

Schwientek, Jan; Seidel, Tobias; Küfer, Karl-Heinz

Article — Published Version

A transformation-based discretization method for solving general semi-infinite optimization problems

Mathematical Methods of Operations Research

Provided in Cooperation with:

Springer Nature

Suggested Citation: Schwientek, Jan; Seidel, Tobias; Küfer, Karl-Heinz (2020) : A transformation-based discretization method for solving general semi-infinite optimization problems, Mathematical Methods of Operations Research, ISSN 1432-5217, Springer, Berlin, Heidelberg, Vol. 93, Iss. 1, pp. 83-114,
<https://doi.org/10.1007/s00186-020-00724-8>

This Version is available at:

<https://hdl.handle.net/10419/288377>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by/4.0/>



A transformation-based discretization method for solving general semi-infinite optimization problems

Jan Schwientek¹ · Tobias Seidel¹ · Karl-Heinz Küfer¹

Received: 6 August 2019 / Revised: 26 June 2020 / Published online: 22 September 2020
© The Author(s) 2020

Abstract

Discretization methods are commonly used for solving standard semi-infinite optimization (SIP) problems. The transfer of these methods to the case of general semi-infinite optimization (GSIP) problems is difficult due to the \mathbf{x} -dependence of the infinite index set. On the other hand, under suitable conditions, a GSIP problem can be transformed into a SIP problem. In this paper we assume that such a transformation exists globally. However, this approach may destroy convexity in the lower level, which is very important for numerical methods. We present in this paper a solution approach for GSIP problems, which cleverly combines the above mentioned two techniques. It is shown that the convergence results for discretization methods in the case of SIP problems can be transferred to our *transformation-based discretization method* under suitable assumptions on the transformation. Finally, we illustrate the operation of our approach as well as its performance on several examples, including a problem of volume-maximal inscription of multiple variable bodies into a larger fixed body, which has never before been considered as a GSIP test problem.

Keywords Semi-infinite optimization · Discretization · Coordinate transformation · Design centering · Inscribing

Mathematics Subject Classification 90C34 · 90C30 · 65K05

✉ Jan Schwientek
Jan.Schwientek@itwm.fraunhofer.de

¹ Fraunhofer Institute for Industrial Mathematics (ITWM), Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany

1 Introduction

In the present paper we consider *general semi-infinite optimization problems* of the following form:

$$\begin{aligned} \text{GSIP : } & \min_{\mathbf{x} \in M} f(\mathbf{x}) \\ & \text{with } M := \{\mathbf{x} \in X \subseteq \mathbb{R}^m \mid g_i(\mathbf{x}, \mathbf{y}) \leq 0 \text{ for all } \mathbf{y} \in Y(\mathbf{x}), i \in I\}, \end{aligned}$$

where $Y : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$ is a set-valued mapping, $|Y(\mathbf{x})| = \infty$ for at least some $\mathbf{x} \in X$, $I := \{1, \dots, p\}$ is a finite index set and $f, g_i, i \in I$, are real-valued and at least continuous functions. By X we summarize the finite restrictions on the decision variables. Moreover, we assume that X is non-empty and closed and for every $\mathbf{x} \in X$ the so-called *infinite index set* $Y(\mathbf{x})$ is closed and compact.

The infinite index set can, e.g., be given as the solution set of finitely many inequalities:

$$Y(\mathbf{x}) := \{\mathbf{y} \in \mathbb{R}^n \mid u_j(\mathbf{x}, \mathbf{y}) \leq 0, j \in J\},$$

where $J := \{1, \dots, q\}$ is a finite index set and $u_j, j \in J$, are again real-valued and at least continuous functions. For the special case, where $Y(\mathbf{x})$ does not depend on \mathbf{x} , the problem above is called a *standard* or *ordinary* semi-infinite optimization problem and is abbreviated by SIP.

One of the keys for both the theoretical and numerical treatment of semi-infinite optimization problems lies in their bi-level structure. The parametric *lower level problems* of a semi-infinite problem are given by

$$Q_i(\mathbf{x}) : \max_{\mathbf{y} \in \mathbb{R}^n} g_i(\mathbf{x}, \mathbf{y}) \text{ s.t. } \mathbf{y} \in Y(\mathbf{x}). \quad (1)$$

The decision variables \mathbf{x} of the semi-infinite problem are the parameters of the lower level problems and the index variables \mathbf{y} of the semi-infinite problem are the decision variables of these problems. By

$$\varphi_i(\mathbf{x}) := \max_{\mathbf{y} \in Y(\mathbf{x})} g_i(\mathbf{x}, \mathbf{y})$$

we denote the so-called *optimal value functions* of the semi-infinite optimization problem. Obviously,

$$M = \{\mathbf{x} \in X \mid \varphi_i(\mathbf{x}) \leq 0, i \in I\},$$

but the functions $\varphi_i, i \in I$, are only given implicitly and may be non-differentiable even if the functions $g_i, i \in I$, and $u_j, j \in J$, are linear.

