General Convex Relaxations of Implicit Functions and Inverse Functions

Huiyi Cao[1] and Kamil A. Khan[1*]

[1]Department of Chemical Engineering, McMaster University, 1280 Main Street West, Hamilton, L8S 4L8, Ontario, Canada.

*Corresponding author(s). E-mail(s): kamilkhan@mcmaster.ca; Contributing authors: caoh14@mcmaster.ca;

Abstract

Convex relaxations of nonconvex functions provide useful bounding information in applications such as deterministic global optimization and reachability analysis. In some situations, the original nonconvex functions may not be known explicitly, but are instead described implicitly by nonlinear equation systems. In these cases, established convex relaxation methods for closed-form functions are not directly applicable. This article presents a new general strategy to construct convex relaxations for such implicit functions. These relaxations are described as convex parametric programs whose constraints are convex relaxations of the original residual function. This relaxation strategy is straightforward to implement, produces tight relaxations in practice, is particularly efficient to carry out when monotonicity properties can be exploited, and does not assume the existence or uniqueness of an implicit function on the entire intended domain. Unlike all previous approaches to the authors' knowledge, this new approach permits any relaxations of the residual function; it does not require the residual relaxations to be factorable or to be obtained from a McCormick-like traversal of a computational graph. This new convex relaxation strategy is extended to inverse functions, compositions involving implicit functions, feasible-set mappings in constraint satisfaction problems, and solutions of parametric ODEs. Based on a proof-of-concept implementation in Julia, numerical examples are presented to illustrate the convex relaxations produced for various implicit functions and optimal-value functions.

Keywords: Implicit functions, Nonconvex optimization, Convex underestimators, McCormick relaxations, Constraint satisfaction problems

1 Introduction

Branch-and-bound algorithms for deterministic global optimization require guaranteed lower bounds on the solution of a nonconvex nonlinear program (NLP) on particular subsets of the search space. This bounding information is typically obtained by generating and minimizing a convex relaxation of the original NLP to its global optimum with a local NLP solver [1]; constructing this relaxation requires furnishing appropriate relaxations of the objective function and constraint functions. For a function described explicitly by a closed-form expression, several established relaxation techniques can effectively generate corresponding convex relaxations. In particular, if a nonconvex function is twice-continuously differentiable, we may construct its convex relaxations using αBB relaxations [2], which involve adding a sufficiently large convex quadratic term to the original function. If the nonconvex function is a finite composition of known intrinsic functions from a library, such as the functions that can be represented on a typical scientific calculator, then the function is said to be *factorable*, and we can construct its convex relaxations using McCormick's relaxation method [1, 3]. This relaxation method generates accurate and computationally cheap convex underestimators [4]. Several open-source implementations of this approach are available, such as the C++ library MC++ [5] and the Julia package McCormick.jl [6]. However, if no closed-form expression for the original nonconvex function is known, then the convex relaxation methods mentioned previously are not directly applicable.

Thus, as will be formalized in Section 3 below, this article considers a function $\boldsymbol{x} : \mathbb{R}^{n_p} \to \mathbb{R}^{n_x}$ that is defined implicitly so as to satisfy the equation:

$$\boldsymbol{f}(\boldsymbol{x}(\boldsymbol{p}), \boldsymbol{p}) \equiv \boldsymbol{0},$$

where $\boldsymbol{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_p}$ is a known *residual function*. Such *implicit functions* $\boldsymbol{x}$ appear in many research areas and applications [7], such as the ellipse equation in physics and astronomy, the van der Waals equation of state in thermodynamics, and the equality constraints in mathematical programming [8]. A closed-form expression is typically not available for the implicit function $\boldsymbol{x}$, so its convex relaxations cannot be constructed using the αBB or McCormick relaxations. This article seeks improved dedicated convex relaxation techniques for implicit functions.

Several existing approaches have been developed to address this problem. One major category of these approaches is based on applying a fixed-point iteration solver to the original nonlinear equation system, and then relaxing these closed-form iterations. Scott et al. [9] developed *generalized McCormick (GM) relaxations* based on McCormick's relaxation method, permitting convex and concave relaxations of a function's inputs to be used as arguments [10]. Using this property, Scott et al. [9] introduced an approach to construct convex relaxations for implicit functions by applying GM to finitely many fixed-point iterations of an equation-solving method [11]. Stuber et al. [10] later showed that this approach may not provide any refinement over known *a priori* interval bounds, which may limit the applicability of this approach. To address this issue, they proposed an improved successive fixed-point iteration approach to construct convex relaxations for implicit functions by relaxing iterations based on the Mean Value Theorem [10]. This approach was employed to relax the equality constraints in NLPs as inequality constraints in order to reduce the NLP dimensionality; they argue that this is particularly useful in global optimization. Notably, this approach only applies to factorable residual functions with GM relaxations, and assumes unique solutions of the corresponding nonlinear equation system. It also requires additional *a priori* knowledge of the Jacobian of the residual function, in the form of interval bounds and convex relaxations. Khan et al. [4] applied Stuber et al.'s approach to construct differentiable relaxations for implicit functions, using *differentiable McCormick (DM) relaxations* [4, 12] in place of GM. Wilhelm et al. [13] adapted Stuber et al.'s approach to generate convex relaxations for the numerical solutions of parametric ordinary differential equations (ODEs) after discretizing them with implicit ODE solution methods.

A second category of implicit function relaxation approaches is based on *reverse McCormick (RM) propagation* proposed by Wechsung et al. [8]. RM is similar to the standard McCormick relaxations for factorable functions, except that it carefully propagates the computed convex and concave relaxations backward through the function's computational graph, like the reverse mode of automatic differentiation [14]. Each backward step through the computational graph involves applying new set intersection rules. Moreover, RM is also applicable to constraint satisfaction problems (CSPs) containing both equality and inequality constraints, and allows convex relaxations to be constructed for a point-to-set mapping of system parameters to the corresponding feasible regions. Unlike Stuber et al.'s approach, Wechsung et al.'s approach does not assume the existence nor the uniqueness of a solution. Nevertheless, implementing this relaxation method is a nontrivial coding task; to our knowledge, no off-the-shelf implementation is currently available. Such an implementation would require generating each function's computational graph and then stepping through it forward and backward while applying RM rules to each operation.

In this work, we propose a strategy to generate convex and concave relaxations for implicit functions using parametric programming. These relaxations are described by convex optimization problems whose constraints are convex relaxations of the original residual function. This approach appears to be completely novel, and does not appear to be a special case of either the general Tsoukalas-Mitsos relaxations [15] or the auxiliary variable method (AVM) [16], though we note that both of these approaches also employ embedded convex optimization problems. Our new approach is extended to construct convex relaxations for inverse functions, compositions involving implicit functions, and point-to-set mappings describing parametric CSPs. This new approach is also applied in the setting of *optimization-based bounds tightening (OBBT)* to tighten *a priori* interval bounds on the range of an implicit function, which can in turn further tighten the resulting convex relaxations.

Our new approach does not require the underlying implicit function to be uniquely defined, or to exist everywhere on the intended domain. It also appears to be simpler to implement and automate than previous methods, and is efficient to carry out when certain monotonicity or linearity properties can be exploited. Moreover, unlike all previous approaches to our knowledge, our new approach does not require the supplied residual function relaxations to be factorable or to be obtained by traversing the original residual function's computational graph. In our new approach, any convex relaxation techniques may be employed to relax the residual function, such as standard McCormick relaxations [1, 3], αBB relaxations [2], convex envelopes, the Scott-Barton relaxations for parametric ordinary differential equations [17], and convex relaxation approaches based on black-box sampling [18]. In principle, they are also compatible with convex relaxation approaches without much precedent in global optimization applications, such as Fenchel conjugates and Moreau-Yosida regularizations [19]. By contrast, established methods are limited to one particular relaxation method, such as GM in [9, 10] and RM in [8]. Lastly, the convex and concave relaxations generated with our new approach are comparable to these established methods in tightness, and are shown to produce tighter relaxations in various numerical examples. In general, tight enclosures are beneficial in deterministic global optimization and reachability analysis.

This article is structured as follows. Section 2 introduces the mathematical background underlying this work. In Section 3, we formulate our new strategy, demonstrate its correctness, and discuss its computational complexity and convergence properties as the underlying domain shrinks. We extend this approach to relax inverse functions, and to relax compositions involving implicit functions by combining our approach with the multivariate McCormick relaxations of Tsoukalas and Mitsos [15]. Section 4 extends this new strategy to parametric CSPs and discusses how parametric sensitivity information might be obtained. In Section 5, we adapt the new convex relaxation strategy to improve the tightness of interval bounds for implicit functions and CSPs via OBBT. Finally, a proof-of-concept Julia implementation of our results is described in Section 6, and numerical examples are presented to illustrate our new approach.

2 Background

This section summarizes the mathematical background underlying this work, and echoes the background presented in [20]. The following notation conventions are used in this article. Vectors are denoted with boldface lower-case letters (e.g. $\boldsymbol{x} \in \mathbb{R}^n$). Given vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, inequalities such as $\boldsymbol{x} < \boldsymbol{y}$ or $\boldsymbol{x} \leq \boldsymbol{y}$ are to be interpreted componentwise. Throughout this article, convexity of a vector-valued function $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^m$ refers to convexity of all components f_i, and concavity is analogous. An *interval* in $\mathbb{R}^n$ is a nonempty subset of $\mathbb{R}^n$ of the form $\{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a} \le \boldsymbol{x} \le \boldsymbol{b}\}$, which is also denoted as $[\boldsymbol{a}, \boldsymbol{b}]$. $\mathbb{IR}^n$ denotes the set of all intervals in $\mathbb{R}^n$. Let $\mathbb{N}$ denote the set $\{1, 2, 3, \ldots\}$ of natural numbers.

Next, we introduce convex and concave relaxations of functions.

Definition 1. Consider a convex set $P \subset \mathbb{R}^{n_p}$ and a subset $Q \subset P$. Consider the extended real numbers $\bar{\mathbb{R}} := \mathbb{R} \cup \{-\infty, +\infty\}$, and a function $\boldsymbol{\phi} : Q \to \bar{\mathbb{R}}^m$. Then:

1. $\boldsymbol{\phi}^{\mathrm{cv}} : P \to \mathbb{R}^m$ is a *convex relaxation* of $\boldsymbol{\phi}$ on P if
   - $\boldsymbol{\phi}^{\mathrm{cv}}(\boldsymbol{p}) \le \boldsymbol{\phi}(\boldsymbol{p})$ for all $\boldsymbol{p} \in Q$, and
   - $\boldsymbol{\phi}^{\mathrm{cv}}$ is convex on P.
2. $\boldsymbol{\phi}^{\mathrm{cc}} : P \to \mathbb{R}^m$ is a *concave relaxation* of $\boldsymbol{\phi}$ on P if
   - $\boldsymbol{\phi}^{\mathrm{cc}}(\boldsymbol{p}) \ge \boldsymbol{\phi}(\boldsymbol{p})$ for all $\boldsymbol{p} \in Q$, and
   - $\boldsymbol{\phi}^{\mathrm{cc}}$ is concave on P.

We permit $Q \ne P$ here, in order to cover relaxations of implicit functions that are not defined everywhere within the residual function's domain.

As summarized in Section 1, several methods have been established to generate convex relaxations for closed-form factorable functions automatically. The αBB relaxation method [2] constructs convex relaxations for twice-continuously differentiable functions, and involves adding a sufficiently large convex quadratic term to the original function. Another approach is McCormick's relaxation method and its variants [1, 3, 4, 9, 12, 15].

Next, we summarize a sufficient condition for an optimal-value function to be convex. The following definition and proposition are adapted from [21].

Definition 2. Let $P \subset \mathbb{R}^{n_p}$ be a convex set. A *point-to-set* map $S^R : \mathbb{R}^{n_p} \rightrightarrows \mathbb{R}^{n_x}$ assigns a subset of $\mathbb{R}^{n_x}$ to each element of $\mathbb{R}^{n_p}$. S^R is *convex* on P if, for all $\boldsymbol{p}_1, \boldsymbol{p}_2 \in P$ and $\lambda \in (0, 1)$, the Minkowski sum $\lambda S^R(\boldsymbol{p}_1) + (1 - \lambda) S^R(\boldsymbol{p}_2)$ is a subset of $S^R(\lambda \boldsymbol{p}_1 + (1 - \lambda)\boldsymbol{p}_2)$. Moreover, S^R is a *convex relaxation* of an arbitrary point-to-set map $S : \mathbb{R}^{n_p} \rightrightarrows \mathbb{R}^{n_x}$ on P if, for all $\boldsymbol{p} \in P$, $S(\boldsymbol{p}) \subseteq S^R(\boldsymbol{p})$ and S^R is convex on P.

Proposition 2.1 (Corollary 2.1 in [21])**.** Consider two convex sets $P \subset \mathbb{R}^{n_p}$ and $X \subset \mathbb{R}^{n_x}$, two convex functions $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \to \mathbb{R}$ and $\boldsymbol{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_g}$, and an affine function $\boldsymbol{h} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_h}$. Let $C : \mathbb{R}^{n_p} \rightrightarrows \mathbb{R}^{n_x}$ be a point-to-set map such that, for each $\boldsymbol{p} \in P$,

$$C(\boldsymbol{p}) = \{\boldsymbol{x} \in X \mid \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{p}) \le \boldsymbol{0}, \quad \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{p}) = \boldsymbol{0}\}.$$

For each $\boldsymbol{p} \in P$, consider a general parametric optimization problem

$$\min_{\boldsymbol{x}} \; f(\boldsymbol{x}, \boldsymbol{p}), \qquad \text{subject to} \;\; \boldsymbol{x} \in C(\boldsymbol{p}).$$

Define an optimal-value function $f^* : \mathbb{R}^{n_p} \to \mathbb{R}$ such that for each $\boldsymbol{p} \in P$,

$$f^*(\boldsymbol{p}) = \begin{cases} \inf_{\boldsymbol{x}}\{f(\boldsymbol{x}, \boldsymbol{p}) \mid \boldsymbol{x} \in C(\boldsymbol{p})\}, & \text{if } C(\boldsymbol{p}) \neq \emptyset, \\ +\infty, & \text{if } C(\boldsymbol{p}) = \emptyset. \end{cases}$$

Then, the function f^* is convex on P.

Finally, we summarize the *directional derivative*, which provides local radial sensitivity information for a function. The following definition is adapted from [22, Section 3.1].

Definition 3. Let $P \subset \mathbb{R}^{n_p}$ be a convex set. Let $\boldsymbol{\phi} : P \to \mathbb{R}^n$ be a function. If for every $\boldsymbol{d} \in P$ the limit

$$\boldsymbol{\phi}'(\boldsymbol{z}_0;\ \boldsymbol{d}) = \lim_{\lambda \downarrow 0} \frac{1}{\lambda}(\boldsymbol{\phi}(\boldsymbol{z}_0 + \lambda \boldsymbol{d}) - \boldsymbol{\phi}(\boldsymbol{z}_0))$$

exists, then $\boldsymbol{\phi}$ is said to be *directionally differentiable* at $\boldsymbol{z}_0$ and the function $\boldsymbol{\phi}'(\boldsymbol{z}_0;\ \cdot)$ is the *directional derivative* mapping of $\boldsymbol{\phi}$ at $\boldsymbol{z}_0$.

3 Convex Relaxations of Implicit Functions

In this section, we present a new formulation for generating convex and concave relaxations for an implicit function using parametric programming. These implicit function relaxations are then generalized to cover compositions of implicit functions with known inner functions, and to cover inverse functions. Convergence properties and computational complexity are discussed.

