

COMPUTING OPTIMALITY CERTIFICATES FOR CONVEX MIXED-INTEGER NONLINEAR PROBLEMS

KATRIN HALBIG, LUKAS HÜMBS, FLORIAN RÖSEL, LARS SCHEWE, DIETER WENINGER

ABSTRACT. Every optimization problem has a corresponding verification problem that checks whether a given optimal solution is in fact optimal. In the literature there are a lot of such ways to verify optimality for a given solution, e.g., the branch-and-bound tree. To simplify this task, Baes et al. introduced optimality certificates for convex mixed-integer nonlinear programs and proved that these are bounded in the number of integer variables. We introduce an algorithm to compute the certificates and conduct computational experiments. Through the experiments we show that the optimality certificates can be surprisingly small.

1. INTRODUCTION

With every optimization problem we have an associated verification problem: Is a given solution actually optimal? In many cases this problem is as difficult as the original optimization problem. One way of dealing with this problem is to produce a *certificate* as part of the solution algorithm. The idea is to compute a mathematical object with short encoding length in addition to the solution that aids verification. The classical example for this is the dual solution of a linear program. But for many problems it is unclear what would be a “good” certificate.

One class of problems, where this is unclear is the class of mixed-integer nonlinear programs (MINLP). In this article, we are specifically dealing with convex MINLPs, that is, MINLPs whose continuous relaxation is a convex optimization problem. There exist a number of solution methods like Branch-and-Bound, Outer Approximation, Extended Cutting Plane, Extended Supporting Hyperplane or Generalized Benders Decomposition (for a state-of-the-art survey of solution methods and software for convex MINLPs, see Kronqvist et al. 2019). From all of these methods one can extract a certificate, for instance for branch-and-bound, one can record all branching decisions and, thus, reconstruct the full tree. But these certificates are large, both in theory and typically also in practice.

In Baes et al. (2016), the authors proposed a new type of certificate for convex MINLP. The authors propose an extension to the standard KKT-conditions to optimization problems with a convex objective function, functional constraints and integrality constraints for some variables. In their case, the certificate, is a set of points, each having a set of Lagrange multipliers. A result of Baes et al. (2016) is that the number of points which are at most necessary to form such an optimality certificate is 2^n , where $n \in \mathbb{N}$ is the number of integer variables. While this is still not small, it is smaller than a full enumeration of all integer points for real-world mixed-integer problems. To verify their certificate, it is necessary to solve one integer linear program and one convex program per certificate point. We note that this certificate has been generalized to infinite-dimensional problems in Jahn and Knossalla (2018). Furthermore the geometric approach of Baes et al. (2016) is also used in Basu et al. (2017) where objective functions are convex, but general feasible sets, are investigated.

Date: April 7, 2022.

2010 Mathematics Subject Classification. 90-08, 90C11, 90C30, 90C26, 90C46.

Key words and phrases. 638 Programming, Integer, Nonlinear 639 Programming, Integer, Theory 628 Programming, Integer, Algorithms .

Left open by Baes et al. (2016) was the question how to compute their certificates efficiently. Beyond that, it was open how their worst-case bound relates empirically to the true certificate size of mixed-integer nonlinear problems as they appear in classical problem classes or applications.

Our contribution is that we a) propose an algorithm to compute the optimality certificates introduced in this manuscript, b) evaluate the results on instances from the MINLPLib and MIPLIB, and c) provide sharper bounds on certificate sizes for specific problem classes and structured problems while retaining the same worst-case bound. To achieve these results, we will work with a slight generalization of the certificate proposed in Baes et al. (2016) that allows us to be more flexible in the algorithms and helps us to prove tighter bounds for specific problems.

The structure of the paper is the following. The subsequent Section 2 formally introduces the problem. The following Section 3 generalizes the optimality certificate introduced in Baes et al. (2016) while an algorithm to compute such generalized optimality certificates is introduced in Section 4. In Section 5 we prove tighter upper bounds on the certificate size for some classes of optimization problems. In Section 6 we briefly discuss similarities and differences to other optimality certificates. Computational experiments are presented in Section 7. In the last Section 8 we give an outlook on the topic.

2. PROBLEM STATEMENT

In this section, we introduce our notation, define the core concepts, and state our assumptions that hold throughout this article.

We consider mixed-integer convex problems of the form

$$\min_{z,y} f(z, y) \tag{1a}$$

$$\text{s.t. } g(z, y) \leq 0, \tag{1b}$$

$$z \in \mathbb{Z}^n, y \in \mathbb{R}^d, \tag{1c}$$

where $f: \mathbb{R}^{n+d} \rightarrow \mathbb{R}$ is a convex function and $g: \mathbb{R}^{n+d} \rightarrow \mathbb{R}^J$ for $J \in \mathbb{N}$ is a component-wise convex function, i.e., each $g_j: \mathbb{R}^{n+d} \rightarrow \mathbb{R}$ with $j = 1, \dots, J$ is a convex function. We denote $x := (z, y)$.

In general for optimization problems there are various ways to certify optimality for a certain optimal solution. A classical example for such an optimality certificate is a solved Karush-Kuhn-Tucker system (Kuhn and Tucker 1951) for convex optimization problems.

However, Problem (1) contains integer variables, which makes it much more difficult to specify an optimality certificate. Nevertheless most algorithms that solve such optimization problems to optimality determine an optimality certificate implicitly. For example, a branch-and-bound algorithm generates a branch-and-bound tree with leaves either being pruned, infeasible or being solved to optimality. Knowing all the leaves is sufficient to proof that a feasible solution is optimal by solving one convex optimization problem per leaf.

Unfortunately, the number of leaves in a branch-and-bound tree is not bounded by the number of integer variables only. We later give examples for this in Section 6.

As mentioned in the introduction, Baes et al. (2016) introduced a certificate that is bounded in the number of integer variables. Their certificate consists of pairs of mixed-integer points and halfspaces in the space of integer variables. This enables to proof the optimality for one given solution by solving one continuous optimization problem per point and one linear integer problem.

In line with Baes et al. (2016), we make the following assumptions throughout the paper.

Assumption 2.1 (Baes et al. 2016). *The set of continuous minima of Problem (1) exists and is bounded.*

This assumption is needed as we use the boundedness of the sub-level sets later on and these sets are unbounded in case the set of continuous minima is unbounded. Thus without the assumption, we would not be able to form the optimality certificates.

Definition 2.2 (Baes et al. 2016). *The constraints $g(z, y) \leq 0$ fulfill the mixed-integer Slater condition, if for every point $(z, y) \in \mathbb{Z}^n \times \mathbb{R}^d$ with $g(z, y) \leq 0$ there exists a $\tilde{y} \in \mathbb{R}^d$ such that $g(z, \tilde{y}) \leq 0$ and $g_j(z, \tilde{y}) < 0$ for every function g_j with $j = 1, \dots, J$ that is not affine linear.*

Assumption 2.3 (Baes et al. 2016). *We assume Problem (1) fulfills the mixed-integer Slater condition.*

We need to make this assumption because in order to form the optimality certificates one needs to solve the KKT-systems corresponding to the convex problems that arise when the integer variables are fixed to some value and are considered to be continuous. Thus the mixed-integer Slater condition ensures that we can solve the KKT-systems corresponding to these problems. This implies that fulfilling any other regularity condition for the problems with fixed integer variables would also be fine.

One key idea of the certificate is to find a compact encoding of a set of integer points that cannot be extended to optimal solutions. In our case, these sets will be complements of a polyhedron that does not contain integer feasible points in its interior. Hence, we will use the following definition as a short-hand:

Definition 2.4 (Mixed-integer-free polyhedron). *A polyhedron $\mathcal{P} \subseteq \mathbb{R}^{n+d}$ is called mixed-integer-free if its interior does not contain any mixed-integer point $(z, y) \in \mathbb{Z}^n \times \mathbb{R}^d$.*

As pointed out later in this article, generalized optimality certificates consist of a collection of half spaces, which are induced by suspension points and normal vectors, which are defined as follows.

Definition 2.5 (Induced half space, suspension point, normal vector). *The induced half-space of a point $\tilde{x} \in \mathbb{R}^{n+d}$ and a vector $\tilde{h} \in \mathbb{R}^{n+d}$ is given by*

$$\mathcal{H}(\tilde{x}, \tilde{h}) := \{x \in \mathbb{R}^{n+d} \mid \tilde{h}^\top (x - \tilde{x}) < 0\}. \quad (2)$$

We call \tilde{x} a suspension point of $\mathcal{H}(\tilde{x}, \tilde{h})$ and $\tilde{h} \in \mathbb{R}^{n+d}$ the corresponding normal vector.

Note that $\mathcal{H}(\tilde{x}, \tilde{h})$ represents an open half space for reasons we give later. By $\tilde{\mathcal{H}}(\tilde{x}, \tilde{h})$ we denote the associated closed half-space. Using Definition 2.5 and the following two lemmas we give a geometric intuition into the optimality certificates. Lemma 2.6 is used later on to construct half-spaces that are defined in the space of integer variables, but are valid for all feasible mixed-integer points.

Lemma 2.6. *Let a set $\mathcal{G} \subseteq \mathbb{R}^{n+d}$ be given and define a respective superset as $\text{elim } \mathcal{G} = \{(z, y) \in \mathbb{R}^{n+d} \mid \exists (\tilde{z}, \tilde{y}) \in \mathcal{G}, \tilde{z} = z\}$. Then $\text{elim } \mathcal{G} \subseteq \tilde{\mathcal{H}}(\tilde{x}, h)$ if*

- (i) $\mathcal{G} \subseteq \tilde{\mathcal{H}}(\tilde{x}, h)$
- (ii) $h_e = 0$ for all $e \in \{n+1, \dots, n+d\}$.

The halfspace $\mathcal{H}(\tilde{x}, h)$ supports $\text{elim } \mathcal{G}$ if $\tilde{x} \in \mathcal{G}$.

Proof. Let h and $\tilde{x} = (\tilde{z}, \tilde{y})$ satisfy the two properties of the lemma. Now let $x = (z, y) \in \text{elim } \mathcal{G}$. By definition of $\text{elim } \mathcal{G}$, there exists a point $\tilde{x} = (\tilde{z}, \tilde{y}) \in \mathcal{G}$ with $\tilde{z} = z$. From the first property it follows that $\tilde{x} \in \tilde{\mathcal{H}}(\tilde{x}, h)$. Hence, $h^\top (\tilde{x} - \tilde{x}) \leq 0$. Split $h^\top = [h_z^\top \ h_y^\top]$. Then the second property is equivalent to $h_y^\top = 0$. Hence, $h_z^\top (z - \tilde{z}) \leq 0$ holds. This inequality holds also for x , hence $x \in \tilde{\mathcal{H}}(\tilde{x}, h)$ holds. The second claim follows directly from $\mathcal{G} \subseteq \text{elim } \mathcal{G}$. \square

Lemma 2.7 shows that conic combinations of valid normal vectors h are also valid, which will be used to generate certificate half-spaces later on.

Lemma 2.7. *Let $\mathcal{G} \subseteq \mathbb{R}^{n+d}$, $\tilde{x} \in \mathcal{G}$ and $h_1, \dots, h_I \in \mathbb{R}^n$. Let $\mathcal{G} \subseteq \bar{\mathcal{H}}(\tilde{x}, h_i)$ for all $i \in \{1, \dots, I\}$. Then for all $h \in \text{cone}(h_1, \dots, h_I)$ the inclusion $\mathcal{G} \subseteq \bar{\mathcal{H}}(\tilde{x}, h)$ holds.*

Proof. Let $x \in \mathcal{G}$. As $h \in \text{cone}(h_1, \dots, h_I)$, there exists a $\lambda \in \mathbb{R}^I$ with $\lambda \geq 0$ such that $h = \sum_{1 \leq i \leq I} \lambda_i h_i$. By assumption $h_i^\top (x - \tilde{x}) \leq 0$ for all $i \in \{1, \dots, I\}$. Then

$$h^\top (x - \tilde{x}) = \sum_{i=1}^I \lambda_i h_i^\top (x - \tilde{x}) \leq \sum_{i=1}^I \lambda_i 0 \leq 0.$$

□

A key idea necessary to build the following optimality certificates is the set of continuous-relaxed feasible points, which have a better objective value than the point in consideration.

Definition 2.8 (Sublevel-set). *Let \tilde{x} be a feasible point of Problem (1), then the sublevel-set at \tilde{x} is given as*

$$\mathcal{S}(\tilde{x}) := \{x \in \mathbb{R}^{n+d} \mid f(x) < f(\tilde{x}), x \in \mathcal{G}\},$$

where

$$\mathcal{G} := \{x \in \mathbb{R}^{n+d} \mid g_j(x) \leq 0 \text{ for all } j = 1, \dots, J\}$$

is the continuous-relaxed feasible set of Problem (1).

As a side note, sublevel-sets of convex optimization problems are convex.

Now we can state the intuition behind the optimality certificates. Given a convex mixed-integer nonlinear problem an optimality certificate for a point \tilde{x} is given through a mixed-integer free polyhedron. To be more precise, a polyhedron that contains all points that have a better objective value than point \tilde{x} , while disregarding the integrality constraint, i.e., the sublevel-set. This implies that – in case this polyhedron does not contain any mixed-integer points – \tilde{x} must be optimal to the mixed-integer problem. However, the question remains how to obtain such a polyhedron. These polyhedra \mathcal{P} in question are constructed such that $\mathcal{P} = \text{elim } \mathcal{P}$ as then they do not contain a mixed-integer point. Furthermore, they are constructed by the intersection of halfspaces \mathcal{H} . Therefore the vectors h defining the polyhedron \mathcal{P} contain zero entries in the continuous dimensions by Lemma 2.6. Then Lemma 2.7 shows that the vectors h can be constructed out of a conic combination of vectors that lead to valid inequalities. By convexity any vector h in the subgradient at a point x leads to a valid constraint for the optimization problem at hand. Thus the certificates consist of suspension points x_1, \dots, x_I with corresponding vectors h_1, \dots, h_I where h_i is a conic combination of vectors of the subgradient of x_i .

Thus we are given an optimal point \tilde{x} together with suspension points x_1, \dots, x_I and corresponding vectors h_1, \dots, h_I . These suspension points x_1, \dots, x_I and corresponding vectors h_1, \dots, h_I then define halfspaces $\mathcal{H} := \mathcal{H}(x_i, h_i)$ that each contain the feasible set in case x_i is infeasible or – in case x_i is feasible – contain the sublevel-set of x_i . This implies that the intersection of the halfspaces \mathcal{H} forms the aforementioned polyhedron \mathcal{P} .

The following Theorem 2.9, directly taken from Baes et al. (2016), specifies the aforementioned optimality certificate. Thus this theorem offers a way to construct a corresponding halfspace to each mixed-integer point through a suspension point and corresponding normal vector which is a conic combination of subgradients. Furthermore, it gives an upper bound on the size that only depends on the number of integer variables (but is exponential in that number).