A natural and widespread solution technique for standard semi-infinite optimization problems is discretizing the infinite index set Y , solving the induced finite optimization problem, and refining the discretization (see e.g. Blankenship and Falk 1976; Reemtsen

1991; Reemtsen and Goerner 1998; López and Still 2007). The direct transfer of this approach to the case of a GSIP problem is possible in principle, but problematic: One problem in this case is the \mathbf{x} -dependency of the infinite index set. Thus, the discretization has to depend on \mathbf{x} , too. To ensure the closedness of the feasible set of the discretization-induced optimization problems, the discretizations must even depend at least continuously on \mathbf{x} . How such a discretization can be constructed is shown in Still (2001b). Due to additional conditions (see Still 2001b), which must be met for convergence of these methods in the case of GSIP problems, they are difficult to implement. And, to our knowledge, there are no numerical experiences with these discretization methods so far.

On the other hand, under suitable assumptions, any general semi-infinite optimization problem can be at least locally converted into a standard one (see Weber 1999; Still 1999). Such a transformation is of practical value only in cases where it is defined globally. We assume in the following that such a global transformation is given. For problems which have a geometric background, i.e. the infinite index set is a geometrically simple body or arises from geometrically simple bodies via intersection or union, such globally defined transformations can often be specified (see Floudas and Stein 2007; Steuermann 2011). Furthermore, these examples often have convex lower level problems. Unfortunately, the transfer to a SIP can destroy convexity of these problems. Then, heavy-weighted methods of global optimization must be applied to solve the resulting SIP (see Steuermann 2011).

We present in this paper a numerical method for solving general semi-infinite optimization problems, which can maintain convex lower-level problems. It combines the transformation into a standard semi-infinite problem with discretization techniques for such problems. In the algorithm we switch between two problems: the original GSIP problem and the SIP problem induced by the transformation. Starting from a given initial point, an initial discretization of the infinite index set of the GSIP problem is calculated and transferred into a discretization of the infinite index set of the induced SIP problem. Subsequently, a sequence of discretized SIP problems is solved, in which the discretization is refined from step to step. However, the lower level problems of the induced SIP problem are not solved for refinement, rather the lower level problems of the GSIP problem are solved. Then, the global solutions of these lower level problems are transferred into points of the index set of the induced SIP problem. It can be shown that the latter ones are global solutions, too. This is necessary for the correctness of the discretization method.

Numerical examples show that one only needs to add relatively few points to achieve feasibility within a given tolerance. Therefore this method is also suited for complex problems like the inscription of multiple bodies into a fixed design.

Early convergence results for discretization methods solving SIP problems are for example given by Reemtsen (1991) and Reemtsen and Goerner (1998). These results focus on the convergence to feasible points and the convergence of global solutions. Results regarding convergence rates of local solutions were presented by Still (2001a). We will show in Sect. 3 how these results carry over to the transformation-based discretization method. Moreover, we add an assumption to the main statement in Still (2001a) and present a counterexample for the case that the assumption is not fulfilled. This assumption is missing in Still (2001a).

A different reformulation approach to a SIP problem and subsequent discretization is followed by Mitsos and Tsoukalas (2015). There, a GSIP is reformulated using disjunctive semi-infinite constraints. The problem is then used to construct better upper and lower bounds for the original problem iteratively. This is done by a discretization strategy which adaptively adds discretization points. The disjunctive programs resulting from the discretization are solved globally using a solver for mixed integer nonlinear problems. In Djelassi et al. (2019) the strategy is extended to problems with equality constraints in the lower level. Instead of using a reformulation to a mixed integer program, in Kirst and Stein (2019) the authors develop tailored strategies to solve the disjunctive problems. An early attempt to globally solve GSIP problems was followed by Lemonidis (2008). In this work an inner and an outer approximation of the variable index set is calculated which doesn't depend on the optimization variables anymore. Then, the same techniques as presented in Bhattacharjee et al. (2005) are applied to the standard semi-infinite optimization problems. These techniques provide lower and upper bounds, which can be used for global optimization. Furthermore, in Lemonidis (2008) it is shown for the SIP application kinetic model reduction (KMR) how GSIP problems with box-shaped infinite index sets can be transferred into an equivalent SIP problem by a linear coordinate transformation (convex combination of the lower and upper bounds).