In the remainder of this section, consider a *residual function* $\boldsymbol{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_x}$, and the following system of equations:

$$\boldsymbol{f}(\boldsymbol{z}, \boldsymbol{p}) = \boldsymbol{0}. \tag{1}$$

Consider a convex compact set $P \subset \mathbb{R}^{n_p}$, and let $Q \subset P$ be the set of $\boldsymbol{p} \in P$ for which the equation (1) has at least one solution $\boldsymbol{z}$. The following assumption formalizes an implicit function that will later be relaxed.

Assumption 1. Suppose that the the following conditions hold:

1. The set Q is nonempty, so there is a meaningful *implicit function* $\boldsymbol{x} : Q \to \mathbb{R}^{n_x}$ that satisfies:
$$\boldsymbol{f}(\boldsymbol{x}(\boldsymbol{p}), \boldsymbol{p}) = \boldsymbol{0}. \tag{2}$$
2. There is a known interval $X \in \mathbb{IR}^{n_x}$ for which (2) holds and $\boldsymbol{x}(\boldsymbol{p}) \in X$ for every $\boldsymbol{p} \in Q$.

Assumption 1 does not require the implicit function $\boldsymbol{x}$ to be uniquely defined by (2); there may be many valid choices of $\boldsymbol{x}$. Condition 2 in Assumption 1 supposes that we know how to bound the range of the particular implicit function $\boldsymbol{x}$ that we are considering. Multiple numerical methods are available to construct the range estimate X, including the interval Newton method [23] and the interval Krawczyk method [24]; these methods both require $\boldsymbol{f}$ to be Lipschitz continuous [24, Theorem 5.1.8].

Incidentally, the semi-local implicit function theorem [24] provides a sufficient condition for the uniqueness of the implicit function $\boldsymbol{x}$, though we do not require uniqueness in this work. Roughly, that theorem requires existence of a nonsingular partial derivative $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{z}}$ on $X \times P$ [7]. This result was later extended to Lipschitz continuous functions in [25, Theorem 7.1.1], using generalized derivative constructions.

3.1 Main Result

Under Assumption 1, the following theorem constructs new convex and concave relaxations for the implicit function $\boldsymbol{x}$.

Theorem 3.1. Suppose that Assumption 1 holds. Let $\boldsymbol{f}^{\mathrm{cv}}, \boldsymbol{f}^{\mathrm{cc}} : X \times P \to \mathbb{R}^{n_x}$ be convex and concave relaxations of $\boldsymbol{f}$ on $X \times P$, respectively. Define $\boldsymbol{x}^{\mathrm{cv}}, \boldsymbol{x}^{\mathrm{cc}} : P \to \bar{\mathbb{R}}^{n_x}$ such that, for each $i \in \{1, \ldots, n_x\}$ and $\boldsymbol{p} \in P$,

$$x_i^{\mathrm{cv}}(\boldsymbol{p}) = \inf_{\boldsymbol{\xi} \in X} \xi_i \quad \text{subject to} \quad \boldsymbol{f}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \boldsymbol{0} \leq \boldsymbol{f}^{\mathrm{cc}}(\boldsymbol{\xi}, \boldsymbol{p}), \tag{3a}$$

$$x_i^{\mathrm{cc}}(\boldsymbol{p}) = \sup_{\boldsymbol{\xi} \in X} \xi_i \quad \text{subject to} \quad \boldsymbol{f}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \boldsymbol{0} \leq \boldsymbol{f}^{\mathrm{cc}}(\boldsymbol{\xi}, \boldsymbol{p}). \tag{3b}$$

If these optimization problems are infeasible, then set $x_i^{\mathrm{cv}}(\boldsymbol{p}) := +\infty$ and $x_i^{\mathrm{cc}}(\boldsymbol{p}) := -\infty$ for each i by convention.

Then, $\boldsymbol{x}^{\mathrm{cv}}$ is a convex relaxation of $\boldsymbol{x}$ on P, and $\boldsymbol{x}^{\mathrm{cc}}$ is a concave relaxation of $\boldsymbol{x}$ on P.

Proof Following Definition 1, we will show that $\boldsymbol{x}^{\mathrm{cv}}(\boldsymbol{p}) \leq \boldsymbol{x}(\boldsymbol{p})$ for all $\boldsymbol{p} \in Q$, and that $\boldsymbol{x}^{\mathrm{cv}}$ is convex on P. The claims regarding $\boldsymbol{x}^{\mathrm{cc}}$ follow from analogous arguments, which are omitted here.

Under Assumption 1, for each $\boldsymbol{p} \in Q$, $\boldsymbol{x}(\boldsymbol{p})$ is feasible in the right-hand side optimization problem in (4), so the feasible region of (4) is nonempty. Thus, for any $i \in \{1, \ldots, n_x\}$ and $\boldsymbol{p} \in Q$,

$$\begin{aligned} x_i^{\mathrm{cv}}(\boldsymbol{p}) &= \inf_{\boldsymbol{\xi} \in X} \{\xi_i \mid \boldsymbol{f}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \boldsymbol{0} \leq \boldsymbol{f}^{\mathrm{cc}}(\boldsymbol{\xi}, \boldsymbol{p})\} \\ &\leq \inf_{\boldsymbol{\xi} \in X} \{\xi_i \mid \boldsymbol{f}(\boldsymbol{\xi}, \boldsymbol{p}) = \boldsymbol{0}\} \\ &\leq x_i(\boldsymbol{p}). \end{aligned}$$

Collecting these inequalities for all i, we obtain $\boldsymbol{x}^{\mathrm{cv}}(\boldsymbol{p}) \leq \boldsymbol{x}(\boldsymbol{p})$ for all $\boldsymbol{p} \in Q$.

Next, we verify the convexity of $\boldsymbol{x}^{\mathrm{cv}}$. Define $\boldsymbol{\phi} : X \times P \to \mathbb{R}^{n_x} \times \mathbb{R}^{n_x}$ such that $\boldsymbol{\phi}(\boldsymbol{\xi}, \boldsymbol{p}) = (\boldsymbol{f}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}), -\boldsymbol{f}^{\mathrm{cc}}(\boldsymbol{\xi}, \boldsymbol{p}))$ for each $\boldsymbol{\xi} \in X$ and $\boldsymbol{p} \in P$. Since $\boldsymbol{f}^{\mathrm{cv}}$ and $\boldsymbol{f}^{\mathrm{cc}}$ are respectively convex and concave, it follows that $\boldsymbol{\phi}$ is convex. For each $i \in \{1, \ldots, n_x\}$ and $\boldsymbol{p} \in P$, (3a) is equivalent to

$$x_i^{\mathrm{cv}}(\boldsymbol{p}) = \inf_{\boldsymbol{\xi} \in X} \xi_i \quad \text{subject to} \quad \boldsymbol{\phi}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \boldsymbol{0}. \tag{4}$$

Since the objective function of (4) is linear, X and P are convex, and $\boldsymbol{\phi}$ is convex on $X \times P$, the convexity of x_i^{cv} on P follows from Proposition 2.1. □

The implicit function relaxations provided by Theorem 3.1 place no restrictions on the choice of residual relaxations $\boldsymbol{f}^{\mathrm{cv}}/\boldsymbol{f}^{\mathrm{cc}}$, beyond requiring them to be valid relaxations. In particular, these residual relaxations need not be continuous at the boundaries of their domains, they need not be obtained by a McCormick-like procedure that traverses the computational graph of $\boldsymbol{f}$, and they need not be factorable themselves. To our knowledge, this generality is unprecedented among approaches for relaxing implicit functions, and permits the use of non-factorable relaxations such as known convex envelopes, the Scott-Barton ODE relaxations [17], Fenchel conjugates, and Moreau-Yosida regularizations [19], and convex relaxation approaches based on black-box sampling [18].

As with the Tsoukalas-Mitsos relaxations of products of nontrivial functions [15], if an implicit function is deemed to be common in global optimization algorithms and based on a simple residual function, then developing closed-form solutions for the parametric convex optimization problems (3) may be viable and useful. The subsequent corollaries will illustrate this idea when monotonicity or linearity may be exploited.

Observe that the optimization problems in (3a) and (3b) are convex optimization problems. Thus, the relaxations $\boldsymbol{x}^{\mathrm{cv}}$ and $\boldsymbol{x}^{\mathrm{cc}}$ may be evaluated using local NLP solvers such as IPOPT [26] and CONOPT [27]. Since evaluating $\boldsymbol{x}(\boldsymbol{p})$ involves solving a nonlinear equation system of similar size to the NLPs (3a) and (3b), we expect that the computational cost of evaluating $\boldsymbol{x}^{\mathrm{cv}}(\boldsymbol{p})$ and $\boldsymbol{x}^{\mathrm{cc}}(\boldsymbol{p})$ with NLP solvers is on the order of n_x times the cost of evaluating $\boldsymbol{x}(\boldsymbol{p})$. Computational complexity will be discussed in more detail in Section 3.6 below.

3.2 Exploiting Monotonocity and Low Range Dimension

The following corollaries of Theorem 3.1 show that the implicit function relaxations (3) are particularly simple to evaluate in certain cases, when we can exploit either monotonicity or linearity of the residual function relaxations, or low range dimension of the implicit function.

Corollary 3.2. Consider the setup of Theorem 3.1, suppose that $n_x = 1$, denote X as $[x^{\mathrm{L}}, x^{\mathrm{U}}]$, and consider some $\boldsymbol{p} \in P$. Suppose that f^{cv} and f^{cc} are continuous, and choose $\xi^{\mathrm{A}} \in \arg\min_{x\in X} f^{\mathrm{cv}}(x, \boldsymbol{p})$ and $\xi^{\mathrm{B}} \in \arg\max_{x\in X} f^{\mathrm{cc}}(x, \boldsymbol{p})$.

If either $0 < f^{\mathrm{cv}}(\xi^{\mathrm{A}}, \boldsymbol{p})$ or $f^{\mathrm{cc}}(\xi^{\mathrm{B}}, \boldsymbol{p}) < 0$, then the optimization problem (3a) is infeasible and $x^{\mathrm{cv}}(\boldsymbol{p}) = +\infty$. Otherwise, define a quantity $x^{\mathrm{A}} \in [x^{\mathrm{L}}, \xi^{\mathrm{A}}]$ as follows.

- If $f^{\mathrm{cv}}(x^{\mathrm{L}}, \boldsymbol{p}) \leq 0$, then set $x^{\mathrm{A}} := x^{\mathrm{L}}$.
- Otherwise, if $f^{\mathrm{cv}}(\xi^{\mathrm{A}}, \boldsymbol{p}) = 0$, then set $x^{\mathrm{A}} := \min\{x \in X : f^{\mathrm{cv}}(x, \boldsymbol{p}) \leq 0\}$.
- Otherwise, set x^{A} to be the unique root of $f^{\mathrm{cv}}(\cdot, \boldsymbol{p})$ on $[x^L, \xi^{\mathrm{A}}]$.

Define a quantity $x^{\mathrm{B}} \in [x^{\mathrm{L}}, \xi^{\mathrm{B}}]$ as follows.

- If $f^{\mathrm{cc}}(x^{\mathrm{L}}, \boldsymbol{p}) \geq 0$, then set $x^{\mathrm{B}} := x^{\mathrm{L}}$.
- Otherwise, if $f^{\mathrm{cc}}(\xi^{\mathrm{B}}, \boldsymbol{p}) = 0$, then set $x^{\mathrm{B}} := \min\{x \in X : f^{\mathrm{cc}}(x, \boldsymbol{p}) \geq 0\}$.
- Otherwise, set x^{B} to be the unique root of $f^{\mathrm{cc}}(\cdot, \boldsymbol{p})$ on $[x^{L}, \xi^{\mathrm{B}}]$.

Then $x^{\mathrm{cv}}(\boldsymbol{p}) = \max(x^{\mathrm{A}}, x^{\mathrm{B}})$.

Proof This follows immediately from Theorem 3.1 and [1, Lemma 3.1]. □

In the typical setting of an implicit function theorem, one is expected to know certain monotonicity properties of the residual function. In this vein, observe that the above corollary simplifies significantly if the problem (3a) is known to be feasible, and f^{cv} and f^{cc} are each known to be strictly monotonically increasing or strictly monotonically decreasing. In this case:

- if f^{cv} is strictly monotonically increasing, then $\xi^{\mathrm{A}} = x^{\mathrm{L}}$, and so $x^{\mathrm{A}} = x^{\mathrm{L}}$.
- If f^{cv} is strictly monotonically decreasing, then $\xi^{\mathrm{A}} = x^{\mathrm{U}}$, so:
  - if $f^{\mathrm{cv}}(x^{\mathrm{L}}, \boldsymbol{p}) \leq 0$, then $x^{\mathrm{A}} = x^{\mathrm{L}}$;
  - otherwise x^{A} is the unique root of $f^{\mathrm{cv}}(\cdot, \boldsymbol{p})$ on X.
- If f^{cc} is strictly monotonically increasing, then $\xi^{\mathrm{B}} = x^{\mathrm{U}}$, so:
  - if $f^{\mathrm{cc}}(x^{\mathrm{L}}, \boldsymbol{p}) \geq 0$, then $x^{\mathrm{B}} = x^{\mathrm{L}}$;
  - otherwise x^{B} is the unique root of $f^{\mathrm{cc}}(\cdot, \boldsymbol{p})$ on X.
- If f^{cc} is strictly monotonically decreasing, then $\xi^{\mathrm{B}} = x^{\mathrm{L}}$, and so $x^{\mathrm{B}} = x^{\mathrm{L}}$.

According to Nesterov [28], Newton's method for equation-solving is particularly efficient for finding roots of monotonic univariate functions that are either convex or concave, as is the case here. We also remark that it may be possible to exploit monotonicity in Theorem 3.1 when $n_x > 1$ by an analogous approach.

Moving on, observe that if $\boldsymbol{f}^{\mathrm{cv}}, \boldsymbol{f}^{\mathrm{cc}}$ are affine with respect to $\boldsymbol{p}$ for each fixed $\boldsymbol{z}$, then Theorem 3.1 describes $\boldsymbol{x}^{\mathrm{cv}}$ and $\boldsymbol{x}^{\mathrm{cc}}$ as the solutions of linear programs (LPs) that may be efficiently solved by standard methods. Such affine relaxations could be constructed by evaluating subgradients of nonlinear relaxations of $\boldsymbol{f}$ using either automatic differentiation [1, 29, 30] or black-box sampling [18]. Since every such LP would have n_x decision variables, $4n_x$ inequality constraints, and no equality constraints, this LP has an accessible closed-form solution when n_x is small. The following corollaries illustrate this notion when n_x is either 1 or 2, and Example 2 in Section 6 will demonstrate their application.

Corollary 3.3. Consider the setup of Theorem 3.1, suppose that $n_x = 1$, denote X as $[x^{\mathrm{L}}, x^{\mathrm{U}}]$, and choose $\boldsymbol{p} \in P$. Suppose that the functions $f^{\mathrm{cv}}(\cdot, \boldsymbol{p})$ and $f^{\mathrm{cc}}(\cdot, \boldsymbol{p})$ are both affine. Then, $x^{\mathrm{cv}}(\boldsymbol{p})$ in (3a) can be evaluated by the following procedure; a similar procedure evaluates $x^{\mathrm{cc}}(\boldsymbol{p})$ instead.

1. Define a set $S_1 := \{x^{\mathrm{L}}, x^{\mathrm{U}}\}$.
2. If the affine function $f^{\mathrm{cv}}(\cdot, \boldsymbol{p})$ is not constant, then compute a root of $f^{\mathrm{cv}}(\cdot, \boldsymbol{p})$ on X and append that root to S_1.

