informs
https://pubsonline.informs.org/journal/ijoc

INFORMS JOURNAL ON COMPUTING
Articles in Advance, pp. 1–20
ISSN 1091-9856 (print), ISSN 1526-5528 (online)

Feasibility Verification and Upper Bound Computation in Global Minimization Using Approximate Active Index Sets

Christian Füllner,[a,*] Peter Kirst,[b] Hendrik Otto,[a] Steffen Rebennack[a]

[a] Institute for Operations Research, Stochastic Optimization, Karlsruhe Institute of Technology, 76199 Karlsruhe, Germany; [b] Operations Research and Logistics, Wageningen University & Research, 6708 PB Wageningen, Netherlands

*Corresponding author

Contact: christian.fuellner@kit.edu, https://orcid.org/0000-0003-2485-6199 (CF); peter.kirst@wur.nl, https://orcid.org/0000-0002-8472-3569 (PK); hendrik.otto94@gmail.com (HO); steffen.rebennack@kit.edu, https://orcid.org/0000-0002-8501-2785 (SR)

Received: May 22, 2023
Revised: December 13, 2023; March 1, 2024; March 4, 2024
Accepted: March 9, 2024
Published Online in Articles in Advance: May 3, 2024

https://doi.org/10.1287/ijoc.2023.0162

Copyright: © 2024 The Author(s)

Abstract. We propose a new upper bounding procedure for global minimization problems with continuous variables and possibly nonconvex inequality and equality constraints. Upper bounds are crucial for standard termination criteria of spatial branch-and-bound (SBB) algorithms to ensure that they can enclose globally minimal values sufficiently well. However, whereas for most lower bounding procedures from the literature, convergence on smaller boxes is established, this does not hold for several methods to compute upper bounds even though they often perform well in practice. In contrast, our emphasis is on the convergence. We present a new approach to verify the existence of feasible points on boxes, on which upper bounds can then be determined. To this end, we resort to existing convergent feasibility verification approaches for purely equality and box constrained problems. By considering carefully designed modifications of subproblems based on the approximation of active index sets, we enhance such methods to problems with additional inequality constraints. We prove that our new upper bounding procedure finds sufficiently good upper bounds so that termination of SBB algorithms is guaranteed after a finite number of iterations. Our theoretical findings are illustrated by computational results on a large number of standard test problems. These results show that compared with interval Newton methods from the literature, our proposed method is more successful in feasibility verification for both, a full SBB implementation (42 instead of 26 test problems) and exhaustive sequences of boxes around known feasible points (120 instead of 29 test problems).

History: Accepted by Antonio Frangioni, Area Editor for Design & Analysis of Algorithms–Continuous.

Open Access Statement: This work is licensed under a Creative Commons Attribution 4.0 International License. You are free to copy, distribute, transmit and adapt this work, but you must attribute this work as "*INFORMS Journal on Computing*. Copyright © 2024 The Author(s). https://doi.org/10.1287/ijoc.2023.0162, used under a Creative Commons Attribution License: https://creativecommons.org/licenses/by/4.0/."

Supplemental Material: The software that supports the findings of this study is available within the paper and its Supplemental Information (https://pubsonline.informs.org/doi/suppl/10.1287/ijoc.2023.0162) as well as from the IJOC GitHub software repository (https://github.com/INFORMSJoC/2023.0162). The complete IJOC Software and Data Repository is available at https://informsjoc.github.io/.

Keywords: branch-and-bound • deterministic upper bounds • feasibility verification • active index sets • global optimization • convergence

1. Introduction

In this article, we propose a new method to determine valid upper bounds for the globally optimal value v^* of nonconvex minimization problems:

$$
\begin{aligned}
P(B): \quad v^* := \min_{x \in \mathbb{R}^n} \quad & f(x) \\
\text{s.t.} \quad & g_i(x) \le 0, \quad \forall i \in I, \\
& h_j(x) = 0, \quad \forall j \in J, \\
& x \in B.
\end{aligned}
$$

We assume that all functions $f, g_i, i \in I,$ and $h_j, j \in J,$ are continuously differentiable on an open set containing the box B, but we do not require f and g_i to be convex or h_j to be linear. Therefore, the feasible set

$$M(B) := \{x \in B | g_i(x) \le 0, i \in I, h_j(x) = 0, j \in J\}$$

is possibly nonconvex. We assume $M(B) \neq \emptyset$. The box B is defined by $B = \{x \in \mathbb{R}^n | \underline{b} \le x \le \overline{b}\}$ with $\underline{b}, \overline{b} \in \mathbb{R}^n, \underline{b} < \overline{b}$ and with all inequalities being defined componentwise. We denote the cardinality of the constraint sets by $p := |I|$ and $q := |J|$.

Furthermore, we assume that the linear independence constraint qualification (LICQ) is satisfied at least in all globally minimal points x^* of $P(B)$. LICQ is considered to be a mild assumption (Jongen et al. 1986), even if it may not be satisfied for some degenerate problems of practical interest. Moreover, in nonlinear optimization, the assumption of LICQ even in all locally minimal points is standard for convergence proofs.

Determining upper bounds to v^* is an essential component of many algorithms in global nonlinear optimization, especially in spatial branch-and-bound (SBB) algorithms, which iteratively branch the original box B into smaller subboxes X, on which then subproblems $P(X)$ are solved and bounds are computed. SBB algorithms are the state-of-the-art approach to solve problems of the form $P(B)$ to global optimality. Their application goes back to Falk and Soland (1969). Since then, various extensions and related methods have been proposed, such as branch-and-reduce (Ryoo and Sahinidis 1995, 1996), symbolic branch-and-bound (Smith and Pantelides 1997, 1999), and branch-and-cut (Tawarmalani and Sahinidis 2005) methods. For extensive surveys on deterministic global optimization, we refer to reviews (Floudas and Gounaris 2009, Tuy et al. 2013) and monographs (Horst and Tuy 1996, Floudas 2000, Tawarmalani and Sahinidis 2002b, Locatelli and Schoen 2013). A recent review on domain reduction techniques is provided in Puranik and Sahinidis (2017). Well-known implementations of state-of-the-art SBB solvers are ANTIGONE (Misener and Floudas 2014), BARON (Sahinidis 1996, Tawarmalani and Sahinidis 2004), Couenne (Belotti et al. 2009), LINDOGlobal (Lin and Schrage 2009), and SCIP (Achterberg 2009, Vigerske and Gleixner 2018).

In fact, computing valid upper and lower bounds to v^* is crucial for the *convergence* of such algorithms, as they iteratively approximate v^* by such bounds and terminate if the gap is sufficiently close. Therefore, they rely on bounds that are valid and improve with smaller box sizes.

However, although there has been substantial research in global optimization on developing efficient lower bounding procedures that yield converging bounds on smaller boxes (Adjiman et al. 1998a, b; Tawarmalani and Sahinidis 2005), much less focus has been on (convergent) upper bounding procedures.

Upper bounds to the globally minimal value v^* are often constructed by explicitly evaluating the objective function in feasible points of $P(B)$ or by applying local solvers, which implicitly make use of such evaluation. In this context, most articles from the literature either address the problem of accelerating global solvers by finding good feasible points early in the solution process without the aim of yielding a proven deterministic global solver, or they propose upper bounding techniques which provide sufficiently good upper bounds for many practical applications but are not guaranteed to ensure convergence (Bonami et al. 2009, Berthold and Gleixner 2014). One such approach is solving the nonconvex problem $P(B)$, or some subproblem $P(X)$, locally (Androulakis et al. 1995).

Because exact feasibility is hard to ensure, a common and related concept is to accept so-called ε_f-feasible points, that is, points $x \in B$ with $g_i(x) \le \varepsilon_f, i \in I,$ and $|h_j(x)| \le \varepsilon_f, j \in J,$ with $\varepsilon_f > 0$. However, this concept does not work in general to obtain upper bounds, even for ε_f close to zero, as discussed by Tuy (2010) and Kirst et al. (2015).

Alternatively, in some algorithms, to obtain upper bounds for v^*, the strategy is to compute upper bounds $\overline{v}(X)$ for the objective function over whole subboxes X of B, for instance by using interval arithmetic (Neumaier 1990, Moore et al. 2009). Such a bound $\overline{v}(X)$ is only a valid upper bound for v^* if the associated box X contains a feasible point, though. Otherwise, we cannot rule out the case $f(x) \le \overline{v}(X) < v^*$ for all $x \in X$. Consequently, this approach is closely related to feasibility verification.

For nonconvex problems, determining feasible points, or at least verifying their existence, is an NP-hard problem. As mentioned, conventional solution methods may fail to detect feasible points and thus fail to construct valid upper bounds. Given the previous arguments, such solvers are not guaranteed to terminate for arbitrary nonconvex problems of type $P(B)$. In particular, this is true for obtaining upper bounds by solving the nonconvex problem locally, even though this approach provides sufficiently good upper bounds *on many* practical applications.

For this reason, to ensure convergence of SBB methods, more sophisticated approaches are required to verify the existence of feasible points and to compute valid upper bounds. In the context of rigorous upper bound determination and constraint satisfaction, such feasibility verification approaches are proposed in Domes and Neumaier (2015) and Kearfott (1998, 2014). They are based on computing approximately feasible points, for example, by using conventional nonlinear solvers and then verifying the existence of feasible points in specifically constructed boxes around such points using interval Newton methods. Those approaches are not guaranteed to identify feasible points, however, as they rely on heuristics to construct appropriate boxes, which need to be sufficiently small to allow for feasibility verification but at the same time sufficiently large to cover feasible points. In our emphasis on the convergence of the upper bounding procedure, our work clearly differs from these methods.

For the case of purely inequality and box constrained problems, a convergent upper bounding procedure is presented in Kirst et al. (2015) based on perturbing infeasible iterates along Mangasarian-Fromovitz directions. Because reformulating equality constraints by two reverse inequality constraints destroys the Mangasarian-Fromovitz constraint qualification (MFCQ), it is not straightforward to extend this approach to equality constrained problems.

For solely equality and box constrained problems, a convergent upper bounding procedure is proposed by Füllner et al. (2021) based on a generalization of Miranda's theorem (Miranda 1940). For convenience, in the remainder of this article, we refer to this method as the *Miranda-based method*. Its main idea is to verify the existence of feasible points in boxes $X \subseteq B$ based on sign conditions on facets of the boxes and then to compute upper bounds for v^* by means of established upper bounding procedures on such boxes.

This method, however, does not allow for inequality constraints in problems $P(B)$. Additionally, for proven convergence, it requires that the constraints of the original box B are *strictly* satisfied in all feasible points. This is not guaranteed to be true for all problems $P(B)$ in general. In Füllner et al. (2021), it is therefore suggested to represent the original box constraints as inequality constraints $g_i(x) \leq 0$ and to introduce additional, less strict box constraints $x \in \widetilde{B}$ with $B \subset \widetilde{B}$. This reformulation ensures that the strict feasibility assumption is satisfied. However, with such reformulation, it is all the more required to apply a feasibility verification method that can handle problems with not only box and equality constraints but also inequality constraints.

A natural approach to apply feasibility verification procedures, such as the one from Füllner et al. (2021), to problems with inequality constraints is to reformulate those constraints as equality constraints. This can be achieved by introducing slack variables $y_i, i \in I$ and then replacing inequality constraints $g_i(x) \leq 0$ with $g_i(x) + y_i^2 = 0$ for all $i \in I$ (Jongen and Stein 2003). However, this slack variable approach has some huge computational drawbacks, in particular if incorporated into a branch-and-bound algorithm, which we thoroughly discuss in this article.