Our focus is different from these approaches. We try to avoid a reformulation using disjunctive constraints. Our approach allows to maintain helpful structure, e.g. convexity of the lower level problems, which comes with the GSIP problem. And, we try to avoid solving problems by global solvers.

The paper is organized as follows. In Sect. 2 we show how a GSIP problem can be reformulated into a SIP one. We point out why it is beneficial to solve the lower level problem of the given GSIP. Then, we introduce the algorithm. Section 3 contains results about convergence to a feasible point or global solutions, along with rates of convergence. Section 4 discusses the conditions necessary to retain convexity of the induced SIP lower level problems and of the discretized SIP problems. In Sect. 5 we illustrate the application of our method and its performance on three classes of examples: some native GSIP problems, a design centering problem, and a problem of inscribing two (or more) variable bodies volume-maximal into a larger fixed container. Section 6 concludes the paper and points out future directions of research.

2 The transformation-based discretization method (TDM)

Transformation: In principle, under suitable assumptions, each GSIP problem can be transformed, at least *locally*, into an equivalent SIP problem (see Still 1999; Weber 1999 for details). However, such a transformation is only of practical value in cases where this transformation is defined *globally*. The ideal situation is the following:

Assumption 1 (*Existence of transformation*) There exist a nonempty, compact set $Z \subseteq \mathbb{R}^{\bar{n}}$ and a continuous function $\mathbf{t} : X \times Z \rightarrow \mathbb{R}^n$, such that $\mathbf{t}(\mathbf{x}, Z) = Y(\mathbf{x})$ for all $\mathbf{x} \in X$.

Given Assumption 1, for every $i \in I$, the feasibility condition

$$g_i(\mathbf{x}, \mathbf{y}) \leq 0 \text{ for all } \mathbf{y} \in Y(\mathbf{x}),$$

can be equivalently written as

$$\tilde{g}_i(\mathbf{x}, \mathbf{z}) := g_i(\mathbf{x}, \mathbf{t}(\mathbf{x}, \mathbf{z})) \leq 0 \text{ for all } \mathbf{z} \in Z, \tag{2}$$

which is of ordinary semi-infinite type. Thus, the feasible sets of the GSIP problem and the induced SIP problem coincide. For the rest of the paper we assume that Assumption 1 is fulfilled.

For an interval, or more general box-shaped, index set $Y(\mathbf{x}) = [a(\mathbf{x}), b(\mathbf{x})]$ with $a(\cdot) < b(\cdot)$, such a transformation can be easily constructed by convex combination of the interval ends resp. box vertices. In the case of higher dimensional index sets such a transformation exists for example, if they are star-shaped (see Still 1999 for details). But often the application itself already suggests such a transformation [see for example the design centering problems in Steuermann (2011)].

Remark 1 In general, it won't be possible to parametrize the set $Y(\mathbf{x})$ by one function. However, it is sufficient for the following considerations, that the set $Y(\mathbf{x})$ can be partitioned such that for every subset $Y_l(\mathbf{x}), l \in \{1, \dots, s\}$, a set Z_l and a function \mathbf{t}_l according to Assumption 1 exist. The results and the algorithm presented in the following carry over to this case directly.

We denote the transformation-induced standard semi-infinite optimization problem by

$$\begin{aligned} \widetilde{\text{SIP}} : \quad & \min_{\mathbf{x} \in \widetilde{M}} f(\mathbf{x}) \\ & \text{with } \widetilde{M} := \{\mathbf{x} \in X \subseteq \mathbb{R}^m \mid \tilde{g}_i(\mathbf{x}, \mathbf{z}) \leq 0 \text{ for all } \mathbf{z} \in Z, i \in I\}, \end{aligned}$$

where the functions \tilde{g}_i are defined as in (2).

By

$$\tilde{Q}_i(\mathbf{x}) := \max_{\mathbf{z} \in Z} \tilde{g}_i(\mathbf{x}, \mathbf{z}), \quad i \in I,$$

and

$$\tilde{\varphi}_i(\mathbf{x}) := \max_{\mathbf{z} \in Z} \tilde{g}_i(\mathbf{x}, \mathbf{z}), \quad i \in I,$$

we denote the lower level problems and the optimal value functions of the (transformation-)induced SIP problem, respectively. We show in the following that the optimal value functions $\tilde{\varphi}_i$ and φ_i coincide for every $i \in I$.