3. If the affine function $f^{cc}(\cdot, \boldsymbol{p})$ is not constant, then compute a root of $f^{cc}(\cdot, \boldsymbol{p})$ on X and append that root to S_1.
4. Compute $x^{cv}(\boldsymbol{p})$ as the least element $\xi \in S_1$ for which $f^{cv}(\xi, \boldsymbol{p}) \leq 0 \leq f^{cc}(\xi, \boldsymbol{p})$. If no such element exists, then set $x^{cv}(\boldsymbol{p}) := +\infty$.

Proof This follows immediately from Theorem 3.1. □

In Step 4 of the above corollary's evaluation procedure, observe that S_1 will have no more than four elements, and so an optimization solver is not required to carry out this step.

Corollary 3.4. Consider the setup of Theorem 3.1, suppose that $n_x = 2$, denote X as $[\boldsymbol{x}^L, \boldsymbol{x}^U]$, and choose $\boldsymbol{p} \in P$. Suppose that the functions $\boldsymbol{f}^{cv}(\cdot, \boldsymbol{p})$ and $\boldsymbol{f}^{cc}(\cdot, \boldsymbol{p})$ are both affine. Then, for any $\boldsymbol{p} \in P$, $\boldsymbol{x}^{cv}(\boldsymbol{p})$ in (3a) and $\boldsymbol{x}^{cc}(\boldsymbol{p})$ are be evaluated by the following steps.

1. Remove any redundant equations among the following eight linear equations in $\boldsymbol{\xi} \in \mathbb{R}^2$.

$$\begin{aligned} f_1^{cv}(\boldsymbol{\xi}, \boldsymbol{p}) &= 0, & \xi_1 - x_1^L &= 0, \\ f_2^{cv}(\boldsymbol{\xi}, \boldsymbol{p}) &= 0, & \xi_2 - x_2^L &= 0, \\ f_1^{cc}(\boldsymbol{\xi}, \boldsymbol{p}) &= 0, & \xi_1 - x_1^U &= 0, \\ f_2^{cc}(\boldsymbol{\xi}, \boldsymbol{p}) &= 0, & \xi_2 - x_2^U &= 0. \end{aligned}$$

   Let E denote the collection of remaining equations.
2. Define S_2 to be the empty set.
3. For each pair of linear equations in E (there are $\binom{8}{2} = 28$ such pairs), solve this pair for $\boldsymbol{\xi} \in \mathbb{R}^2$ (if possible). If this solution $\boldsymbol{\xi}$ satisfies

$$\boldsymbol{f}^{cv}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \mathbf{0} \leq \boldsymbol{f}^{cc}(\boldsymbol{\xi}, \boldsymbol{p}),$$

   then append $\boldsymbol{\xi}$ to S_2.
4. If S_2 is empty, then set $x_i^{cv}(\boldsymbol{p}) := +\infty$ and $x_i^{cc}(\boldsymbol{p}) := -\infty$ for each $i \in \{1, 2\}$. Otherwise, for each $i \in \{1, 2\}$, evaluate:

$$\begin{aligned} x_i^{cv}(\boldsymbol{p}) &:= \min\{\xi_i : \boldsymbol{\xi} \in S_2\}, \\ x_i^{cc}(\boldsymbol{p}) &:= \max\{\xi_i : \boldsymbol{\xi} \in S_2\}. \end{aligned}$$

Proof This result follows from Theorem 3.1, noting that if an LP has a bounded feasible set, then its solution is attained at an extreme point of its feasible set. □

In this corollary, S_2 can never include more than 28 points. Thus, by inspection, each step of this evaluation approach is tractable.

3.3 Convergence as Domain Shrinks

To be useful in branch-and-bound methods for global optimization, a scheme of convex relaxations should converge to the original function as the domain P is shrunk to a singleton set. This section demonstrates this convergence when the new implicit-function relaxations of Theorem 3.1 are coupled with a convergent interval method for generating the range estimate X. As noted after Assumption 2 below, such interval methods do indeed exist. In the following assumption, limits of sets are defined in terms of the Hausdorff metric.

The following assumption roughly requires that, as we choose smaller and smaller subsets of P, converging on some element of Q, then our supplied interval bounds on the implicit function $\boldsymbol{x}$ must also converge.

Assumption 2. Consider the setup of Theorem 3.1. For each $q \in \mathbb{N}$, consider sets $\Pi(q) \subset P$ and $\Xi(q) \subset X$, a convex relaxation $\boldsymbol{f}^{\mathrm{cv}}_{(q)} : \Xi(q) \times \Pi(q) \to \mathbb{R}^{n_x}$ of $\boldsymbol{f}$ on $\Xi(q) \times \Pi(q)$, and a concave relaxation $\boldsymbol{f}^{\mathrm{cc}}_{(q)} : \Xi(q) \times \Pi(q) \to \mathbb{R}^{n_x}$ of $\boldsymbol{f}$ on $\Xi(q) \times \Pi(q)$. For some $\bar{\boldsymbol{p}} \in Q$, assume that both of the following conditions hold:

1. For each sufficiently large $q \in \mathbb{N}$, the set $\Pi(q) \cap Q$ is nonempty.
2. For each $q \in \mathbb{N}$ and $\boldsymbol{p} \in \Pi(q) \cap Q$, there exists $\boldsymbol{\xi} \in \Xi(q)$ for which $\boldsymbol{f}(\boldsymbol{\xi}, \boldsymbol{p}) = \mathbf{0}$.
3. $\lim_{q\to\infty} \Xi(q) = [\boldsymbol{x}(\bar{\boldsymbol{p}}), \boldsymbol{x}(\bar{\boldsymbol{p}})]$.

Observe that Conditions 1 and 2 of Assumption 2 are trivially satisfied if $\bar{\boldsymbol{p}} \in \Pi(q)$ and $\boldsymbol{x}(\bar{\boldsymbol{p}}) \in \Xi(q)$ for each $q \in \mathbb{N}$. Moreover, Condition 3 of Assumption 2 implies that the supplied interval bounds $\Xi(q)$ of the implicit function's range converge as $q \to \infty$. If the implicit function $\boldsymbol{x}$ is unique, and if $\lim_{q\to\infty} \Pi(q) = [\bar{\boldsymbol{p}}, \bar{\boldsymbol{p}}]$, then such bounds might be constructed automatically by several established interval methods, including the interval Newton method [31] and the interval Krawczyk method [24]. These particular methods require $\boldsymbol{f}$ to be Lipschitz continuous. However, if there are multiple solutions $\boldsymbol{z} \in X$ of (1) when $\boldsymbol{p} := \bar{\boldsymbol{p}}$, then, by construction, the implicit function relaxations of Theorem 3.1 will enclose all of them, and Condition 3 of Assumption 2 is unlikely to be satisfied if established interval methods are used to generate the sets $\Xi(q)$. This "nonuniqueness gap" can be averted by ensuring that the implicit function is indeed unique, perhaps by shrinking X and/or by appending additional equations to the system (1) to specify which single solution is intended.

Theorem 3.5. Under Assumption 2, for each $q \in \mathbb{N}$, let $\boldsymbol{x}^{\mathrm{cv}}_{(q)}$ and $\boldsymbol{x}^{\mathrm{cc}}_{(q)}$ denote the implicit function relaxations described in Theorem 3.1 with $\Pi(q)$ in place of P, with $\Xi(q)$ in place of X, and with $\boldsymbol{f}^{\mathrm{cv}}_{(q)}/\boldsymbol{f}^{\mathrm{cc}}_{(q)}$ in place of $\boldsymbol{f}^{\mathrm{cv}}/\boldsymbol{f}^{\mathrm{cc}}$. Then, for each $i \in \{1, \ldots, n_x\}$,

$$\liminf_{q\to\infty} \inf_{\boldsymbol{p}\in\Pi(q)} x^{\mathrm{cv}}_{(q),i}(\boldsymbol{p}) = x_i(\bar{\boldsymbol{p}}), \qquad \text{and} \qquad \limsup_{q\to\infty} \sup_{\boldsymbol{p}\in\Pi(q)} x^{\mathrm{cc}}_{(q),i}(\boldsymbol{p}) = x_i(\bar{\boldsymbol{p}}).$$

Proof The limit involving $\boldsymbol{x}^{\text{cv}}_{(q)}$ will be demonstrated; the limit involving $\boldsymbol{x}^{\text{cc}}_{(q)}$ follows from an analogous argument. Pick some $i \in \{1, \ldots, n_x\}$. For each $q \in \mathbb{N}$, denote $\Xi(q)$ as $[\boldsymbol{\xi}^{\text{L}}(q), \boldsymbol{\xi}^{\text{U}}(q)]$, and define:

$$\hat{x}_i^*(q) := \inf_{\boldsymbol{p} \in \Pi(q)} x^{\text{cv}}_{(q),i}(\boldsymbol{p}).$$

So, for each $q \in \mathbb{N}$,

$$\hat{x}_i^*(q) = \inf_{\boldsymbol{p} \in \Pi(q), \boldsymbol{\xi} \in \Xi(q)} \{\xi_i \mid \boldsymbol{f}^{\text{cv}}_{(q)}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \mathbf{0} \leq \boldsymbol{f}^{\text{cc}}_{(q)}(\boldsymbol{\xi}, \boldsymbol{p})\}.$$

By Assumption 2, for each sufficiently large $q \in \mathbb{N}$, the set $\{\boldsymbol{\xi} \in \Xi(q) \mid \boldsymbol{p} \in \Pi(q), \boldsymbol{f}(\boldsymbol{\xi}, \boldsymbol{p}) = \mathbf{0}\}$ will be nonempty. Thus:

$$\begin{aligned}
\liminf_{q \to \infty} \hat{x}_i^*(q) &= \liminf_{q \to \infty} \inf_{\boldsymbol{p} \in \Pi(q), \boldsymbol{\xi} \in \Xi(q)} \{\xi_i \mid \boldsymbol{f}^{\text{cv}}_{(q)}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \mathbf{0} \leq \boldsymbol{f}^{\text{cc}}_{(q)}(\boldsymbol{\xi}, \boldsymbol{p})\} \\
&\leq \liminf_{q \to \infty} \inf_{\boldsymbol{p} \in \Pi(q), \boldsymbol{\xi} \in \Xi(q)} \{\xi_i \mid \boldsymbol{f}(\boldsymbol{\xi}, \boldsymbol{p}) = \mathbf{0}\} \\
&\leq \liminf_{q \to \infty} \sup_{\boldsymbol{\xi} \in \Xi(q)} \xi_i \\
&= \liminf_{q \to \infty} \xi_i^{\text{U}}(q) \\
&= x_i(\bar{\boldsymbol{p}}),
\end{aligned}$$

and:

$$\begin{aligned}
\liminf_{q \to \infty} \hat{x}_i^*(q) &= \liminf_{q \to \infty} \inf_{\boldsymbol{p} \in \Pi(q), \boldsymbol{\xi} \in \Xi(q)} \{\xi_i \mid \boldsymbol{f}^{\text{cv}}_{(q)}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \mathbf{0} \leq \boldsymbol{f}^{\text{cc}}_{(q)}(\boldsymbol{\xi}, \boldsymbol{p})\} \\
&\geq \liminf_{q \to \infty} \inf_{\boldsymbol{\xi} \in \Xi(q)} \xi_i \\
&= \liminf_{q \to \infty} \xi_i^{\text{L}}(q) \\
&= x_i(\bar{\boldsymbol{p}}),
\end{aligned}$$

as required. □

Though not required by Assumption 2, we suspect that analogous convergence of the supplied relaxations of $\boldsymbol{f}$ may improve the rate of convergence. Several established relaxation methods produce relaxations that converge rapidly to the original function as $q \to \infty$; these methods include the McCormick relaxations [32], the αBB relaxations [33], various piecewise-affine variants of these [34], and recent relaxations of parametric ODE solutions [17, 35], and certain sampling-based affine relaxations of any of these [18].

3.4 Relaxing Composite Implicit Functions

In this section, we extend Theorem 3.1 to generate convex and concave relaxations for a composition of an implicit outer function with a known inner function, supposing that the convex and concave relaxations of the inner function are available. This construction proceeds by combining (3) with the Tsoukalas-Mitsos relaxations of composite functions [15]. Coupling a relaxation method with a convergent interval method is an established step in global optimization applications [32].

Theorem 3.6. Consider the setup of Theorem 3.1, and assume additionally that $\boldsymbol{f}$ is continuous and that $Q = P$ (i.e. an implicit function $\boldsymbol{x}$ is defined on P). Suppose there is a compact convex set $W \subset \mathbb{R}^{n_w}$, a continuous function $\boldsymbol{r} : W \to P$, and convex/concave relaxations $\boldsymbol{r}^{\mathrm{cv}}, \boldsymbol{r}^{\mathrm{cc}} : W \to P$ of $\boldsymbol{r}$ on W. Then there is an implicit function $\boldsymbol{y} : W \to X$ for which

$$\boldsymbol{f}(\boldsymbol{y}(\boldsymbol{w}), \boldsymbol{r}(\boldsymbol{w})) = \boldsymbol{0}, \qquad \forall \boldsymbol{w} \in W. \tag{5}$$

Define functions $\boldsymbol{y}^{\mathrm{cv}}, \boldsymbol{y}^{\mathrm{cc}} : W \to X$ so that, for each $i \in \{1, \ldots, n_w\}$ and $\boldsymbol{w} \in W$,

$$\begin{aligned} y_i^{\mathrm{cv}}(\boldsymbol{w}) = \min_{\boldsymbol{\xi} \in X,\, \boldsymbol{p} \in P} \ & \xi_i \\ \text{subject to} \ & \boldsymbol{f}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}) \le \boldsymbol{0} \le \boldsymbol{f}^{\mathrm{cc}}(\boldsymbol{\xi}, \boldsymbol{p}), \\ & \boldsymbol{r}^{\mathrm{cv}}(\boldsymbol{w}) \le \boldsymbol{p} \le \boldsymbol{r}^{\mathrm{cc}}(\boldsymbol{w}), \\ \text{and} \qquad y_i^{\mathrm{cc}}(\boldsymbol{w}) = \max_{\boldsymbol{\xi} \in X,\, \boldsymbol{p} \in P} \ & \xi_i \\ \text{subject to} \ & \boldsymbol{f}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}) \le \boldsymbol{0} \le \boldsymbol{f}^{\mathrm{cc}}(\boldsymbol{\xi}, \boldsymbol{p}), \\ & \boldsymbol{r}^{\mathrm{cv}}(\boldsymbol{w}) \le \boldsymbol{p} \le \boldsymbol{r}^{\mathrm{cc}}(\boldsymbol{w}). \end{aligned}$$

Then $\boldsymbol{y}^{\mathrm{cv}}$ is a convex relaxation of $\boldsymbol{y}$ on W, and $\boldsymbol{y}^{\mathrm{cc}}$ is a concave relaxation of $\boldsymbol{y}$ on W.

Proof Since $\boldsymbol{x}$ is assumed to exist on P, the composition $\boldsymbol{y} := \boldsymbol{x} \circ \boldsymbol{r}$ satisfies (5). The remaining claims follow immediately from applying [15, Theorem 2] to the composition $\boldsymbol{x} \circ \boldsymbol{r}$, with $\boldsymbol{x}$ relaxed according to Theorem 3.1. □

As before, observe that the optimization problems defining $\boldsymbol{y}^{\mathrm{cv}}, \boldsymbol{y}^{\mathrm{cc}}$ become LPs if we employ affine relaxations of the residual function $\boldsymbol{f}$ and the inner function $\boldsymbol{r}$.