Therefore, in this article, we present a different approach of feasibility verification for general nonconvex problems $P(B)$. For this approach, we draw on existing convergent feasibility verification methods for solely equality and box constrained problems, such as the one presented in Füllner et al. (2021) but enhance them to problems $P(B)$ with additional inequality constraints. To this end, we consider some carefully designed approximations $\overline{P}(X)$ of the subproblems $P(X)$, containing only equality and box constraints. Such problems are then accessible to existing feasibility verification methods.

The main idea of our approximation is to consider inequality constraints as equality constraints as long as they are possibly active within a given box X. This is determined by using so-called *approximate active index sets* (Kirst et al. 2015). Because these sets converge to the actual active index set of feasible points $x \in X$ for sufficiently small boxes X, it is possible to verify the existence of feasible points also for the original problem. Moreover, we show that even with using approximating problems $\overline{P}(X)$, the case of false positive feasibility verification for $P(B)$ can be ruled out. Thus, valid upper bounds for v^* can be computed.

We prove that standard SBB algorithms from global optimization terminate under mild assumptions if our new upper bounding procedure is applied. This is what we mean by the term *convergent upper bounding procedure*. To our best knowledge, our proposed method is the first proven convergent upper bounding procedure in this context.

1.1. Contribution

Summarizing, the main contributions of this article are as follows:

1. We present a deterministic method that verifies the existence of feasible points of continuous nonconvex problems $P(B)$ on sufficiently small boxes $X \subseteq B$, presuming a known approach to verify such existence for solely equality and box constrained problems. For this method, a reformulation based on the concept of approximate active index sets is crucial.
2. Although our proposed approach can incorporate different feasibility verification methods for equality and box constrained problems, in particular, it can be used to generalize the Miranda-based method by Füllner et al. (2021) to the case with additional inequality constraints.
3. By applying established upper bounding techniques to boxes X that are proven to contain feasible points, valid upper bounds for the globally minimal value v^* of $P(B)$ can be computed. We prove that this upper bounding procedure yields convergent upper bounds in the sense that convergence of standard SBB algorithms from global optimization is ensured, if our new method is incorporated.
4. Our theoretical findings are confirmed by computational results on standard test problems.

- Our proposed reformulation approach for inequality constraints performs significantly better than the alternative approach to use slack variables. Feasibility verification and termination of a standard SBB algorithm are successful for 42 of 70 test problems within two hours in contrast to only one test problem for the slack variable approach.
- Applying the convergent Miranda-based method from Füllner et al. (2021) for feasibility verification of equality and box constrained problems within our framework performs superior to interval Newton-based approaches from the literature without proven convergence. For a full SBB algorithm, 42 instead of 26 problems (of 70 test problems) terminate successfully. Applied to an exhaustive sequence of boxes around an optimal point of the respective test problem, the Miranda-based method verifies feasibility for 106 (or with some modification 120) of 130 test problems, whereas interval Newton methods are only successful in 29 cases.

1.2. Structure

The remainder of this article is structured as follows. In Section 2, we present some general assumptions and some results on feasibility verification for solely equality and box constrained problems, which provide the basis for our enhanced method. Moreover, we discuss the Miranda-based method from Füllner et al. (2021) as one specific feasibility verification method that satisfies our requirements. In Section 3, we present our extended method, which can additionally handle inequality constraints and give a proof of convergence. Afterward, we discuss its incorporation into the SBB framework in Section 4, with special focus on determining upper bounds and proving convergence. In Section 5, our computational results are presented, with additional material provided in an Online Appendix. We conclude this article with a brief summary and an outlook on future research topics in Section 6.

The notation in this article is as follows. The gradient of a function $f_j, j \in J$, is denoted by ∇f_j. For boxes $X = [\underline{x}_1, \overline{x}_1] \times \cdots \times [\underline{x}_n, \overline{x}_n]$, the midpoint of X is denoted by $\mathrm{mid}(X) := (\overline{x} + \underline{x})/2$, while $\mathrm{w}(X) := \|\overline{x} - \underline{x}\|_2$ denotes the diagonal length of X. By $\underline{X}_i := \{x \in X \,|\, x_i = \underline{x}_i\}$ and $\overline{X}_i := \{x \in X \,|\, x_i = \overline{x}_i\}$ for $i = 1, \ldots, n$, we denote the lower and upper facets of X, respectively. The interior of some set S is denoted by $\mathrm{int}(S)$.

2. Feasibility Verification for Purely Equality and Box Constrained Problems

One key idea of our upper bounding procedure for general continuous nonconvex problems $P(B)$ is to rely on feasibility verification techniques for purely equality and box constrained problems, for example, the Miranda-based method from Füllner et al. (2021). The only requirements for such technique are formally described in the following.

A sequence of boxes $(X_k)_{k \in \mathbb{N}}$ is called an *exhaustive sequence of boxes*, if for all $k \in \mathbb{N}$ we have $X_k \subset X_{k-1}$, $X_k \neq \emptyset$ and $\lim_{k \to \infty} \mathrm{w}(X_k) = 0$ (Horst and Tuy 1996). With this concept, we are ready to state our main assumption.

Assumption 1 (Algorithm (VER)). *We assume the existence of an Algorithm (VER) such that*

- *For all problems $P(B)$ with only equality and box constraints,*
- *For all exhaustive sequences of boxes $(X_k)_{k \in \mathbb{N}}$ such that a globally minimal point $x^* \in X_k$ for all $k \in \mathbb{N}$*

There exists some sequence of positive values $(\delta_k)_{k \in \mathbb{N}}$ with $\lim_{k \to \infty} \delta_k = 0$ and some $\hat{k} \in \mathbb{N}$ such that for all $k \geq \hat{k}$ the existence of a feasible point $\tilde{x}$ with $\|x^* - \tilde{x}\|_2 \leq \delta_k$ is verified by Algorithm (VER).

Moreover, in case that the existence of a feasible point is verified for some index k, we assume Algorithm (VER) to return δ_k.

In principle, Assumption 1 allows for general choices of Algorithm (VER). However, to the best of our knowledge, the only existing procedure that is *proven* to fulfill this requirement is the Miranda-based method proposed in Füllner et al. (2021). However, even for this procedure, thus far no rigorous implementation is available. By *rigorous* we mean an implementation that takes rounding errors due to finite precision of floating point arithmetic into account, for example, by means of interval arithmetic or infinite precision toolboxes.

Because the Miranda-based method is a reasonable choice for Algorithm (VER) and also applied in our computational experiments in Section 5, we briefly outline some of its main ideas in this section. It is based on Miranda's theorem, which is introduced and proven in Miranda (1940). The theorem provides a sufficient condition for the existence of zeros of nonlinear equality systems inside some box X based on signs of the involved functions on facets of X.

As the theorem is directly applicable only to systems of equations and thus problems satisfying $q = n$, an extension to the more general case $q \leq n$ is formulated and proven in Füllner et al. (2021). This is stated in Theorem 1.

Theorem 1 (Theorem 3.2 in Füllner et al. 2021). *Let $X \subseteq \mathbb{R}^n$ be a box and let function $h = (h_1, \ldots, h_q)$ be continuous on X with $q \leq n$. Let there exist a set of indices $S := \{s_1, \ldots, s_q\}$ with $S \subseteq \{1, \ldots, n\}$ so that h satisfies the conditions*

$$h_j(x) \leq 0, \quad \forall x \in \underline{X}_{s_j},$$

$$h_j(x) \geq 0, \quad \forall x \in \overline{X}_{s_j}$$

for all $j \in J$. Then $h(x) = 0$ has a solution in X.

Remark 1. The Miranda-based method cannot be applied to systems of equations with $q > n$. In such a case, however, LICQ is violated in all feasible points.

However, even the extension in Theorem 1 provides only a sufficient, but not necessary, condition for the existence of a feasible point inside a box X. In Füllner et al. (2021), three cases are identified, in which the box X contains a feasible point $\hat{x}$, but the conditions of Theorem 1 are not satisfied. These cases are illustrated in Figure 1. In the following, we briefly discuss each one.

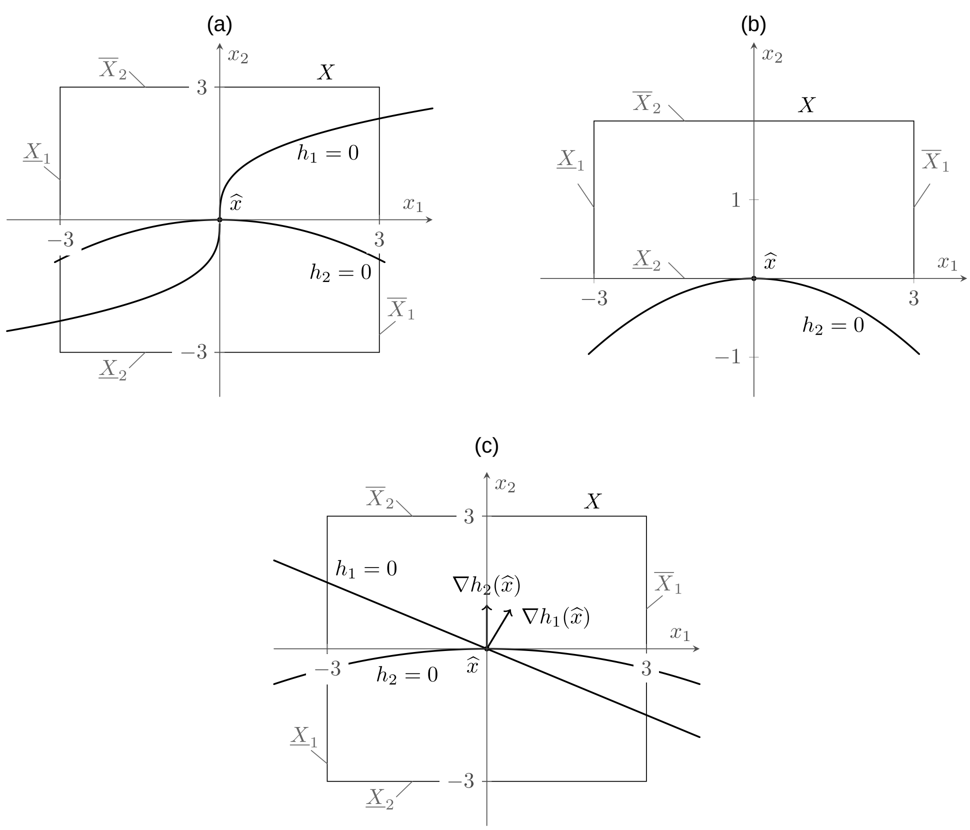
(C1) **The box *X* is too large.** Even if the considered box X contains a feasible point, if it is too large, Theorem 1 may not be applicable to verify the existence of such a point. This case is displayed in Figure 1 (C1), which shows the box X and the level curves of h_1 and h_2 to level 0. Although

$$\max_{x \in \underline{X}_2} h_2(x) < 0 \qquad \text{and} \qquad \min_{x \in \overline{X}_2} h_2(x) > 0,$$

the function h_1 intersects with both facets $\underline{X}_1$ and $\overline{X}_1$. Thus, the conditions of Theorem 1 are not satisfied. In contrast, in the case that the box X is of less width in x_2-direction, the conditions are fulfilled.

(C2) **The feasible point $\hat{x}$ is located on a facet.** If $\hat{x}$ is located on a facet of X, Theorem 1 may not be satisfied either. This case is displayed in Figure 1 (C2), where $\hat{x}$ is located on $\underline{X}_2$. As constraint h_2 solely takes nonnegative values on both facets, the existence of a feasible point cannot be deduced by Theorem 1. In contrast to case (C1), this issue persists for smaller subboxes as long as $\hat{x}$ remains located on a facet.