Discretization: A widespread approach for solving SIPs is discretizing the infinite index set. Therefore, for a subset \dot{Z} of Z , we introduce the optimization problem

$$\begin{aligned} \widetilde{\text{SIP}}(\dot{Z}) : \quad & \min_{\mathbf{x} \in \widetilde{M}(\dot{Z})} f(\mathbf{x}) \\ & \text{with } \widetilde{M}(\dot{Z}) := \{\mathbf{x} \in X \subseteq \mathbb{R}^m \mid \tilde{g}_i(\mathbf{x}, \mathbf{z}) \leq 0 \text{ for all } \mathbf{z} \in \dot{Z}, i \in I\}. \end{aligned}$$

If $\dot{Z} \subseteq Z$ is a finite set, $\widetilde{\text{SIP}}(\dot{Z})$ is called *discretized SIP problem*.

Discretization methods which adaptively add points to the discretization are typically very successful. An algorithm for SIP problems using this methodology was first introduced by Blankenship and Falk (1976). The points which are added are the global solutions of the lower level problem. This is important for guaranteeing feasibility. Unfortunately, the transformation may destroy given convexity in the lower level problem (see Kaplan and Tichatschke 1997; Still 2001b). More precisely, if $g_i(\mathbf{x}, \mathbf{y})$ is convex in \mathbf{y} , $\tilde{g}_i(\mathbf{x}, \mathbf{z})$ can be non-convex in \mathbf{z} . We illustrate this effect by the following example.

Example 1 We consider a so-called squircle as infinite index set, i.e.

$$Y(\mathbf{x}) = \{\mathbf{y} \in \mathbb{R}^2 \mid (y_1 - x_1)^4 + (y_2 - x_2)^4 \leq x_3^4\},$$

with $\mathbf{x} \in X := \mathbb{R}^2 \times \mathbb{R}_{\geq 0}$. Furthermore, let $g_1(\mathbf{x}, \mathbf{y}) := y_2 - y_1^2 \leq 0$ be the semi-infinite constraint. Obviously, $Y(\mathbf{x})$ is a convex set for all \mathbf{x} and $g_1(\mathbf{x}, \mathbf{y})$ is concave in \mathbf{y} (for all \mathbf{x}). Hence, for all \mathbf{x} the (lower level) problem $Q_1(\mathbf{x}) : \max_{\mathbf{y} \in Y(\mathbf{x})} g_1(\mathbf{x}, \mathbf{y})$ is a convex one (see Fig. 1a for a graphical visualization). The representation of a circle as image of the set $[0, 1] \times [0, \pi]$ can be generalized to this case (see for example Jaklič et al. 2000):

$$\mathbf{t}_{\text{sq}}(\mathbf{x}, \mathbf{z}) = \begin{pmatrix} x_1 + z_1 x_3 \text{sgn}(\sin(2z_2)) |\sin(2z_2)|^{\frac{1}{2}} \\ x_2 + z_1 x_3 \text{sgn}(\cos(2z_2)) |\cos(2z_2)|^{\frac{1}{2}} \end{pmatrix}.$$

While $Z = [0, 1] \times [0, \pi]$ is also a convex set, the function

$$\begin{aligned} \tilde{g}_1(\mathbf{x}, \mathbf{z}) & := g_1(\mathbf{x}, \mathbf{t}(\mathbf{x}, \mathbf{z})) \\ & = x_1 + z_1 x_3 \text{sgn}(\sin(2z_2)) |\sin(2z_2)|^{\frac{1}{2}} \\ & \quad - (x_2 + z_1 x_3 \text{sgn}(\cos(2z_2)) |\cos(2z_2)|^{\frac{1}{2}})^2 \end{aligned}$$

is not concave in \mathbf{z} for all \mathbf{x} , e.g. not in $\mathbf{x} = (0.8, 0.5, 1)^T$ as one can see in Fig. 1b. There are two different local solutions for the transformed lower level problem $\widetilde{Q}_1(\mathbf{x}) : \max_{\mathbf{z} \in Z} \tilde{g}(\mathbf{x}, \mathbf{z})$. Hence, the lower level problems of $\widetilde{\text{SIP}}$ may be non-convex in general.