3.5 Relaxations of Inverse Functions

Since implicit functions are closely related to inverse functions, Theorem 3.1 may be adapted to relax inverse functions instead. Given convex compact sets $P, X \subset \mathbb{R}^{n_x}$, suppose that $\boldsymbol{v} : X \to P$ is an invertible function. So, there exists an inverse function $\boldsymbol{v}^{-1} : P \to X$ of $\boldsymbol{v}$ for which, for each $\boldsymbol{p} \in P$,

$$\boldsymbol{v}(\boldsymbol{v}^{-1}(\boldsymbol{p})) = \boldsymbol{p}.$$

The inverse function $\boldsymbol{v}^{-1}$ may also be written as an implicit function satisfying the following equation system in the form (2):

$$\boldsymbol{v}(\boldsymbol{v}^{-1}(\boldsymbol{p})) - \boldsymbol{p} = \boldsymbol{0}, \qquad \forall \boldsymbol{p} \in P.$$

Hence, convex and concave relaxations of $\boldsymbol{v}^{-1}$ on P may be constructed by adapting (3) as follows.

Corollary 3.7. Let $\boldsymbol{v}^{\mathrm{cv}}, \boldsymbol{v}^{\mathrm{cc}} : X \to P$ be convex and concave relaxations of $\boldsymbol{v}$ on X, respectively. Consider functions $\boldsymbol{v}^{-\mathrm{cv}}, \boldsymbol{v}^{-\mathrm{cc}} : P \to X$ such that, for each $i \in \{1, \ldots, n_x\}$ and $\boldsymbol{p} \in P$,

$$v_i^{-\mathrm{cv}}(\boldsymbol{p}) = \min_{\boldsymbol{\xi} \in X} \ \xi_i \quad \text{subject to} \quad \boldsymbol{v}^{\mathrm{cv}}(\boldsymbol{\xi}) \leq \boldsymbol{p} \leq \boldsymbol{v}^{\mathrm{cc}}(\boldsymbol{\xi}), \tag{6a}$$

$$\text{and} \qquad v_i^{-\mathrm{cc}}(\boldsymbol{p}) = \max_{\boldsymbol{\xi} \in X} \ \xi_i \quad \text{subject to} \quad \boldsymbol{v}^{\mathrm{cv}}(\boldsymbol{\xi}) \leq \boldsymbol{p} \leq \boldsymbol{v}^{\mathrm{cc}}(\boldsymbol{\xi}). \tag{6b}$$

Then, $\boldsymbol{v}^{-\mathrm{cv}}$ is a convex relaxation of the inverse function $\boldsymbol{v}^{-1}$ on P, and $\boldsymbol{v}^{-\mathrm{cc}}$ is a concave relaxation of $\boldsymbol{v}^{-1}$ on P.

Proof It suffices to show that the hypotheses in Theorem 3.1 hold with $\boldsymbol{v}^{-1}$ in place of $\boldsymbol{x}$, with $\boldsymbol{v}^{-\mathrm{cv}}, \boldsymbol{v}^{-\mathrm{cc}}$ respectively in place of $\boldsymbol{x}^{\mathrm{cv}}, \boldsymbol{x}^{\mathrm{cc}}$, with $\boldsymbol{f}(\boldsymbol{z}, \boldsymbol{p}) \equiv \boldsymbol{v}(\boldsymbol{z}) - \boldsymbol{p}$, and with relaxations of $\boldsymbol{f}$ to be furnished below. Since $\boldsymbol{v}$ is invertible and $\boldsymbol{v}^{-1}$ is its inverse, Assumption 1 is satisfied with $Q := P$.

Now, define functions $\bar{\boldsymbol{f}}^{\mathrm{cv}}, \bar{\boldsymbol{f}}^{\mathrm{cc}} : X \times P \to P$ such that, for each $\boldsymbol{p} \in P$,

$$\bar{\boldsymbol{f}}^{\mathrm{cv}}(\boldsymbol{z}, \boldsymbol{p}) = \boldsymbol{v}^{\mathrm{cv}}(\boldsymbol{z}) - \boldsymbol{p},$$
$$\bar{\boldsymbol{f}}^{\mathrm{cc}}(\boldsymbol{z}, \boldsymbol{p}) = \boldsymbol{v}^{\mathrm{cc}}(\boldsymbol{z}) - \boldsymbol{p}.$$

Observe that the constraints $\boldsymbol{v}^{\mathrm{cv}}(\boldsymbol{\xi}) \leq \boldsymbol{p} \leq \boldsymbol{v}^{\mathrm{cc}}(\boldsymbol{\xi})$ in (6) are equivalent to $\bar{\boldsymbol{f}}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \mathbf{0} \leq \bar{\boldsymbol{f}}^{\mathrm{cc}}(\boldsymbol{\xi}, \boldsymbol{p})$. Since $\boldsymbol{v}^{\mathrm{cv}}$ and $\boldsymbol{v}^{\mathrm{cc}}$ are convex and concave relaxations of $\boldsymbol{v}$ on X, respectively, it follows that $\bar{\boldsymbol{f}}^{\mathrm{cv}}$ and $\bar{\boldsymbol{f}}^{\mathrm{cc}}$ are respective convex and concave relaxations of $\boldsymbol{f}$. Thus, all hypotheses of Theorem 3.1 are satisfied, and this theorem yields the claimed result. □

As was the case in Theorem 3.1, observe that the optimization problems in (6a) and (6b) are convex NLPs, which may be solved with local NLP solvers. Moreover, if $\boldsymbol{v}^{\mathrm{cv}}, \boldsymbol{v}^{\mathrm{cc}}$ are chosen to be affine or piecewise-affine relaxations, then (6a) and (6b) may be formulated as linear programs (LPs) and solved efficiently.

3.6 Computational Complexity

The computational expense of evaluating our implicit function relaxations depends heavily on how the convex optimization problems (3) in Theorem 3.1 are solved. An evaluation of the $(\boldsymbol{x}^{\mathrm{cv}}(\boldsymbol{p}), \boldsymbol{x}^{\mathrm{cc}}(\boldsymbol{p}))$ pair will involve solving $2n_x$ optimization problems, each with n_x decision variables, bound constraints on each decision variable, and $2n_x$ additional convex inequality constraints. In this section we discuss this computational expense qualitatively and somewhat roughly; corresponding CPU times for certain numerical experiments are reported in Section 6.

In general, we expect the cost of evaluating our relaxations to be independent of the domain dimension n_p of the implicit function, since $\boldsymbol{p}$ is held constant in our relaxation formulation. By inspection, this is also the case for the established implicit function relaxation approaches by Stuber et al. [10] and Wechsung et al. [8].

If a local nonlinear programming (NLP) solver is employed directly to solve (3), then we would expect the computational expense of evaluating the new relaxations to be dominated by the evaluations of $\boldsymbol{f}^{\mathrm{cv}}$ and $\boldsymbol{f}^{\mathrm{cc}}$ required by the NLP solver, along with any subgradients, gradients, and/or Hessians. In general, if an implicit function is well-defined, we would expect each NLP solve to incur comparable computational expense to evaluating the implicit function $\boldsymbol{x}$ by a nonlinear equation-solve.

Like our implicit function relaxations, the Tsoukalas-Mitsos relaxations of composite functions [15] are described as optimal-value functions for certain parametric convex optimization problems based on supplied relaxations. In Tsoukalas and Mitsos's development, however, these relaxations are intended to be obtained as a general closed-form solution that covers all choices of $\boldsymbol{p}$ and X, so that the end user can employ this closed-form solution without requiring a numerical optimization solver. As we showed in Section 3.1, monotonicity properties of $\boldsymbol{f}^{\mathrm{cv}}$ and $\boldsymbol{f}^{\mathrm{cc}}$ can be exploited to obtain such closed-form solutions in (3) when $n_x = 1$. For simple implicit functions that occur often in applications, it may be worth proceeding analogously to [15], to obtain closed-form solutions for the optimization problems appearing in (3) in advance.

As was discussed in Section 3.1, useful affine relaxations $\boldsymbol{f}^{\mathrm{cv}}$ and $\boldsymbol{f}^{\mathrm{cc}}$ in (3) may be constructed from subgradients or black-box samples of supplied nonlinear relaxations. Roughly, let $\mathcal{C}$ denote the computational cost of evaluating such a nonlinear relaxation once. Following a complexity analysis of automatic differentiation (AD) by Griewank and Walther [14], we expect that evaluating a subgradient by the reverse AD mode [29] would cost approximately $10\mathcal{C}$, while the simpler forward AD mode [1] would cost approximately $3n_x\mathcal{C}$. The black-box sampling approach of Song et al. [18] constructs an affine relaxation from $(2n_x + 1)$ nonlinear relaxation evaluations, and therefore costs approximately $2n_x\mathcal{C}$. Once affine relaxations are employed throughout (3), the optimization problems in this formulation become linear programs (LPs) that are efficiently solved in practice. As we have already shown, solving this LP is trivial when $n_x = 1$, and when $n_x = 2$ the LP has a closed-form solution based on solving 28 linear equation systems in two variables, and could presumably be solved even faster using the simplex method.

We also note that if the residual function $\boldsymbol{f}$ is quadratic in $\boldsymbol{z}$ for each fixed $\boldsymbol{p}$, and if αBB relaxations [2] $\boldsymbol{f}^{\mathrm{cv}}/\boldsymbol{f}^{\mathrm{cc}}$ are employed, then the optimization problems in (3) become convex quadratically-constrained quadratic programs (QCQPs), which are also efficiently solved.

By comparison, previous implicit function relaxation approaches [8, 10] proceed by performing successive tightening iterations, each of which involves traversing the residual function's computational graph once. In the case of [10], these iterations are directly analogous to the iterations of a nonlinear equation solver used to evaluate $\boldsymbol{x}$. Hence, we expect that the computational expense of these approaches does not scale directly with n_x, but instead with the length of the residual function's computational graph, and the number of tightening iterations desired.

4 Convex Relaxations of Constraint Satisfaction Problems

In this section, we generalize the convex relaxation methodology described in Theorem 3.1 to constraint satisfaction problems (CSPs). Relaxations will be presented for point-to-set mappings describing parametric CSPs, and the directional derivatives of these relaxations will be considered as well. Throughout this section, consider continuously differentiable mappings $\boldsymbol{g} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_g}$ and $\boldsymbol{h} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_h}$. Unlike the function $\boldsymbol{f}$ considered in Section 3, the dimensions of the codomains of $\boldsymbol{g}$ and $\boldsymbol{h}$ are arbitrary and may be distinct from n_x. Given known intervals $X \in \mathbb{IR}^{n_x}$ and $P \in \mathbb{IR}^{n_p}$, consider the following CSP:

$$\begin{aligned} \min_{\boldsymbol{z}\in X,\ \boldsymbol{p}\in P} \quad & 0 \\ \text{subject to} \quad & \boldsymbol{g}(\boldsymbol{z},\boldsymbol{p}) \leq \boldsymbol{0}, \\ & \boldsymbol{h}(\boldsymbol{z},\boldsymbol{p}) = \boldsymbol{0}. \end{aligned} \tag{7}$$

Let the set of $\boldsymbol{z}$-values in X be expressed as a point-to-set map Ξ from $\mathbb{R}^{n_p}$ to $\mathbb{R}^{n_x}$ such that, for each $\boldsymbol{p} \in P$,

$$\Xi(\boldsymbol{p}) := \{\boldsymbol{\xi} \in X \mid \boldsymbol{g}(\boldsymbol{\xi},\boldsymbol{p}) \leq \boldsymbol{0},\ \boldsymbol{h}(\boldsymbol{\xi},\boldsymbol{p}) = \boldsymbol{0}\}. \tag{8}$$

Observe that Ξ generalizes the implicit functions $\boldsymbol{x}$ considered in Section 3.1.

Let $\boldsymbol{g}^{\mathrm{cv}} : X \times P \to \mathbb{R}^{n_g}$ be a convex relaxation of $\boldsymbol{g}$ on $X \times P$, and let $\boldsymbol{h}^{\mathrm{cv}}, \boldsymbol{h}^{\mathrm{cc}} : X \times P \to \mathbb{R}^{n_h}$ be respective convex and concave relaxations of $\boldsymbol{h}$ on $X \times P$, respectively. Define $\boldsymbol{\xi}^{\mathrm{cv}}, \boldsymbol{\xi}^{\mathrm{cc}} : P \to \mathbb{R}^{n_x}$ such that, for each $i \in \{1, \ldots, n_x\}$ and $\boldsymbol{p} \in P$,

$$\begin{aligned} \xi_i^{\mathrm{cv}}(\boldsymbol{p}) = \quad \min_{\boldsymbol{\xi}\in X} \quad & \xi_i \\ \text{subject to} \quad & \boldsymbol{g}^{\mathrm{cv}}(\boldsymbol{\xi},\boldsymbol{p}) \leq \boldsymbol{0}, \\ & \boldsymbol{h}^{\mathrm{cv}}(\boldsymbol{\xi},\boldsymbol{p}) \leq \boldsymbol{0} \leq \boldsymbol{h}^{\mathrm{cc}}(\boldsymbol{\xi},\boldsymbol{p}), \end{aligned} \tag{9a}$$

$$\begin{aligned} \xi_i^{\mathrm{cc}}(\boldsymbol{p}) = \quad \max_{\boldsymbol{\xi}\in X} \quad & \xi_i \\ \text{subject to} \quad & \boldsymbol{g}^{\mathrm{cv}}(\boldsymbol{\xi},\boldsymbol{p}) \leq \boldsymbol{0}, \\ & \boldsymbol{h}^{\mathrm{cv}}(\boldsymbol{\xi},\boldsymbol{p}) \leq \boldsymbol{0} \leq \boldsymbol{h}^{\mathrm{cc}}(\boldsymbol{\xi},\boldsymbol{p}). \end{aligned} \tag{9b}$$

The optimization problems in (9a) and (9b) are convex NLPs, which are typically easier to solve to global optimality than the original nonconvex CSP (7).

Define an interval-valued point-to-set map $\Xi^R : P \rightrightarrows \mathbb{R}^{n_x}$ such that, for each $\boldsymbol{p} \in P$,

$$\Xi^R(\boldsymbol{p}) \equiv [\boldsymbol{\xi}^{\text{cv}}(\boldsymbol{p}), \boldsymbol{\xi}^{\text{cc}}(\boldsymbol{p})].$$

We will show that Ξ^R is a convex relaxation of Ξ on P in the sense of Definition 2.

Theorem 4.1. Let Q be a subset of P such that for each $\boldsymbol{p} \in Q$, $\Xi(\boldsymbol{p})$ is nonempty. Suppose that Q is nonempty. Then, Ξ^R is a convex relaxation of Ξ on P.

Proof According to Definition 2, we will proceed by showing that $\Xi(\boldsymbol{p}) \subseteq \Xi^R(\boldsymbol{p})$ for each $\boldsymbol{p} \in P$, and that Ξ^R is convex on P.