Theorem 2.9 (Baes et al. 2016). *Let the set of continuous minima of Problem (1) exist and be finite. And let the constraints $g(z, y) \leq 0$ fulfill the mixed-integer slater condition. Then the point $\hat{x} \in \mathbb{Z}^n \times \mathbb{R}^d$ is optimal for the mixed-integer constrained problem (1) if and only if $g(\hat{x}) \leq 0$ and there exist $I \leq 2^n$ points $x_1 = \hat{x}, x_2, \dots, x_I \in \mathbb{Z}^n \times \mathbb{R}^d$ and I vectors*

$u_1, \dots, u_I \in \mathbb{R}_+^{J+1}$ with corresponding $h_{i,J+1} \in \partial f(x_i)$ and $h_{ij} \in \partial g_j(x_i)$ for $j = 1, \dots, J$ and $i = 1, \dots, I$ such that the following five conditions hold:

- (i) If $g(x_i) \leq 0$ then $f(x_i) \geq f(\widehat{x})$, $u_{i,J+1} > 0$ and $u_{ij}g_j(x_i) = 0$ for $j = 1, \dots, J$.
- (ii) If $g(x_i) \not\leq 0$ then $u_{i,J+1} = 0$ and $u_{ij} = 0$ for all $j \notin C_i := \{1 \leq j \leq J \mid g_j(x_i) = \max_{1 \leq l \leq J} g_l(x_i)\}$.
- (iii) $1 \leq |\text{supp}(u_i)| \leq d + 1$ for $i = 1, \dots, I$.
- (iv) $\{x \in \mathbb{R}^{n+d} \mid \sum_{j=1}^{J+1} u_{ij}h_{ij}^\top(x - x_i) < 0 \text{ for all } i = 1, \dots, I\} \cap (\mathbb{Z}^n \times \mathbb{R}^d) = \emptyset$.
- (v) $\sum_{j=1}^{J+1} u_{ij}h_{ij} \in \mathbb{R}^n \times \{0\}^d$ for $i = 1, \dots, I$.

The theorem guarantees the existence of a system of 2^n feasible solutions or *suspension points*.

We proceed with explaining the underlying ideas of Theorem 2.9 in detail in the following Section 3, to state the definition of generalized optimality certificates and give a few results how to determine these in the following.

3. GENERALIZED OPTIMALITY CERTIFICATE

The certificate presented in the last section has two major drawbacks: The rules of constructing the half spaces corresponding to suspension points are overly restrictive; additionally the theorem is a pure existence result and does not offer a way to construct a set with cardinality 2^n of suspension points with corresponding half spaces. To overcome this, one main goal of this paper is to make it possible to compute the optimality certificates. Hence, we introduce a generalized version of the optimality certificates introduced by Baes et al. (2016), which we refer to as *generalized optimality certificate* in the following.

The generalized version allows for more flexibility in choosing the corresponding half spaces. To be more clear, in contrast to Baes et al. (2016), we allow that more than $d + 1$ vectors can be used to generate a normal vector, that defines the half space in question. We would like to note that this may allow for certificates with fewer suspension points and corresponding normal vectors. This is also demonstrated in some of the examples later.

Because we introduce the generalized optimality certificates in order to do our computational experiments, we establish a procedure to derive generalized optimality certificates. This is done using dual multipliers and follows directly after the definition of our generalized optimality certificates. With the help of this procedure, we will be able to state an algorithm to constructively determine generalized optimality certificates in the next section.

Now, if one finds a feasible point \widehat{x} of Problem (1) for which the corresponding sublevel-set or a set where this is contained in does not contain any feasible point, this can be seen as a certificate of optimality. In the optimality certificate stated in the previous section, these supersets have been open polyhedra defined by the intersection of the half spaces corresponding to the suspension points. We follow this idea and show how to construct a mixed-integer-free polyhedron that completely contains the sublevel-set $\mathcal{S}(\widehat{x})$ and does not contain any mixed-integer point. Then the only difference is that more freedom in choosing the half spaces exists. The emerging polyhedron serves as an optimality certificate for point \widehat{x} .

As a short prerequisite, we need to define the feasible- and infeasible-fiber-subproblems. As a side note, these will also be important for the calculation of the normal vectors $\{h_1, \dots, h_I\}$ later on.

Definition 3.1 (Feasible- and infeasible-fiber-subproblem). *For a point $\tilde{z} \in \mathbb{Z}^n$ the set $\{\tilde{z}\} \times \mathbb{R}^d$ is called the fiber of \tilde{z} . Based on this, we define the feasible-fiber-subproblem for*

$\tilde{z} \in \mathbb{Z}^n$ as

$$\min_{z,y} f(z, y) \quad (3a)$$

$$\text{s.t. } z = \tilde{z} \quad (3b)$$

$$g(z, y) \leq 0 \quad (3c)$$

$$z \in \mathbb{R}^n, y \in \mathbb{R}^d \quad (3d)$$

and the infeasible-fiber-subproblem for $\tilde{z} \in \mathbb{Z}^n$ as

$$\min_{z,y,q} q \quad (4a)$$

$$\text{s.t. } z = \tilde{z} \quad (4b)$$

$$g(z, y) \leq q \quad (4c)$$

$$q \in \mathbb{R}_{\geq 0}, z \in \mathbb{R}^n, y \in \mathbb{R}^d. \quad (4d)$$

If Problem (3) is feasible, we call the corresponding fiber a feasible fiber, otherwise an infeasible fiber.

The general ideas behind the optimality certificates in Baes et al. (2016) stay the same, thus the generalized optimality certificates also consist of tuples of suspension points and normal vectors, which together induce halfspaces. The intersection of these halfspaces defines a mixed-integer-free polyhedron. Therefore the structure of Baes et al. (2016) stays intact.

Furthermore, we also distinguish between *feasible* and *infeasible suspension points*. The idea is that for a feasible suspension point the corresponding halfspace is chosen such that it contains the sublevel-set of the suspension point. For an infeasible suspension point the corresponding halfspace is chosen such that it contains the feasible set. This implies that every point, which is feasible and has a better objective value than all suspension points is inside the intersection of these halfspaces. If this polyhedron is mixed-integer-free, then all feasible suspension points with the best objective values are optimal solutions of the initial optimization problem. This is also the structure in Baes et al. (2016). Nevertheless, we are convinced that due to less restrictive limitations in the election of corresponding half spaces, our certificates can be computed more easily.

For improved practicability of the optimality certificates we want to introduce the concept of *support sets*. For a large proportion of mixed-integer optimization problems emerging in practical applications, there are parts of \mathbb{Z}^n that are not relevant to the optimization problem, e.g., because they are known to be infeasible or not optimal. Take for example an arbitrary binary program, where the feasible region of the “integer” variables is just the 0-1-cube, which is way smaller than \mathbb{Z}^n . For this reason, we make regions which can in some sense “easily” excluded from the optimization process irrelevant for the corresponding optimality certificates. This new structure enables also to make some interesting theoretical observations, for instance more precise bounds for specific problem structures (see Section 5 and 6).

Definition 3.2 (Support set). *Given Problem (1) where \mathcal{X}^* is the set of optimal solutions to Problem (1) we say a set $\mathcal{Z} \subseteq \mathbb{Z}^n$ with $\mathcal{X}^* \cap (\mathcal{Z} \times \mathbb{R}^d) \neq \emptyset$ is a valid support set.*

In case the use of a support set is reasonable for the optimization problem under consideration, i.e., $\mathcal{Z} \neq \mathbb{Z}^n$, we replace the integrality constraints $z \in \mathbb{Z}^n$ by $z \in \mathcal{Z}$. For better readability we use \mathcal{Z} and $\mathcal{Z} \times \mathbb{R}^d$ interchangeably.

Furthermore, we like to note that the support set – in theory – can be any integer set. In the following, however, we always use the support set $\{z \in \mathbb{Z}^n \mid g_j(z, 0) \leq 0 \text{ for all } j \in \tilde{J}\}$, whereby $\tilde{J} := \{j \in J \mid g_j(z, y) = g_j(z, 0^d) \text{ for all } z \in \mathbb{Z}^n, y \in \mathbb{R}^d\}$, i.e., we consider the set defined by all constraints that do not depend on continuous variables.

3.1. Defining the generalized optimality certificate. Now that we have all definitions that are needed to introduce an optimality certificate we can formally define our notion of the aforementioned generalized optimality certificate. Here we restrict ourselves to the following case, which is very common in the MINLP literature.

Assumption 3.3. *The set of feasible fibers in the support set \mathcal{Z} is finite, i.e.,*

$$|\mathcal{Z} \cap \{z \in \mathbb{Z}^n \mid \exists y \in \mathbb{R}^d : g(z, y) \leq 0\}| < \infty.$$

Now that we introduced all necessary definitions and ideas we can define the generalized optimality certificate, which certifies optimality for a given feasible solution of the underlying optimization problem. We prove this shortly after the definition and an example which may give the reader a better intuition of optimality certificates.

Definition 3.4 (Generalized optimality certificate). *We call a set of suspension points x_1, \dots, x_I together with normal vectors h_1, \dots, h_I a generalized optimality certificate for Problem (1) if*

- (i) $h_i \in \mathbb{R}^n \times \{0\}^d$ for all $i = 1, \dots, I$,
- (ii) $\mathcal{H}(x_i, h_i) \supseteq \mathcal{S}(x_i)$ for all feasible x_i ,
- (iii) $\mathcal{H}(x_i, h_i) \supseteq \mathcal{G}$ for all infeasible x_i ,
- (iv)

$$\bigcap_{i=1, \dots, I} \mathcal{H}(x_i, h_i) \cap \mathcal{Z} \times \mathbb{R}^d = \emptyset, \quad (5)$$

where $\mathcal{S}(x_i)$ is the sublevel-set at x_i , \mathcal{G} is the continuous-relaxed feasible set and $\mathcal{H}(x_i, h_i)$ is the induced open half-space of x_i and h_i .

As previously stated, the main difference between the certificate introduced in the last section and the generalized certificate is a distinct level of flexibility in choosing the normal vector h . This is illustrated by the following example, which tries to find the most upper-right point in a circle on a plane.

Example 3.5. *Let a problem be given as*

$$\min_z \sum_{p=1}^2 -z_p \quad (6a)$$

$$\text{s.t. } \sum_{p=1}^2 z_p^2 \leq 1 \quad (6b)$$

$$z \in \mathbb{Z}^n. \quad (6c)$$

Then the suspension point $(2, 0)$ leads to the normal vector $\lambda(1, 0)$ with some multiplier $\lambda > 0$. However, the normal vector $(1, -1)$ is also valid, but can not be generated by dual variables. These vectors are depicted in Figure 1 and visualize that the normal vectors could be chosen more freely as they do not depend on the subgradient of the constraints that are not fulfilled.

For the sake of completeness, the following theorem states, that whenever a generalized optimality certificate is known, the best among the feasible suspension points is an optimal solution of the corresponding optimization problem. Hence, the certificate we introduced really does what it should do, i.e., Definition 3.4 indeed *certifies* an optimal solution at the effort of checking if the system of certificate points has properties (i) to (iv) in Definition 3.4. We want to explicitly exclude making statements for optimization problems without a feasible solution.

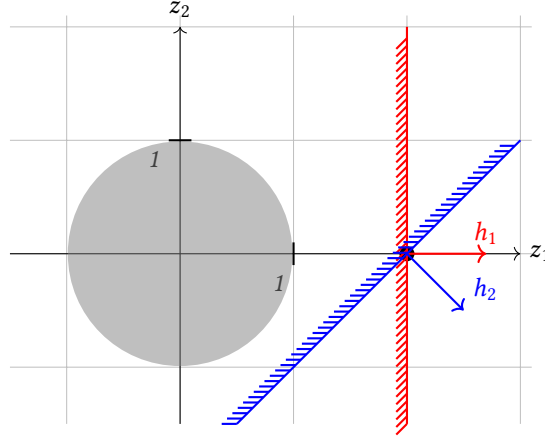


FIGURE 1. Feasible set of Problem (6a) and possible normal vectors for suspension point $(2, 0)$: h_1 the normal vector generated by Baes et al. (2016) and h_2 an example for a normal vector of a generalized certificate. Thus h_2 can not be derived using subgradients – but it is a valid normal vector to a generalized optimality certificate.

Theorem 3.6. *If Problem (1) has an optimal solution, then a feasible suspension point of a generalized optimality certificate with best objective value is an optimal solution.*

Proof. Let \hat{x} be an optimal solution to the optimization problem (1), then this point is either a suspension point or not. In case it is a suspension point, the theorem holds directly. In the latter case, this point has to be cut off by some certificate constraint, because the optimal solution is a mixed-integer point by feasibility. Because the induced halfspaces in the certificates contain the sublevel-sets of their respective suspension points it follows that there is a suspension point that has a better or equal optimal value than the point \hat{x} . Then this suspension point is the optimal point addressed in the theorem. \square

Before we present the results which are relevant for the computations of generalized optimality certificates, we present some very basic properties of the certificates. These are not directly relevant for the further results of this section, but they may give the reader a good intuition of the objects under consideration. This is why we present it on this point in the paper. All normal vectors corresponding to suspension points must have zero entries in all continuous dimensions. Hence the halfspaces which are defined by the normal vectors and the fiber corresponding to the suspension point are disjoint. Additionally they have to contain the sublevel-set of the suspension point. Therefore, it is impossible to find appropriate normal vectors for feasible suspension points which are not optimal on their own fiber. This result is recorded in the following lemma.

Lemma 3.7. *Let $((z_i, y_i), h_i)$ be a certificate point. If the point has Properties (i) and (ii) in Definition 3.4, then y_i is an optimal solution of Problem (3)(z_i), i.e., the fiber problem corresponding to z_i .*

Proof. Assume that y_i is not an optimal solution of Problem (3)(z_i). Then there exists a point \tilde{y}_i feasible for Problem (3)(z_i) and fulfilling $f(z_i, \tilde{y}_i) < f(z_i, y_i)$. Hence, this point is in the sublevel-set, i.e., $(z_i, \tilde{y}_i) \in \mathcal{S}(z_i, y_i)$. Additionally, from Property (i) follows that $h_i^\top((z_i, \tilde{y}_i) - (z_i, y_i)) = (\tilde{h}_i, 0^d)^\top(0^n, (\tilde{y}_i - y_i)) = 0$. Hence, the point is not in the induced half space $\mathcal{H}((z_i, y_i), h_i)$. This implies that the point does not have Property (ii), which is the desired result. \square

In the following we keep the notation $(\check{h}, 0^d) = h$ for the ease of notation.

As a side note, the inequality “ $<$ ” in Definition (2) cannot be replaced by “ \leq ”, as then it would be possible to form a certificate without a feasible point. This is demonstrated by the following example.

Example 3.8 (Infeasible certificate). *Given a problem*

$$\begin{aligned} & \min_{z_1, z_2} 0 \\ & \text{s.t. } z_1 = 0 \\ & \quad z_2 = 0 \\ & \quad z \in \mathbb{Z}^2. \end{aligned}$$

We use the suspension point $(1, 0)$ with the normal vectors $(0, 1)$ and $(0, -1)$. This would be a certificate with the definition using “ \leq ” as the arising halfspaces \mathcal{H} are $\{z \in \mathbb{Z}^2 \mid z_2 \leq 0\}$ and $\{z \in \mathbb{Z}^2 \mid z_2 \geq 0\}$ which contain the only feasible point $(0, 0)$ and would form a valid certificate. However this certificate would not contain the optimal solution.