Figure 1. Cases Where a Feasible Point Cannot Be Verified Using Theorem 1 (Füllner et al. 2021)



Notes. (a) Box X is to too large. (b) Feasible point on facet $\underline{X}_2$. (c) Gradients not directed toward facets.

(C3) **Gradients in feasible point $\hat{x}$ are not directed toward the facets.** Miranda's theorem exploits that the functions $h_j, j \in J$, take different signs on the associated opposing facets of X. The simplest setting for which this property is satisfied, requires sufficiently small boxes X (see (C1)), a feasible point $\hat{x}$ in the interior of X (see (C2)) and the gradients $\nabla h_j(\hat{x})$ directed toward the facets (i.e., $\nabla h_j(\hat{x}) = \lambda e_j$ with $\lambda > 0$ and e_j a unit vector with only zero entries except for a one at the j-th component), thus causing opposing signs on those facets. In contrast, if some of the constraint gradients $\nabla h_j(x)$ do not have unit direction in $\hat{x}$, the conditions of Theorem 1 may not be satisfied, even independent of the size of X. This case is displayed in Figure 1 (C3). In this example, $\nabla h_1(\hat{x})$ does not have unit direction, and thus h_1 intersects with the facets $\underline{X}_1$ and $\overline{X}_1$. As a result, Theorem 1 is not applicable.

The Miranda-based method deals with all three cases by appropriate transformations. Briefly summarized, (C1) is ruled out for sufficiently small boxes X, (C2) is taken care of by artificially enlarging the boxes X before checking if Theorem 1 is satisfied, and (C3) is taken care of by a coordinate transformation. For details, see Füllner et al. (2021). These transformations ensure that under certain assumptions, the conditions of Theorem 1 are satisfied for sufficiently small boxes X and that the existence of feasible points can be proven by applying Theorem 1.

For this result to hold, certain assumptions have to be satisfied (Füllner et al. 2021). The box constraints $x \in B$ are required to be strictly satisfied by all feasible points, and as already discussed in Section 1, LICQ is assumed to be satisfied in all globally minimal points. This LICQ requirement rules out the occurrence of case (C2) for globally minimal points x^*, as in case (C2) LICQ is always violated. However, it is still reasonable to take care of (C2) algorithmically, because, when integrated into a spatial branch-and-bound algorithm, this allows us to verify the existence of suboptimal feasible points $\hat{x}$ using Theorem 1, for which LICQ is not presumed to hold.

Another requirement is that the sequence of boxes in Assumption 1 is not only exhaustive but also nondeformed in the sense of the following definition.

Definition 1 (From Füllner et al. 2021). Let $(X_k)_{k\in\mathbb{N}}$ be a sequence of boxes with the maximum ratio of the length of box edges

$$t^k := \frac{\max_{i=1,\ldots,n}(\overline{x}_i^k - \underline{x}_i^k)}{\min_{i=1,\ldots,n}(\overline{x}_i^k - \underline{x}_i^k)},$$

bounded above by a constant $\overline{t} < \infty$. Then we call $(X_k)_{k\in\mathbb{N}}$ a *nondeformed sequence of boxes*.

This can be regarded to be a mild assumption. For example, using the standard branching technique within spatial branch-and-bound algorithms, dividing along a longest edge, it is naturally satisfied (Füllner et al. 2021).

Summarized, we obtain the following result.

Proposition 1. *The Miranda-based method from Füllner et al.* (2021) *satisfies Assumption* 1 *provided that the following additional assumptions hold*:

(A1) *The box constraints of problem P(B) are strictly satisfied.*
(A2) *LICQ is satisfied in all globally minimal points.*
(A3) *All exhaustive sequences of boxes are nondeformed*.

Proof. The assertion follows immediately from lemma 4.10, theorem 4.11, and corollary 4.12 in Füllner et al. (2021). □

It is worth mentioning that the last part of Assumption 1 regarding the return of some valid $\delta_k > 0$ is not actually required for feasibility verification itself, but turns out to be crucial later in the branch-and-bound context when upper bounds for the globally minimal value v^* are computed. Such upper bounds can be computed on boxes $Y_k \subseteq B$, in which the existence of a feasible point is verified. These boxes Y_k can, but must not necessarily, coincide with the boxes X_k, which are the input of Algorithm (VER). Using the returned $\delta_k > 0$, an appropriate such box can be constructed by

$$Y_k = [\underline{x}_1 - \delta_k, \overline{x}_1 + \delta_k] \times \cdots \times [\underline{x}_n - \delta_k, \overline{x}_n + \delta_k]. \tag{1}$$

This follows directly from Assumption 1.

In fact, the Miranda-based method from Füllner et al. (2021) does not return δ_k explicitly. Instead, it directly returns a box Y_k, which is guaranteed to contain a feasible point. However, by relation (1), for different choices of Algorithm (VER), the weaker requirement in Assumption 1 is sufficient.

3. Extension to Problems with Inequality Constraints

To extend Algorithm (VER) to problems $P(B)$ with additional inequality constraints, we aim at reducing this case to the purely equality and box constrained one from the previous section. Two different approaches to achieve this are proposed in the following two sections. We show that both techniques verify the existence of feasible points for sufficiently small boxes and thus can be applied to obtain upper bounds for v^*. This yields a convergent upper bounding procedure in the sense that sufficiently good upper bounds are found within a finite number of iterations when inserted into a branch-and-bound framework, as we show in Section 4.

3.1. Slack Variable Approach

One way to transform $P(B)$ into a problem that is only constrained by equality constraints is to replace all inequality constraints $g_i(x) \leq 0$, $i \in I$, by equality constraints using a slack variable reformulation.

This is done by introducing additional free variables y_i, $i \in I$, and replacing the constraints $g_i(x) \leq 0$ by equality constraints of the form $g_i(x) + y_i^2 = 0$ for all $i \in I$, as discussed in Jongen and Stein (2003). Then, for every globally minimal point x^* of $P(B)$, there exist globally minimal points (x^*, y^*) of the transformed problem with the same optimal value v^*. As both $y_i^* = \sqrt{-g_i(x^*)}$ and $y_i^* = -\sqrt{-g_i(x^*)}$ are possible solutions for each $y_i^*, i \in I$, the number of globally minimal points of the reformulated problem may increase exponentially with the number of introduced slack variables and thus with the number of inequality constraints in $P(B)$ (Jongen and Stein 2003). It also grows exponentially in the number of globally minimal points of the original formulation. This is computationally unfavorable, in particular for a branch-and-bound algorithm. First, the dimension n of the problem is increased. Second, multiple globally minimal points may lead to long execution times of the algorithm, as they prevent boxes from being pruned and thus counteract a confined and deep search. These theoretical observations are supported by computational tests in Section 5.

To apply some Algorithm (VER) satisfying Assumption 1 to the reformulated problem with slack variables, the box B has to be extended to include components associated with the slack variables y_i. As these variables are free, bounds have to be derived for such extension. This is possible by computing a lower bound $\ell_{g_i}(B)$ for the values of g_i on B, for example, by interval arithmetic. Then, y_i has to satisfy the box constraint $y_i \in [-\ell_{g_i}(B), \ell_{g_i}(B)]$.

Finally, let us stress that at least in theory it is also possible to use linear slack variables of the form $g_i(x) + y_i = 0$ with the additional requirement $y_i \geq 0$ for all $i \in I$. This, however, contradicts Assumption 1. Therefore, the box constraints for the slack variables need to be slightly relaxed such that we also allow for negative values of y_1 close to zero. Unfortunately, this is equivalent to accepting ε_f-feasible points. In addition, this may easily lead to a situation similar to case (C2) in Figure 1 and may introduce comparable difficulties.

3.2. Using Approximate Active Index Sets

In this section, we propose an alternative reformulation approach to obtain purely equality and box constrained problems without the requirement of slack variables. Our approach is based on the concept of active indices and certain approximations that are described in Section 3.2.1. Using this, we state our extended feasibility verification method enhancing Algorithm (VER) in Section 3.2.2 together with a proof of convergence.

3.2.1. Approximating Active Index Sets. In a given feasible point $\hat{x} \in M(B)$, inequality constraints $g_i(x) \leq 0$ can either hold with equality, that is, $g_i(\hat{x}) = 0$, or with strict inequality, that is, $g_i(\hat{x}) < 0$. The former constraints are called *active* in $\hat{x}$, whereas the latter are called *inactive* in $\hat{x}$. The indices $i \in I$ of all inequality constraints that are active in $\hat{x}$ are contained in the active index set in that point, which is defined as follows.

Definition 2 (Active Index Set). Let $\hat{x} \in M$ be a feasible point of $P(B)$. Then the *active index set* (or set of active indices) I_0 in $\hat{x}$ is defined as

$$I_0(\hat{x}) := \{i \in I | g_i(\hat{x}) = 0\}.$$

A key element of our approach is the observation that a problem with inequality constraints can *locally* be viewed as a purely equality constrained optimization problem where the inequality constraints that are active at the optimal point x^* are treated as equality constraints (and those that are not active at x^* are omitted). This interpretation does only work locally, however, for instance on small boxes.

Unfortunately, the active index sets for different points in a box might differ and are not known in advance. To circumvent this difficulty, we consider certain approximations of the set of active indices that turn out to be helpful for our purpose. To achieve this, again, we consider exhaustive sequences $(X_k)_{k\in\mathbb{N}}$ of boxes with $x^* \in X_k$ for all $k \in \mathbb{N}$ and x^* a globally minimal point of $P(B)$. Additionally, we make use of bounds to the function values of all constraints on a given box X_k. In this context, we need the following concepts.

Definition 3 (M-Independent Bounding Procedures from Kirst et al. 2015). A function $\ell_f(\cdot)$ on an exhaustive sequence of boxes $(X_k)_{k\in\mathbb{N}}$ is called an M-independent lower bounding procedure for the function f, if it satisfies

$$\ell_f(X_k) \leq \min_{x\in X_k} f(x)$$

for all subboxes $X_k \subset B$. Analogously, a function $u_f(\cdot)$ on an exhaustive sequence of boxes $(X_k)_{k\in\mathbb{N}}$ is called an M-independent upper bounding procedure for the function f, if it satisfies

$$u_f(X_k) \geq \max_{x\in X_k} f(x)$$

for all subboxes $X_k \subset B$.

Furthermore, such a procedure is called convergent if it satisfies

$$\lim_{k\to\infty} \ell_f(X_k) = \lim_{k\to\infty} \min_{x\in X_k} f(x) \qquad \text{or} \qquad \lim_{k\to\infty} u_f(X_k) = \lim_{k\to\infty} \max_{x\in X_k} f(x),$$

respectively.

With these concepts, we are ready to describe the following approximation of the set of active indices, which is crucial for our proposed method.

Definition 4 (Approximate Active Index Set Based on Kirst et al. 2015). Let $X_k \subset \mathbb{R}^n$ be a box and $P(X_k)$ the problem $P(B)$ restricted to X_k. Let $\ell_{g_i}(X_k)$ and $u_{g_i}(X_k)$, $i \in I$, denote convergent M-independent lower and upper bounding procedures of $g_i, i \in I$, applied to X_k. Then the set

$$\bar{I}_0(X_k) := \{i \in I \mid \ell_{g_i}(X_k) \leq 0 \leq u_{g_i}(X_k)\}$$

is called an approximate active index set on X_k.

The set $\bar{I}_0(X_k)$ contains the indices of all inequality constraints that could be satisfied with equality in a feasible point in X_k. It is determined by constructing lower and upper bounds to the values of all inequality constraints.