First, choose any $\boldsymbol{p} \in P$. If $\boldsymbol{p} \notin Q$, then $\Xi(\boldsymbol{p}) = \emptyset$ and therefore $\Xi(\boldsymbol{p}) \subset \Xi^R(\boldsymbol{p})$. Otherwise, $\Xi(\boldsymbol{p})$ is nonempty, and we may consider some arbitrary $\boldsymbol{z} \in \Xi(\boldsymbol{p})$. For any $i \in \{1, \ldots, n_x\}$,

$$\begin{aligned} \xi_i^{\text{cv}}(\boldsymbol{p}) &= \min_{\boldsymbol{\xi} \in X} \{\xi_i \mid \boldsymbol{g}^{\text{cv}}(\boldsymbol{\xi}, \boldsymbol{p}) \le \boldsymbol{0}, \quad \boldsymbol{h}^{\text{cv}}(\boldsymbol{\xi}, \boldsymbol{p}) \le \boldsymbol{0} \le \boldsymbol{h}^{\text{cc}}(\boldsymbol{\xi}, \boldsymbol{p})\} \\ &\le \min_{\boldsymbol{\xi} \in X} \{\xi_i \mid \boldsymbol{g}(\boldsymbol{\xi}, \boldsymbol{p}) \le \boldsymbol{0}, \quad \boldsymbol{h}(\boldsymbol{\xi}, \boldsymbol{p}) = \boldsymbol{0}\} \\ &\le z_i. \end{aligned}$$

It is analogous to show that $\xi_i^{\text{cc}}(\boldsymbol{p}) \ge z_i$ for each $i \in \{1, \ldots, n_x\}$. Hence, $\boldsymbol{\xi}^{\text{cv}}(\boldsymbol{p}) \le \boldsymbol{z} \le \boldsymbol{\xi}^{\text{cc}}(\boldsymbol{p})$, and so $\boldsymbol{z} \in \Xi^R(\boldsymbol{p})$. Thus, $\Xi(\boldsymbol{p})$ is a subset of $\Xi^R(\boldsymbol{p})$ for each $\boldsymbol{p} \in P$.

Next, we demonstrate the convexity of Ξ^R on P. Define $\boldsymbol{\phi} : X \times P \to \mathbb{R}^{n_g} \times \mathbb{R}^{n_h} \times \mathbb{R}^{n_h}$ such that, for each $\boldsymbol{\xi} \in X$ and $\boldsymbol{p} \in P$, $\boldsymbol{\phi}(\boldsymbol{\xi}, \boldsymbol{p}) = (\boldsymbol{g}^{\text{cv}}(\boldsymbol{\xi}, \boldsymbol{p}), \boldsymbol{h}^{\text{cv}}(\boldsymbol{\xi}, \boldsymbol{p}), -\boldsymbol{h}^{\text{cc}}(\boldsymbol{\xi}, \boldsymbol{p}))$, which is convex on $X \times P$. For each $i \in \{1, \ldots, n_x\}$, (9a) is equivalent to

$$\xi_i^{\text{cv}}(\boldsymbol{p}) = \min_{\boldsymbol{\xi} \in X} \ \xi_i \quad \text{subject to} \quad \boldsymbol{\phi}(\boldsymbol{\xi}, \boldsymbol{p}) \le \boldsymbol{0}. \tag{10}$$

Observe that any point $\boldsymbol{\xi} \in \Xi(\boldsymbol{p})$ is feasible in the optimization problem (10). Since the objective function of (10) is linear, $\boldsymbol{\phi}$ is convex on $X \times P$, and X, P are convex, the convexity of ξ_i^{cv} on P follows from Proposition 2.1. It is analogous to show that ξ_i^{cc} is concave on P.

Consider any $\boldsymbol{p}_A, \boldsymbol{p}_B \in P$ and $\lambda \in (0, 1)$. The convexity of $\boldsymbol{\xi}^{\text{cv}}$ and the concavity of $\boldsymbol{\xi}^{\text{cc}}$ ensure that

$$\begin{aligned} \lambda \boldsymbol{\xi}^{\text{cv}}(\boldsymbol{p}_A) + (1-\lambda)\boldsymbol{\xi}^{\text{cv}}(\boldsymbol{p}_B) &\ge \boldsymbol{\xi}^{\text{cv}}(\lambda \boldsymbol{p}_A + (1-\lambda)\boldsymbol{p}_B), \\ \lambda \boldsymbol{\xi}^{\text{cc}}(\boldsymbol{p}_A) + (1-\lambda)\boldsymbol{\xi}^{\text{cc}}(\boldsymbol{p}_B) &\le \boldsymbol{\xi}^{\text{cc}}(\lambda \boldsymbol{p}_A + (1-\lambda)\boldsymbol{p}_B). \end{aligned}$$

Consider any $\boldsymbol{z}_{\boldsymbol{p}_A} \in \Xi(\boldsymbol{p}_A)$ and $\boldsymbol{z}_{\boldsymbol{p}_B} \in \Xi(\boldsymbol{p}_B)$. $\Xi(\boldsymbol{p})$ being a subset of $\Xi^R(\boldsymbol{p})$ for each $\boldsymbol{p} \in P$ ensures that $\boldsymbol{z}_{\boldsymbol{p}_A} \in \Xi^R(\boldsymbol{p}_A)$ and $\boldsymbol{z}_{\boldsymbol{p}_B} \in \Xi^R(\boldsymbol{p}_B)$. Then,

$$\begin{aligned} \lambda \boldsymbol{z}_{\boldsymbol{p}_A} + (1-\lambda)\boldsymbol{z}_{\boldsymbol{p}_B} &\ge \lambda \boldsymbol{\xi}^{\text{cv}}(\boldsymbol{p}_A) + (1-\lambda)\boldsymbol{\xi}^{\text{cv}}(\boldsymbol{p}_B) \ge \boldsymbol{\xi}^{\text{cv}}(\lambda \boldsymbol{p}_A + (1-\lambda)\boldsymbol{p}_B), \\ \lambda \boldsymbol{z}_{\boldsymbol{p}_A} + (1-\lambda)\boldsymbol{z}_{\boldsymbol{p}_B} &\le \lambda \boldsymbol{\xi}^{\text{cc}}(\boldsymbol{p}_A) + (1-\lambda)\boldsymbol{\xi}^{\text{cc}}(\boldsymbol{p}_B) \le \boldsymbol{\xi}^{\text{cc}}(\lambda \boldsymbol{p}_A + (1-\lambda)\boldsymbol{p}_B), \end{aligned}$$

which shows that

$$\lambda \boldsymbol{z}_{\boldsymbol{p}_A} + (1-\lambda)\boldsymbol{z}_{\boldsymbol{p}_B} \in \Xi^R(\lambda \boldsymbol{p}_A + (1-\lambda)\boldsymbol{p}_B).$$

Since $\lambda, \boldsymbol{z}_{\boldsymbol{p}_A}, \boldsymbol{z}_{\boldsymbol{p}_B}$ were arbitrarily chosen, and since $\lambda \boldsymbol{z}_{\boldsymbol{p}_A} + (1-\lambda)\boldsymbol{z}_{\boldsymbol{p}_B}$ is an arbitrary point in the Minkowski sum $\lambda \Xi^R(\boldsymbol{p}_A) + (1-\lambda)\Xi^R(\boldsymbol{p}_B)$, it follows that

$$\lambda \Xi^R(\boldsymbol{p}_A) + (1-\lambda)\Xi^R(\boldsymbol{p}_B) \subset \Xi^R(\lambda \boldsymbol{p}_A + (1-\lambda)\boldsymbol{p}_B).$$

Thus, according to Definition 2, Ξ^R is convex on P. □

4.1 Directional Derivatives

In Theorem 4.1, we constructed convex and concave functions to enclose the point-to-set mapping defined by a CSP, generalizing the earlier implicit function relaxations of Theorem 3.1. In global optimization methods based on a branch-and-bound framework, determining useful lower bounds requires minimizing convex relaxations, and methods for computing these minima typically require sensitivity information that describes how those relaxations vary with $\boldsymbol{p}$. We conjecture that it may be possible to describe subgradients of the mappings $\boldsymbol{\xi}^{\mathrm{cv}}, \boldsymbol{\xi}^{\mathrm{cc}}$ in general, since these relaxations are particularly well-behaved due to the linear objective functions and convex inequality constraints of (9). However, such a subgradient description does not appear to follow immediately from existing parametric sensitivity theory.

In lieu of a resolution to this this subgradient conjecture, we observe that directional derivatives of the mappings $\boldsymbol{\xi}^{\mathrm{cv}}, \boldsymbol{\xi}^{\mathrm{cc}}$ are described by [36] under additional second-order optimality assumptions; these assumptions are somewhat onerous but are standard in parametric programming. Similar to the proof of Theorem 4.1, consider a function $\boldsymbol{\phi} : X \times P \to \mathbb{R}^{n_g} \times \mathbb{R}^{n_h} \times \mathbb{R}^{n_h}$ such that, for each $\boldsymbol{\xi} \in X$ and $\boldsymbol{p} \in P$, $\boldsymbol{\phi}(\boldsymbol{\xi}, \boldsymbol{p}) = (\boldsymbol{g}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}), \boldsymbol{h}^{\mathrm{cv}}(\boldsymbol{\xi}, \boldsymbol{p}), -\boldsymbol{h}^{\mathrm{cc}}(\boldsymbol{\xi}, \boldsymbol{p}))$. Then, (9) becomes

$$\xi_i^{\mathrm{cv}}(\boldsymbol{p}) = \min_{\boldsymbol{\xi} \in X} \ \xi_i \quad \text{subject to} \quad \boldsymbol{\phi}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \mathbf{0}, \tag{11a}$$

$$\xi_i^{\mathrm{cc}}(\boldsymbol{p}) = \max_{\boldsymbol{\xi} \in X} \ \xi_i \quad \text{subject to} \quad \boldsymbol{\phi}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \mathbf{0}. \tag{11b}$$

At this point, [36, Theorem 2] describes directional derivatives of $\boldsymbol{\xi}^{\mathrm{cv}}, \boldsymbol{\xi}^{\mathrm{cc}}$ as the solutions of convex quadratic programs, provided that the assumptions of this theorem are satisfied. Crucially, these assumptions do not include convexity, and so we have not exploited the convexity of the relaxed CSP here at all. This supports our subgradient conjecture above; we expect that making full use of the relaxed CSP's convexity would provide useful subgradient information, but this would be a nontrivial theoretical development.

Directional derivatives themselves are nevertheless useful in a lower-bounding setting, though not as useful as subgradients. If n_p is 1 or 2, then convex analysis theory [37] shows that valid subgradients may be constructed from 1 or 4 directional derivative evaluations, respectively. If $n_p \geq 3$, then directional derivatives and subgradients are related somewhat more tenuously through standard results [38].

5 Tightening Interval Bounds

In the previous sections, convex and concave relaxations of implicit functions and CSPs were constructed within known interval bounds X. In this section, we adapt the formulation (9) in the setting of optimization-based bounds tightening (OBBT), to generate new interval bounds for implicit functions and CSPs that are at least as tight as the original bounds. These tighter intervals can in turn be used to construct relaxations of implicit functions and CSPs that are tighter than those constructed with the original intervals, based on the fact that αBB or McCormick relaxations of the original residual function will converge quickly to the residual function as its interval subdomain shrinks [32].

For context, given a convex relaxation of the feasible region of an NLP, classical OBBT methods generate tighter bounds for each variable by minimizing and maximizing each variable [39]. OBBT is commonly employed in various global optimization algorithms to tighten bounds in the nodes of a spatial branch-and-bound tree [2, 40, 41]. Examples 1 and 4 in Section 6 below will illustrate the approach of this section for tightening the interval bounds of implicit functions.

Throughout this section, we adopt the setup of (7) and (8), except we now allow the domain interval X to be varied. Thus, we denote dependence on X with a superscript where appropriate.

Define $\Xi^{B,X} \equiv [\boldsymbol{\xi}^{L,X}, \boldsymbol{\xi}^{U,X}] \in \mathbb{IR}^{n_x}$ such that for each $i \in \{1, \ldots, n_x\}$,

$$\begin{aligned} \xi_i^{L,X} = \quad & \min_{\boldsymbol{\xi}\in X,\, \boldsymbol{p}\in P} \quad \xi_i \\ & \text{subject to} \quad \boldsymbol{g}^{\mathrm{cv},X}(\boldsymbol{\xi}, \boldsymbol{p}) \le \mathbf{0}, \\ & \qquad\qquad\quad \boldsymbol{h}^{\mathrm{cv},X}(\boldsymbol{\xi}, \boldsymbol{p}) \le \mathbf{0} \le \boldsymbol{h}^{\mathrm{cc},X}(\boldsymbol{\xi}, \boldsymbol{p}), \end{aligned} \tag{12a}$$

$$\begin{aligned} \xi_i^{U,X} = \quad & \max_{\boldsymbol{\xi}\in X,\, \boldsymbol{p}\in P} \quad \xi_i \\ & \text{subject to} \quad \boldsymbol{g}^{\mathrm{cv},X}(\boldsymbol{\xi}, \boldsymbol{p}) \le \mathbf{0}, \\ & \qquad\qquad\quad \boldsymbol{h}^{\mathrm{cv},X}(\boldsymbol{\xi}, \boldsymbol{p}) \le \mathbf{0} \le \boldsymbol{h}^{\mathrm{cc},X}(\boldsymbol{\xi}, \boldsymbol{p}). \end{aligned} \tag{12b}$$

We will show that, given an initial interval X that contains $\Xi(\boldsymbol{p})$ for all $\boldsymbol{p} \in P$, (12) describes refined interval bounds that are as least as tight as X.

Theorem 5.1. Let $\Xi^{R,X}(\boldsymbol{p}) \equiv [\boldsymbol{\xi}^{\mathrm{cv},X}, \boldsymbol{\xi}^{\mathrm{cc},X}]$ be a solution of (9). Then, $\Xi^{B,X} \equiv [\boldsymbol{\xi}^{L,X}, \boldsymbol{\xi}^{U,X}]$ in (12) satisfies the following inclusions. For all $\boldsymbol{p} \in P$,

$$\Xi(\boldsymbol{p}) \subseteq \Xi^{R,X}(\boldsymbol{p}) \subseteq \Xi^{B,X} \subseteq X.$$

Proof Theorem 4.1 yields the first inclusion. Next, from (9) and (12), observe that, for any $i \in \{1, \ldots, n_x\}$,

$$\xi_i^{L,X} = \min_{\boldsymbol{p}\in P}\ \xi_i^{\mathrm{cv},X}(\boldsymbol{p}), \quad \text{and} \quad \xi_i^{U,X} = \max_{\boldsymbol{p}\in P}\ \xi_i^{\mathrm{cc},X}(\boldsymbol{p}).$$

Hence, $\Xi^{R,X}(\boldsymbol{p}) \subseteq \Xi^{B,X}$ for all $\boldsymbol{p} \in P$. Lastly, since (12) guarantees that $\boldsymbol{\xi}^{L,X}, \boldsymbol{\xi}^{U,X} \in X$, it follows that $\Xi^{B,R} \subseteq X$. □

Theorem 5.1 may be invoked repeatedly to iteratively tighten intervals that enclose the ranges of implicit functions and the point-to-set mappings in CSPs. Let an interval $\Xi^{B,0}$ be an initial interval bound on Ξ from (8), in place of X. Next, for each $k \in \{1, 2, \ldots\}$, define $\Xi^{B,k} \equiv [\boldsymbol{\xi}^{L,k}, \boldsymbol{\xi}^{U,k}]$ inductively in terms of $\Xi^{B,k-1}$ as follows. For each $i \in \{1, \ldots, n_x\}$, let

$$\begin{aligned}
\xi_i^{L,k} = \quad & \min_{\boldsymbol{\xi} \in \Xi^{B,k-1}, \boldsymbol{p} \in P} \quad \xi_i \\
& \text{subject to} \quad \boldsymbol{g}^{\mathrm{cv},\Xi^{B,k-1}}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \boldsymbol{0}, \qquad (13a) \\
& \qquad \boldsymbol{h}^{\mathrm{cv},\Xi^{B,k-1}}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \boldsymbol{0} \leq \boldsymbol{h}^{\mathrm{cc},\Xi^{B,k-1}}(\boldsymbol{\xi}, \boldsymbol{p}), \\
\xi_i^{U,k} = \quad & \max_{\boldsymbol{\xi} \in \Xi^{B,k-1}, \boldsymbol{p} \in P} \quad \xi_i \\
& \text{subject to} \quad \boldsymbol{g}^{\mathrm{cv},\Xi^{B,k-1}}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \boldsymbol{0}, \qquad (13b) \\
& \qquad \boldsymbol{h}^{\mathrm{cv},\Xi^{B,k-1}}(\boldsymbol{\xi}, \boldsymbol{p}) \leq \boldsymbol{0} \leq \boldsymbol{h}^{\mathrm{cc},\Xi^{B,k-1}}(\boldsymbol{\xi}, \boldsymbol{p}).
\end{aligned}$$

Theorem 5.1 illustrates that $\Xi^{B,k} \subseteq \Xi^{B,k-1} \subseteq \cdots \subseteq \Xi^{B,0}$. Thus, (13) represents a method to iteratively compute interval bounds on the feasible-set mappings of CSPs that are at least as tight as an initial bound. Since implicit functions may be represented as CSPs with equality constraints, this approach may also be used to tighten known interval bounds on the range of the implicit functions considered in Section 3.

