 aluminum alloy dissimilar joints—process modeling and optimization using desirability approach. The International Journal of Advanced Manufacturing Technology **86**(1), 1095–1112 (2016). doi:10.1007/s00170-015-8223-7
52. Rajesh, P., Nagaraju, U., Gowd, G.H., Vardhan, T.V.: Experimental and parametric studies of nd: Yag laser drilling on austenitic stainless steel. The International Journal of Advanced Manufacturing Technology **93**(1), 65–71 (2017). doi:10.1007/s00170-015-7639-4
53. Rump, S.M., Kashiwagi, M.: Implementation and improvements of affine arithmetic. Nonlinear Theory and Its Applications, IEICE **6**(3), 341–359 (2015). doi:10.1587/nolta.6.341

54. Sahinidis, N.V.: BARON: A general purpose global optimization software package. Journal of Global Optimization **8**(2), 201–205 (1996). doi:10.1007/BF00138693
55. Sahinidis, N.V.: BARON 21.1.13: Global Optimization of Mixed-Integer Nonlinear Programs, *User's Manual* (2017). URL `https://www.minlp.com/downloads/docs/baron%20manual.pdf`
56. Sahlodin, A.M., Chachuat, B.: Discretize-then-relax approach for convex/concave relaxations of the solutions of parametric ODEs. Applied Numerical Mathematics **61**(7), 803 – 820 (2011). doi:10.1016/j.apnum.2011.01.009
57. Schweidtmann, A.M., Bongartz, D., Grothe, D., Kerkenhoff, T., Lin, X., Najman, J., Mitsos, A.: Deterministic global optimization with gaussian processes embedded. Mathematical Programming Computation **13**(3), 553–581 (2021). doi:10.1007/s12532-021-00204-y
58. Schweidtmann, A.M., Bongartz, D., Huster, W.R., Mitsos, A.: Deterministic global process optimization: Flash calculations via artificial neural networks. In: A.A. Kiss, E. Zondervan, R. Lakerveld, L. Özkan (eds.) Computer Aided Chemical Engineering, vol. 46, pp. 937–942. Elsevier, Amsterdam, Netherlands (2019). doi:10.1016/b978-0-12-818634-3.50157-0
59. Schweidtmann, A.M., Huster, W.R., Lüthje, J.T., Mitsos, A.: Deterministic global process optimization: Accurate (single-species) properties via artificial neural networks. Computers & Chemical Engineering **121**, 67–74 (2019). doi:10.1016/j.compchemeng.2018.10.007
60. Schweidtmann, A.M., Mitsos, A.: Deterministic global optimization with artificial neural networks embedded. Journal of Optimization Theory and Applications **180**(3), 925–948 (2018). doi:10.1007/s10957-018-1396-0
61. Scott, J.K., Barton, P.I.: Improved relaxations for the parametric solutions of odes using differential inequalities. Journal of Global Optimization **57**(1), 143–176 (2013). doi:10.1007/s10898-012-9909-0
62. Scott, J.K., Stuber, M.D., Barton, P.I.: Generalized McCormick relaxations. Journal of Global Optimization **51**(4), 569–606 (2011). doi:10.1007/s10898-011-9664-7
63. Singer, A.B.: Global dynamic optimization. Ph.D. thesis, Massachusetts Institute of Technology (2004)
64. Stolfi, J., Figueiredo, L.H.D.: An introduction to affine arithmetic. TEMA - Tendências em Matemática Aplicada e Computacional **4**(3), 297–312 (2003). doi:10.5540/tema.2003.04.03.0297
65. Stuber, M.D., Scott, J.K., Barton, P.I.: Convex and concave relaxations of implicit functions. Optimization Methods and Software **30**(3), 424–460 (2015). doi:10.1080/10556788.2014.924514
66. Tawarmalani, M., Sahinidis, N.V.: A polyhedral branch-and-cut approach to global optimization. Mathematical Programming **103**(2), 225–249 (2005). doi:10.1007/s10107-005-0581-8
67. Tsoukalas, A., Mitsos, A.: Multivariate McCormick relaxations. Journal of Global Optimization **59**(2-3), 633–662 (2014). doi:10.1007/s10898-014-0176-0
68. Wang, C., Wilhelm, M.E., Stuber, M.D.: Semi-Infinite optimization with hybrid models. Industrial & Engineering Chemistry Research **61**(15), 5239–5254 (2022). doi:10.1021/acs.iecr.2c00113
69. Wang, E., Zhang, Q., Shen, B., Zhang, G., Lu, X., Wu, Q., Wang, Y.: Intel Math Kernel Library, pp. 167–188. Springer International Publishing, New York, NY (2014). doi:10.1007/978-3-319-06486-4_7
70. Watson, H.A., Vikse, M., Gundersen, T., Barton, P.I.: Optimization of single mixed-refrigerant natural gas liquefaction processes described by nondifferentiable models. Energy **150**, 860–876 (2018). doi:https://doi.org/10.1016/j.energy.2018.03.013
71. Wechsung, A., Scott, J.K., Watson, H.A.J., Barton, P.I.: Reverse propagation of McCormick relaxations. Journal of Global Optimization **63**(1), 1–36 (2015). doi:10.1007/s10898-015-0303-6
72. Wilhelm, M.E., Gottlieb, R.X., Stuber, M.D.: PSORLab/McCormick.jl (2020). doi:10.5281/ZENODO.4278415. URL `https://github.com/PSORLab/McCormick.jl`
73. Wilhelm, M.E., Le, A.V., Stuber, M.D.: Global optimization of stiff dynamical systems. AIChE Journal **65**, e16,836 (2019). doi:10.1002/aic.16836
74. Wilhelm, M.E., Stuber, M.D.: EAGO.jl: easy advanced global optimization in Julia. Optimization Methods and Software pp. 1–26 (2020). doi:10.1080/10556788.2020.1786566
75. Wilhelm, M.E., Wang, C., Stuber, M.D.: Convex and concave envelopes of artificial neural network activation functions for deterministic global optimization. Journal of Global Optimization (2022). doi:10.1007/s10898-022-01228-x
76. Yue, Z., Huang, C., Zhu, H., Wang, J., Yao, P., Liu, Z.: Optimization of machining parameters in the abrasive waterjet turning of alumina ceramic based on the response surface methodology. The International Journal of Advanced Manufacturing Technology **71**(9-12), 2107–2114 (2014). doi:10.1007/s00170-014-5624-y
77. Zorn, K., Sahinidis, N.V.: Global optimization of general non-convex problems with intermediate bilinear substructures. Optimization Methods and Software **29**(3), 442–462 (2014). doi:10.1080/10556788.2013.783032