Those lower and upper bounds for the values of $g_i(x), i \in I$, on a given box $X_k \subset B$ allow conclusions regarding the satisfiability of the constraints. If $u_{g_i}(X_k) < 0$ for some constraint $i \in I$, the constraint can be omitted from the problem $P(X_k)$, as it is strictly satisfied for all $x \in X_k$ and thus obsolete. Furthermore, if $\ell_{g_i}(X_k) > 0$ for at least one $i \in I$, the problem $P(X_k)$ has no feasible point, because at least one inequality constraint is violated for all $x \in X_k$. The remaining constraints $i \in I$ with $\ell_{g_i}(X_k) \leq 0 \leq u_{g_i}(X_k)$ are contained in the approximate active index set $\bar{I}_0(X_k)$.

Our approach to extend Algorithm (VER) to inequality constraints is based on locally treating the constraints $g_i(x) \leq 0$ with $i \in \bar{I}_0(X_k)$ as equality constraints $g_i(x) = 0$ for a given problem $P(X_k)$. As the resulting problem is only equality and box constrained, Algorithm (VER) can be applied to verify the existence of feasible points.

The active index set $I_0(\hat{x})$ in a feasible point $\hat{x} \in X_k$ is a subset of the approximate active index set $\bar{I}_0(X_k)$, but both sets do not have to coincide. However, in exhaustive sequences of boxes, the approximation becomes better with smaller boxes, such that for sufficiently small boxes X_k with $\hat{x} \in X_k$ both sets are equal. This is stated in Lemma 1.

Lemma 1 (Lemma 4.9 in Kirst et al. 2015). *Let $(X_k)_{k\in\mathbb{N}}$ be an exhaustive sequence of boxes and let $\hat{x} \in X_k$ for all $k \in \mathbb{N}$. Then there exists some $\hat{k} \in \mathbb{N}$, such that $\bar{I}_0(X_k) = I_0(\hat{x})$ for all $k \geq \hat{k}$.*

Using these techniques we are ready to state our extended feasibility verification method.

3.2.2. Verifying Feasible Points.

Our proposed method is described formally in Algorithm 1, where for notational convenience we omit the index k.

Algorithm 1 (Feasibility Verification for Problems $P(B)$ Including Inequality Constraints)
Input: Problem $P(X)$ with $X \subseteq B$
1: Determine lower bounds $\ell_{g_i}(X)$ for all $i \in I$ and $\ell_{h_j}(X)$ for all $j \in J$ on the box X using some convergent M-independent lower bounding procedure.
2: Determine upper bounds $u_{g_i}(X)$ for all $i \in I$ and $u_{h_j}(X)$ for all $j \in J$ on the box X using some convergent M-independent upper bounding procedure.
3: **if** $\exists i \in I$ with $\ell_{g_i}(X) > 0$ or $\exists j \in J$ with $(\ell_{h_j}(X) > 0$ or $u_{h_j}(X) < 0)$ **then**
4: Set flag = `no feasible point possible in` X.
5: **else**

6: Determine the approximate active index set

$$\bar{I}_0(X) := \{i \in I \,|\, \ell_{g_i}(X) \le 0 \le u_{g_i}(X)\}$$

with $|\bar{I}_0(X)| = p_0$.
7: Determine problem $\overline{P}(X)$ based on $\bar{I}_0(X)$.
8: Apply Algorithm (VER) to problem $\overline{P}(X)$.
9: **if** existence of a feasible point is verified **then**
10: With $\delta > 0$ returned by Algorithm (VER) construct a box

$$Y = [\underline{x}_1 - \delta, \overline{x}_1 + \delta] \times \cdots \times [\underline{x}_n - \delta, \overline{x}_n + \delta].$$

11: Set flag = `existence of feasible point verified`.
12: **else**
13: Set flag = `no successful verification`.
Ensure: flag and, if verification successful, box Y containing a feasible point.

First, for a given box X_k, some convergent M-independent lower and upper bounding procedures, for example, interval arithmetic (Neumaier 1990) or centered forms (Krawczyk and Nickel 1982), are applied to the functions $g_i, i \in I$, and $h_j, j \in J$. This way, it is checked whether the existence of a feasible point in X_k can be directly ruled out (steps 1–4).

If this is not the case, the approximate active index set $\bar{I}_0(X_k)$ is determined and the corresponding inequality constraints $g_i(x) \le 0, i \in \bar{I}_0(X_k)$, are treated as equality constraints (steps 6 and 7). This yields the new constraint system $h_j(x) = 0, j \in \bar{J}(X_k)$. By reindexing, the index set can be defined as $\bar{J}(X_k) = \{1, \ldots, q, q+1, \ldots, q + p_0^k\}$ with $p_0^k := |\bar{I}_0(X_k)|$ the number of approximately active equality constraints. We then have $h_{q+\ell} = g_{i_\ell}$, where the subscript $\ell = 1, \ldots, p_0^k$ indicates the approximately active equality constraints.

In this way, the subproblem $P(X_k)$ is transformed to the problem

$$\overline{P}(X_k): \quad \min_{x \in \mathbb{R}^n} \quad f(x) \quad \text{s.t.} \quad h_j(x) = 0, \quad \forall j \in \bar{J}(X_k), \quad x \in X_k.$$

After this transformation, Algorithm (VER) can be applied because no inequality constraints, except for box constraints, are present anymore (step 8).

By Assumption 1, Algorithm (VER) returns a valid value δ_k in the case of successful feasibility verification. Using Relation (1), a box Y_k can then be constructed, which is guaranteed to contain a feasible point (steps 9–14).

In the remainder of this section, we discuss the implications of this transformation and prove that feasibility verification is guaranteed to be achieved for sufficiently small boxes. To this end, let $\overline{M}(X_k)$ denote the feasible set and $\bar{v}(X_k)$ the optimal value of the transformed problem $\overline{P}(X_k)$, and $M(X_k)$ and $v(X_k)$ those of the original problem restricted to X_k.

First of all, we can conclude that $\overline{M}(X_k)$ is a subset of $M(X_k)$ by construction of $\bar{J}$. This implies that all feasible points of $\overline{P}(X_k)$ are also feasible for $P(X_k)$ and for $P(B)$.

Lemma 2. *The feasible set $\overline{M}(X_k)$ of $\overline{P}(X_k)$ is a subset of the feasible set $M(X_k)$ of $P(X_k)$.*

Considering some exhaustive sequence of boxes $(X_k)_{k\in\mathbb{N}}$ with $x^* \in X_k$ for all $k \in \mathbb{N}$, in case that k is not sufficiently large, the approximate active index set $\bar{I}_0(X_k)$ and the active index set $I_0(x^*)$ do not necessarily coincide. This means that some inequality constraints are treated as active inequality constraints, that is, as equality constraints, on the box X_k, although they are not actually active in the globally minimal point $x^* \in X_k$. For $\bar{I}_0(X_k) \ne I_0(x^*)$, the feasible set $\overline{M}(X_k)$ might even be empty.

However, as we have $\overline{M}(X_k) \subseteq M(X_k)$ by Lemma 2, it is ruled out that Algorithm 1 verifies the existence of a feasible point of $\overline{P}(X_k)$ that is not feasible for the original problem $P(X_k)$ (no *false positives*). Therefore, treating nonactive inequality constraints as active may only prevent feasible points of $P(X_k)$, such as x^*, to be detected if they are not feasible for $\overline{P}(X_k)$. For sufficiently large k, that is, for sufficiently small boxes X_k, this cannot happen, as we show now (no *false negatives* for sufficiently large k).

In Assumption 1, we assume an exhaustive sequence of boxes $(X_k)_{k\in\mathbb{N}}$ with $x^* \in X_k$ for all $k \in \mathbb{N}$ and x^* a globally minimal point of $P(B)$. This is crucial to ensure verification of the existence of feasible points of $P(B)$ sufficiently close to x^* for sufficiently large k. Considering $\overline{P}(X_k)$ instead of $P(X_k)$ in each iteration, for sufficiently large k, such verification is possible because the approximate active index set $\overline{I}_0(X_k)$ converges to $I_0(x^*)$.

Lemma 3. *Let $(X_k)_{k\in\mathbb{N}}$ be an exhaustive sequence of boxes with $X_k \subseteq B$ for all $k \in \mathbb{N}$. Furthermore, let x^* be a globally minimal point of $P(X_k)$ with $x^* \in X_k$ for all $k \in \mathbb{N}$. Then there exists some $\hat{k} \in \mathbb{N}$ so that for all $k \geq \hat{k}$, x^* is also a globally minimal point of $\overline{P}(X_k)$.*

Proof. With Lemma 2, we have $\overline{M}(X_k) \subseteq M(X_k)$. Because $P(X_k)$ and $\overline{P}(X_k)$ share the same objective function $f(x)$, this implies $\overline{v}(X_k) \geq v(X_k) = f(x^*)$ for all $k \in \mathbb{N}$.

With Lemma 1, there exists some $\hat{k} \in \mathbb{N}$ such that $\overline{I}_0(X_k) = I_0(x^*)$ for all $k \geq \hat{k}$. Thus, we have $x^* \in \overline{M}(X_k)$ for all $k \geq \hat{k}$. It follows $f(x^*) \geq \overline{v}(X_k)$ for all $k \geq \hat{k}$. Combining this with $\overline{v}(X_k) \geq f(x^*)$, we obtain $v(X_k) = \overline{v}(X_k)$ for all $k \geq \hat{k}$ and the assertion follows. □

With Lemma 3, we can now state the main result of this section.

Theorem 2. *Applying Algorithm 1 to some problem P(B), for all exhaustive sequences of boxes $(X_k)_{k\in\mathbb{N}}$ with a globally minimal point $x^* \in X_k$ for all $k \in \mathbb{N}$, there exists some sequence of positive values $(\delta_k)_{k\in\mathbb{N}}$ with $\lim_{k\to\infty} \delta_k = 0$ and some $\widetilde{k} \in \mathbb{N}$ such that for all $k \geq \widetilde{k}$ the existence of a feasible point $\widetilde{x}$ with $\|x^* - \widetilde{x}\|_2 \leq \delta_k$ is verified.*

Proof. Following from Lemma 2, it is impossible that Algorithm 1 verifies the existence of a feasible point of $\overline{P}(X_k)$ that is not feasible for $P(X_k)$. Moreover, by Lemma 3, there exists some $\hat{k} \in \mathbb{N}$ such that x^* is also a globally minimal point of $\overline{P}(X_k)$ for all $k \in \mathbb{N}$ with $k \geq \hat{k}$. Thus, the exhaustive sequence of boxes $(X_k)_{k\in\mathbb{N}}$ also contracts around a globally minimal point of $\overline{P}(X_k)$.

As the transformed problem $\overline{P}(X_k)$ is only constrained by equality and box constraints, Algorithm (VER) can be applied to it. All components of Assumption 1 with $P(X_{\hat{k}})$ in the role of $P(B)$ are fulfilled. With this, the existence of some $\widetilde{k} \in \mathbb{N}$ as stated in the assertion follows. □

Theorem 2 states that Algorithm 1 verifies the existence of a feasible point of $P(B)$ for some sufficiently small subbox $X_k \subseteq B$. In particular, the Miranda-based method from Füllner et al. (2021) can be applied as Algorithm (VER) in Algorithm 1 to ensure feasibility verification for sufficiently small boxes.