It is straight-forward that the optimality certificate introduced by Baes et al. (2016), which is stated through Theorem 2.9, is a special case of a generalized optimality certificate introduced in Definition 3.4. The support set in this case is given less restrictive as in our setting as $\mathcal{Z} = \mathbb{Z}^n \times \mathbb{R}^d$, which is not finite as required for Assumption 3.3 holding true. Nevertheless, we can formulate the following corollary.

Corollary 3.9. *Every optimality certificate stated through Theorem 2.9 in combination with $\mathcal{Z} = \mathbb{Z}^n \times \mathbb{R}^d$ is a generalized optimality certificate.*

According to this, generalized optimality certificates which are bounded in size exist for all optimization problems for which the above assumptions hold. This is recorded in the next corollary.

Corollary 3.10 (Generalized Certificate). *There is a certificate according to Definition 3.4 with at most 2^n many inequalities where n is the number of integer variables.*

In the remainder of the section, we formulate certain results for generalized optimality certificates, which are mainly required to introduce an algorithm to compute generalized optimality certificates, but are also very interesting standing alone.

3.2. Constructing a generalized optimality certificate. The next two propositions state that a normal vector for a given suspension point can generally be linearly combined of subgradients of the objective function and the active or violated constraints at the suspension points. The first proposition corresponds to feasible suspension points, the second to infeasible ones. Nevertheless, the propositions do not (yet) offer a constructive way to derive the linear coefficients, but they are a good start point to elaborate a constructive way to obtain valid normal vectors. The first Proposition 3.13 motivates the procedure how to compute corresponding normal vectors to feasible suspension points because it shows that the normal vectors can be obtained solving a system of equations. With an abuse of notation, for any functions f we write from now on, shortly ∇f for an arbitrary element in the subgradient ∂f of the potentially non-differentiable function f . The following two lemmas allow us to formalize the intuition behind the optimality certificates.

Lemma 3.11. *Let \tilde{x} with $g_j(\tilde{x}) \geq 0$ and $h \in \partial g_j(\tilde{x})$. Then $\mathcal{G} \subseteq \mathcal{H}(\tilde{x}, h)$.*

Proof. Let $x \in \mathcal{G}$. Hence, $g_j(x) \leq 0$ holds. From the subgradient inequality, we can deduce that

$$0 \geq g_j(x) \geq g_j(\tilde{x}) + h^\top(x - \tilde{x}) \geq h^\top(x - \tilde{x}).$$

□

Lemma 3.12. *Let $\tilde{x} \in \mathcal{G}$ and $h \in \partial f(\tilde{x})$. Then $\mathcal{S}(\tilde{x}) \subseteq \mathcal{H}(\tilde{x}, h)$.*

Proof. Let $x \in \mathcal{S}(\tilde{x})$. Hence, $f(\tilde{x}) \geq f(x)$ holds. From the subgradient inequality, we can deduce that

$$f(x) \geq f(\tilde{x}) + h^\top(x - \tilde{x}).$$

Hence,

$$0 \geq f(x) - f(\tilde{x}) \geq h^\top(x - \tilde{x}).$$

□

Now we can state how the certificate half spaces can be generated using subgradients in the feasible and infeasible case.

Proposition 3.13. *Let \tilde{x} be a feasible point of Problem (1), and let $A(\tilde{x}) \subset \{1, \dots, J\}$ be the set of active constraints in \tilde{x} . Let $\tilde{h} \in \mathbb{R}^n \times \{0\}^d$ with $\tilde{h} \neq 0^{n+d}$ be a vector such that there are vectors $h_j \in \bigcup_{j \in A(\tilde{x})} \partial g_j(\tilde{x}) \cup \partial f(\tilde{x})$ such that \tilde{h} is a conic combination of the h_j . Then \tilde{h} and \tilde{x} together fulfill Properties (i) and (ii) of Definition 3.4 and \tilde{x} is a feasible suspension point.*

Proof. Lemmas 3.11 and 3.12 show that the subgradients define valid constraints while Lemma 2.7 shows that the conic combinations of the subgradients lead to constraints that are valid as well. Then Lemma 2.6 shows that the arising constraint supports the set $\mathcal{S}(\tilde{x})$. And Property (i) is fulfilled by prerequisite. □

Analogously, normal vectors corresponding to infeasible suspension points can be linearly combined of subgradients of violated constraints.

Proposition 3.14. *Let (\tilde{x}) be an infeasible point of Problem (1), and let $\mathcal{V}(\tilde{x}) := \{j \in \{1, \dots, J\} \mid g_j(\tilde{x}) > 0\}$ be the set of violated constraints. Let $\tilde{h} \in \mathbb{R}^n \times \{0\}^d$ with $\tilde{h} \neq 0^{n+d}$ be a vector such that there are vectors $h_j \in \bigcup_{j \in \mathcal{V}(\tilde{x})} \partial g_j(\tilde{x})$ such that \tilde{h} is a conic combination of the h_j . Then \tilde{h} and \tilde{x} together fulfill Properties (i) and (iii) of Definition 3.4 and (\tilde{x}) is an infeasible suspension point.*

Proof. Lemma 3.11 shows that the subgradients define valid constraints while Lemma 2.7 shows that the conic combinations of the subgradients lead to constraints that are valid as well. And Property (i) is fulfilled by prerequisite. □

We pursue the approach to combine normal vectors of subgradients of constraints and the objective function to formulate the next lemmas, that allow to derive normal vectors to corresponding suspension points via observing certain dual variables.

More specifically, we show that to determine valid normal vectors it is only necessary to observe the Lagrange multipliers of solved fiber subproblems. This comes in handy because then one can simply use the dual variables given by standard solvers to obtain the corresponding normal vectors. Thus one does not have to solve systems of equations for that and can instead use an out-of-the-box solution.

3.2.1. *Computing dual multipliers and normal vectors.* This allows us to prove two lemmas stating that a valid certificate plane can be constructed of the Lagrange Multipliers of Constraints (3b) for feasible certificate points or of Constraints (4b) for infeasible certificate points.

Lemma 3.15. *The dual multipliers of Constraints (3b) of Problem (3) for a feasible fiber point $\tilde{z} \in \mathcal{Z}$ define a valid certificate vector $h \in \mathbb{R}^{n+d}$ for the point, i.e., Property (ii) holds for (\tilde{z}, h) .*

Proof. Let \tilde{z} be a feasible fiber point. By Assumption 2.2, our fiber problem (3) for \tilde{z} satisfies Slater's condition. Let $\tilde{x} = (\tilde{z}, \tilde{y})$ be an optimal solution. Hence, by Rockafellar (1970,

Theorem 28.3) there exist $\lambda \geq 0$, $\nabla f(\tilde{x})$, $\nabla g_j(\tilde{x})$ – where we denote specific elements of the respective subgradients with ∇ and unit vectors as e_p – and $\eta \in \mathbb{R}^n$ such that

$$\begin{aligned} 0_{n+d} &= \nabla f(\tilde{x}) + \sum_{j=1}^J \lambda_j \nabla g_j(\tilde{x}) + \sum_{p=1}^n \eta_p \begin{pmatrix} e_p \\ 0_d \end{pmatrix} \\ 0 &\geq g(\tilde{x}) \perp \lambda \geq 0. \end{aligned}$$

Hence,

$$h := \begin{bmatrix} -\eta \\ 0 \end{bmatrix} = \nabla f(\tilde{x}) + \sum_{j=1}^J \lambda_j \nabla g_j(\tilde{x}).$$

Thus it is possible to form a vector from a conic combination such that its entries in the continuous dimensions are zero. Additionally from Lemma 2.7 it follows that h is a valid certificate vector as it is a conic combination of subgradients. \square

Then we can state the following observation.

Observation 3.16. *The vector h corresponding to suspension point \tilde{x} is zero if and only if \tilde{x} is an optimal solution of the continuous relaxation to the underlying mixed-integer problem (1).*

Conversely, we show how the dual variables corresponding to infeasible fibers are used to generate normal vectors.

Lemma 3.17. *The dual multipliers of Constraints (4b) of Problem (4) for an infeasible fiber point $\tilde{z} \in \mathcal{Z}$ define a valid certificate plane $h \in \mathbb{R}^{n+d}$ for the point, i.e., Property (iii) holds for (\tilde{z}, h) .*

Proof. Let e_0 be the unit vector corresponding to variable q . Let \tilde{z} be an infeasible fiber point. By assumption, our fiber problem for \tilde{z} satisfies Slater's condition. Let $\tilde{x} = (z, y)$, \tilde{q} be an optimal solution to the fiber subproblem (4). Hence, there exist $\lambda \geq 0$, $\eta \in \mathbb{R}^n$ and $\nabla f(\tilde{x})$, $\nabla g_j(\tilde{x})$ – where we again denote specific elements of the respective subgradients with ∇ – such that

$$0_{n+d} = e_0 + \sum_{p=1}^n \eta_p e_p + \sum_{j \in A(\tilde{x})} \lambda_j (\nabla g_j(\tilde{x}) - e_0) = \left(1 - \sum_{j \in A(\tilde{x})} \lambda_j\right) e_0 + \sum_{p=1}^n \eta_p e_p + \sum_{j \in A(\tilde{x})} \lambda_j \nabla g_j(\tilde{x}).$$

With $A(\tilde{x})$ denoting the set of active constraints. Now, as the values of $g_j(\tilde{x})$ do not depend on q , we can split this relation into two conditions:

$$\begin{aligned} 1_{n+d} &= \sum_{j \in A(\tilde{x})} \lambda_j, \\ 0_{n+d} &= \sum_{p=1}^n \eta_p e_p + \sum_{j \in A(\tilde{x})} \lambda_j \nabla g_j(\tilde{x}). \end{aligned}$$

From the second relation, we obtain as in the previous lemma that

$$h := \begin{bmatrix} -\eta \\ 0 \end{bmatrix} = \sum_{j \in A(\tilde{x})} \lambda_j \nabla g_j(\tilde{x}).$$

Then it follows from Lemma 2.7 that h is a valid certificate vector. \square

After establishing this lemma we now showed how to obtain the normal vectors corresponding to feasible or infeasible suspension points. This means that we now can introduce an algorithm that computes optimality certificates for given problems.

4. DETERMINING OPTIMALITY CERTIFICATES

In order to empirically investigate the results obtained so far, we first introduce Algorithm 1 that returns a certificate for a given Problem (1). This certificate does not necessarily have an appropriately small size. Hence we introduce Algorithm 2 which takes a possibly too large certificate and reduces its size to at most 2^n points. Although it is pretty straight-forward to check if a certificate is valid, we formally introduce Algorithm 3 for this purpose.

4.1. Computing a certificate. From Assumption 3.3, we know that explicitly stated with the help of an appropriate convex representable support set \mathcal{Z} , there is a finite number of potentially feasible fibers. An appropriate subset of these forms together with the constraints defining the support set a certificate. The strategy of Algorithm 1 we propose is to go through the fiber set in a dedicated order, calculate the corresponding certificate plane, and to check if the polyhedron defined by all certificate planes so far is empty. In a nutshell, Algorithm 1 repeatedly solves an optimization problem. This contains all certificate planes determined so far as constraints. It is defined on the support set \mathcal{Z} , i.e., the space of integer variables of Problem (1), and maximizes the distance to all certificate planes found so far and inserts for a feasible solution of the optimization problem a new certificate plane. Algorithm 1 stops if the maximal distance to the certificate planes is equal to 0.

Formally, the problem of the $(I + 1)$ -th iteration of Algorithm 1 is

$$\max_{\varepsilon, z} \varepsilon \tag{7a}$$

$$\text{s.t. } \check{h}_i^\top (z - z_i) + \varepsilon \leq 0 \quad \text{for all } i = 1, \dots, I \tag{7b}$$

$$0 \leq \varepsilon \leq \tilde{\varepsilon}, z \in \mathcal{Z}, \tag{7c}$$

whereby (z_i, \check{h}_i) are for $i = 1, \dots, I$ the integer parts of the certificate points determined in previous iterations, and $\tilde{\varepsilon}$ is an arbitrary positive real.

Once found an optimal solution of Problem (7), Algorithm 1 determines a certificate plane by solving Problem (3), or, in case this is infeasible, by solving Problem (4) in connection, determines KKT-Multipliers for the fixing Constraints (3b)/(4b) and derives a new certificate plane $\check{h}_{I+1} \neq 0$ corresponding to a feasible solution of Problem (7). The vector $\check{h}_{I+1} \neq 0$ can not become zero, as then the KKT-system of the entire problem would have been solved and therefore the continuous relaxation would yield a solution to the mixed-integer problem. Then no certificate is needed.

In the following we proof that Algorithm 1 calculates a generalized certificate.

Corollary 4.1. *Algorithm 1 returns a generalized certificate in a finite number of steps.*

Proof. We proof first that the algorithm terminates. Problem (7)'s feasible region is a subset of the support set \mathcal{Z} . This is a finite set. The algorithm picks a point z_I in this feasible region and inserts a constraint (7b) into Problem (7). This constraint implies that the point z_I from now on has objective value 0 since the inserted constraint reduces to $\varepsilon \leq 0$ for $z = z_I$. For this reason, the algorithm leaves the while-loop after at most $|\mathcal{Z}|$ iterations. Thus the algorithm terminates.

It remains to show that the points which are returned form a valid certificate. To check if a collection of suspension points and normal vectors, $(x_i, h_i)_{i=1, \dots, I}$ is a generalized certificate, we have to make sure that the properties from Definition 3.4 hold for the point system, to be precise, that Property (i) holds for all normal vectors, Property (ii) holds for all feasible certificate points, Property (iii) holds for all infeasible certificate points and Property (iv) holds for the system of certificate points.

Property (i) is correct by construction of the $h_i = (\check{h}_i, 0^d)$.

Algorithm 1 Determine a valid certificate $(x_i, h_i)_{i=1, \dots, I}$ for $I \in \mathbb{N}$.

Input A Problem (1)

Output An optimality certificate for Problem (1)

- 1: $I \leftarrow 0$.
 - 2: **while** Problem (7)'s optimal value > 0 **do**
 - 3: $I \leftarrow I + 1$
 - 4: Set z_I to a solution of Problem (7) with strictly positive value.
 - 5: Solve Problem (3)(z_I) and if it is infeasible, solve Problem (4)(z_I).
 - 6: Set y_I to an optimal solution of either Problem (3)(z_I) or, if it is infeasible, of Problem (4)(z_I).
 - 7: Set x_I to (z_I, y_I) .
 - 8: Set \check{h}_I to the dual multipliers of (3b) or (4b) and set h_I to $(\check{h}_I, 0^d)$.
 - 9: Set $h_I \leftarrow \frac{h_I}{\|h_I\|_2}$.
 - 10: Add a new constraint (7b) for (z_I, \check{h}_I) to Problem (7).
 - 11: **return** $(x_i, h_i)_{i=1, \dots, I}$.
-

Property (iv): Assume there exists a $\tilde{x} = (\tilde{z}, \tilde{y}) \in \mathcal{P}$. \tilde{z} can be extended to a feasible solution for (7) with a strictly positive value of $\tilde{\varepsilon} = -\min_{i=1, \dots, I} \check{h}_i^\top (z_i - \tilde{z})$. But the while-loop of the algorithm is not left when the optimal value of (7) is not 0. A contradiction.