The loss of convexity can cause numerical difficulties in solving the lower level problems. In the worst case time-consuming global optimization routines have to be used to solve these problems. That’s why we consider the original convex lower level problems in the algorithm. Then, we can map these solutions to solutions of the

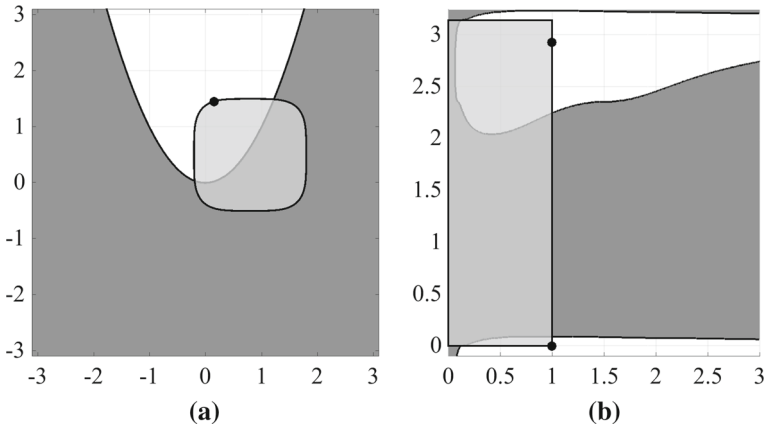


Fig. 1 Transformation destroys convexity in the lower level [dark gray - points with $g_1(x, y) \leq 0$ resp. $\tilde{g}_1(x, y) \leq 0$, light gray—feasible set $Y(x)$ resp. Z , circles - local solutions]; **a** original problem $Q_1(x)$ with unique optimal solution. **b** Transformed problem $\tilde{Q}_1(x)$ with two local solutions

possible non-convex lower level problems of $\widetilde{\text{SIP}}$.

The following lemma states that this is a feasible approach. From it also follows that the transformation does not change the feasible set defining functions $\varphi_i, i \in I$.

Lemma 1 Let $\mathbf{x} \in X$ and $i \in I$. The point $\mathbf{z}^* \in Z$ is a global solution of $\tilde{Q}_i(\mathbf{x})$ if and only if $\mathbf{y}^* = \mathbf{t}(\mathbf{x}, \mathbf{z}^*)$ is a global solution of $Q_i(\mathbf{x})$.

Proof Let \mathbf{y}^* be a global solution of $Q_i(\mathbf{x})$, i.e. $g_i(\mathbf{x}, \mathbf{y}^*) \geq g_i(\mathbf{x}, \mathbf{y})$ for all $\mathbf{y} \in Y(\mathbf{x})$. This relation is under $\mathbf{y}^* = \mathbf{t}(\mathbf{x}, \mathbf{z}^*)$ equivalent to

$$\tilde{g}_i(\mathbf{x}, \mathbf{z}^*) = g_i(\mathbf{x}, \mathbf{t}(\mathbf{x}, \mathbf{z}^*)) \geq g_i(\mathbf{x}, \mathbf{t}(\mathbf{x}, \mathbf{z})) = \tilde{g}_i(\mathbf{x}, \mathbf{z})$$

for all $\mathbf{z} \in Z$. Thus, \mathbf{z}^* is a global solution of $\tilde{Q}_i(\mathbf{x})$. The reverse applies under analogous considerations and the surjectivity of \mathbf{t} . \square

The above lemma states that one can calculate a global solution of the generally non-convex problem $\tilde{Q}_i(\mathbf{x})$ by computing a global solution of the convex problem $Q_i(\mathbf{x})$ and mapping this one via the transformation $\mathbf{t}(\mathbf{x}, \cdot)$ into Z . Using an adaptive discretization as in Blankenship and Falk (1976), these considerations lead to the following algorithm:

Algorithm 1 Transformation-based discretization method (TDM) for GSIP problems

- 1: Choose a starting point $\mathbf{x}^0 \in X$ and a tolerance $\alpha \geq 0$.
- 2: Choose/Compute an initial discretization $\dot{Y}_0(\mathbf{x}^0) \subseteq Y(\mathbf{x}^0)$ and determine \dot{Z}_0 such that $\mathbf{t}(\mathbf{x}^0, \dot{Z}_0) = \dot{Y}_0(\mathbf{x}^0)$.
- 3: Set $k = 0$.
- 4: **repeat**
- 5: Compute a solution \mathbf{x}^{k+1} of the discretized SIP problem $\widetilde{\text{SIP}}(\dot{Z}_k)$ using \mathbf{x}^k as starting point.
- 6: Let $Z_{new} = \emptyset$.
- 7: **for** $i \in I$ **do**
- 8: Compute a global solution \mathbf{y}_i^{k+1} of $Q_i(\mathbf{x}^{k+1})$.
- 9: **if** $g_i(\mathbf{x}^{k+1}, \mathbf{y}_i^{k+1}) > \alpha$ **then**
- 10: Determine \mathbf{z}_i^{k+1} such that $\mathbf{t}(\mathbf{x}^{k+1}, \mathbf{z}_i^{k+1}) = \mathbf{y}_i^{k+1}$ and set
 $\dot{Z}_{new} = \dot{Z}_{new} \cup \{\mathbf{z}_i^{k+1}\}$.
- 11: **end if**
- 12: **end for**
- 13: Set $\dot{Z}_{k+1} = \dot{Z}_k \cup \dot{Z}_{new}$ and replace k by $k + 1$.
- 14: **until** $\max_{i \in I} \{g_i(\mathbf{x}^k, \mathbf{y}_i^k)\} \leq \alpha$.
- 15: **return** $\mathbf{x}_\alpha^* := \mathbf{x}^k$.

If, in Step 2, an initial discretization $\dot{Y}_0(\mathbf{x}^0)$ of $Y(\mathbf{x}^0)$ is not at hand, one can be obtained by solving the lower level problems and transferring the solutions (see Steps 8 and 10). A feasible starting point for Step 8 can be calculated using the transformation $\mathbf{t}(\mathbf{x}, \cdot)$ from any feasible point of Z .

While we demand to calculate a global solution in Step 8, we can either calculate a global or a local solution in Step 5. As we will see in the proof of Theorem 1, the global solution of the lower level problem is important to guarantee feasibility. The type of solution of the discretized problems determines the type of solution calculated for the GSIP problem.

If the feasible set of $\widetilde{\text{SIP}}(\dot{Z}_k)$ becomes empty, also the feasible set M of the GSIP problem is empty and Algorithm 1 can be stopped.

With regard to one of the main applications of semi-infinite optimization, the inscribing of one or multiple designs, the infinite index sets are often simple geometric objects, such as (hyper)ellipsoids or (hyper)boxes, but also objects composed of geometrically simple objects. For these, (re-)parameterization, e.g. in the form of polar, cylindrical or spherical coordinates, are often well known. On the other hand, infinite index sets, especially in this context, can often be written as a translation, rotation, or scaling of fixed objects. Hence, the transformation is known, too. Finally, under suitable conditions, there are “generic” transformations. For example, any star-shaped set can be described by the star center and a radius (function) depending on the distance to the center and the direction. In Still (1999) conditions are stated under which such a constructive transformation exists for infinite index sets.

3 Convergence results

There is a huge variety of convergence results for adaptive discretization methods solving ordinary semi-infinite optimization problems. The goal of this section is to show that the results can be carried over to the transformation-based discretization method under suitable assumptions on the transformation.

We start by assuming that we can guarantee finding global solutions in Step 5 of Algorithm 1. Similar results regarding the convergence in the case of ordinary semi-infinite optimization problems can be found for example in Blankenship and Falk (1976), Reemtsen (1991) and Reemtsen and Goerner (1998). The next theorem summarizes the convergence properties for GSIP problems.

Theorem 1 *Let the (initial) feasible set $\tilde{M}(\dot{Z}_0)$ be compact and the feasible set M of GSIP be non-empty. Let the tolerance $\alpha = 0$. Then, either Algorithm 1 terminates after a finite number of steps, or it holds:*

- (i) *The problems GSIP and $\widetilde{\text{SIP}}(\dot{Z}_k), k \in \mathbb{N}_0$, have global solutions $\mathbf{x}^* \in M$ and $\mathbf{x}^{k+1} \in \tilde{M}(\dot{Z}_k), k \in \mathbb{N}_0$.*
- (ii) *The sequence $\{\mathbf{x}^k\}_{k \in \mathbb{N}_0}$ has at least one accumulation point and each such accumulation point is a global solution of GSIP.*
- (iii) *If $\mathbf{x}^* \in M$ is a unique global solution of GSIP, then it holds*

$$\lim_{k \rightarrow \infty} \|\mathbf{x}^k - \mathbf{x}^*\|_2 = 0.$$

- (iv) *For a global solution \mathbf{x}^* of GSIP it holds $f(\mathbf{x}^k) \leq f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^*), k \in \mathbb{N}_0$, and $\lim_{k \rightarrow \infty} f(\mathbf{x}^k) = f(\mathbf{x}^*)$.*