Corollary 1. *If Assumption 1,* (A1)–(A3), *is satisfied, then the Miranda-based method from Füllner et al.* (2021) *can be used as Algorithm (VER) in Algorithm 1 to verify the existence of feasible points of problem P(B). Then, considering some exhaustive sequence of boxes with $x^* \in X_k$ for all $k \in \mathbb{N}$ and x^* a globally minimal point, there exists some $\hat{k} \in \mathbb{N}$ such that for all $k \geq \hat{k}$ the existence of a feasible point $\widetilde{x}$ with $\|x^* - \widetilde{x}\|_2 \leq \delta_k$ is verified.*

Proof. This directly follows from Proposition 1 and Theorem 2. □

3.3. On Strict Satisfaction of Box Constraints

We should point out again that choosing the Miranda-based method from Füllner et al. (2021) as Algorithm (VER) in Algorithm 1 requires all box constraints to be strictly satisfied by feasible points to allow for their verification; see (A1) in Proposition 1 (this is a sufficient condition for its convergence, but not necessary for feasibility verification in every case). Usually, for some arbitrary problem $P(B)$, it is not known in advance if this assumption is satisfied. However, it can always be guaranteed to be satisfied by adapting problem $P(B)$ appropriately. More precisely, the initial box constraints $x \in B$ can be considered as standard inequality constraints $g_i(x) \leq 0$. To retain the structure of the original problem, then additional, but less strict, box constraints $x \in \widetilde{B}$ can be introduced. As Corollary 1 shows, Algorithm 1 can then be applied for feasibility verification. In this sense, (A1) in Proposition 1 can be considered a weak assumption.

However, it is important to mention that in some cases, the proposed reformulation of the original box constraints comes with the drawback that active box constraints, if interpreted as standard inequality constraints, increase the size of $\overline{I}_0(X_k)$, and thus may favor that $|\overline{I}_0(X_k)| + q > n$, which in the light of Remark 1 rules out feasibility verification using the Miranda-based method. We take up on this observation in Section 5.

Finally, from this reformulation perspective, our proposed feasibility verification method Algorithm 1 is not only suited for problems of form $P(B)$, which naturally contain inequality constraints. It is also suited for purely equality and box constrained problems, for which inequality constraints are artificially introduced by some reformulation, for example, to ensure strict satisfaction of box constraints. We also address this in our computational tests in Section 5.

4. Computing Upper Bounds Within an SBB Framework

In this section, we use our presented feasibility verification method to determine upper bounds on the returned box Y, in which the existence of a feasible point is verified. This upper bounding procedure is incorporated into an SBB algorithm. As we show, the upper bounds are convergent in the sense that sufficiently good upper bounds are found within a finite number of iterations and convergence of the SBB method is guaranteed.

As noted earlier, if the considered problem is nonconvex, convergence of these methods from the literature often cannot be guaranteed, as they might fail to determine valid upper bounds for the globally minimal value v^*. However, incorporating our proposed Algorithm 1 into the SBB framework can guarantee (the determination of) sufficiently good upper bounds, and thus the algorithm converges. Convergence of lower bounds $\ell(X_k)$ commonly used in the literature is usually ensured in the sense that it satisfies

$$\lim_{k\to\infty} \ell(X_k) = \lim_{k\to\infty} \min_{x\in M(X_k)} f(x). \tag{2}$$

A standard SBB framework is formally stated in Algorithm 2. It is primarily designed to approximate the minimal value v^* of $P(B)$. In the remainder of this section, we briefly explain each step of the algorithm and give a proof of convergence.

Algorithm 2 (Convergent SBB Algorithm)

Input: Problem $P(B)$
1: **Initialize**
2: Set the iteration counter to $k = 0$.
3: Choose a tolerance $\varepsilon > 0$.
4: Set the lower bound for v^* to $\ell_0 = -\infty$.
5: Set the upper bound for v^* to $u_0 = +\infty$.
6: Set $\mathcal{L} = \{(B, l_0)\}$.
7: **while** $u_k - \ell_k \le \varepsilon$ or $\mathcal{L} = \emptyset$ **do**
8: Increase the iteration counter k by one.
9: Choose a pair $(X_k, \ell(X_k))$ from $\mathcal{L}$ with the smallest $\ell(X_k)$ and remove it.
10: Divide X_k along its longest edge into subboxes X_k^1 and X_k^2.
11: **for** $l = 1, 2$ **do**
12: Determine a lower bound $\ell_f(X_k^l)$ for the minimal value of $P(B)$ on X_k^l using some lower bounding procedure, with $\ell(X_k^l) = +\infty$ for an infeasible problem.
13: If $\ell(X_k^l) < +\infty$ and $\ell(X_k^l) \le u_{k-1}$, add the pair $(X_k^l, \ell(X_k^l))$ to $\mathcal{L}$. Otherwise, continue with step 11.
14: Apply Algorithm 1 to $P(X_k^l)$ to verify the existence of feasible points.
15: **if** Feasibility verification successful **then**
16: Compute an upper bound $u_k^l := u_f(Y_k^l)$ for f over box Y_k^l, which is returned by Algorithm 1.
17: **else**
18: Set $u_k^l := +\infty$.
19: Compute the best upper bound for v^* as $u_k = \min\{u_{k-1}, u_k^l\}$.
20: Save the box Y_k related to u_k as best known box X_k^*.
21: Remove all pairs $(\widetilde{X}, \ell(\widetilde{X}))$ with $\ell(\widetilde{X}) > u_k$ from $\mathcal{L}$.
22: **if** $\mathcal{L} \neq \emptyset$ **then**
23: Update the lower bound for v^* to $\ell_k = \min\{\ell(\widetilde{X}) \,|\, (\widetilde{X}, \ell(\widetilde{X})) \in \mathcal{L}\}$.

Output: Bounds ℓ_k and u_k approximating v^*. Box X_k^* containing a feasible point and corresponding to the best known upper bound u_k.

After initialization, in a first step, the box with the smallest corresponding lower bound $\ell_f(X_k)$ is chosen and removed from $\mathcal{L}$, as it is the most promising box from a minimization perspective (step 9). This box is then divided along its longest edge into two subboxes X_k^1 and X_k^2 (step 10). In general, more sophisticated box division strategies are possible as well, as long as they yield nondeformed exhaustive sequences of boxes.

Next, both boxes X_k^1 and X_k^2 are examined one after another (step 11). First, a lower bound $\ell_f(X_k^l)$ for the minimal value of the objective function on X_k^l is determined (step 12). Typically, this is considered the crucial step within an SBB method. Most commonly, the lower bound $\ell_f(X_k^l)$ is computed by solving special convex relaxations of $P(X_k^l)$, which are obtained by replacing the functions in $P(X_k^l)$ with convex envelopes (Tawarmalani and Sahinidis 2002a), convex underestimators (Adjiman et al. 1998a, b) or LP relaxations (Tawarmalani and Sahinidis 2002b). The tuple $(X_k^l, \ell_f(X_k^l))$ is then appended to $\mathcal{L}$, if it cannot be ruled out immediately (step 13).

Following that, it is attempted to determine an upper bound $u_f(X_k^l)$. Upper bounds are usually determined by evaluating the objective function f of $P(B)$ in a feasible point. If such a feasible point x_k^l is known inside X_k^l, then $f(x_k^l)$ serves as a valid upper bound for the minimal value of P. As opposed to standard approaches, in this article we propose to apply our feasibility verification method Algorithm 1 as a basis to determine valid upper bounds $u_f(X_k^l)$ for v^* (step 14). In case the existence of a feasible point can be verified in some box Y_k returned by Algorithm 1, such an upper bound can be computed by applying standard convergent upper bounding methods, for instance, interval arithmetic in the simplest case, over the entire box Y_k (steps 15–18). After this, the best known upper bound u_k and the corresponding box X_k^* are updated if possible (steps 19 and 20).

Once the upper bound is updated, all elements $(\widetilde{X}, \ell(\widetilde{X}))$ in $\mathcal{L}$ with $\ell(\widetilde{X}) > u_k$ are removed from $\mathcal{L}$ as the corresponding boxes cannot contain a globally minimal point of P (step 21). This is called *fathoming* or *pruning* and is crucial to limit the size of the branching tree.

As a last step (steps 22 and 23), the best known lower bound is updated to

$$\ell_k = \min\{\ell(\widetilde{X}) | (\widetilde{X}, \ell(\widetilde{X})) \in \mathcal{L}\}.$$

Using Equation (2), it can be ensured that the overall lower bound ℓ_k in Algorithm 2 converges to the globally minimal value of $P(B)$. The algorithm terminates if $u_k - \ell_k \leq \varepsilon$ (step 24); otherwise, the iteration counter is increased to $k = k + 1$ and the next iteration starts (step 8).

In SBB methods, a single point is usually stored as the currently best known solution. In contrast, here we adapt this by saving the whole box X_k^* because it is guaranteed to contain a feasible point.

We can prove now that Algorithm 2 terminates after a finite number of iterations. We start with a feasibility verification result.

Theorem 3 (Feasibility Verification). *Assume that in Algorithm 2 some convergent lower bounding procedure in the sense of Equation (2) is used. Moreover, for two boxes X and Y with $X \subset Y$, we assume $\ell(X) \geq \ell(Y)$.*

Then, if the infinite branch-and-bound procedure corresponding to $\varepsilon = 0$ does not terminate, the existence of feasible points in arbitrary small boxes around globally minimal point x^* is verified.

Proof. The proof is by contradiction. Assuming $\ell(X) \geq \ell(Y)$ for two boxes X and Y with $X \subset Y$, it holds that $\lim_{k\to\infty} \ell_k = v^*$ (Horst and Tuy 1996).

Next, we consider an exhaustive sequence of boxes $(X_{k_\nu})_{\nu\in\mathbb{N}}$ with $x^* \in X_{k_\nu}$ for all $\nu \in \mathbb{N}$ generated by Algorithm 2. Then, according to Theorem 2, there exists some $\hat{\nu}$ so that for every $\nu > \hat{\nu}$, a feasible point in some box Y_k is verified in Algorithm 1 that is called by Algorithm 2.

With $\lim_{\nu\to\infty} \delta_{k_\nu} = 0$, we also have $\lim_{\nu\to\infty} \mathrm{w}(Y_{k_\nu}) = 0$ for such boxes Y_{k_ν}. This proves that if the infinite branch-and-bound procedure does not terminate, the existence of feasible points is verified in arbitrary small boxes around x^*. □

Based on this result, convergence of Algorithm 2 can be proven.

Theorem 4 (Convergence of Algorithm 2). *Let the assumptions of Theorem 3 hold. Moreover, assume that some convergent upper bounding procedure is used to compute upper bounds at the objective function in step* 16 *of Algorithm 2 if a feasible point is verified. Then, if the infinite branch-and-bound procedure corresponding to $\varepsilon = 0$ does not terminate, we have $\lim_{k\to\infty} \ell_k = \lim_{k\to\infty} u_k = v^*$.*

Proof. As in the proof of Theorem 3, it follows that $\lim_{k\to\infty} \ell_k = v^*$. Additionally, from Theorem 3, by considering the exhaustive sequence of boxes $(X_{k_\nu})_{\nu\in\mathbb{N}}$ with $x^* \in X_{k_\nu}$ for all $\nu \in \mathbb{N}$ generated by Algorithm 2, we have $\lim_{\nu\to\infty} \mathrm{w}(Y_{k_\nu}) = 0$ for boxes Y_{k_ν}, in which the existence of feasible points is verified.

Because some convergent upper bounding procedure is used in Algorithm 2, it follows that $\lim_{\nu\to\infty} u_f(X_{k_\nu}) = \lim_{\nu\to\infty} \max_{x\in X_{k_\nu}} f(x)$. As the sequence $(X_{k_\nu})_{\nu\in\mathbb{N}}$ is exhaustive, with $x^* \in X_{k_\nu}$ for all $\nu \in \mathbb{N}$, this proves the assertion. □

With the same reasoning as before, the following result can be derived with respect to the finite termination of Algorithm 2 for some $\varepsilon > 0$.