Property (ii): Let x_i be a feasible point of Problem (1). Since the corresponding certificate plane h_i is set to the duals of Constraints (3b), we can apply Lemma 3.15, yielding the required result.

Property (iii): Let x_i be an infeasible point of Problem (1). Since the corresponding certificate plane h_i is set to the duals of Constraints (4b), we can apply Lemma 3.17, yielding the required result. \square

Beyond that, the certificate calculated by Algorithm 1 has interesting properties, as stated in the next lemma.

Lemma 4.2. *The polyhedron \mathcal{P} defined by the certificate calculated by Algorithm 1 has the same dimension as $\text{conv}(\mathcal{Z})$ or is the empty set.*

Proof. We proof this by induction over I .

Induction start: We start with the polyhedron $\text{conv}(\mathcal{Z})$ with the right dimension. Induction step: Let \mathcal{P}_{I-1} denote the polyhedron defined after $I - 1$ steps of the while loop of the algorithm. Assume it has the right dimension. The point z_I added in the while loop of the algorithm has to be in the relative interior of \mathcal{P}_{I-1} : The euclidean distance of a point z_i to a the hyperplane induced by a constraint (7b) for $\tilde{i}, \tilde{i} \in \{1, \dots, I - 1\}$, i.e., $\check{h}_{\tilde{i}}^\top (z - z_{\tilde{i}}) = 0$ is exactly $\frac{\check{h}_{\tilde{i}}^\top}{\|\check{h}_{\tilde{i}}\|_2} (z_{\tilde{i}} - z_i)$; since the polyhedron is restricted by finitely many of these hyperplanes, an open ball with radius being equal to the minimum over all these distances,

$$\epsilon := \min_{\tilde{i} \in \{1, \dots, I-1\}} \frac{\check{h}_{\tilde{i}}^\top}{\|\check{h}_{\tilde{i}}\|_2} (z_{\tilde{i}} - z_i)$$

is contained in the polyhedron, and hence the point is in the relative interior of the polyhedron. Whatever hyperplane containing z_i we insert, exactly half of the ball is fully contained in the new polyhedron, if the hyperplane is not defined by normal vector 0. This half ball contains another ball with radius $\frac{\epsilon}{2}$, i.e., the polyhedron defined by the intersection of \mathcal{P}_{I-1} and the new certificate plane remains full dimensional. If the normal vector is 0, the resulting polyhedron is the empty set. \square

4.2. **Reducing the size of a certificate.** To reduce a certificate calculated by Algorithm 1, we propose Algorithm 2. Given a valid generalized certificate of size possibly larger than

Algorithm 2 Reduce certificate size to $I \leq 2^n$.

Input A certificate to Problem (1)

Output A reduced certificate to Problem (1)

- 1: **for** $i = 1, \dots, I$ **do**
 - 2: Replace Constraint (7b) for (z_i, \check{h}_i) in (7) by $\check{h}_i^\top (z - z_i) \geq 0$.
 - 3: **if** Optimal value of (7) > 0 **then**
 - 4: Replace $\check{h}_i^\top (z - z_i) \geq 0$ by Constraint (7b) for (z_i, \check{h}_i) in (7).
 - 5: **else**
 - 6: Remove $\check{h}_i^\top (z - z_i) \geq 0$ from (7).
 - 7: Remove (x_i, h_i) .
 - 8: **return** (z_i, h_i) for all $i = 1, \dots, I$
-

2^n , it returns a certificate of size at most 2^n . To show this, we need the following auxiliary lemma.

Lemma 4.3. *Let $(z_i, \check{h}_i)_{i=1, \dots, I}$ be a collection of vectors in $\mathbb{Z}^n \times \mathbb{R}^n$. Let the optimal value of (7) be 0. Then, for all $i' = 1, \dots, I$, either the optimal value of the problems*

$$\max \varepsilon \tag{8a}$$

$$\text{s.t. } \check{h}_i^\top (z - z_i) + \varepsilon \leq 0 \quad \text{for all } i = 1, \dots, I : i \neq i' \tag{8b}$$

$$\varepsilon \geq 0, z \in \mathcal{Z} \tag{8c}$$

and

$$\max \varepsilon \tag{9a}$$

$$\text{s.t. } h_i^\top (z - z_i) + \varepsilon \leq 0 \quad \text{for all } i = 1, \dots, I : i \neq i' \tag{9b}$$

$$\check{h}_{i'}^\top (z - z_{i'}) \geq 0 \tag{9c}$$

$$\varepsilon \geq 0, z \in \mathcal{Z} \tag{9d}$$

coincide or the optimal value of (8) is 0 and problem (9) is infeasible.

Proof. The optimal value of (7), as well as the optimal value of (9), are lower bounds for the optimal value of (8) since the latter is a relaxation of the prior ones.

Case (9) is infeasible: Assume we have a point $(\widehat{z}, \widehat{\varepsilon})$ feasible for (8) with $\widehat{\varepsilon} > 0$. Since (7) has optimal value 0, the point is not in the open halfspace $\check{h}_{i'}^\top (z - z_{i'}) < 0$. This point is feasible for (9). A contradiction.

Case (9) is feasible. If the optimal value of (8) is 0, we are done, all optimal values are equal. If not, let $(\widehat{z}, \widehat{\varepsilon})$ denote an optimal solution of (8). Hence $\widehat{\varepsilon}$ is the corresponding optimal value. It is also feasible for (9), since otherwise \widehat{z} can be completed to a feasible solution of (7), with a strictly positive value of $\widetilde{\varepsilon} := \min(\widehat{\varepsilon}, \check{h}_{i'}^\top (z_{i'} - \widehat{z})) > 0$. Hence, the optimal values of (8) and (9) coincide. \square

Lemma 4.3 justifies Line 2 of Algorithm 2. The following corollary can be shown now.

Corollary 4.4. *Algorithm 2 reduces the size of a given certificate $(x_i, h_i)_{i=1, \dots, I}$ to at most 2^n , while n is the number of integer variables.*

Proof. Let the input of the algorithm be a valid optimality certificate $(x_i, h_i)_{i=1, \dots, I}$ with $I \in \mathbb{N}$ possibly greater than 2^n .

First we show that the for-loop of the algorithm preserves the certificate property. In the loop no certificate points are added, so every certificate point is valid. We apply

Lemma 4.3 to show that the interior of the certificate polyhedron remains integer-free, since the algorithm is constructed to only remove certificate points for which this holds.

It remains to show that the algorithm reduces the size of the certificate to at most 2^n . We refer to a corollary from Doignon (1973) (slightly adapted in formulation).

Corollary 4.5. *If a finite collection of at least 2^n convex sets of a lattice \mathcal{L}^n has an empty intersection, then there exists a subcollection of 2^n members with an empty intersection.*

Assume now that the returned certificate's size is greater than 2^n . Then, the application of Proposition 4.5 guarantees the existence of a subset of 2^n certificate points with empty intersection. For this reason, there is a certificate point which has not been removed by the algorithm and for which the removal of the corresponding constraint would not have raised the objective value of (7) over 0. A contradiction. \square

4.3. Verifying a certificate. For the sake of completeness we want to propose a simple procedure to verify if a certificate is valid. The procedure has two steps. The first is to evaluate if the polyhedron defined by the certificate points and hyperplanes, $(x_i, h_i)_{i=1, \dots, I}$ is empty. It is to check if the optimal value of the linear (M)IP (7) is equal to 0. The second step is to check whether (z_i, h_i) defines a valid certificate plane for all $i = 1, \dots, I$. This can be done applying Algorithm 3. Algorithm 3 checks for each certificate point if the

Algorithm 3 Check certificate planes

Input A certificate to Problem (1)

Output A Boolean indicating whether the certificate is valid

```

1: retval  $\leftarrow$  True
2: Remove integrality constraints of (1).
3: for  $i = 1, \dots, I$  do
4:   Add constraint  $h_i^\top (x - x_i) \geq 0$  to Problem (1) and solve it.
5:   if Objective value is smaller than  $f(x_i)$  for feasible points or problem is not
     infeasible for infeasible points then
6:     retval  $\leftarrow$  False
7: return retval

```

sublevel-set corresponding to the certificate point is contained in the feasible halfspace of the certificate plane.

Observation 4.6. *Algorithm 3 applied to a set of vectors (x_i, h_i) in $(\mathbb{Z}^n \times \mathbb{R}^d) \times (\mathbb{R}^n \times \{0\}^d)$ returns true if and only if every point in the set with feasible x_i has Property (ii) and every point in the set with infeasible x_i has Property (iii).*

5. SIZE OF OPTIMALITY CERTIFICATES

The subject of this section are statements on the size of the generalized certificate (Definition 3.4). First, we will utilize mathematical results regarding lattices to prove a statement about convex MINLPs with certain support sets, reducing the upper bound on necessary certificate bounds considerably. Then we will derive upper bounds for certificate sizes of linear MIPs. Finally, we comment on certificate sizes if the underlying problem has a block structure.

5.1. Using lattices to reduce the size of a certificate. A relevant subclass of convex MINLPs has the property, that there exist some linear equality constraints which have nonzero coefficients only for integer variables.

A popular example for that is the linear regression problem with subset selection constraints (see, e.g., Konno and Yamamoto 2009), i.e., a linear regression problem for

which only $k \in \mathbb{N}$ of the regression coefficients are to be nonzero. This problem can be modeled as a convex MINLP with the described properties.

Since this problem class is relevant, we derive some tighter upper bounds on certificate sizes for this kind of problems in the following. As background, we need some basic facts about lattices, which are taken from Bertsimas and Weismantel (2005).

Definition 5.1 (Lattice). *Let $B = [b_1, \dots, b_m] \in \mathbb{Q}^{n \times m}$ be a matrix whose column vectors b_1, \dots, b_m are linearly independent. The set*

$$\mathcal{L} = \mathcal{L}(B) = \{z \in \mathbb{R}^n \mid z = Bv \text{ with } v \in \mathbb{Z}^m\}$$

is called the lattice generated by B . The matrix B is called a basis of $\mathcal{L}(B)$.

In the following, we will need that we can always decompose the basis into a standard form.

Lemma 5.2. *Let $A \in \mathbb{Z}^{\tilde{n} \times n}$ be a matrix of full row rank and let $b \in \mathbb{Z}^{\tilde{n}}$ and $\mathcal{F} := \{z \in \mathbb{Z}^n \mid Az = b\}$. Then it holds that*

- (i) *There exists a unimodular matrix $U \in \mathbb{Z}^{n \times n}$ such that $AU = [L, 0]$ with $L \in \mathbb{Z}^{\tilde{n} \times \tilde{n}}$ lower triangular and invertible.*
- (ii) *$\mathcal{F} \neq \emptyset$ if and only if $L^{-1}b \in \mathbb{Z}^{\tilde{n}}$.*
- (iii) *If $\mathcal{F} \neq \emptyset$, then every element z of \mathcal{F} is of the form*

$$z = U_1 L^{-1} b + U_2 \bar{z}, \quad \bar{z} \in \mathbb{Z}^{n-\tilde{n}},$$

and U_1, U_2 are appropriate submatrices of U , i.e., $U = [U_1, U_2]$.

- (iv) *$\mathcal{L} = \{z \in \mathbb{Z}^n \mid Az = 0\}$ is a lattice.*

We can apply Lemma 5.2 directly in the special case that the equality constraints in our MINLP only contain integer variables. This yields the following result.

Lemma 5.3. *Let $\tilde{\mathcal{Z}}$ be an intersection convex subset of \mathbb{Z}^n . Consider the set $\mathcal{Z} := \{z \in \tilde{\mathcal{Z}} : a_j^\top z = b_j, j = 1, \dots, \tilde{n}\}$, for some $a_j \in \mathbb{Z}^n$ linear independent and $b_j \in \mathbb{Z}$. Then, either \mathcal{Z} is empty or there exist $L \in \mathbb{Z}^{\tilde{n} \times \tilde{n}}$, lower triangular, $U_1 \in \mathbb{R}^{n \times \tilde{n}}, U_2 \in \mathbb{R}^{n \times n-\tilde{n}}, [U_1, U_2]$ unimodular, such that the convex MINLP*

$$\begin{aligned} & \min f(z, y) \\ & \text{s.t. } g(z, y) \leq 0 \\ & z \in \mathcal{Z}, y \in \mathbb{R}^d \end{aligned}$$

can be reformulated with $n - \tilde{n}$ integer variables, as

$$\begin{aligned} & \min f(U_1 L^{-1} b + U_2 z, y) \\ & \text{s.t. } g(U_1 L^{-1} b + U_2 z, y) \leq 0 \\ & z \in \{\tilde{z} \in \mathbb{Z}^{n-\tilde{n}} \mid U_1 L^{-1} b + U_2 \tilde{z} \in \tilde{\mathcal{Z}}\}, y \in \mathbb{R}^d. \end{aligned}$$

This optimization problem is a convex MINLP.

This allows us to tighten the general bound for convex MINLP with equality constraints in the support set.

Theorem 5.4. *Let the support set \mathcal{Z} for the domain of the integer variables z of a convex MINLP (1) have a superset given by a system of \tilde{n} linear, linear independent, integral equations defined on the integer variables. Then, the maximum size of a generalized optimality certificate reduces to $2^{n-\tilde{n}}$.*

Proof. This follows directly from Lemma 5.3.

Let $z_1, \dots, z_I \in \mathbb{Z}^{n-\tilde{n}}$ be suspension points of a certificate for the transformed problem and let $\tilde{h}_1, \dots, \tilde{h}_I \in \mathbb{R}^{n-\tilde{n}}$ the corresponding normal vectors. U_2 has a full column rank, so we

can explicitly define the pseudo inverse matrix $U_2^\dagger := (U_2^\top U_2)^{-1} U_2^\top$. A certificate for the original problem is given by $\tilde{z}_i := U_1 L^{-1} b + U_2 z_i$ and $\tilde{h}_i := (U_2^\dagger)^\top \check{h}_i$.