6 Numerical Examples

In this section, we illustrate the new results of the previous sections by constructing convex and concave relaxations, as well as improved interval bounds, for various implicit functions and parametric ODEs. These approaches were implemented in the programming language Julia [42]. The McCormick.jl package [6] was used to construct convex relaxations of nonconvex factorable functions following either the standard McCormick relaxations [1, 9] or the differentiable McCormick relaxations [4, 12], and was also used to construct the established implicit function relaxations of [10] for comparison. All convex nonlinear programs were solved with IPOPT v3.13.2 [26] via JuMP v0.21.4 [43]. Nonlinear equations were solved with the NLsolve.jl package. The CPU times reported in this section were recorded using the BenchmarkTools.jl package. The numerical results reported below were obtained by running this implementation on a Windows 10 machine with a 3.6 GHz AMD Ryzen 5 2600X CPU and 8 GB memory.

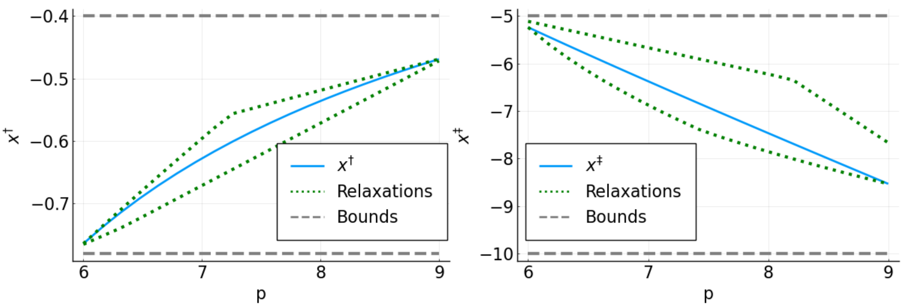
6.1 Relaxing Implicit Functions

The following example is adapted from [10, Example 3.26] and [8, Example 1].

Example 1. Let $P := [6, 9]$, and consider a function $f(z, p) = z^2 + pz + 4$ where the parameter p is an element of P. According to the quadratic formula, for each $p \in P$, there are two real roots z^* of the equation $f(z, p) = 0$. It was reported in [10] that $X^{\dagger,0} = [-0.78, -0.4]$ and $X^{\ddagger,0} = [-10.0, -5.0]$ are respective interval bounds of these two real roots. In both $X^{\dagger,0}$ or $X^{\ddagger,0}$, there is a single real root z^* of $f(z, p) = 0$ for each $p \in P$, so we have two injective implicit functions $x^\dagger : P \to X^{\dagger,0}$ and $x^\ddagger : P \to X^{\ddagger,0}$ such that $f(x^\dagger(p), p) = 0$ and $f(x^\ddagger(p), p) = 0$.

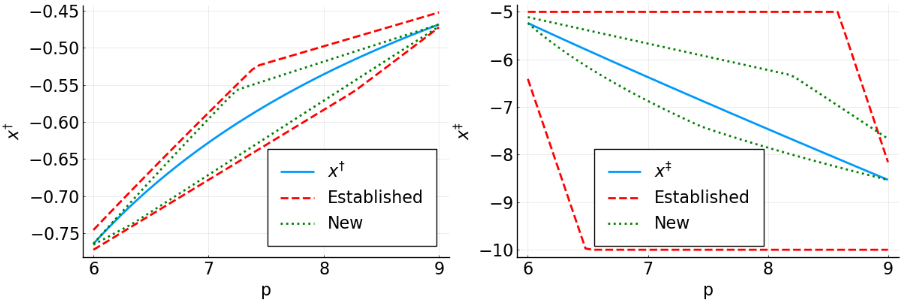
We generated convex and concave relaxations of $x^\dagger$ and $x^\ddagger$ on P using Theorem 3.1, and compared them with relaxations constructed using the method established in [10]. The convex and concave relaxations of f were constructed with standard McCormick relaxations [1, 9]. The minimization and maximization problems in (3) were solved at each $p \in P$ by two approaches: applying IPOPT to the NLP formulations (3), and alternatively applying the equation-solver NLsolve.jl according to the discussion below Corollary 3.2, after confirming the strict monotonicity of f^{cv} and f^{cc}. The resulting relaxations are depicted in Figure 1, and are evidently valid relaxations of the implicit functions $x^\dagger$ and $x^\ddagger$ on P. Figure 2 depicts our relaxations together with corresponding relaxations proposed by Stuber et al. [10] and implemented in the McCormick.jl package [6]. As shown in this figure, our relaxations appear to be significantly tighter in this case. Our average CPU time for each evaluation of $x^{\mathrm{cv}}(p)$ or $x^{\mathrm{cc}}(p)$ was 19.01 μs using NLsolve.jl, and 0.015 s using IPOPT; it seems that IPOPT has some overhead when applied to small problems. The average CPU time to evaluate Stuber et al's relaxations was 3.87 μs.

This example was also considered by Wechsung et al. [8]; comparing our Figure 1(a) with [8, Figure 3(b)], we conclude that our new relaxations are tighter in this case than relaxations constructed by one iteration of RM propagation.



(a) (b)

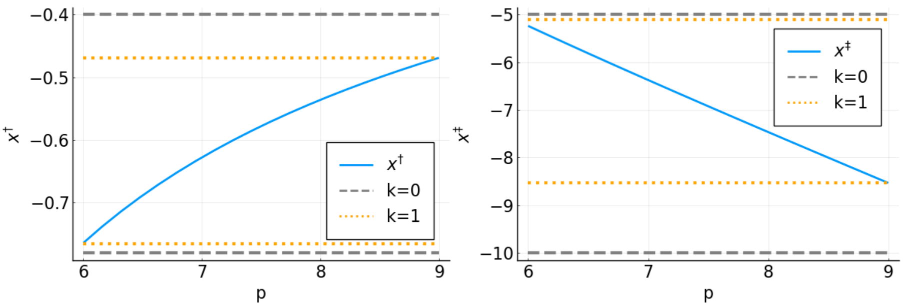
Fig. 1: The implicit functions $x^\dagger$ and $x^\ddagger$ in Example 1 (solid), along with their interval bounds (dashed) reported in [10] and new convex and concave relaxations (dotted) described by Theorem 3.1.



(a) (b)

Fig. 2: The implicit functions $x^\dagger$ and $x^\ddagger$ in Example 1 (solid), our new implicit function relaxations from Figure 1 (dotted), and analogous relaxations by Stuber et al. [10] (dashed).

Next, we constructed improved interval bounds of $x^\dagger$ and $x^\ddagger$ on P separately. Since an implicit function may be considered as a CSP with equality constraints only, we applied the formulation in (13) with $k = 1$ to generate interval bounds that are tighter than the original interval bounds $X^{\dagger,0}$ and $X^{\ddagger,0}$. As shown in Figure 3, these improved interval bounds are significantly tighter than the original bounds.

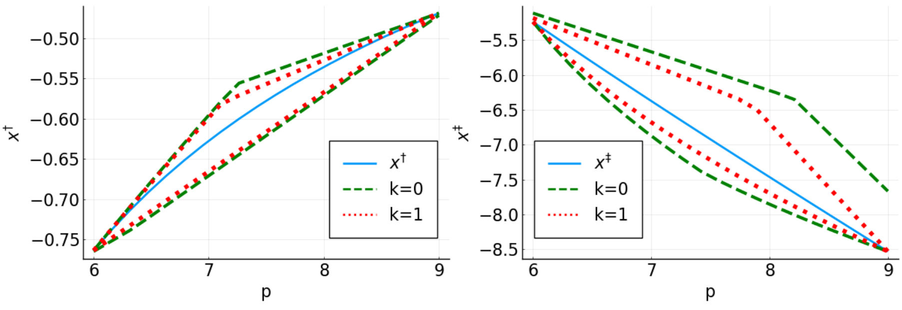


(a) (b)

Fig. 3: The implicit functions $x^\dagger$ and $x^\ddagger$ in Example 1 (solid), along with their original interval bounds $X^{\dagger,0}$ and $X^{\ddagger,0}$ (dashed) and improved interval bounds $X^{\dagger,1}$ and $X^{\ddagger,1}$ (dotted) on P, plotted as functions of p.

Furthermore, we used the improved interval bounds $X^{\dagger,1}$ and $X^{\ddagger,1}$ to generate improved relaxations for $x^\dagger$ and $x^\ddagger$, respectively, on P. These relaxations are plotted in Figure 4, along with the original relaxations constructed with $X^{\dagger,0}$ and $X^{\ddagger,0}$. This illustrates that tighter interval bounds do translate tighter convex and concave relaxations. We note that RM propagation may also employ an iterative approach to generate tighter relaxations. As shown in [8,

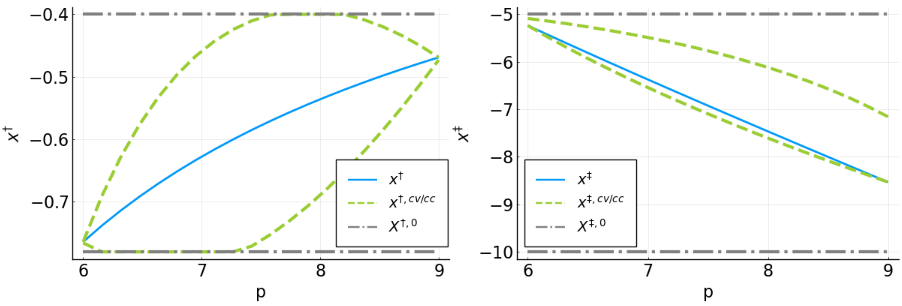
Figure 3(b)], ten iterations of RM propagation can generate tighter relaxations than our relaxations depicted in Figure 4(a).



(a) (b)

Fig. 4: The implicit functions $x^{\dagger}$ and $x^{\ddagger}$ in Example 1 (solid), along with their relaxations constructed on $X^{\dagger,0}$ and $X^{\ddagger,0}$ (dashed) and improved relaxations constructed on $X^{\dagger,1}$ and $X^{\ddagger,1}$ (dotted) on P, plotted as functions of p.

In addition to McCormick relaxations, we also used αBB relaxations [2] to construct convex and concave relaxations of f. The resulting convex and concave relaxations of x on $X^{\dagger,0}$ and $X^{\ddagger,0}$ are illustrated in Figure 5. This illustrates the versatility of our relaxation approach; any valid convex and concave relaxations of f can be used in (3), while the established method in [10] is limited to GM relaxations. For these αBB-based implicit function relaxations, the average CPU time for each evaluation of $x^{\text{cv}}(p)$ or $x^{\text{cc}}(p)$ was 0.018 s.



(a) (b)

Fig. 5: The implicit functions $x^{\dagger}$ and $x^{\ddagger}$ in Example 1 (solid), along with their interval bounds (dot-dashed), and new convex and concave relaxations (dashed) based on αBB relaxations of the residual function f.

The following example considers a thermodynamic equation of state that may exhibit multiple roots.

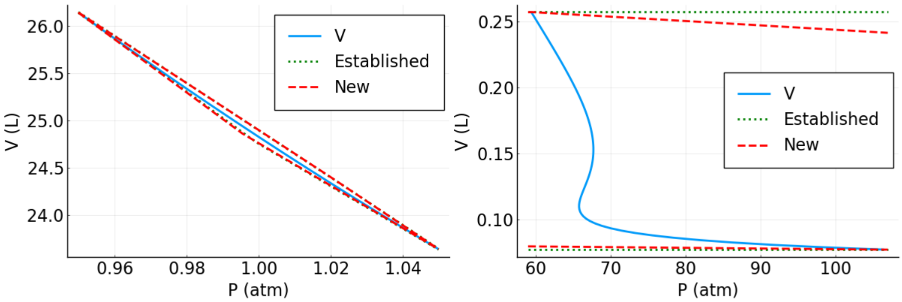
Example 2. The van der Waals equation of state is a physical property model for describing the behavior of non-ideal gases in chemical engineering. This equation suggests the following relationship between pressure P (atm), volume V (L), temperature T (K), and amount of gas n (mol):

$$f(P,V) := \left(P + a\frac{n^2}{V^2}\right)(V - nb) - nRT = 0, \tag{14}$$

where $R = 0.082\,057\,4\,\frac{\mathrm{L\,atm}}{\mathrm{K\,mol}}$ is the gas constant, and a, b are van der Waals constants. We study the behavior of 1 mole of carbon dioxide gas (with van der Waals constants: $a = 3.610\,\frac{\mathrm{L^2\,atm}}{\mathrm{mol^2}}$, and $b = 0.0429\,\frac{\mathrm{L}}{\mathrm{mol}}$ [44]), undergoing reversible isothermal compression at $T = 297.77$ K. Suppose that we would like to compute guaranteed bounds on the volumes obtained during this conversion, which may be used to verify that a process operates safely. Since (14) defines P as a cubic function of V, it is not practical to obtain a closed-form expression for the implicit function defining V in terms of P.

Thus, we use Theorem 3.1 to construct convex and concave relaxations of V in terms of P, with P (measured in atm) chosen from the domains $[0.95, 1.05]$ and $[59, 107]$. In the first of these pressure regimes, the van der Waals equation defines V uniquely in terms of P. In the second of these regimes, for certain values of P, the van der Waals equation suggests three different choices of V instead. (The well-known Maxwell construction can resolve this nonuniqueness, but we ignore this construction in our development here.) The interval bounds X that enclose V on $[0.95, 1.05]$ and $[59, 107]$ are set to $[23.5, 26.5]$ and $[0.07722, 0.2574]$ (measured in L), respectively. Convex and concave relaxations of f were constructed as McCormick relaxations. Our resulting convex and concave relaxations of volume V are illustrated in Figures 6a and 6b, compared against the prior relaxations of Stuber et al. [10]. In the pressure regime $[0.95, 1.05]$, both methods produce similar relaxations. In the pressure regime $[59, 107]$, our approach produces tighter relaxations that are still somewhat far from the actual van der Waals volumes. We suppose this slackness results from weakeness of the McCormick relaxations of f in this regime, due to extreme slope changes in the graphs of the various individual terms in (14). In this case, the average CPU time for each evaluation of our implicit function relaxations was 0.018 s using IPOPT (and the formulation in Theorem 3.1) and 19.5 μs using NLsolve.jl (and the monotonicity-based formulation after Corollary 3.2); evaluating Stuber et al.'s relaxations took 12.17 μs on average, and solving the original van der Waals equation to compute V took 9.15 μs using NLsolve.jl.