Corollary 2 (Finite Termination of Algorithm 2). *Let the assumptions of Theorem 3 hold. Moreover, assume that some convergent upper bounding procedure is used to compute upper bounds at the objective function in step* 16 *of Algorithm 2 if a feasible point is verified. Then, for any $\varepsilon > 0$, Algorithm 2 terminates after finitely many iterations.*

Recall that the Miranda-based method from Füllner et al. (2021) satisfies all required assumptions for Algorithm (VER) in Algorithm 1. Therefore, in particular for this specific choice, convergence of the branch-and-bound method Algorithm 2 is assured.

Corollary 3. *Assume that the Miranda-based method from Füllner et al. (2021) is used as Algorithm (VER) within Algorithm 1 to verify the existence of feasible points of problem P(B) and that* (A1)–(A3) *are satisfied. Moreover, let all assumptions from Theorem 3 and Theorem 4 be satisfied. Then,*

(a) *If the infinite branch-and-bound procedure corresponding to* $\varepsilon = 0$ *in Algorithm* 2 *does not terminate, we have* $\lim_{k\to\infty} \ell_k = \lim_{k\to\infty} u_k = v^*$.

(b) *For any* $\varepsilon > 0$, *Algorithm* 2 *terminates after finitely many iterations.*

5. Computational Tests

In this section, we present computational results for applying the proposed feasibility verification method, Algorithm 1, both as a standalone method and incorporated into a convergent SBB method (presented in Algorithm 2) to several test problems from the *COCONUT benchmark* library (Shcherbina et al. 2003). We compare our proposed method with existing feasibility verification ideas from the literature, which we discuss now.

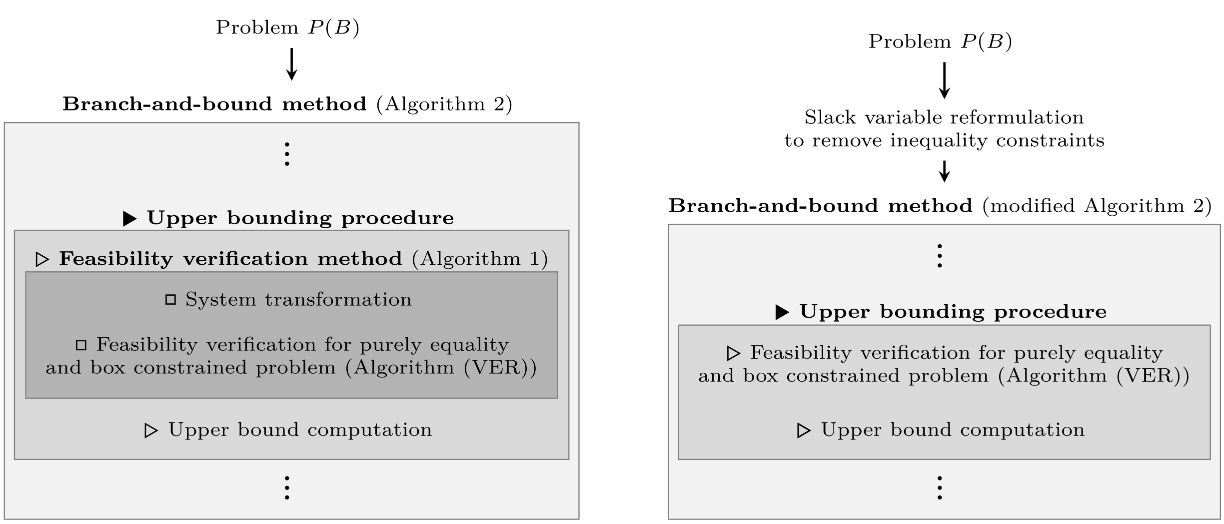
5.1. Test Setting and Applied Methods

As discussed in Section 3.1, it is also possible to use slack variables to transform an inequality constrained problem to one with only equality and box constraints. To analyze the performance of Algorithm 1, for comparison, we also apply this approach in our computational tests. In this case, we directly use Algorithm (VER) for feasibility verification within Algorithm 2 instead of Algorithm 1. This means that Algorithm 1 is replaced by Algorithm (VER) in steps 14 and 16 of Algorithm 2. To further emphasize the differences of both approaches, their main components are illustrated in Figure 2.

For both approaches, we consider different choices for the role of Algorithm (VER) to verify the existence of feasible points in the obtained systems with only equality and box constraints:

- `MIR`: The Miranda-based method from Füllner et al. (2021); see Section 2.
- `MIR_H`: The Miranda-based method from Füllner et al. (2021) but without the computationally costly matrix inversion step. Without this step, the method loses its convergence guarantees, but as a feasibility verification heuristic can be applied much faster.
- `KRAW`: The Krawczyk operator (Krawczyk 1969), an interval Newton operator, is applied to the equation system on the current box X_k and its midpoint $\mathrm{mid}(X_k)$. If its image $K(X_k, \mathrm{mid}(X_k))$ is contained in the box interior $\mathrm{int}(X_k)$, then the existence of a feasible point in X_k is verified. As the Krawczyk operator can only be applied to systems with an equal number of variables and equations, it can be required to transform the system first. To do this, we fix variables based on a heuristic by Kearfott, which uses the null space of the Jacobian of the equality constraints and is described in Domes and Neumaier (2015).
- `NARR`: This approach is similar to the one proposed by Kearfott (1998) and also described in Domes and Neumaier (2015). The problem is solved locally on X_k using a nonlinear solver. Then, using deviations of 10^{-5} around this local solution, a *narrow* box is constructed. As for `KRAW`, the Krawczyk operator is applied to this artificial box to verify the existence of feasible points. We use both an SLSQP solver (Kraft 1988) and Ipopt (Wächter and Biegler 2006) as local solvers in this context and indicate this by `NARR_S` and `NARR_I`.

Figure 2. Proposed Algorithm 2 (Left) and Its Modification Using the Slack Variable Approach (Right)



With respect to these choices, we should make some important remarks. First, even if local solutions are used in NARR, this approach is *not* equal to the common approach to obtain upper bounds by evaluating a local solution in the objective function. As explained in detail in Section 1, a local solution may only be approximately feasible and therefore is not guaranteed to yield valid upper bounds for v^*. In NARR, the idea is to verify the existence of a true feasible point in a neighborhood of a local solution and to then compute an upper bound on this neighborhood.

Second, recall that MIR requires the box constraints of the test problems to be strictly satisfied, see (A1) in Proposition 1. As discussed in Section 3.3, for this reason, test problems with initial box constraints are adapted appropriately. More precisely, the original box constraints $[\underline{x}, \overline{x}]$ are replaced by the slightly larger boxes $[\underline{x}-1, \overline{x}+1]$ and instead considered standard inequality constraints $g_i(x) \leq 0$. Similar to MIR, interval Newton methods such as KRAW are not guaranteed to successfully verify the existence of feasible points that are located on the boundary of some box X. Therefore, we use the same reformulation approach for all choices of Algorithm (VER).

Third, we should emphasize that MIR is the only choice for Algorithm (VER) that provably satisfies the conditions in Assumption 1 and thus may be used to obtain a guaranteed *convergent* branch-and-bound algorithm. The other choices are used for comparison with illustrate the efficacy and efficiency of using such a feasibility verification method.

Finally, in addition to KRAW and NARR, more sophisticated feasibility verification methods have been proposed in the literature (Domes and Neumaier 2015). However, these methods are still heuristic and require a lot of implementation effort. We can also not draw on computational results from Domes and Neumaier (2015) for comparison because there the feasibility verification methods are not incorporated into an SBB method. Instead, the authors use a complex framework to first compute a larger set of approximately feasible points for problems $P(B)$. Then, similar to NARR, they construct narrow boxes around these approximately feasible points to verify the existence of true feasible points. However, this is not done iteratively on boxes X_k as they occur in Algorithm 2.

Importantly, although we do not present a branch-and-bound implementation competitive to state-of-the-art software, such as BARON (Sahinidis 1996), our computational results serve mainly illustrative purposes and as a proof of concept. Thus, for most test problems and applications, state-of-the-art SBB implementations should be significantly superior in performance. The difference is that our focus is on theoretical convergence guarantees, which are not provided by such solvers in general.

Our implementation is kept as simple as possible to focus on the effects of our proposed new techniques. However, for this reason, it only allows for the solution of low-dimensional problems in reasonable time without further tuning. Therefore, we complement our computational analysis by a second batch of tests, containing also some problems of higher dimension. Here, to illustrate the efficacy of our proposed method, we do not consider the complete branch-and-bound tree, but limit the search space to an exhaustive sequence of boxes which provably contains a globally minimal point of the respective test problem.

5.2. Implementation Details

The SBB framework consisting of the Miranda-based method, Algorithms 1 and 2, and the Krawczyk operator is implemented in *Python* 3.7. For numeric operations, both *Numpy* and *Scipy* are used. For interval arithmetic operations, we use the *IntvalPy* package (Androsov 2021). To solve nonlinear problems locally, we use both an SLSQP solver from the *scipy.minimize* package and Ipopt. Our implementation is executed on a Windows machine with a 3.2-GHz Intel Xeon CPU and 64 GB of RAM.

Because SBB algorithms require the problems to be box constrained, the initial starting box B is set to $[-10{,}000, 10{,}000]^n$ for problems where no box constraints are specified in the problem definition. Before the first iteration of Algorithm 2, a standard *optimality-based bounds tightening* (OBBT) is used to decrease the size of the starting box (Gleixner et al. 2017).

The termination criterion of Algorithm 2 is set to $u_k - \ell_k \leq 0.1$. Moreover, the computation is stopped if convergence is not reached after 10,000 iterations or two hours of computation time (excluding the time for OBBT). Setting a time limit is standard in computational tests for SBB methods and allows us to assess the performance in terms of both the convergence behavior of Algorithm 2 and the computational cost of the compared feasibility verification methods. Setting an iteration limit allows us to filter out instances for which no successful feasibility verification is possible after a reasonable amount of branching. This may provide insight on the convergence behavior and the guarantees of the chosen verification approach.

Further implementation details specific to approach MIR can be found in the computational section of Füllner et al. (2021).

The data, code, and results of the computational experiments are available in the *IJOC* GitHub repository (Füllner et al. 2024).

5.3. Test Problems

We perform our computational tests on a set of global optimization problems from the COCONUT benchmark library (Shcherbina et al. 2003). These problems are constrained by a combination of equality, inequality, and box constraints.

Recall that by reformulating the original box constraints to satisfy (A1) in Proposition 1, inequality constraints are artificially introduced to all test problems. Because this already ensures $I \neq \emptyset$, we are able to test our proposed Algorithm 2 even for instances from the COCONUT library that do not contain inequality constraints in the first place.

A complete list of all test problems including their dimension and constraint numbers, is presented in the Online Appendix.

5.4. Tests for the SBB Method

As addressed previously, we first test an SBB method (Algorithm 2) for problems of moderate dimension. We consider a subset of 70 problem instances from the complete problem set, with a maximum dimension of $n = 6$. The results for these tests are summarized in Table 1 and illustrated in Figures 3 and 4. The full results are available in the Online Appendix.