Consider some point \tilde{z} in the convex hull of \mathcal{Z} contained in the strict (feasible) sublevel-set of a certificate point \tilde{z}_i of the original optimization problem. Since $U_1 L^{-1} b + U_2 z$ is an bijection between the affine hull of \mathcal{Z} and $\mathbb{R}^{n-\bar{n}}$, there exists a unique point \tilde{z}' in $\mathbb{R}^{n-\bar{n}}$ such that $\tilde{z} = U_1 L^{-1} b + U_2 \tilde{z}'$ and we know that $\tilde{z}_i = U_1 L^{-1} b + U_2 z_i$. It holds that \tilde{z}'/z_i in the reformulated optimization problem has the same objective value than \tilde{z}/\tilde{z}_i in the original one, \tilde{z}' has to be in the half-space defined by (z_i, \check{h}_i) . Hence,

$$0 > \check{h}_i^\top (\tilde{z}' - z_i) = \check{h}_i^\top U_2^\dagger U_2 (\tilde{z}' - z_i) = \quad (10a)$$

$$((U_2^\dagger)^\top \check{h}_i)^\top ((U_1 L^{-1} b + U_2 \tilde{z}') - (U_1 L^{-1} b + U_2 z_i)) = \tilde{h}_i^\top (\tilde{z} - \tilde{z}_i). \quad (10b)$$

Furthermore, the polytope defined by $(\tilde{z}_i, \tilde{h}_i)_{i=1, \dots, I}$ is integer free: assume that there is an integer point in the certificate polytope, then we can apply the same transformations as in (10) to deduce the existence of an integer point in the certificate polyhedron of the reformulated problem. A contradiction.

Further it holds that the best certificate point in the reformulated problem is also a certificate point for the original problem and both are the integer optimal solution for their corresponding problems.

Hence, $(\tilde{z}_i, \tilde{h}_i)_{i=1, \dots, I}$ defines a certificate of the original problem, with size at most $2^{n-\bar{n}}$ since this is a valid bound for the certificate of the reformulated problem. \square

In summary, we have shown that when we add linearly independent linear equality constraints, each additional constraint consisting of only integer variables reduces the maximum certificate size by half.

Next, we derive upper bounds for linear MIPs which rather depend on the number of constraints involved than the number of integer variables, which might, in various special cases, be tighter than the bounds which can be obtained by the number of integer variables.

5.2. Upper bounds for certificate sizes of linear problems. In this section upper bounds on the size of a certificate are examined. We start with an upper bound for the simplest subclass of problems, the pure integer linear problems. Using these insights, we can give an upper bound on the size of a generalized certificate (see Definition 3.4) for mixed-integer linear problems.

We consider mixed-integer linear problems of the form

$$\min_{z, y} c^\top (z, y) \quad (11a)$$

$$\text{s.t. } A(z, y) \leq b, \quad (11b)$$

$$z \in \mathbb{Z}^n, y \in \mathbb{R}^d, \quad (11c)$$

whereby $c \in \mathbb{R}^{n+d}$, $A \in \mathbb{R}^{J \times (n+d)}$, and $b \in \mathbb{R}^J$. Bounds on variables are explicitly added as constraints.

Lemma 5.5. *For a pure integer linear problem with J constraints and optimal solution \hat{x} exists an optimality certificate according to Theorem 2.9 consisting of at most $J + 1$ points.*

Proof. Suppose that we have an optimality certificate according to Theorem 2.9. As the number of continuous variables d equals zero, for each multiplier vector $u_i \in \mathbb{R}_+^{J+1}$, $i = 1, \dots, I$, the condition (iii) $|\text{supp}(u_i)| = 1$ holds. That is, each vector u_i has exactly one positive entry. Since $\dim(u_i) = J + 1$, at most $J + 1$ of them are linear independent.

Suppose now that the certificate has $I > J + 1$ points. Then, there exist two suspension points x_1, x_2 with corresponding linear dependent u_1, u_2 . Let $u_1 = \lambda u_2$ with $\lambda > 0$, and let

entry $l \in \{1, \dots, J+1\}$ be positive. Moreover, for the subgradients holds $h_{1l} = h_{2l}$ since all constraints are linear.

The corresponding hyperplanes are

$$\sum_{j=1}^{J+1} u_{1j} h_{1j}^\top (x - x_1) < 0 \quad \text{and} \quad \sum_{j=1}^{J+1} u_{2j} h_{2j}^\top (x - x_2) < 0, \quad (12)$$

and with the above observations follows:

$u_{1l} h_{1l}^\top (x - x_1) < 0 \iff \lambda u_{2l} h_{2l}^\top (x - x_1) < 0 \iff \lambda u_{2l} h_{2l}^\top x < \lambda u_{2l} h_{2l}^\top x_1 \iff u_{2l} h_{2l}^\top x < u_{2l} h_{2l}^\top x_1$ and $u_{2l} h_{2l}^\top (x - x_2) < 0 \iff u_{2l} h_{2l}^\top x < u_{2l} h_{2l}^\top x_2$, and therefore (12) holds if and only if $u_{2l} h_{2l}^\top x < \min(u_{2l} h_{2l}^\top x_1, u_{2l} h_{2l}^\top x_2)$. Thus one of the two points can be removed. This procedure can be repeated until only $J+1$ points are left which form the certificate. \square

This implies that the number of necessary certificate points does not directly depend on the computational complexity of a problem. As in this case, e.g., the NP-hard Knapsack Problem (see, e.g., Martello et al. 1990) will need maximal $2n+2$ certificate points. $2n+1$ points corresponding to constraints ($2n$ bound constraints, 1 linear constraint) and 1 optimal point corresponding to the objective function.

We further illustrate that the size of an optimality certificate can be smaller than 2^n in the case of linear integer problems via the following example.

Example 5.6. Consider the problem

$$\begin{aligned} \min_z \quad & z_1 + z_2 \\ \text{s.t.} \quad & z_1 + z_2 \geq -\frac{1}{2} \\ & z_1, z_2 \in \mathbb{Z}. \end{aligned}$$

We only need suspension points $(0, 0)$ and $(0, -1)$ with the respective normal vectors $(1, 1)$ and $(-1, -1)$. This gives an optimality certificate of size 2^1 instead of 2^2 .

In the following, we extend Lemma 5.5 for mixed-integer linear problems. For this, we need two auxiliary lemmas that follow directly from standard arguments. The first lemma relates extreme rays of a cone to the extreme rays of its embedding in a higher dimensional space. In the second lemma, we give an upper bound on the number of extreme rays of such a cone.

Lemma 5.7. Given a matrix $M \in \mathbb{R}^{\alpha \times \beta}$ and a vector $m \in \mathbb{R}^\alpha$. Let u^* be an extreme ray of cone $K_1 = \{u \in \mathbb{R}^\beta \mid Mu = 0, u \geq 0\}$. Then, the extended vector $\begin{bmatrix} u^* \\ 0 \end{bmatrix}$ is an extreme ray of the cone

$$K_2 = \left\{ \begin{bmatrix} u \\ \tilde{u} \end{bmatrix} \in \mathbb{R}^{\beta+1} \mid [M, m] \begin{bmatrix} u \\ \tilde{u} \end{bmatrix} = 0, \begin{bmatrix} u \\ \tilde{u} \end{bmatrix} \geq 0 \right\}.$$

Proof. Since $u^* \in K_1$ follows $[M, m] \begin{bmatrix} u^* \\ 0 \end{bmatrix} = 0$, $\begin{bmatrix} u^* \\ 0 \end{bmatrix} \geq 0$ and, therefore, $\begin{bmatrix} u^* \\ 0 \end{bmatrix}$ is element of K_2 . Suppose that $\begin{bmatrix} u^* \\ 0 \end{bmatrix}$ is not an extreme ray of K_2 . Then there exist $v, w \in K_2$, so that $\begin{bmatrix} u^* \\ 0 \end{bmatrix} = \mu v + (1-\mu)w$, with $0 < \mu < 1$, $v \neq \lambda w$, $\lambda > 0$. So follows $0 = \mu v_{\beta+1} + (1-\mu)w_{\beta+1}$ which holds if and only if $v_{\beta+1} = w_{\beta+1} = 0$. Then there exist $v^*, w^* \in \mathbb{R}^\beta$ such that $v = \begin{bmatrix} v^* \\ 0 \end{bmatrix}$ and $w = \begin{bmatrix} w^* \\ 0 \end{bmatrix}$ and thus $v^*, w^* \in K_1$. Moreover, $u^* = \mu v^* + (1-\mu)w^*$ and thus u^* cannot be an extreme ray of K_1 , which is a contradiction. \square

Lemma 5.8. Given a matrix $M \in \mathbb{R}^{\alpha \times \beta}$ with $\alpha < \beta$. The number p of extreme rays of the cone $\{u \in \mathbb{R}^\beta \mid Mu = 0, u \geq 0\}$ is bounded from above by

$$p \leq \binom{\beta}{\beta - 1 - \text{rank}(M)}. \quad (13)$$

In particular, p can be bounded by

$$p \leq \begin{cases} \binom{\beta}{\alpha+1} & \alpha \leq \lfloor \frac{\beta-1}{2} \rfloor, \\ \binom{\beta}{\lfloor \frac{\beta}{2} \rfloor} & \text{otherwise.} \end{cases}$$

Proof. The first inequality follows from the fact that every extreme ray u^* of a polyhedral cone in \mathbb{R}^β can be defined by $\beta - 1$ linearly independent constraints binding at u^* . Due to the properties of the binomial coefficient the second inequality holds. \square

The following statement extends Lemma 5.5 for the case of mixed-integer linear problems.

Theorem 5.9. *Let a mixed-integer linear problem (11) and an optimality certificate according to Theorem 2.9 be given. h_j is the unique gradient of constraint j and h_j^d denotes the entries corresponding to the d continuous variables. Let p be the number of extreme rays of the cone C*

$$C = \{u \in \mathbb{R}^{J+1} \mid \sum_{j=1}^{J+1} u_j h_j^d = 0^d, u \geq 0\}.$$

Then, there exists a certificate according to Theorem 2.9 with at most p points, with

$$p \leq \binom{J+1}{J - \text{rank}([h_1^d, \dots, h_{J+1}^d])}.$$

Proof. Since we only have linear constraints the subdifferential contains only the unique gradient and it is independent of the suspension point x_i , i.e., $h_j := h_{i,j}$. Moreover, due to (iii) $1 \leq |\text{supp}(u_i)| \leq d + 1$ holds. That is, each u_i has at most $d + 1$ non-zero entries. For every certificate point i holds for u_i

$$\sum_{j=1}^{J+1} u_{ij} h_j \in \mathbb{R}^n \times \{0\}^d, \quad u_{ij} \geq 0$$

and thus

$$\sum_{j=1}^{J+1} u_{ij} h_j^d = 0^d, \quad u_{ij} \geq 0.$$

Let p be the number of extreme rays of this cone and let $v_q, q = 1, \dots, p$, be the extreme rays. For every certificate point i , the corresponding u_i is a convex combination of extreme rays, i.e.,

$$u_i = \sum_{q=1}^p \lambda_q^i v_q, \quad \lambda^i \in \mathbb{R}_{\geq 0}^p.$$

And we can choose this convex combination in such a way that the number of extreme rays t_i needed is bounded by d , i.e., $|\text{supp}(\lambda^i)| = t_i \leq d$. If $\lambda_q^i \neq 0$, only entries of v_q , which are non-zero in u_i , can be non-zero. This implies that the non-zero entries of the extreme ray correspond to constraints which are violated at the current suspension point. Hence, every extreme ray v_q with $\lambda_q^i \neq 0$ also leads to a valid normal vector for the suspension point. Then, take t_i copies of this certificate point and the corresponding extreme rays as hyperplanes. The original hyperplane is removed. This procedure also leads to a certificate. According to Lemma 5.7 the extreme rays for one certificate point are a subset of the extreme rays of

$$\{u_j \in \mathbb{R}^{J+1} \mid \sum_{j=1}^{J+1} u_j h_j^d = 0^d, u_j \geq 0\},$$

by setting $[M, m] = [h_1^d, \dots, h_{J+1}^d]$, $\beta := J$, and $\alpha := d$. As in the proof of Lemma 5.5 parallel hyperplanes can get removed. So there exists a certificate with at most p points, whereby p can be estimated by Lemma 5.8. \square

TABLE 1. Upper bounds on the size of a generalized certificate (J : number of constraints, n : number of integer variables).

	integer	mixed-integer
linear	$J + 1$ (Lemma 5.5)	$\binom{J+1}{J - \text{rank}([h_1^d, \dots, h_{J+1}^d])}$ (Theorem 5.9)
convex	2^n (Example 6.3)	2^n (Example 6.3)

Remark 5.10. *If we have a pure integer linear problem, $\text{rank}([h_1^d, \dots, h_{J+1}^d]) = \text{rank}(0) = 0$ holds and thus there exist $\binom{J+1}{J} = J+1$ possibilities and the formula is the same as in Lemma 5.5.*

Remark 5.11. *Due to Corollary 3.9 all upper bounds also hold for the generalized optimality certificate.*

We summarize our results about upper bounds in Table 1.

5.3. Influence of a block structure on the size of a certificate. Optimization problems with block structure occur frequently in different areas, see, e.g., (Martin 1999; Geißler et al. 2018; Schewe et al. 2020). If different blocks are connected by constraints, this is called *bordered block diagonal form* (Borndörfer et al. 1998). A bordered block diagonal form is desirable, because it can guide and speed up a solution process for optimization problems. We want to investigate in this section what influence the presence of such a structure has on the properties of an optimality certificate. To do this, we take a two-step approach. First, we consider the simple case of *uncoupled blocks*, where the individual blocks are not connected. We then look at the more interesting case of coupled blocks by examining the bordered block diagonal form. In order to exploit the block structures to form smaller certificates we give a new definition of a generalized certificate for problems with block structure.

We consider the convex mixed-integer nonlinear problem

$$\min_x \sum_{w=1}^W f_w(x_w) \quad (14a)$$

$$\text{s.t. } g_w(x_w) \leq 0, \quad \text{for all } w = 1, \dots, W \quad (14b)$$

$$x_w \in \mathbb{Z}^{n_w} \times \mathbb{R}^{d_w} \quad (14c)$$

where W is the number of blocks. It is obvious that one can determine an optimality certificate for each block independently, since the blocks are not coupled. From the individual optimality certificates, one can then simply create an optimality certificate for the entire problem. Thus, one obtains an optimality certificate that is linear in W and not exponential.

Observation 5.12. *Let Problem (14) be given, then the problem decomposes into independent problems $\min\{f_w(x_w) \mid g_w(x_w) \leq 0, x_w \in \mathbb{Z}^{n_w} \times \mathbb{R}^{d_w}\}$ for $w \in \{1, \dots, W\}$, which can be optimized independently.*

This observation implies that in case we are given a problem of Form (14), one can form separate certificates for each block $w \in \{1, \dots, W\}$ in order to certify the optimality of the entire problem.

Thus we give the definition of a *block-structured generalized optimality certificate* which is necessary to use – and not the generalized optimality certificates like before – as exemplified by Example 5.15.

Definition 5.13 (Block-structured generalized optimality certificate). *Let Problem (14) be given, then the $w \in \{1, \dots, W\}$ generalized optimality certificates x_1^w, \dots, x_l^w together with normal vectors h_1^w, \dots, h_l^w for each block $w \in \{1, \dots, W\}$ form a block-structured generalized optimality certificate.*

Lemma 5.14. *Let Problem (14) be given, then the corresponding block-structured generalized optimality certificate has at most size $\sum_{w=1}^W 2^{n_w}$.*

Proof. For each block $w = 1, \dots, W$ on its own it is possible to find an optimality certificate with at most the size of 2^{n_w} . \square

We need to consider block-structured generalized optimality certificates and not generalized optimality certificates as in the rest of the paper. This is illustrated by the following example.