Proof For the discretizations $\dot{Z}_k, k \in \mathbb{N}_0$, of Z calculated by Algorithm 1, the following relation is valid:

$$\dot{Z}_0 \subseteq \dots \subseteq \dot{Z}_k \subseteq \dot{Z}_{k+1} \subseteq \dots \subseteq Z.$$

Hence, it holds for the feasible sets of the discretized SIP-problems $\widetilde{\text{SIP}}(\dot{Z}_k), k \in \mathbb{N}_0$, and the feasible set of $\widetilde{\text{SIP}}$:

$$\tilde{M}(\dot{Z}_0) \supseteq \dots \supseteq \tilde{M}(\dot{Z}_k) \supseteq \tilde{M}(\dot{Z}_{k+1}) \supseteq \dots \supseteq \tilde{M} = M. \tag{3}$$

The functions $\tilde{g}_i, i \in I$, are continuous as compositions of continuous functions and, hence, the sets $\tilde{M}(\dot{Z}_k), k \in \mathbb{N}_0$, and \tilde{M} are closed sets. Because of (3) they are bounded and, thus, are compact. Because of its continuity the function f takes its minimum on the sets $M = \tilde{M}$ and $\tilde{M}(\dot{Z}_k), k \in \mathbb{N}_0$. Hence, assertion (i) is shown.

For (ii) note that \mathbf{x}^k and $\mathbf{z}_i^k, i \in I$, are sequences in $\tilde{M}(\dot{Z}_0)$ and Z respectively. Both sets are compact by assumption. Hence, there is at least one accumulation point. For every converging subsequence $\{\mathbf{x}^{k_l}\}_{l \in \mathbb{N}}$ we can choose a subsequence such that $\mathbf{z}_i^{k_l}, i \in I$ is convergent as well. Using continuity of \tilde{g}_i and optimality of $\mathbf{z}_i^{k_l}$ for $\tilde{Q}_i(\mathbf{x}^{k_l})$,

we have for $i \in I$ and $\mathbf{z} \in Z$

$$\tilde{g}_i \left(\lim_{l \rightarrow \infty} \mathbf{x}^{k_l}, \mathbf{z} \right) \leq \lim_{l \rightarrow \infty} \tilde{g}_i \left(\mathbf{x}^{k_l}, \mathbf{z}_i^{k_l} \right) = \tilde{g}_i \left(\lim_{l \rightarrow \infty} \mathbf{x}^{k_{l+1}}, \lim_{l \rightarrow \infty} \mathbf{z}_i^{k_l} \right) \leq 0.$$

Thus, every accumulation point is feasible. Moreover, because $\mathbf{x} \in M \Rightarrow \mathbf{x} \in \tilde{M}(\dot{Z}_{k_l})$ by (3), we also have for $\mathbf{x} \in M$,

$$f \left(\lim_{l \rightarrow \infty} \mathbf{x}^{k_l} \right) = \lim_{l \rightarrow \infty} f \left(\mathbf{x}^{k_l} \right) \leq f(\mathbf{x}),$$

which shows optimality.

The remaining assertions (iii) and (iv) follow easily by (ii) and relation (3). □

If the $\widetilde{\text{SIP}}$ -defining functions f and $\tilde{g}_i, i \in I$, are non-convex in \mathbf{x} , it can be costly to guarantee the calculation of global solutions. In the literature there also exist results regarding the case, that only local solutions can be calculated (see, e.g., Reemtsen 1994). The same techniques can be used here. Therefore, assume that for every $k \in \mathbb{N}_0$, the current iterate \mathbf{x}^k is a local minimizer of $\widetilde{\text{SIP}}(\dot{Z}_k)$ with radius $r_k > 0$, i.e. for all \mathbf{x} in $\tilde{M}(\dot{Z}_k)$ with $\|\mathbf{x} - \mathbf{x}^k\| < r_k$ the following holds:

$$f(\mathbf{x}^k) \leq f(\mathbf{x}).$$

If the radii do not converge towards zero, i.e.

$$r := \inf_{k \in \mathbb{N}_0} r_k > 0,$$

then every accumulation point \mathbf{x}^* of the sequence $\{\mathbf{x}^k\}_{k \in \mathbb{N}_0}$ is a local minimizer of GSIP. This can be proven by restricting the analysis to a ball of radius $r/2$ around \mathbf{x}^* . For large k , the current iterates are global solutions within this ball.