Overall, with our proposed method `MIR`, the existence of feasible points is verified in 46 of 70 test cases during 10,000 iterations or two hours. In most of these cases, computing a valid upper bound causes termination of Algorithm 2 soon after. In contrast, using the slack variable approach for `MIR` yields no successful feasibility verification. Similar observations can be made for `MIR_I`, `KRAW`, and `NARR_I`, where our proposal of exploiting approximate active index sets performs much better than the slack variable approach. In total, we observe only *one* successful experiment using the latter approach. This observation is consistent with our theoretical conjecture in Section 3.1 that our new method of reformulating inequality constrained problems has several computational advantages compared with the slack variable approach.

Furthermore, looking at the total number of successful verifications and terminations of Algorithm 2, we can see that `MIR` performs better than all the alternatives that we tested, with `NARR_I` performing worst (it is also for this reason that we only consider `NARR_S` for the slack variable approach). This shows that in addition to its theoretical convergence guarantees under certain assumptions, `MIR` also performs well computationally compared with similar approaches from the literature. However, in a few cases, `KRAW` and `NARR_S` are successful, whereas `MIR` is not.

One drawback is that `MIR` requires considerable computational effort, especially for high dimensions, as it requires the inversion of Jacobian matrices. Therefore, the iterations take much longer than for `KRAW`. We can see, however, that the related heuristic `MIR_H` is only slightly less successful than `MIR` in feasibility verification while reducing the average computation time to a level similar to `KRAW`. Finally, we observe that also `NARR_S` and `NARR_I` have significant computational overhead compared with `KRAW` and `MIR_H`, as here the time limit is way more often reached before the iteration limit, implying that the feasibility verification approach slows down the overall solution process.

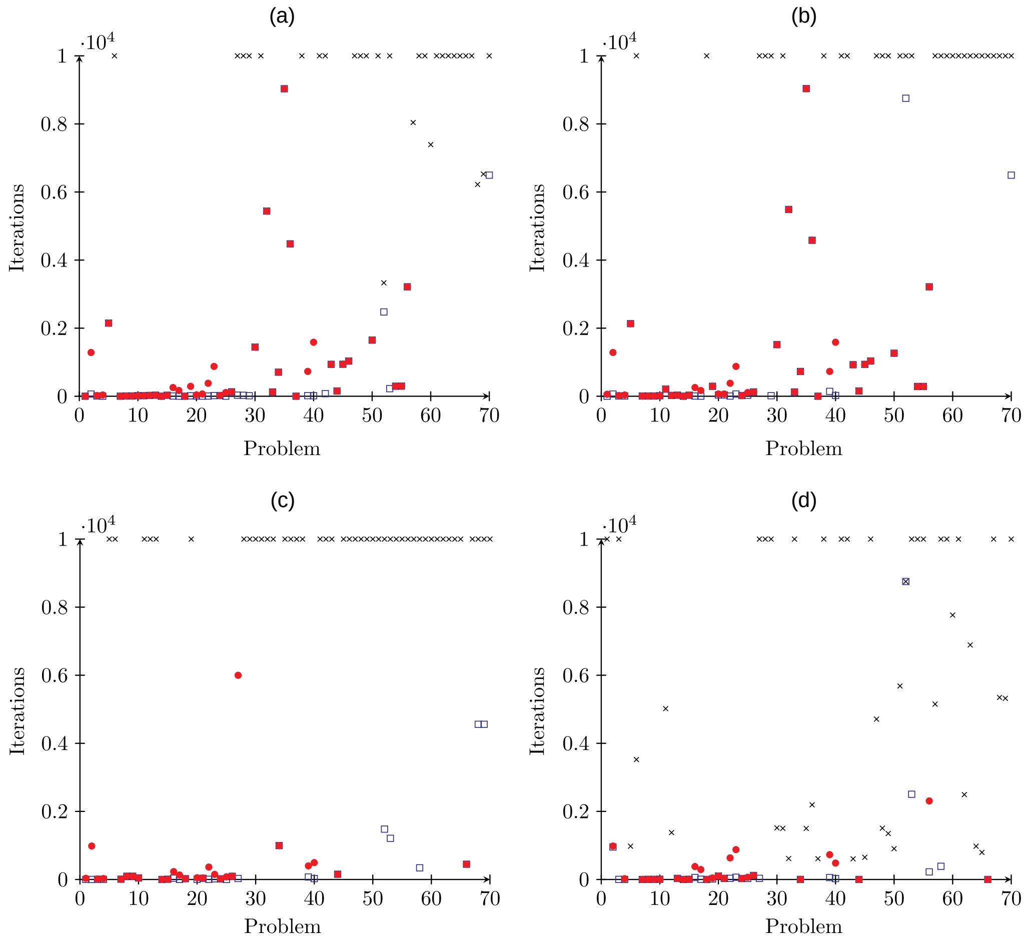
Another observation is, that with increasing dimension, our proposed approach becomes less successful. There are different possible explanations for this behavior. First, we assume that LICQ is satisfied at least in the globally minimal points of the considered test problems. This might not be the case for some instances in our test set, and thus feasibility verification may be prevented.

Second, it is possible that 10,000 iterations are simply not enough to verify the existence of feasible points for these instances due to slow convergence of our simple SBB implementation. This is also why we limited our computational tests to low-dimensional test instances. According to our theoretical results, Algorithm 1 should manage to verify the existence of feasible points, and Algorithm 2 should terminate after an unknown, but finite number of iterations also for the test instances where no success is reported in our previous tests. To identify whether this is

Table 1. Summary of Computational Results for SBB Tests

Criterion	Algorithm 2					Modified Algorithm 2			
	`MIR`	`MIR_H`	`KRAW`	`NARR_S`	`NARR_I`	`MIR`	`MIR_H`	`KRAW`	`NARR_S`
Total number of instances	70	70	70	70	70	70	70	70	70
Successful feasibility verification	46	44	29	29	23	0	0	0	1
Successful termination	42	41	26	26	23	0	0	0	1
Iteration limit reached	23	29	44	18	11	48	70	70	16
Time limit reached	5	0	0	26	35	32	0	0	53

Figure 3. (Color online) Iteration Numbers for Algorithm 2 for All 70 Test Instances



Notes. Squares signify feasibility verification, and dots signify successful termination. × denotes that termination was not successful due to reaching the iteration limit (10,000 iterations) or the time limit (two hours). (a) MIR. (b) MIR_H. (c) KRAW. (d) NARR_S.

the case, we perform a second batch of tests, where we decouple our analysis from the general performance of the SBB implementation. We discuss this in detail in the following section.

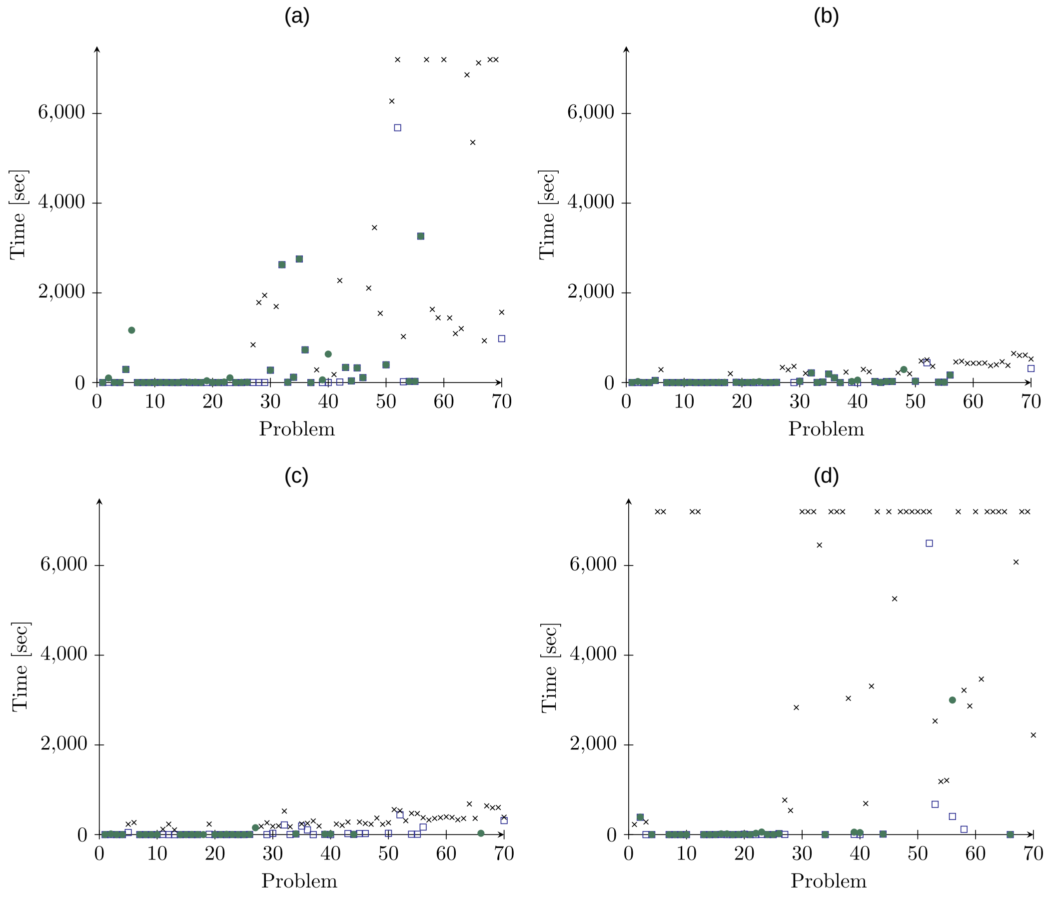
5.5. Tests for Exhaustive Sequences of Boxes

In the second batch of computational tests, we artificially limit the search space of Algorithm 2 to only examine subproblems of the considered test instance that contain a globally minimal point. This is possible, as (approximate) globally minimal points of all test problems are documented in the COCONUT benchmark library. This means that we restrict the search space to exhaustive sequences of boxes $(X_k)_{k\in\mathbb{N}}$ with $x^* \in X_k$ for all $k \in \mathbb{N}$ and x^* a globally minimal point, which we already used for our theoretical results in Sections 2 and 3. This is achieved by modifying Algorithm 2 in such a way that the branching step (step 10) remains the same, but only boxes are examined and chosen for further branching that contain x^*. By doing this, we consider only one single path through the branch-and-bound tree and avoid its exponential growth. Thus, the number of considered boxes and subproblems is reduced significantly. This allows us to examine the convergence behavior of our proposed feasibility verification method (Algorithm 1) more specifically, without further tuning of our branch-and-bound method. Moreover, it allows us to test Algorithm 2 for instances of higher dimension. Thus, in this case, we consider 131 test problems up to dimension $n = 75$.

The obtained results for this second batch of tests are summarized in Table 2 and illustrated in Figures 3 and 4. The full results are available in the Online Appendix.

For 106 of 131 test problems, MIR terminates and the existence of a feasible point is verified, confirming the efficacy of our proposed approach. Because of the speedup per iteration, MIR_H even performs slightly better. All

Figure 4. (Color online) Solution Times for Algorithm 2 for All 70 Test Instances



Notes. Squares signify feasibility verification, and dots signify successful termination. × denotes that termination was not successful due to reaching the iteration limit (10,000 iterations) or the time limit (two hours). (a) MIR. (b) MIR_H. (c) KRAW. (d) NARR_S.

methods from the literature perform significantly worse. In particular, NARR_S and NARR_I exhibit high computational cost, often reaching the time limit, whereas KRAW reaches the iteration limit without successful feasibility verification in most cases. This may be an indicator for no verification guarantees even for sufficiently small boxes in contrast to our proposed approach.