Example 5.15. *Problem (18) can be split into blocks where each block only consists of one dimension. Thus we certify optimality by introducing half-spaces*

$$\begin{aligned} (-1)^\top (z_p - 0) &< 0 \\ (1)^\top (z_p - 1) &< 0 \end{aligned}$$

for all $p \in \{1, \dots, n\}$. This encloses a mixed-integer free polyhedron and certifies optimality for the entire problem.

We like to note that this is technically not an optimality certificate in the sense of Definition 3.4 as here $(\frac{1}{2}, 1.1)$ would be cut off by the certificate even though it is contained in the sublevel set of all optimality certificates. However, it is a block-structured generalized optimality certificate and therefore, in order to certify optimality, one does not need 2^n certificate half-spaces.

Furthermore, the proof of Lemma 5.14 shows that any smaller optimality certificate for one block can also be exploited for an optimality certificate for the entire problem. Thus, using the bounds shown in Section 5.2, further statements about optimality certificates can be achieved.

Building on Lemma 5.14, we now examine the case of problems with bordered block diagonal form. For this we consider the problem

$$\min_{z, y} \sum_{w=1}^W f_w(z_w, y_w) \tag{15a}$$

$$\text{s.t. } g_w(z_w, y_w) \leq 0 \quad \text{for all } w = 1, \dots, W \tag{15b}$$

$$G(z_1, \dots, z_W) \leq v \tag{15c}$$

$$(z_w, y_w) \in \mathbb{Z}^{n_w} \times \mathbb{R}^{d_w}, \tag{15d}$$

where $v \in \mathbb{R}^k$, $k \in \mathbb{N}$, $l := \sum_{w=1}^W n_w$, and a function $G : \mathbb{Z}^l \rightarrow \mathbb{R}^k$. Note that we are considering in problem (15) a special form of a bordered block diagonal form, since there are only integer variables present within the *coupling constraint* (15c). Examples of problems with this type of coupling constraint are competitive market equilibrium problems (Bikhchandani and Mamer 1997; Bikhchandani, Ostroy, et al. 2002) and the p -median problem (Daskin and Maass 2015).

Definition 5.16. *Let \bar{x} be a point of Problem (15) then we define the projection of the sublevel set of projection of $\mathcal{S}(\bar{x})$ into the dimensions of block w as*

$$\begin{aligned} \mathcal{S}_w(\bar{x}) := & \{x_w \in \mathbb{Z}^{n_w} \times \mathbb{R}^{d_w} \mid \exists x : G(z_1, \dots, z_w, \dots, z_W) \leq v, g_w(z_w, y_w) \leq 0, \\ & f(x_1, \dots, x_w, \dots, x_W) < \min_{\bar{x}} f(\bar{x}_1, \dots, \bar{x}_w, \dots, \bar{x}_W) \text{ for all } w = 1, \dots, W\}. \end{aligned}$$

Lemma 5.17. *Let \tilde{x} be a point of Problem (15) and let \tilde{x}_w be the corresponding variable from block w . Then there is a corresponding normal vector h_w to \tilde{x}_w . Such that the projection $\mathcal{S}_w(\tilde{x})$ then is contained in the set $\mathcal{H}_w(\tilde{x}_w, h_w) := \{x_w \in \mathbb{Z}^{n_w} \times \mathbb{R}^{d_w} \mid h_w^\top(x_w - \tilde{x}_w) < 0\}$:*

$$(16a)$$

$$\mathcal{S}_w(\tilde{x}) \subset \mathcal{H}_w(\tilde{x}_w, h_w). \quad (16b)$$

Proof. As the projection of the sublevel-set $\mathcal{S}_w(\tilde{x})$ is convex there is a corresponding normal vector h_w fulfilling the Relation (16a). \square

Hence, we can deduce the following bound on problems in block bordered diagonal form where all coupling constraints only contain integer variables.

Theorem 5.18. *Let Problem (15) be given with support set $\{z \in \mathbb{Z}^n \mid G(z_1, \dots, z_w, \dots, z_W) \leq v\}$ then there exists a block-structured generalized optimality certificate of size $\sum_{w=1}^W 2^{n_w}$, where W corresponds to the number of blocks.*

Proof. Lemma 5.17 shows that for feasible suspension points there are corresponding normal vectors. In case a suspension point is infeasible the corresponding inequality that contains all feasible points for the block does not cut off any feasible point of the entire Problem (15). Therefore certificates for each block can be formed and therefore block-structured generalized optimality certificate exists for the entire problem. \square

Here the size of the block-structured generalized optimality certificate can be smaller as well in case there are smaller certificates for the blocks.

6. COMPARISON TO OTHER OPTIMALITY CERTIFICATES

In this section we show that the optimality certificates that were introduced in this paper may lead to smaller certificates than the original certificates in Baes et al. (2016) and we show that the branch and bound algorithm may yield a partial optimality certificate.

6.1. Original certificate. In this subsection we study lower bounds on the size of a certificate according to Theorem 2.9, show that a generalized certificate can be smaller than this bound, and give an example where the size of both certificates must be 2^n .

For this reason, we introduce Algorithm 4, which calculates a lower bound on the size of a certificate according to Theorem 2.9. This algorithm initializes at first a counter t with one and a set X with the optimal solution \hat{x} of the optimization problem (1). Then, the algorithm checks for every constraint of the input problem whether omitting it would yield a better optimal objective value. If this is the case, counter t is increased by one and a feasible solution x^j with better objective value is added to X . Finally, t is divided by $d + 1$ to take the influence of continuous variables into account. A lower bound on the size of a certificate introduced by Baes et al. (2016) is given by return value t as stated in the next lemma.

Lemma 6.1. *Let a mixed-integer convex problem (1) with optimal solution \hat{x} be given and let a certificate according to Theorem 2.9 exist. Then, Algorithm 4 determines a lower bound t of points in a certificate according to Theorem 2.9.*

Proof. To get a mixed-integer-free polyhedron, every point in X has to get cut off by a certificate point with corresponding hyperplane. Let x_i with hyperplane $h_i := \sum_{j=1}^{J+1} u_{ij} h_{ij}$ be a valid certificate point according to Theorem 2.9. Choose $x^l \in X$ arbitrarily. Due to the properties in 2.9, the convexity of the constraints and the objective function, and the construction of x^l in Algorithm 4, hyperplane h_i can cut off point x^l if and only if $u_{il} \neq 0$. Moreover, there is a 1-1-correspondence between points in X and constraints or the objective function. Since property (iii), $1 \leq |\text{supp}(u_i)| \leq d + 1$, has to hold, hyperplane

Algorithm 4 Computing a lower bound on the size of a certificate**Input** A Problem (1) with optimal solution \widehat{x} **Output** Minimum number t of points in a certificate

```

1:  $t \leftarrow 1$ .
2:  $X \leftarrow \{\widehat{x}\}$ . ▷ Add optimal solution.
3: for  $j = 1, \dots, J$  do
4:   Solve (1) without constraint  $g_j(x) \leq 0$ .
5:   Let  $x^j$  be an optimal solution or a feasible solution with  $f(x^j) < f(\widehat{x})$ .
6:   if  $f(x^j) < f(\widehat{x})$  then
7:      $t \leftarrow t + 1$ .
8:      $X \leftarrow X \cup \{x^j\}$ .
9:  $t \leftarrow \lceil \frac{t}{d+1} \rceil$  ▷ Consider continuous variables.
10: return  $t$ 

```

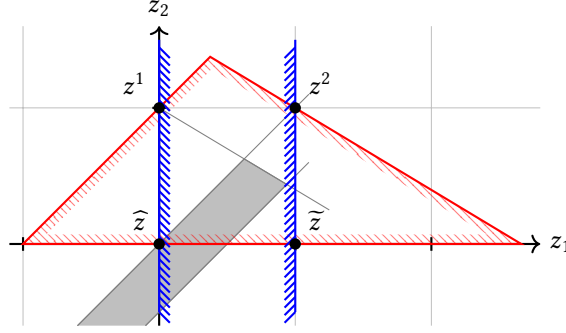


FIGURE 2. Feasible set of Problem (17), the certificate generated by Baes et al. (2016) (red), and a generalized certificate (blue).

h_i at point x_i can cut off maximal $d + 1$ points of X . Thus, minimal $\frac{|X|}{d+1}$ certificate points are necessary. \square

The following example visualizes Lemma 6.1. Moreover, it shows, that these bounds are not transferable to generalized certificates, since the generalized certificate can be smaller.

Example 6.2. Consider the following pure integer linear program with three constraints.

$$\min_z -z_2 \tag{17a}$$

$$\text{s.t. } -z_1 + z_2 \leq 0 \tag{17b}$$

$$0.5 - z_1 + z_2 \leq 0 \tag{17c}$$

$$-1 + 0.6z_1 + z_2 \leq 0 \tag{17d}$$

$$z \in \mathbb{Z}^2. \tag{17e}$$

The optimal solution is $\widehat{x} = (0, 0)$. If constraint (17b) or (17d) are removed, the objective value decreases. So, Algorithm 4 states, that a certificate according to Theorem 2.9 needs at least three points. Whereas there is a generalized certificate that requires only two hyperplanes. See Figure 2.

However, the following example shows that the bound of 2^n for the number of necessary hyperplanes can not be reduced.

Example 6.3.

$$\min_z \sum_{p=1}^n (z_p - \frac{1}{2})^2 \quad (18a)$$

$$\text{s.t. } z \in \mathbb{Z}^n. \quad (18b)$$

In this problem the certificate of minimal size is given with the suspension points $z_i = \{0, 1\}^n$, where the normal vectors h_i are given via $h_{i,p} = 2z_{i,p} - 1$. This stems from the fact that the sublevel-sets for all points are n -dimensional balls around the center $(\frac{1}{2}, \dots, \frac{1}{2})$. This shows that the aforementioned bound can not be reduced.

The previous Example 6.3 also shows that reducing the feasible set to only cover binary variables ($z \in \{0, 1\}^n$) does not reduce the necessary size of the certificate. Furthermore, setting the support set \mathcal{Z} to $\{0, 1\}^n$ does not change this as then still all points in the support set are needed to form the certificate. However, in the case of pure integer linear problems the support set \mathcal{Z} can be set at the feasible region of the considered problem. This decreases the size of a certificate to only one inequality as this is sufficient to form the mixed-integer free polyhedron.

6.2. Branch-and-Bound. In this subsection we show that the branch-and-bound algorithm yields valid certificate half-spaces, but does not form an entire certificate in general. As the branch-and-bound algorithm yields mixed-integer feasible points in its search tree these points can be used to generate a certificate. However, these points and normal vectors do not necessarily form a generalized certificate.

6.2.1. Partial certificate through branch-and-bound. Some nodes in the branch and bound tree can be used to generate valid certificate half-spaces. So we are given an optimization Problem (1). Let \hat{z} be a suspension point of the certificate. Then we compute the normal vector through

$$\nabla f(\hat{z}, y) + \sum_{j \in J} \lambda_j \nabla g_j(\hat{z}, y) = h \quad (19a)$$

with $h \in \mathbb{R}^n \times 0^d$, see Lemma 3.15. Now let $\hat{z} \in \mathbb{Z}^n$ be a mixed-integer feasible solution to a branch-and-bound node, i.e., which branches e.g.

$$\min_{z, y} f(z, y) \quad (20a)$$

$$\text{s.t. } g(z, y) \leq 0 \quad (20b)$$

$$z_p \leq b_p \quad \text{for all } p \in \mathcal{B} \quad (20c)$$

$$z \in \mathbb{R}^n, y \in \mathbb{R}^d. \quad (20d)$$

with $b \in \mathbb{Z}^n$ and \mathcal{B} being the set of branched variables.

We use this to show that the arising problems help form a certificate as feasible nodes of the branch-and-bound – together with corresponding dual variables – lead to valid certificate half-spaces. This is explained precisely in the following lemma.

Lemma 6.4. *Given a Node (20) in a branch-and-bound algorithm with optimal solution (\hat{z}, \hat{y}) fulfilling the mixed-integer constraint $\hat{z} \in \mathbb{Z}^n$, then the dual variables η corresponding to the fixing constraint (20c) form the corresponding normal vector h to the point (\hat{z}, \hat{y}) in the following way. For the integral dimensions of h we set*

$$h_p = \begin{cases} -\eta_p, & \text{if } p \in \mathcal{B} \\ 0, & \text{else} \end{cases} \quad \text{for all } p \in 1, \dots, n$$

and for the continuous dimensions we set $h_e = 0$.

Proof. We need to show that the \tilde{h} that we compute through the procedure given in Lemma 6.4 fulfills the properties of Definition 3.4. We do that by applying Lemma 3.15. The corresponding KKT-system to the Node (20) is

$$\begin{aligned} \nabla f(\hat{z}, y) + \sum_{j=1}^J \lambda_j \nabla g_j(\hat{z}, y) + \sum_{p \in \mathcal{B}} \eta_p \begin{pmatrix} e_p \\ 0_d \end{pmatrix} &= 0_{n+d} \\ 0 &\geq g(\hat{z}, y) \perp \lambda \geq 0 \\ 0 &\geq z - b \perp \eta \geq 0. \end{aligned}$$

However, solving this system leads to a solution to the KKT-systems arising from the feasible-fiber subproblems (3), see Lemma 3.15. This infers that the duals η can be used to generate the normal vectors h with the described procedure. \square

This implies that solving an optimization Problem (1) via a branch-and-bound algorithm while using KKT-systems to solve the nodes may lead to an optimality certificate where the feasible nodes are the suspension points and the corresponding normal vectors are automatically given as described above.

An example for yielding an entire certificate instead of just some hyperplanes for feasible suspension points is given by Problem (18), as no node will be pruned, because all suspension points in the certificate have the same objective value.

6.2.2. *Branch-and-bound does not yield an entire generalized certificate.* To observe that the branch-and-bound algorithm does not yield an entire generalized certificate we give the following example:

Example 6.5.

$$\begin{aligned} \min_z (z_1 - \frac{1}{10})^2 + (z_2 - \frac{1}{2})^2 \\ \text{s.t. } z \in \mathbb{Z}^2. \end{aligned}$$

In the first step we solve continuous-relaxation of the problem which leads to the solution $(\frac{1}{10}, \frac{1}{2})$ with objective value 0. First we branch on z_1 which leads to the solutions $z_1 \leq 0 : (0, \frac{1}{2})$ and $z_1 \geq 1 : (1, \frac{1}{2})$ with objective values $\frac{1}{4}$ and $\frac{5}{4}$. Then we branch on z_2 which, for $z_1 \leq 0$, leads to solution $(0, 0)$ and for $z_1 \geq 1$, $z_1 \geq 1$ to $(1, 0)$ with objective values 0 and 1. This implies that we can prune $z_1 \geq 1$, $z_2 \geq 1$ and $z_1 \geq 1$, $z_2 \leq 0$, as the objective values for $z_1 \leq 0$, $z_2 \geq 1$ and $z_1 \leq 0$, $z_2 \leq 0$ are smaller than for $z_1 \geq 1$.