Furthermore, affine relaxations of f were also used to construct f^{cv} and f^{cc}, using the subgradients of standard McCormick relaxations of f at the midpoint of $[23.5, 26.5] \times [0.95, 1.05]$. In this case, x^{cv} and x^{cc} described in Theorem 3.1 can be evaluated using Corollary 3.3 easily. Neither NLP solvers



(a) (b)

Fig. 6: Volume defined as an implicit function of pressure for a van der Waals gas in Example 2 (solid), along with new convex and concave relaxations (dashed) and relaxations by the prior method of Stuber et al. [10] (dotted), constructed on (a) $P := [0.95, 1.05]$ and (b) $P := [59, 107]$.

nor LP solvers are required here. The constructed relaxations of P are illustrated in Figure 7, along with Stuber et al.'s corresponding relaxations for comparison. The average CPU time of evaluating either our convex relaxation or the concave relaxation at each pressure value was 5.99 μs; these affine-based relaxations are thus faster to evaluate than Stuber et al.'s approach, yet weaker due to outer approximation.

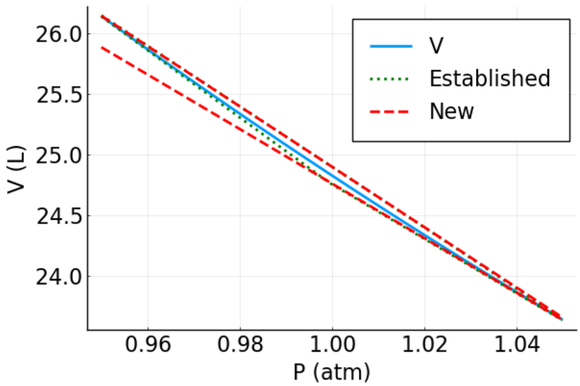


Fig. 7: The implicit function V from Example 2 (solid), along with new convex and concave relaxations (dashed) based on affine relaxations f^{cv} and f^{cc}, along with Stuber et al.'s relaxations [10] for comparison (dotted).

The following example is adapted from [8, Example 5]. This example illustrates that when an implicit function does not exist everywhere on the intended domain, our new relaxation approach still constructs valid convex relaxations.

Example 3. Let $P := [-3, 3]$ and $X := [-10, 10]$, and consider a function $f(z, p) = z^2 - (\sqrt{p^2 - p} - 2)^4$ with $(z, p) \in X \times P$. Let $Q := [-3, 0] \cap [1, 3]$, which is a subset of P. For each $p \in Q$, there are two real roots z^* of the equation $f(z, p) = 0$, and so the equation $f(x(p), p) \equiv 0$ defines a nonunique implicit function $x : Q \to X$. For any $p \in P \backslash Q = (0, 1)$, there is no value z that satisfies $f(z, p) = 0$, and so no implicit function can exist here. Figure 8 illustrates the corresponding convex and concave relaxations of x described by Theorem 3.1. To construct these, the required relaxations of f were obtained using standard McCormick relaxations, and each evaluation of $x^{\mathrm{cv}}(p)$ and x^{cc} took 0.022 s of CPU time on average. Comparing Figure 8 with [8, Figure 7], it seems that our new approach produces tighter relaxations than [8]; we were unable to test this directly since the method of [8] is nontrivial to implement. In particular, our relaxations in Figure 8 coincide with an implicit function at $p = -3$, while this is not the case in [8].

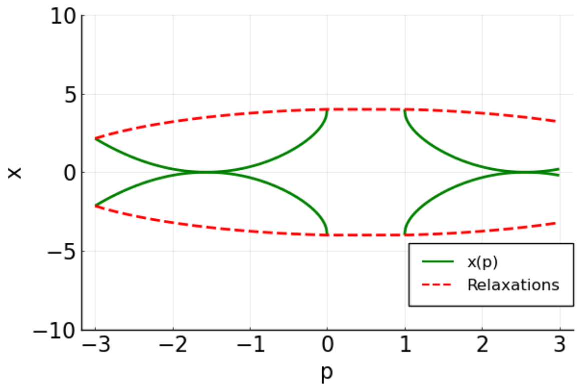


Fig. 8: The nonunique implicit functions x from Example 3 (solid), along with their convex and concave relaxations (dashed) constructed according to Theorem 3.1.

6.2 Relaxing Numerical ODE Solutions

In this section, we construct convex and concave relaxations for implicit functions that are numerical solutions of parametric ordinary differential equations (ODEs), computed using implicit integration methods. Compared with explicit integration methods, implicit integration methods are typically more stable when dealing with stiff ODEs [45]. While methods have been established in [10, 13] to construct convex relaxations for implicit numerical solutions of ODEs, this section introduces an alternative approach that may yield tighter relaxations, and provides a situation in which we must relax multiple related implicit functions in succession. Having tighter convex relaxations would aid deterministic methods for dynamic global optimization. Like the approaches presented in [10, 13], the approach presented this section only relaxes approximate numerical solutions of ODEs; other established methods [17, 46] instead provide relaxations that are guaranteed to enclose the true ODE solution.

For the remainder of this section, define $t_0, t_f \in \mathbb{R}$ such that $t_0 < t_f$, and let $I = (t_0, t_f]$. Given $\boldsymbol{z}^0 \in \mathbb{R}^{n_z}$ and a continuous function $\boldsymbol{u} : I \times P \times \mathbb{R}^{n_z} \to \mathbb{R}^{n_z}$, consider an ODE system:

$$\begin{aligned} \frac{\mathrm{d}\boldsymbol{z}}{\mathrm{d}t}(t,\boldsymbol{p}) &= \boldsymbol{u}(t,\boldsymbol{p},\boldsymbol{z}(t,\boldsymbol{p})), \quad t \in I, \\ \boldsymbol{z}(t_0,\boldsymbol{p}) &= \boldsymbol{z}^0. \end{aligned} \tag{15}$$

According to Peano's Theorem summarized in [47, Theorem 2.1, Chapter II], the ODE (15) has at least one solution. We will use the implicit Euler method to obtain a numerical solution for (15) and generate its convex relaxations using the approach of Section 3. An analogous approach can be applied to other implicit integration methods, such as the Adams–Moulton method and the BDF method. To solve (15) with the implicit Euler method at an arbitrary $\boldsymbol{p} \in P$, we first discretize I into n evenly spaced intervals with length $\Delta t := (t_f - t_0)/n$. For each $m \in \{0, \ldots, n\}$, denote the numerical ODE solution value at the mesh point $t_m := (t_0 + m(\Delta t))$ as $\boldsymbol{z}^m$. Then, (15) can be approximated by the following nonlinear equations for all $m \in \{1, \ldots, n\}$ and $\boldsymbol{p} \in P$:

$$\boldsymbol{z}^m(\boldsymbol{p}) - \boldsymbol{z}^{m-1}(\boldsymbol{p}) - \Delta t\, \boldsymbol{u}(t_m, \boldsymbol{p}, \boldsymbol{z}^m(\boldsymbol{p})) = \boldsymbol{0}. \tag{16}$$

where $\boldsymbol{z}^0(\boldsymbol{p}) = \boldsymbol{z}^0$ is the known initial condition. Observe that (16) defines an implicit function:

$$\boldsymbol{x}(\boldsymbol{p}) \equiv \begin{bmatrix} \boldsymbol{z}^1(\boldsymbol{p}) \\ \vdots \\ \boldsymbol{z}^n(\boldsymbol{p}) \end{bmatrix}$$

in the form of (2) if we define:

$$\boldsymbol{f}((\boldsymbol{\zeta}^1, \ldots, \boldsymbol{\zeta}^n), \boldsymbol{p}) \equiv \begin{bmatrix} \boldsymbol{\zeta}^1 - \boldsymbol{z}^0 - \Delta t\, \boldsymbol{u}(t_1, \boldsymbol{p}, \boldsymbol{\zeta}^1) \\ \boldsymbol{\zeta}^2 - \boldsymbol{\zeta}^1 - \Delta t\, \boldsymbol{u}(t_2, \boldsymbol{p}, \boldsymbol{\zeta}^2) \\ \vdots \\ \boldsymbol{\zeta}^n - \boldsymbol{\zeta}^{n-1} - \Delta t\, \boldsymbol{u}(t_n, \boldsymbol{p}, \boldsymbol{\zeta}^n) \end{bmatrix} \tag{17}$$

Thus, we can use Theorem 3.1 to construct convex and concave relaxations for $\boldsymbol{z}^n$ on P, with $\boldsymbol{z}^n(\boldsymbol{p})$ denoting the ODE solver's attempt to evaluate the true ODE solution $\boldsymbol{z}(t_f, \boldsymbol{p})$.

Let $Z \equiv [\boldsymbol{z}^L, \boldsymbol{z}^U] \in \mathbb{R}^{n_z}$ be a known interval bound for which $\boldsymbol{z}(t,\boldsymbol{p}) \in Z$ for all $(t,\boldsymbol{p}) \in I \times P$. Define $Z^{m,0} \equiv [\boldsymbol{z}^{m,0,L}, \boldsymbol{z}^{m,0,U}] \subseteq \mathbb{IR}^{n_z}$ to be *a priori* known interval bounds of $\boldsymbol{z}^m$ for each $m \in \{1,\ldots,n\}$, where the index m represents the mesh point and "0" represents that this is an *a priori* bound (similar to the notation in Section 5). Since a conservative interval bound Z is known, we may set $Z^{m,0} := Z$ for each $m \in \{1,\ldots,n\}$, and it follows that $\boldsymbol{z}^m(\boldsymbol{p}) \in Z^{m,0}$ for each $m \in \{1,\ldots,n\}$ and $\boldsymbol{p} \in P$. Then, convex and concave relaxations of the terminal numerical ODE solution $\boldsymbol{z}^n$ on P can be computed using Theorem 3.1 as follows. Let $\boldsymbol{f}^{\mathrm{cv},Z^{m,0}}$ and $\boldsymbol{f}^{\mathrm{cc},Z^{m,0}} : \mathbb{R}^{n_z \times n + n_p} \to \mathbb{R}^{n_z \times n}$ be convex and concave relaxations of $\boldsymbol{f}$ in (17), respectively, constructed on the domain $Z^{m,0} \times P$. Then, for each $j \in \{1,\ldots,n_z\}$, Theorem 3.1 yields the following relaxations of $\boldsymbol{z}^n$ on P:

$$
\begin{aligned}
z_j^{n,\mathrm{cv}}(\boldsymbol{p}) = \min_{\substack{\boldsymbol{\zeta}^m \in Z^{m,0},\\ \forall m \in \{1,\ldots,n\}}} \quad & \zeta_j^n, \\
\text{subject to} \quad & f^{\mathrm{cv},Z^{m,0}}_{i+n_z(m-1)}((\boldsymbol{\zeta}^1,\ldots,\boldsymbol{\zeta}^n),\boldsymbol{p}) \\
& \quad \le 0 \le f^{\mathrm{cc},Z^{m,0}}_{i+n_z(m-1)}((\boldsymbol{\zeta}^1,\ldots,\boldsymbol{\zeta}^n),\boldsymbol{p}), \\
& \qquad \forall i \in \{1,\ldots,n_z\}, \quad \forall m \in \{1,\ldots,n\}, \\
z_j^{n,\mathrm{cc}}(\boldsymbol{p}) = \max_{\substack{\boldsymbol{\zeta}^m \in Z^{m,0},\\ \forall m \in \{1,\ldots,n\}}} \quad & \zeta_j^n, \\
\text{subject to} \quad & f^{\mathrm{cv},Z^{m,0}}_{i+n_z(m-1)}((\boldsymbol{\zeta}^1,\ldots,\boldsymbol{\zeta}^n),\boldsymbol{p}) \\
& \quad \le 0 \le f^{\mathrm{cc},Z^{m,0}}_{i+n_z(m-1)}((\boldsymbol{\zeta}^1,\ldots,\boldsymbol{\zeta}^n),\boldsymbol{p}), \\
& \qquad \forall i \in \{1,\ldots,n_z\}, \quad \forall m \in \{1,\ldots,n\}.
\end{aligned}
\tag{18}
$$

Furthermore, we may use the formulation in (13) to construct improved interval bounds $Z^{m,1} \equiv [\boldsymbol{z}^{m,L,1}, \boldsymbol{z}^{m,U,1}]$ of $\boldsymbol{z}^m$ for each $m \in \{1,\ldots,n\}$, where m denotes the mesh point index and 1 denotes one iteration of refinement. As discussed in Section 5, these improved intervals are guaranteed to be at least as tight as the original interval $Z^{m,0}$. In this case, we can use these tighter intervals to generate tighter relaxations for the numerical solutions of ODEs by replacing $Z^{m,0}$ in (18) with $Z^{m,1}$.

This approach yields the following. For each $m \in \{1,\dots,n\}$, consider $Z^{m,1} \equiv [\boldsymbol{z}^{m,L,1}, \boldsymbol{z}^{m,U,1}] \in \mathbb{R}^{n_z}$ such that, for each $j \in \{1,\dots,n_z\}$,

$$\begin{aligned}
z_j^{m,L,1} = &\min_{\substack{\boldsymbol{p}\in P, \boldsymbol{\zeta}^\kappa \in Z^{\kappa,0},\\ \forall \kappa\in\{1,\dots,n\}}} \quad \zeta_j^m, \\
&\text{subject to} \quad f^{\text{cv},Z^{\kappa,0}}_{i+n_z(\kappa-1)}((\boldsymbol{\zeta}^1,\dots,\boldsymbol{\zeta}^n),\boldsymbol{p}) \\
&\qquad\qquad \le 0 \le f^{\text{cc},Z^{\kappa,0}}_{i+n_z(\kappa-1)}((\boldsymbol{\zeta}^1,\dots,\boldsymbol{\zeta}^n),\boldsymbol{p}), \\
&\qquad\qquad\quad \forall i \in \{1,\dots,n_z\}, \quad \forall \kappa \in \{1,\dots,n\}, \\
z_j^{m,U,1} = &\max_{\substack{\boldsymbol{p}\in P, \boldsymbol{\zeta}^\kappa \in Z^{\kappa,0},\\ \forall \kappa\in\{1,\dots,n\}}} \quad \zeta_j^m, \\
&\text{subject to} \quad f^{\text{cv},Z^{\kappa,0}}_{i+n_z(\kappa-1)}((\boldsymbol{\zeta}^1,\dots,\boldsymbol{\zeta}^n),\boldsymbol{p}) \\
&\qquad\qquad \le 0 \le f^{\text{cc},Z^{\kappa,0}}_{i+n_z(\kappa-1)}((\boldsymbol{\zeta}^1,\dots,\boldsymbol{\zeta}^n),\boldsymbol{p}), \\
&\qquad\qquad\quad \forall i \in \{1,\dots,n_z\}, \quad \forall \kappa \in \{1,\dots,n\}.
\end{aligned} \tag{19}$$

Then, Theorem 5.1 implies that $\boldsymbol{z}^m(\boldsymbol{p}) \in Z^{m,1} \subseteq Z^{m,0}$ for each $m \in \{1,\dots,n\}$ and $\boldsymbol{p} \in P$. We now illustrate this approach in a numerical example.