However, the number of instances for which MIR is not successful despite its theoretical properties is considerable. The main reason for this behavior is that our approach to ensure strict feasibility of box constraints, shifting the original box constraints $x \in B$ to the inequality constraints and introducing slightly larger box constraints $x \in \tilde{B}$, may favor LICQ not being satisfied, and in particular the Miranda-based method not being applicable, as discussed in Remark 1 and Section 3.3. This is especially relevant in this case where we only consider one exhaustive sequence of boxes $(X_k)_{k\in\mathbb{N}}$ around x^* and do not take different boxes with less active indices into account.

To confirm this, we perform a last batch of experiments where we analyze all previously unsuccessful problems. The configuration is completely the same as before, with the only difference that the original box constraints $x \in B$

Table 2. Summary of Computational Results for Exhaustive Sequences of Boxes

Criterion	Algorithm 2				
	MIR	MIR_H	KRAW	NARR_S	NARR_I
Total number of instances	131	131	131	131	131
Successful feasibility verification	106	108	29	27	23
Successful termination	106	108	29	27	22
Iteration limit reached	21	22	101	58	44
Time limit reached	4	1	1	46	65

Table 3. Summary of Computational Results for Exhaustive Sequences Without Box Reformulation

Criterion	Algorithm 2				
	MIR	MIR_H	KRAW	NARR_S	NARR_I
Total number of instances	25	25	25	25	25
Successful feasibility verification	14	20	0	0	0
Successful termination	14	20	0	0	0
Iteration limit reached	1	5	23	10	1
Time limit reached	10	0	2	15	24

are maintained. Recall that our reformulation is a sufficient condition to ensure convergence of the Miranda-based method, but not necessary for successful feasibility verification in every case.

The results for these experiments are summarized in Table 3. The full results are again available in the Online Appendix. In line with our hypothesis, MIR is successful for more than half of the previously unsuccessful instances. The remaining instances either still do not satisfy LICQ in x^* in their original formulation, for example, *ladders* (in fact, it violates LICQ in *all* feasible points), or possibly require more than two hours for a successful feasibility verification, as the additional successful terminations for MIR_H indicate.

The computational results can be regarded as a proof of concept for our new feasibility verification and upper bounding procedure. They indicate that with a larger iteration limit, the SBB method (Algorithm 2) should also terminate with the verification of a feasible point after a finite number of iterations. Moreover, this implies that our proposed method may be used to enhance high-performing SBB implementations.

6. Conclusion

In this article, a convergent method for determining valid upper bounds for the globally minimal value v^* of nonconvex minimization problems $P(B)$ with box constraints, equality constraints, and inequality constraints is presented. This method is based on verifying the existence of feasible points in boxes and then computing upper bounds on such boxes. For this purpose, we provide an approach based on approximate active index sets, which transforms a subproblem $P(X)$ with equality, inequality, and box constraints to a related problem with only equality and box constraints. For the obtained problem, feasibility verification can then be achieved by reasonable approaches for equality constrained problems and, in particular, the Miranda-based method laid out in Füllner et al. (2021). We prove that for sufficiently small boxes containing feasible points, feasibility verification for $P(B)$ is assured under certain assumptions. Once the existence of feasible points is verified in some box X, upper bounds can be computed by standard approaches on such box.

Our method is incorporated into an SBB framework and tested on some standard nonconvex global minimization problems. The test results confirm our theoretical results that the proposed method yields convergent upper bounds and shows performance improvements compared with related methods from the literature, both for application in SBB algorithms (42 instead of 26 successful test problems) and on exhaustive sequences of boxes containing a known feasible point (120 instead of 29 successful test problems). However, even using our method, common computational drawbacks of SBB methods, such as slow convergence, cannot be ruled out.

We also test our method against the slack variable approach laid out in Jongen and Stein (2003). Results on a simple test instance indicate that our method performs significantly better than this alternative approach (termination of SBB in 42 instead of 1 of 70 cases).

However, it is worth noting that our method for determining valid and convergent upper bounds does not come without some drawbacks. Most of these drawbacks are related to the Miranda-based feasibility verification method proposed in Füllner et al. (2021). In this regard, our new feasibility verification method has the downside that it heavily relies on this method (or any other feasibility verification method for equality and box constrained problems satisfying Assumption 1).

Additionally, our proposed reformulation to consider the original box constraints as standard inequality constraints and to replace them with less strict box constraints, which is required for our convergence results, may favor cases where more constraints are interpreted as (approximately) active than the dimension of the decision variables. In such a case, however, the Miranda-based method from Füllner et al. (2021) cannot be applied immediately for feasibility verification. This can be considered a potential drawback of this reformulation approach, even though our computational results show that for many instances this reformulation is not required.

Acknowledgments

The authors thank John Alasdair Warwicker for helpful remarks on earlier versions of this manuscript; Ferenc Domes for explanations on the feasibility verification approaches covered in Domes and Neumaier (2015) and for providing Java code for interval Newton methods that was beneficial for our own implementation; and the anonymous referees for precise and substantial remarks that significantly improved the paper.

References

Achterberg T (2009) SCIP: Solving constraint integer programs. *Math. Programming Comput.* 1:1–41.

Adjiman C, Androulakis I, Floudas C (1998a) A global optimization method, αbb, for general twice-differentiable NLPs: II. Implementation and computational results. *Comput. Chemical Engrg.* 22(9):1159–1179.

Adjiman C, Dallwig S, Floudas C, Neumaier A (1998b) A global optimization method, αbb, for general twice-differentiable NLPs: I. Theoretical advances. *Comput. Chemical Engrg.* 22(9):1137–1158.

Androsov A (2021) IntvalPy. Code released on GitHub. https://github.com/AndrosovAS/intvalpy/.

Androulakis I, Maranas C, Floudas C (1995) αbb: A global optimization method for general constrained nonconvex problems. *J. Global Optim.* 7:337–363.

Belotti P, Lee J, Liberti L, Margot F, Wächter A (2009) Branching and bounds tightening techniques for non-convex minlp. *Optim. Methods Software* 24(4–5):597–634.

Berthold T, Gleixner AM (2014) Undercover: A primal MINLP heuristic exploring a largest sub-MIP. *Math. Programming Ser. A* 144:315–346.

Bonami P, Cornuéjols G, Lodi A, Margot F (2009) A feasibility pump for mixed integer nonlinear programs. *Math. Programming Ser. A* 119:331–352.

Domes F, Neumaier A (2015) Rigorous verification of feasibility. *J. Global Optim.* 61:255–278.

Falk JE, Soland RM (1969) An algorithm for separable nonconvex programming problems. *Management Sci.* 15(9):550–569.

Floudas C (2000) Deterministic global optimization: Theory, algorithms and applications. *Nonconvex Optimization and Its Applications* (Kluwer Academic Publishers, Dordrecht, Netherlands), 309–554.

Floudas C, Gounaris C (2009) A review of recent advances in global optimization. *J. Global Optim.* 45:3–38.

Füllner C, Kirst P, Stein O (2021) Convergent upper bounds in global minimization with nonlinear equality constraints. *Math. Programming* 187:617–651.

Füllner C, Kirst P, Otto H, Rebennack S (2024) Repository to "Feasibility verification and upper bound computation using approximate active index sets." https://dx.doi.org/10.1287/ijoc.2023.0162.cd, https://github.com/INFORMSJoC/2023.0162.

Gleixner AM, Berthold T, Müller B, Weltge S (2017) Three enhancements for optimization-based bound tightening. *J. Global Optim.* 67:731–757.

Horst R, Tuy H (1996) *Global Optimization: Deterministic Approaches* (Springer, Berlin).

Jongen HT, Stein O (2003) On the complexity of equalizing inequalities. *J. Global Optim.* 27:367–374.

Jongen HT, Jonker P, Twilt F (1986) Critical sets in parametric optimization. *Math. Programming* 34:333–353.

Kearfott RB (1998) On proving existence of feasible points in equality constrained optimization problems. *Math. Programming* 83:89–100.

Kearfott RB (2014) On rigorous upper bounds to a global optimum. *J. Global Optim.* 59:459–476.

Kirst P, Stein O, Steuermann P (2015) Deterministic upper bounds for spatial branch-and-bound methods in global minimization with nonconvex constraints. *TOP* 23:591–616.

Kraft D (1988) A software package for sequential quadratic programming. Technical report, Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen, Germany.

Krawczyk R (1969) Newton-Algorithmen zur Bestimmung von Nullstellen mit Fehlerschranken. *Computing* 4:187–201.

Krawczyk R, Nickel K (1982) Die zentrische Form in der Intervallarithmetik, ihre quadratische Konvergenz und Inklusionsisotonie. *Computing* 28:117–137.

Lin Y, Schrage L (2009) The global solver in the LINDO API. *Optim. Methods Software* 24:657–668.

Locatelli M, Schoen F (2013) *Global Optimization: Theory, Algorithms and Applications.* MOS-SIAM Series on Optimization (Society for Industrial and Applied Mathematics, Philadelphia).

Miranda C (1940) Un' osservazione su una teorema di Brouwer. *Bollettino dell'Unione Matematica Italiana* 3:5–7.

Misener R, Floudas C (2014) Antigone: Algorithms for continuous/integer global optimization of nonlinear equations. *J. Global Optim.* 59(2–3):503–526.

Moore RE, Kearfott RB, Cloud MJ (2009) *Introduction to Interval Analysis.* Other Titles in Applied Mathematics (Society for Industrial and Applied Mathematics, Philadelphia).

Neumaier A (1990) *Interval Methods for Systems of Equations* (Cambridge University Press, Cambridge, UK).

Puranik Y, Sahinidis NV (2017) Domain reduction techniques for global NLP and MINLP optimization. *Constraints* 22:338–376.

Ryoo H, Sahinidis N (1996) A branch-and-reduce approach to global optimization. *J. Global Optim.* 8:107–138.

Ryoo HS, Sahinidis NV (1995) Global optimization of nonconvex NLPs and MINLPs with applications in process design. *Comput. Chemical Engrg.* 19(5):551–566.

Sahinidis NV (1996) BARON: A general purpose global optimization software package. *J. Global Optim.* 8:201–205.

Shcherbina O, Neumaier A, Sam-Haroud D, Vu XH, Nguyen TV (2003) Benchmarking global optimization and constraint satisfaction codes. *Global Optimization and Constraint Satisfaction* (Springer, Berlin), 211–222.

Smith E, Pantelides C (1997) Global optimisation of non-convex MINLPs. *Comput. Chemical Engrg.* 21(Supplement):S791–S796.

Smith E, Pantelides C (1999) A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex minlps. *Comput. Chemical Engrg.* 23(4–5):457–478.

Tawarmalani M, Sahinidis N (2002a) Convex extensions and envelopes of lower semi-continuous functions. *Math. Programming* 93:247–263.

Tawarmalani M, Sahinidis N (2002b) *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming: Theory, Algorithms, Software, and Applications*, vol. 65. Nonconvex Optimization and Its Applications (Kluwer Academic Publishers, Dordrecht, Netherlands).

Tawarmalani M, Sahinidis N (2004) Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Math. Programming* 99:563–591.

Tawarmalani M, Sahinidis N (2005) A polyhedral branch-and-cut approach to global optimization. *Math. Programming* 103:225–249.

Tuy H (2010) $\mathcal{D}(\mathcal{C})$-optimization and robust global optimization. *J. Global Optim.* 47:485–501.

Tuy H, Rebennack S, Pardalos P (2013) Global optimization. Gass S, Fu M, eds. *Encyclopedia of Operations Research and Management Science* (Springer, Berlin), 650–658.

Vigerske S, Gleixner A (2018) SCIP: Global optimization of mixed-integer nonlinear programs in a branch-and-cut framework. *Optim. Methods Software* 33(3):563–593.

Wächter A, Biegler LT (2006) On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Math. Programming* 106(1):25–57.