This implies we have the multipliers for $(0, 0)$, $(0, 1)$ but this leads to the vectors $(\frac{-2}{10}, -1)$ and $(\frac{-2}{10}, 1)$. Thus, it does not form a generalized certificate. See Figure 3.

Therefore, the branch-and-bound algorithm does not yield an entire generalized certificate.

6.2.3. *Using a branch-and-bound tree as a certificate.* As the branch-and-bound tree certifies optimality we show that considering the generalized optimality certificates is worthwhile because they are bounded by 2^n and the branch-and-bound tree is not bounded by the number of integer variables. Instead it is bounded by the variable bounds of the integer variables. The following example shows that the branch-and-bound tree may contain more nodes than the generalized optimality certificate.

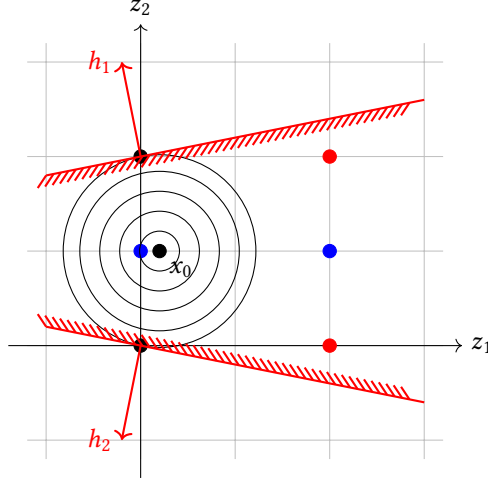


FIGURE 3. Branch-and-bound nodes with corresponding half spaces. Marked with an “ x_0 ” the continuous optimum, blue the solutions after branching at z_1 , solutions, that are pruned, highlighted in red.

Example 6.6. Observe the following problem with its optimum at $z = (97, 103)$.

$$\begin{aligned} \min_z \quad & z_1 + z_2 \\ \text{s.t.} \quad & z_2 = \frac{103}{97} z_1 \\ & z_1 \geq \frac{1}{2} \\ & z \in \mathbb{Z}^2. \end{aligned}$$

The branch-and-bound algorithm has to branch alternating between z_1 and z_2 in order to reach the optimum. This leads to more nodes being computed than needed to form an optimality certificate, as this needs at maximum 2^2 hyperplanes. In order to store the information we need less storage with using the certificate than storing the information through the branch-and-bound algorithm.

7. COMPUTATIONAL RESULTS

In this section we carry out computational experiments in order to investigate the properties that the generalized optimality certificates and Algorithm 1 have with respect to problems stemming from the MINLPLib (Bussieck et al. 2003) and the MIPLIB 2017 (Gleixner et al. 2021).

As in theory the optimality certificates can be exponential in size, one can not expect to compute optimality certificates for all problems – thus we investigate how many certificates can be computed and whether there are properties of the problems that a priori allow to assess whether an optimality certificate can be computed.

We will show that optimality certificates can be computed for realistic instances and that the size of these optimality certificates are mostly considerably lower than the theoretical worst upper bound of 2^n . Furthermore, we show that the computational time for these optimality certificates that could be computed is viable.

Subsequently, we investigate which sizes the optimality certificates have with respect to the number of integer variables. In addition, we would like to know which impact the linearity of mixed-integer linear problems has in comparison to the convexities of the convex mixed-integer nonlinear problems.

TABLE 2. Computational results of Algorithm 1 on MINLPLib (Bussieck et al. 2003)

Statistics	Results		Instances	
	Computation time	# Hyperplanes	# Integer variables	# Continuous Variables
Mean	132.8 s	7316.8	16.4	7366.2
Std	381.8 s	14 696.5	9.8	27 483.9
Min	0.0 s	1	3	3
25%	0.4 s	18.5	10	21
50%	1.5 s	174	14	62
75%	10.3 s	1025.5	19.5	585
Max	1479.8 s	40 947	40	106 705

These main topics are investigated, first, for convex mixed-integer nonlinear problems and, second, for mixed-integer linear problems. Hereby we state the results for each type of problem and then discuss and compare them later on.

7.1. Computational setup. We compute the certificates on the Woody cluster of the RRZE-HPC (Regionales Rechenzentrum Erlangen 2021) with four Xeon E3-1240 v5 CPUs running with 32 GB of RAM and 3.50 GHz. Furthermore we set an internal RAM limit in our code of 20 GB. The underlying optimization problems are solved by using Gurobi 9.1 (Gurobi Optimization, LLC 2021) while the algorithm itself is implemented in Python 3.7.

The optimality certificates are computed for instances from the MIPLIB 2017 (Gleixner et al. 2021) for the mixed-integer linear case and for the convex mixed-integer nonlinear case we compute certificates for problems from the MINLPLib (Bussieck et al. 2003). We downloaded and selected the respective instances using the MIPLibing library (Serra and O’Neil 2020). The code is available at <https://github.com/lhuembs/certificates>.

7.2. Convex mixed-integer nonlinear problems. We picked the instances from the MINLPLib (Bussieck et al. 2003) via the following pattern: The instances contain at least one continuous variable in line with the mixed-integer linear problems. As here only one certificate half-space is needed because these instances have a linear objective function which, together with the support set, forms a certificate. Furthermore we only considered instances that have a proven convex relaxation as otherwise Theorem 3.6 can not be applied. Thus we were left with 67 instances. Of these 67 instances 66 could be solved by Gurobi within our time-limit of 3000 seconds.

We were able to compute 15 certificates for the remaining 66 problems. Thus we could provide certificates of optimality for almost a quarter of all instances. Therefore, as stated before, it could be shown that for many instances optimality certificates could be computed within viable time. In the following we investigate what characterizes the instances that we could compute certificates for and what differs the uncomputable ones from the computable ones. We then reduced the certificates that we could compute via Algorithm 2.

Concerning the question which certificates could be computed: we were able to compute optimality certificates for all problems that contain less than 18 integer variables. On the upper end: The problem, for which we could compute a certificate, that contains the most integer variables has 40 integer variables.

Hereby in Figure 4 one can see that the number of half-spaces needed to form a certificate is not exponential and thus the theoretical bound of 2^n is not attained in general, as already stated in the introduction to this section. In the following we further explain

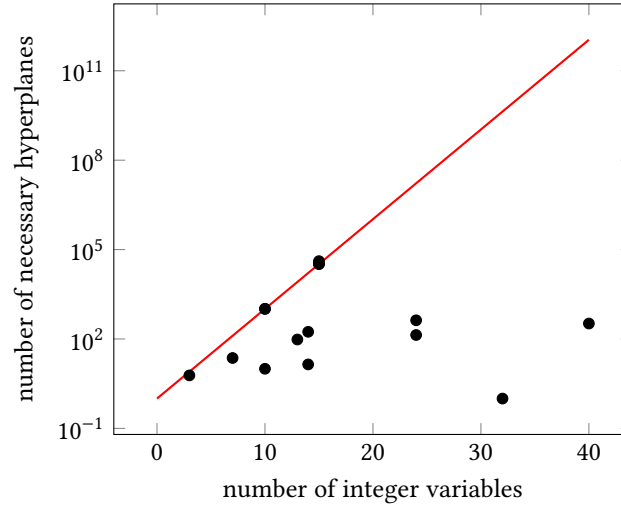


FIGURE 4. Results of computations of Algorithm 1 on MINLPlib (Bussieck et al. 2003); in red exponential function 2^z

for which instances we could compute optimality certificates and what characterizes them. In addition, we show that the optimality certificates could be computed in viable time – when a certificate could be computed.

Most of these certificates could be computed quite fast as the following metrics, given in seconds, show. The mean computational time was 132.8 s. Which is fast considering the fact that the optimality certificates – in theory – may need up to 2^n half-spaces. However, later on we show that the optimality certificates that we computed do not need that many half-spaces and therefore this explains why the computation time is smaller than expected. The standard deviation was 381.8 s which is quite high given the short mean computation times. This can be explained by the fact that often the optimality certificates are extremely small, e.g., only one half-space is needed. Sometimes, on the other hand, the certificates need a considerable number of half-spaces and thus the computation time for these instances is much larger. The minimal computation time that was needed to find a certificate was 0.0 s. This happened when computing the optimality certificate for the instance “st_miqp4”. This instance only needs 6 hyperplanes to form a certificate and has 3 integer variables. Furthermore, the 25% percentile was 0.4 s while the median was 1.5 s. So both of these computation times are very small while the number still more than doubles from 25% percentile to median. The 75% percentile was 10.3 s, which is considerably higher than the median. Thus most of the certificates could be computed quite fast. This is also explained by the small size of most optimality certificates that we could compute. However, the maximal computation time of a certificate that was found was 1479.8 s. In this case, the instance “cvxnonsep_normcon30r”, 40 947 half-spaces were computed. So naturally computing this large number of half-spaces takes more time than the other instances. In the following it will also be shown that this instance has the largest certificate that could be computed.

However, these computational results regarding the computational time show that when an optimality certificate could be computed it normally did not take much time and therefore one can give computing an optimality certificate a try as it is not so expensive regarding the computational time.

The results illustrate that most certificates do not need many certificate points as the mean size of the certificates is 7316.8. Thus this is quite small given the fact that the sizes of optimality certificates grow exponentially in the number of integer variables. Here the

TABLE 3. Computational results of Algorithm 1 on MIPLIB (Gleixner et al. 2021)

Statistics	Results		Instances	
	Computation time	# Hyperplanes	# Integer variables	# Continuous Variables
Mean	3480.3 s	1996.3	3231.0	6319.4
Std	5481.8 s	4508.4	5993.8	9589.3
Min	0.1 s	1	1	30
25%	43.1 s	5.5	3.5	150
50%	1146.6 s	146	174	2467
75%	4530.6 s	254.5	4254.5	7339
Max	20 025.0 s	19 130	22 326	34 324

standard deviation was 14 696.5 which is high given the small mean. However, as we will see, most optimality certificates are small in size while few are quite large. The minimum size that was needed is 1, which happened for the instance “clay0204m”. This instance has 32 integer variables, which shows that the sizes of the certificates can be quite small despite a larger number of integer variables and thus the upper bound on the half-spaces needed of 4 294 967 296 is not obtained. The 25% percentile of the half-spaces needed was 18.5 and the 50% percentile was 174 which are both quite small given that the median number of integer variables in the set of instances that were computed is 14 which would potentially lead to 16 384 half-spaces. The 75% percentile was 1025.5 and the maximal size of a certificate that was found was 40 947 which corresponds to the instance “cvxnonsep_normcon30r” which was mentioned before. This instance has 15 integer variables which is smaller than the number of integer variables that the instance “clay0204m” has. Surprisingly, “clay0204m” needs far fewer half-spaces to form a certificate as stated above. On the basis of the large difference between the maximum and 75% percentile one can also see that this huge number of half-spaces needed in the instance “cvxnonsep_normcon30r” is an outlier. In addition, from Table 2 it can be seen that there are instances that need the upper bound of 2^n certificate half spaces. This is the case of “cvxnonsep_normcon20r” and “squfl015-060” in addition the instances “cvxnonsep_normcon30r”, “squfl015-080” and “squfl010-080” have more certificate half spaces than 2^n , which happened because Algorithm 2, which reduces the certificates, did not finish within the time limit.

Therefore, it could be shown that when an optimality certificate could be computed the resulting optimality certificate it normally was smaller in size than the worst case upper bound of 2^n . Thus storing optimality certificates usually does not take much space and therefore computing them is actually worthwhile.

7.3. Mixed-integer linear problems. We picked the instances from the MIPLIB 2017 Benchmark (Gleixner et al. 2021) via the following pattern: The instances contain at least one continuous variable, as otherwise only one certificate hyperplane is needed, and are feasible. The Benchmark contains 129 instances that meet these criteria. Of these 129 instances Gurobi was able to compute 125 instances within our time limit of 21000 seconds. We set this time limit as there were some instances for which we could find a certificate close to the initial time limit of 3000 seconds. For 31 of these instances we were able to compute an optimality certificate and furthermore we were able to reduce the certificates according to Algorithm 2 in 12 cases.

Therefore again it is worth a try computing optimality certificates as many optimality certificates could be computed withing viable time.

In Figure 5, we can observe that the number of half-spaces needed to form a certificate is far from exponential in the number of integer variables for the examples for which

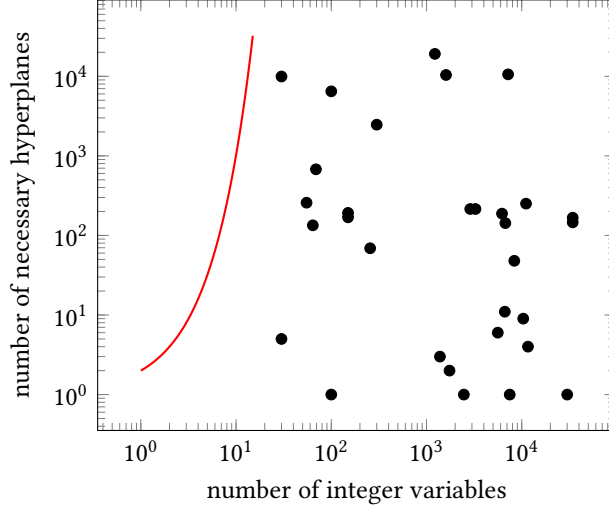


FIGURE 5. Results of computations of Algorithm 1 on MIPLib Benchmark (Gleixner et al. 2021); in red exponential function 2^z

we could compute a certificate. In the following we further explain for which instances we could determine optimality certificates and what characterizes them similar as in the preceding subsection on convex MINLPs. Furthermore we investigate the computation times of the certificates.

The mean computational time was 3480.3 s which is far higher than in the case of MINLPs but here the time limit also was higher. We used a higher time limit in the case of MIPs as here test results showed that some certificates may be computed using a higher computation time and therefore an increase in the time limit was seen as a good way to compute additional certificates. Hereby the standard deviation was 5481.8 s which is quite high given the small mean computation time. In the following we can see that the sizes of the certificates vary widely also as in the preceding subsection regarding mixed-integer nonlinear problems. This then explains the high standard deviation. The minimal computation time that was needed to find a certificate was 0.1 s which happened for the instance “cbs-cta” which only needs one half-space to form a certificate. Also the 25% percentile was quite small 43.1 s similar to the MINLPs. However the median was considerably higher at 1146.6 s. Furthermore the 75% percentile was 4530.6 s and thus by far higher than in the case of convex mixed-integer nonlinear problems. Thus most of the certificates could be computed quite fast but needed more time than in the case of MINLPs. The maximal computation time of a certificate that was found was quite high at 20 025.0 s which was the certificate for the instance “neos-3216931-puriri”. This instance needed 215 half-spaces to form a certificate.