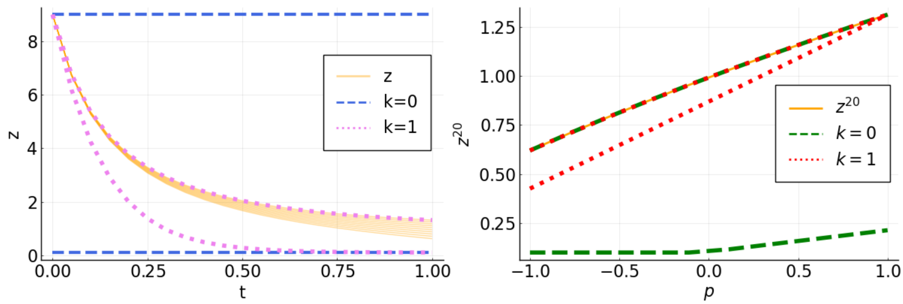
Example 4. Consider the following parametric ODE:

$$\begin{aligned}
\frac{\mathrm{d}z}{\mathrm{d}t}(t,p) &= -z^2 + p, \quad t \in (0,1], \\
z(0,p) &= 9,
\end{aligned} \tag{20}$$

where $p \in P := [-1,1]$.

This system was previously studied in [48, Section 4.1] and [13, Example 1]. Convex and concave relaxations of numerical solutions of this ODE system were generated according to our new approach in Section 6.2 as follows. We first discretize the integration duration $[0,1]$ into 20 intervals, so that $n = 20$ and $\Delta t = (t_f - t_0)/n = 0.05$. Using the implicit Euler method and the notation of Section 6.2, the ODE solution $z(\cdot,p)$ can be numerically approximated by mesh point values $z^1(p),\dots,z^{20}(p)$ for all $p \in P$. In particular, $z^{20}(p)$ is the numerical approximation of the ODE solution $z(t_f,p)$ at the terminal time for all $p \in P$. A known conservative interval bound for the ODE (20) is $Z = [0.1, 9]$ according to [13], so the interval bounds of z^m on P, $Z^{m,0}$, are set to Z for each $m \in \{1,\dots,20\}$. We generated convex and concave relaxations $z^{20,\text{cv},0}(p), z^{20,\text{cc},0}(p)$ on P using (18), where $\boldsymbol{f}^{\text{cv}}, \boldsymbol{f}^{\text{cc}}$ were constructed with GM relaxations. These relaxations are plotted in Figure 9b where $k = 0$, and appear to be valid convex and concave relaxations of $z^{20}(p)$ on P. The average CPU time of computing either $z^{20,\text{cv},0}(p)$ or $z^{20,\text{cc},0}(p)$ at each $p \in P$ is 0.0505 seconds, which is comparable with but slower than the relaxation evaluations reported in [48, Section 4.1].

Next, the formulation in (19) was employed to construct improved interval bounds $Z^{m,1} \equiv [\boldsymbol{z}^{m,L,1}, \boldsymbol{z}^{m,U,1}]$ of $\boldsymbol{z}^m$ for each $m \in \{1, \ldots, 20\}$, where the final superscript 1 denotes one iteration of refinement. The generated lower bounds $\boldsymbol{z}^{1,L,1}, \ldots, \boldsymbol{z}^{20,L,1}$ and upper bounds $\boldsymbol{z}^{1,U,1}, \ldots, \boldsymbol{z}^{20,U,1}$ are plotted as the lower-bounding and upper-bounding trajectories in Figure 9a. Furthermore, these tighter interval bounds were used to generate tighter convex and concave relaxations $z^{20,\mathrm{cv},1}(p), z^{20,\mathrm{cc},1}(p)$ by replacing $Z^{m,0}$ in (18) with $Z^{m,1}$ for each $m \in \{1, \ldots, 20\}$. The improved relaxations are illustrated in Figure 9(b).



(a) (b)

Fig. 9: (a) Interval bounds $Z^{m,0}$ (dashed) and tighter interval bounds $Z^{m,1}$ (dotted), $m \in \{1, \ldots, 20\}$, in Example 4. Solid lines are trajectories of $z(\cdot, p)$ in (20) with different p. (b) The parametric solution of (15) (solid), along with its convex and concave relaxations constructed on conservative interval bounds (dashed) and improved interval bounds (dotted), plotted as a function of p at $t = 1$

Lastly, we compare the convex and concave relaxations illustrated in Figure 9(b) with those constructed with established methods [13, 48]. When $k = 0$, we used very conservative interval bounds $Z^{1,0}, \ldots, Z^{20,0}$ that are much looser than the bounds used in [48]. In this case, the convex relaxation $z^{20,\mathrm{cv},0}$ in Figure 9(b) is looser than the convex relaxation in [48, Figure 5], but the concave relaxation $z^{20,\mathrm{cc},0}$ overlaps with the numerical solution z^{20}, and is significantly tighter than the concave relaxation in [48, Figure 5]. When $k = 1$, we used tighter interval bounds $Z^{1,1}, \ldots, Z^{20,1}$. In this case, the convex and concave relaxations, $z^{20,\mathrm{cv},1}$ and $z^{20,\mathrm{cc},1}$, are both significantly tighter than the relaxations in [48, Figure 5]. Compared with the lower and upper bounds shown in [13, Figure 4, lower left and lower right], the bounds in Figure 9(a) are looser. This is probably due to the difference in numerical integration methods. Instead of the naive implicit Euler method used in this work, more advanced Adams–Moulton (AM) and backward difference formula (BDF) methods were used in the implementation of [13]. Though we expect that the approach in Section 6.2 may be extended to the AM and BDF methods, we do not attempt that here for simplicity. Again, we note that this new approach and the approach from [13] construct relaxations for approximate numerical solutions of ODEs, while the relaxations from [48] are guaranteed relaxations of the true ODE solution.

7 Conclusion

This article has presented a novel approach for generating convex and concave relaxations of implicit functions. These relaxations are described by the convex parametric programs shown in Theorem 3.1, whose constraints are arbitrary convex and concave relaxations of the original residual function. These relaxations can be evaluated particularly efficiently when linearity or monotonicity of the supplied residual relaxations can be exploited. Using the Tsoukalas-Mitsos relaxations of compositions [15], our result was extended to generate relaxations for compositions of outer implicit functions with inner known functions. Our new approach was also extended to construct convex relaxations for inverse functions (Section 3.5) and feasible set mappings in CSPs (Section 4). Section 5 illustrated that tighter interval bounds of implicit functions and feasible regions in CSPs can be obtained by further optimizing their convex relaxations with respect to parameters, in an OBBT setting. These improved interval bounds can then be used to generate tighter relaxations.

Unlike some established methods that construct relaxations for implicit functions and CSPs, our new approach does not assume uniqueness of a solution and does not require the original residual function to be factorable. While the method in [10] requires GM relaxation and the method in [8] requires RM relaxation, our new approach admits any valid convex relaxations of the original residual function, including McCormick relaxations [1, 4, 9], αBB relaxations [2], convex envelopes, and the pointwise best among multiple relaxations. Furthermore, while the established method in [10] depends on one particular nonlinear equation solution approach, namely fixed-point iteration, our new approach may employ various methods to solve the embedded optimization problems, such as LP algorithms and NLP algorithms, or even may even solve these analytically. This optimization-based approach is straightforward to implement, and a proof-of-concept Julia implementation of this approach was developed. As illustrated by the numerical examples in Section 6, our new approach may construct tighter relaxations of implicit functions and parametric ODEs than established methods, thus aiding overarching methods for global optimization or reachability analysis.

Future work may include describing subgradients for the new convex relaxations of implicit functions, to help minimize these relaxations during global optimization, or to construct useful outer approximations. We conjecture that a general, useful subgradient result for our relaxations is possible, yet this development seems nontrivial based on current parametric optimization sensitivity theory. Another potential direction of future research is to consider relax solutions of parametric index-1 differential-algebraic equations, by somehow combining these implicit function relaxations with recent relaxations of parametric ODE solutions. Building a useful library of closed-form relaxations of common implicit functions may be a worthy goal, and is aided by our new results here.

Declarations

Funding

This work was supported by the McMaster Advanced Control Consortium (MACC), and by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN-2017-05944.

Competing interests

The authors have no relevant financial or non-financial interests to disclose.

Data availability

The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Code availability

Our Julia code for our numerical examples is available at https://github.com/kamilkhanlab/implicit-func-relaxations.

Author contributions

All authors contributed to the study conception and design, and to the formulation and proofs of mathematical results. Numerical experiments were performed by Huiyi Cao. All authors wrote, edited, read, and approved the final manuscript.

References

[1] Mitsos, A., Chachuat, B., Barton, P.I.: McCormick-based relaxations of algorithms. SIAM Journal on Optimization **20**(2), 573–601 (2009)

[2] Adjiman, C.S., Dallwig, S., Floudas, C.A., Neumaier, A.: A global optimization method, αBB, for general twice-differentiable constrained NLPs – I. Theoretical advances. Computers & Chemical Engineering **22**(9), 1137–1158 (1998)

[3] McCormick, G.P.: Computability of global solutions to factorable nonconvex programs: Part I–Convex underestimating problems. Mathematical Programming **10**(1), 147–175 (1976)

[4] Khan, K.A., Watson, H.A.J., Barton, P.I.: Differentiable McCormick relaxations. Journal of Global Optimization **67**(4), 687–729 (2017)

[5] Chachuat, B.: MC++: A Toolkit for Bounding Factorable Functions (2014)

[6] Wilhelm, M.E., Stuber, M.D.: EAGO.jl: Easy Advanced Global Optimization in Julia. Optimization Methods and Software, 1–26 (2020)

[7] Dontchev, A.L., Rockafellar, R.T.: Implicit Functions and Solution Mappings: A View from Variational Analysis, 2nd edn. Springer Series in Operations Research and Financial Engineering. Springer, New York (2014)

[8] 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)

[9] Scott, J.K., Stuber, M.D., Barton, P.I.: Generalized McCormick relaxations. Journal of Global Optimization **51**(4), 569–606 (2011)

[10] 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)

[11] Ortega, J.M., Rheinboldt, W.C.: Iterative Solution of Nonlinear Equations in Several Variables. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, New York (2000)

[12] 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)

[13] Wilhelm, M.E., Le, A.V., Stuber, M.D.: Global optimization of stiff dynamical systems. AIChE Journal **65**(12) (2019)

[14] Griewank, A., Walther, A.: Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation, 2nd edn. Society for Industrial and Applied Mathematics, Philadelphia, PA (2008)

[15] Tsoukalas, A., Mitsos, A.: Multivariate McCormick relaxations. Journal of Global Optimization **59**(2-3), 633–662 (2014)

[16] Smith, E.M.B., Pantelides, C.C.: Global optimisation of nonconvex MINLPs. Computers & Chemical Engineering **21**, 791–796 (1997)

[17] 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)

[18] Song, Y., Cao, H., Mehta, C., Khan, K.A.: Bounding convex relaxations of process models from below by tractable black-box sampling. Computers & Chemical Engineering **153**, 107413 (2021)

[19] Hiriart-Urruty, J.B., Lemaréchal, C.: Convex Analysis and Minimization Algorithms I: Fundamentals. Springer, Berlin (1993)

[20] Cao, H., Khan, K.A.: Enclosing reachable sets for nonlinear control systems using complementarity-based intervals. IFAC-PapersOnLine **54**(3), 590–595 (2021)

[21] Fiacco, A.V., Kyparisis, J.: Convexity and concavity properties of the optimal value function in parametric nonlinear programming. Journal of Optimization Theory and Applications **48**(1), 95–126 (1986)

[22] Scholtes, S.: Introduction to Piecewise Differentiable Equations. SpringerBriefs in Optimization. Springer, New York (2012)

[23] Hansen, E., Greenberg, R.: An interval Newton method. Applied Mathematics and Computation **12**(2-3) (1983)

[24] Neumaier, A.: Interval Methods for Systems of Equations. Encyclopedia of Mathematics and Its Applications. Cambridge University Press, Cambridge (1991)

[25] Clarke, F.: Optimization and Nonsmooth Analysis. Classics in Applied Mathematics. SIAM, Philadelphia (1990)

[26] Wächter, A., Biegler, L.T.: On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical Programming **106**(1), 25–57 (2006)

[27] Drud, A.S.: CONOPT - a large-scale GRG code. ORSA Journal on Computing **6**(2), 207–216 (1994)

[28] Nesterov, Y.: Nonsmooth convex optimization. In: Lectures on Convex Optimization. Springer Optimization and Its Applications, pp. 139–240. Springer, Cham (2018)

[29] Beckers, M., Mosenkis, V., Naumann, U.: Adjoint mode computation of subgradients for McCormick relaxations. In: Forth, S., Hovland, P., Phipps, E., Utke, J., Walther, A. (eds.) Recent Advances in Algorithmic Differentiation, pp. 103–113. Springer, Berlin (2012)

[30] Khan, K.A.: Branch-locking ad techniques for nonsmooth composite functions and nonsmooth implicit functions. Optimization Methods and Software **33**(4–6), 1127–1155 (2018)

[31] Hansen, E., Sengupta, S.: Bounding solutions of systems of equations using interval analysis. BIT **21**(2), 203–211 (1981)

[32] Bompadre, A., Mitsos, A.: Convergence rate of McCormick relaxations. Journal of Global Optimization **52**(1), 1–28 (2012)

[33] Nerantzis, D., Adjiman, C.S.: Tighter αBB relaxations through a refinement scheme for the scaled Gerschgorin theorem. Journal of Global Optimization **73**(3), 467–483 (2019)

[34] Cao, H., Khan, K.A.: Bounding nonconvex optimal control problems using pontryagin's minimum principle (Submitted) (2021)

[35] Song, Y., Khan, K.A.: Optimization-based convex relaxations for nonconvex parametric systems of ordinary differential equations. Mathematical Programming **in press** (2021)

[36] Ralph, D., Dempe, S.: Directional derivatives of the solution of a parametric nonlinear program. Mathematical Programming **70**(1-3), 159–172 (1995)

[37] Khan, K.A., Yuan, Y.: Constructing a subgradient from directional derivatives for functions of two variables. Journal of Nonsmooth Analysis and Optimization, 6061 (2020)

[38] Rockafellar, R.T.: Convex Analysis. Princeton University Press, Princeton, NJ (1996)

[39] Gleixner, A.M., Berthold, T., Müller, B., Weltge, S.: Three enhancements for optimization-based bound tightening. Journal of Global Optimization **67**(4), 731–757 (2017)

[40] Zamora, J.M., Grossmann, I.E.: A branch and contract algorithm for problems with concave univariate, bilinear and linear fractional terms. Journal of Global Optimization **14**(3), 217–249 (1999)

[41] 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)

[42] Bezanson, J., Edelman, A., Karpinski, S., Shah, V.: Julia: A fresh approach to numerical computing. SIAM Review **59**(1), 65–98 (2017)

[43] Dunning, I., Huchette, J., Lubin, M.: JuMP: A modeling language for

mathematical optimization. SIAM Review **59**(2), 295–320 (2017)

[44] Atkins, P., de Paula, J.: Physical Chemistry, 8th edn. W. H. Freeman, New York (2006)

[45] Ascher, U.M., Petzold, L.R.: Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations, 1st edn. Society for Industrial and Applied Mathematics, USA (1998)

[46] Sahlodin, A.M., Chachuat, B.: Convex/concave relaxations of parametric ODEs using Taylor models. Computers & Chemical Engineering **35**(5), 844–857 (2011)

[47] Hartman, P.: Ordinary Differential Equations. SIAM, Philadelphia, PA (2002)

[48] 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)