Thus the initial time limit is sufficient for most of the certificates but some could be solved using the higher time limit. Therefore most optimality certificates that could be computed were computed relatively fast. Therefore, like in the case of convex mixed-integer nonlinear problems, trying to compute optimality certificates is worth a try.

The sizes of the certificates are in mean 1996.3 while the standard deviation is 4508.4. Surprisingly, 3 instances only need one hyperplane to form a certificate. This can be seen as even the 25% percentile is 5.5. Furthermore the 50% percentile is small at 146. The 75% percentile still is at 254.5 which is small given the number of integer variables in the instances. The maximal size of a certificate that could be computed was 19 130 which was the case for the instance “app1-1”. Therefore, as a lot of optimality certificates are quite small in size, it is worthwhile computing and storing these optimality certificates.

7.4. Discussion.

7.4.1. *Convex mixed-integer nonlinear problems.* All of the results show that if one finds a certificate at all it will be found fast and the corresponding size of the certificate is quite small. Thus trying to compute optimality certificates is worth a try as often a certificate can be found, and in case it is found, it does not take up many resources.

In addition, we explained earlier on that the number of integer variables heavily influences the algorithm's ability to find an optimality certificate. In the following we further show which parameters influence whether the optimality certificates can be computed.

In the MINLPLib the number of integer variables that are part of a nonlinear function is given. Here we see that we only could compute 3 instances which contain such a variable. Hereby the maximal number of these variables for which we still could compute a certificate is 15. We think that the occurrence of integer variables in nonlinear constraints lead to non-polyhedral sublevel-sets, which makes a description via halfspaces uneconomical. Thus one has to keep this in mind when computing optimality certificates for such instances – one may not be able to generate an optimality certificate. In Section 5 it was shown that linearities can positively influence the number of half-spaces that are needed. Furthermore Problem (18) shows an extreme example for a mixed-integer nonlinear problem where the fact that the sublevel-set is not polyhedral leads to the maximal size of certificates. Therefore linearity in the integer variables positively influences the algorithm's ability to compute certificates and trying to compute optimality certificates for this type of instances is promising.

In addition, the MINLPLib shows the number of linear constraints, where it can be seen that the number of linear constraints also positively influences computability. With the exception of one instance the number of linear constraints is 4 or larger. The rule of thumb is that more linear constraints are better for our algorithm. These observations also can be explained like in the preceding block as also in the case of constraints linearity is positive for computing optimality certificates.

All of these points mentioned also are reflected by the number of nonlinear nonzeros in the Jacobian. Here we could only compute certificates for 2 instances which have more than 0 nonlinear nonzeros. In the entire test set that we aimed to compute certificates for there are about half of the instances greater with a number greater than zero in this identifier. This further shows that – as a rule of thumb – the closer to a linear mixed-integer problem the instance is, the easier it is to compute the corresponding certificate.

The instances in our test set all either have a convex objective and linear constraints or they have a linear objective and convex constraints. Here it shows that even though there are 38 instances with linear objective function and 27 instances with convex objective function we can compute optimality certificates for 12 instances with convex objective function and only for 3 instances with convex constraints. This can be explained by the fact that the objective function is only one function and there – in general – is more than one constraint. Thus there is the possibility to introduce more nonlinearities via the constraints than the objective function.

Furthermore, an interesting observation that could be made is that computing new certificate half-spaces can be done very fast. In general it took less than one second to compute a new one. Even the 99.99% percentile of the computation time to find a new certificate half-space normally took less than ten seconds. This can be explained by the small number of integer variables that the instances in question have. The corresponding master problems (7b) then reduce to small mixed-integer linear problems which can be computed quite fast by Gurobi. The remaining fiber subproblems (3b) and (4b) then reduce to convex problems that are easy to solve as well. Therefore the half-spaces can be computed fast for these instances in question.

To summarize, we can state that the number of integer variables influence the ability to find optimality certificates as predicted by the theory. The second major observation is that linearity positively influences the algorithm. This can be seen by the number of integer variables, the nonlinear nonzeros in the Jacobian and the number of linear constraints. Thus, in case the optimization problem at hand contains only few integer variables or is “close” to being a linear mixed-integer problem an optimality certificate probably can be computed. In case the optimality certificate can be computed it most likely will be computed fast and it will not need many half-spaces.

7.4.2. Mixed-integer linear problems. In the case of mixed-integer linear problems the influence of integer variables on the ability to compute certificates could not be seen. The instance with the most integer variables for which we could compute a certificate is the instance “satellites2-40” with 34 324 integer variables. In this case only 167 certificate half-spaces were needed which further illustrates that the certificates can be quite small regardless of the integralities.

Additionally, we investigated how the type of problem influences our ability to compute certificates. Here it can be said that, e.g., for half of the instances (3 out of 6) with the tag “integer knapsack” a certificate could be computed. This might be attributed to the observation in Section 5 that the knapsack problem can be certified using only one additional half-space. This argument is unfortunately weakened by the fact that for only a quarter of all instances with the tag “knapsack” (5 out of 20) we could compute a certificate.

7.4.3. Comparison. To extend the discussion on the speed with which we can compute certificate half-spaces we can state that in the case of mixed-integer linear problems, as a rule of thumb, it takes longer to compute new half-spaces. Sometimes it takes minutes to find a new one. However, it can be stated that this is harder for the mixed-integer linear problems. We assume this is because of the number of integer variables that the instances have. The mean of the number of integer variables is 1914 in mixed-integer linear case while in the case of convex mixed-integer nonlinear problems this was 30 so the master problem (7b) is easier to solve.

Furthermore, we could observe that in the case of linear mixed-integer problems certificates could be computed for problems with more integer variables. This can be attributed to the observations made from the computations of the convex mixed-integer problems as here – as a rule of thumb – it is more to be able to compute a certificate the closer it is to linearity.

In summary, the following can be concluded. If a certificate can be computed, it will most likely be small and can be computed in a short time, while the problem for which it is to be computed either has only a few integer constraints or has no significant nonlinearities. Therefore for this type of instances trying to compute optimality certificates is very promising.

8. OUTLOOK AND CONCLUSION

In this paper we extended a concept of optimality certificates that was introduced by Baes et al. (2016). This was done in order to be able to compute optimality certificates which was not possible before. Thus we introduced an algorithm that can use any solver as a black box in order to compute the optimality certificates. We showed that the optimality certificates we introduced are at most as large than the previous optimality certificates. The optimality certificates presented in Baes et al. (2016) and the generalized optimality certificates have the same theoretical upper bound, but there are problems where the generalized optimality certificates are smaller than the original optimality certificates. Furthermore, we exploited special structures of optimality problems to extract smaller

upper bounds on the sizes of optimality certificates. This could be done, e.g., for mixed-integer linear problems, problems with equality constraints and problems with block structure.

On the empirical side, we computed optimality certificates for instances from the MIPLIB and MINLPLib. Here it could be shown that often the optimality certificates are smaller than theory tells. In addition we investigated which properties of the problems are correlated with the existence of small optimality certificates, or at least with the ability of our algorithm to compute these.

There still are open questions regarding the optimality certificates. For example, we consider it possible to exploit the non-uniqueness of valid half-spaces corresponding to certificate points to generate even smaller optimality certificates.

In addition, there might be structures in optimization problems that can be exploited in order to obtain smaller certificates.

Furthermore the duality theory in Baes et al. (2016) can be generalized to use our optimality certificates and seems very intriguing.

On the computational side, the optimality certificates have the potential to provide useful dual information, which could speed up for example the branch-and-bound algorithm.

All of these questions are out of scope for this paper but open interesting possibilities for future work.

ACKNOWLEDGMENTS

LH, LS, and DW thank the Deutsche Forschungsgemeinschaft for their support within project A05 and B07 in the “Sonderforschungsbereich/Transregio 154 Mathematical Modelling, Simulation and Optimization using the Example of Gas Networks”. FR wants to thank the German Federal Ministry of Economic Affairs and Energy (BMWi) for the extensive financial support within the research project OPs-TIMAL. In addition, DW and KH want to thank SAP for their long-term financial and personal support.

REFERENCES

- Baes, M., T. Oertel, and R. Weismantel (2016). “Duality for mixed-integer convex minimization”. In: *Mathematical Programming* 158.1-2, pp. 547–564.
- Basu, A., M. Conforti, G. Cornuéjols, R. Weismantel, and S. Weltge (2017). “Optimality certificates for convex minimization and Helly numbers”. In: *Operations research letters* 45.6, pp. 671–674.
- Bertsimas, D. and R. Weismantel (2005). *Optimization over integers*. Vol. 13. Dynamic Ideas Belmont, MA.
- Bikhchandani, S. and J. W. Mamer (1997). “Competitive equilibrium in an exchange economy with indivisibilities”. In: *J. Econom. Theory* 74.2, pp. 385–413. DOI: [10.1006/jeth.1996.2269](https://doi.org/10.1006/jeth.1996.2269).
- Bikhchandani, S., J. M. Ostroy, et al. (2002). “The package assignment model”. In: *Journal of Economic theory* 107.2, pp. 377–406.
- Borndörfer, R., C. Ferreira, and A. Martin (1998). “Decomposing Matrices into Blocks”. In: *SIAM J. Optim.* 9.1, pp. 236–269.
- Bussieck, M. R., A. S. Drud, and A. Meeraus (2003). “MINLPLib—a collection of test models for mixed-integer nonlinear programming”. In: *INFORMS Journal on Computing* 15.1, pp. 114–119.
- Daskin, M. S. and K. L. Maass (2015). “The p-Median Problem”. In: *Location Science*. Ed. by G. Laporte, S. Nickel, and F. Saldanha da Gama. Cham: Springer International Publishing, pp. 21–45. DOI: [10.1007/978-3-319-13111-5_2](https://doi.org/10.1007/978-3-319-13111-5_2).
- Doignon, J.-P. (1973). “Convexity in crystallographical lattices”. In: *J. Geom.* 3, pp. 71–85. DOI: [10.1007/BF01949705](https://doi.org/10.1007/BF01949705).

- Fletcher, R. and S. Leyffer (1994). “Solving mixed integer nonlinear programs by outer approximation”. In: *Math. Programming* 66.3, Ser. A, pp. 327–349. DOI: [10.1007/BF01581153](https://doi.org/10.1007/BF01581153).
- Geißler, B., A. Morsi, L. Schewe, and M. Schmidt (2018). “Solving Highly Detailed Gas Transport MINLPs: Block Separability and Penalty Alternating Direction Methods”. In: *INFORMS Journal on Computing* 30.2, pp. 309–323. DOI: [10.1287/ijoc.2017.0780](https://doi.org/10.1287/ijoc.2017.0780).
- Gleixner, A., G. Hendel, G. Gamrath, T. Achterberg, M. Bastubbe, T. Berthold, P. M. Christophel, K. Jarck, T. Koch, J. Linderoth, M. Lübbecke, H. D. Mittelmann, D. Ozyurt, T. K. Ralphs, D. Salvagnin, and Y. Shinano (2021). “MIPLIB 2017: Data-Driven Compilation of the 6th Mixed-Integer Programming Library”. In: *Mathematical Programming Computation*. DOI: [10.1007/s12532-020-00194-3](https://doi.org/10.1007/s12532-020-00194-3).
- Gurobi Optimization, LLC (2021). *Gurobi Optimizer Reference Manual*. URL: <http://www.gurobi.com>.
- Jahn, J. and M. Knossalla (2018). “Lagrange theory of discrete-continuous nonlinear optimization”. In: *Journal of Nonlinear and Variational Analysis* 2.3, pp. 317–342.
- Konno, H. and R. Yamamoto (2009). “Choosing the best set of variables in regression analysis using integer programming”. In: *Journal of Global Optimization* 44.2, pp. 273–282.
- Kronqvist, J., D. E. Bernal, A. Lundell, and I. E. Grossmann (2019). “A review and comparison of solvers for convex MINLP”. In: *Optimization and Engineering* 20.2, pp. 397–455.
- Kuhn, H. W. and A. W. Tucker (1951). “Nonlinear programming”. In: *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability, 1950*. University of California Press, Berkeley and Los Angeles, pp. 481–492.
- Martello, Silvano, and P. Toth (1990). *Knapsack problems: algorithms and computer implementations*. New York, NY, USA: John Wiley & Sons, Inc.
- Martin, A. (1999). *Integer Programs with Block Structure*. eng. Takustr. 7, 14195 Berlin. Regionales Rechenzentrum Erlangen (2021). *Woodcrest Cluster*. URL: <https://hpc.fau.de/systems-services/systems-documentation-instructions/clusters/woody-cluster/> (visited on 11/12/2021).
- Reznick, B. (1986). “Lattice point simplices”. In: *Discret. Math.* 60, pp. 219–242. DOI: [10.1016/0012-365X\(86\)90015-4](https://doi.org/10.1016/0012-365X(86)90015-4).
- Rockafellar, R. T. (1970). *Convex Analysis*. Princeton University Press. URL: <http://www.jstor.org/stable/j.ctt14bs1ff>.
- Scarf, H. E. (1985). “Integral Polyhedra in Three Space”. In: *Math. Oper. Res.* 10.3, pp. 403–438. DOI: [10.1287/moor.10.3.403](https://doi.org/10.1287/moor.10.3.403).
- Schewe, L., M. Schmidt, and D. Weninger (2020). “A Decomposition Heuristic for Mixed-Integer Supply Chain Problems”. In: *Operations Research Letters*. DOI: [10.1016/j.orl.2020.02.006](https://doi.org/10.1016/j.orl.2020.02.006).
- Sebő, A. (1999). “An Introduction to Empty Lattice Simplices”. In: *Integer Programming and Combinatorial Optimization*. Ed. by G. Cornuéjols, R. E. Burkard, and G. J. Woeginger. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 400–414.
- Serra, T. and R. J. O’Neil (2020). “MIPLIBing: Seamless benchmarking of mathematical optimization problems and metadata extensions”.

(K. Halbig) FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG, DISCRETE OPTIMIZATION, CAUER-STR. 11, 91058 ERLANGEN, GERMANY
 Email address: katrin.halbig@fau.de

(L. Hümbes) FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG, DISCRETE OPTIMIZATION, CAUER-STR. 11, 91058 ERLANGEN, GERMANY
 Email address: lukas.huembs@fau.de

(F. Rösel) FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG, DISCRETE OPTIMIZATION, CAUER-STR. 11, 91058 ERLANGEN, GERMANY

Email address: `florian.roesel@fau.de`

(L. Schewe) UNIVERSITY OF EDINBURGH, SCHOOL OF MATHEMATICS AND MAXWELL INSTITUTE FOR MATHEMATICAL SCIENCES, JAMES CLERK MAXWELL BUILDING, KING'S BUILDINGS, PETER GUTHRIE TAIT ROAD, EDINBURGH, EH9 3FD, UK

Email address: `lars.schewe@ed.ac.uk`

(D. Weninger) FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG, DISCRETE OPTIMIZATION, CAUER-STR. 11, 91058 ERLANGEN, GERMANY

Email address: `dieter.weninger@fau.de`