These results and the assertions of Theorem 1 hold only for a limit point. We have introduced a parameter α as a finite termination criterion. The next Lemma shows that this is a valid approach.

Lemma 2 *Again let the (initial) feasible set $\tilde{M}(\dot{Z}_0)$ be compact. For a tolerance $\alpha > 0$ Algorithm 1 terminates after finitely many steps.*

Proof Assume that the contrary is the case. In this case in every iteration the algorithm has to add at least one point to the discretization. This means that there is an $i \in I$ and sequences $\{\mathbf{x}^k\}_{k \in \mathbb{N}_0} \subseteq \tilde{M}(\dot{Z}_0), \{\mathbf{z}^k\}_{k \in \mathbb{N}_0} \subseteq Z$ with:

$$\tilde{g}_i(\mathbf{x}^k, \mathbf{z}^k) \geq \alpha \tag{4}$$

and for all $\hat{k} > k$:

$$\tilde{g}_i(\mathbf{x}^{\hat{k}}, \mathbf{z}^k) \leq 0. \tag{5}$$

As Z and $\tilde{M}(\dot{Z}_0)$ are both compact there are convergent subsequences with:

$$\begin{aligned} \lim_{l \rightarrow \infty} \mathbf{x}^{k_l} &= \mathbf{x}^* \in \tilde{M}(\dot{Z}_0). \\ \lim_{l \rightarrow \infty} \mathbf{z}^{k_l} &= \mathbf{z}^* \in Z. \end{aligned}$$

Because t is continuous, also \tilde{g}_i is continuous. Together with (4) and (5) we have

$$\begin{aligned} \tilde{g}_i(\mathbf{x}^*, \mathbf{z}^*) &= \tilde{g}_i(\lim_{l \rightarrow \infty} \mathbf{x}^{k_l}, \lim_{l \rightarrow \infty} \mathbf{z}^{k_l}) \\ &= \lim_{l \rightarrow \infty} \tilde{g}_i(\mathbf{x}^{k_l}, \mathbf{z}^{k_l}) \\ &\geq \alpha \\ &> 0 \\ &\geq \lim_{l \rightarrow \infty} \tilde{g}_i(\mathbf{x}^{k_{l+1}}, \mathbf{z}^{k_l}) \\ &= \tilde{g}_i(\lim_{l \rightarrow \infty} \mathbf{x}^{k_{l+1}}, \lim_{l \rightarrow \infty} \mathbf{z}^{k_l}) \\ &= \tilde{g}_i(\mathbf{x}^*, \mathbf{z}^*), \end{aligned}$$

which is a contradiction. Hence, no such sequences can exist and the algorithm terminates after a finite number of steps. □

The following question arises: After k steps, what is the distance between the obtained solution and a local solution of the GSIP problem. In Still (2001a) the distance is bounded for a sequence of solutions for finer discretizations. The transfer of these results will be the topic of the remainder of this section.

To do so, we have to strengthen the initial assumptions. First, in Assumption 2 we assume that the iterates converge towards a strict local minimizer. Second, in Assumption 3 we assume some differentiability of the functions $g_i, i \in I$, and \mathbf{t} as well as an extension of the Mangasarian-Fromovitz Constraint Qualification (MFCQ). Finally, we assume in Theorem 2 that the objective function is locally Lipschitz continuous.

A point $\mathbf{x}^* \in M$ is called a *strict local minimizer of order $\rho = 1$ or $\rho = 2$* , if there is a $q > 0$ and a neighborhood U such that for every $\mathbf{x} \in M \cap U$

$$q \|\mathbf{x}^* - \mathbf{x}\|_2^\rho \leq f(\mathbf{x}) - f(\mathbf{x}^*)$$

holds.

We now assume that the sequence generated by Algorithm 1 converges to a local solution:

Assumption 2 Let \mathbf{x}^* be a strict local minimizer of order $\rho = 1$ or $\rho = 2$ of GSIP in the neighborhood U and $\{\mathbf{x}^k\}_{k \in \mathbb{N}_0}$ a sequence of local minimizers with radius r_k calculated by the transformation-based discretization method (TDM). Furthermore, assume that $\|\mathbf{x}^k - \mathbf{x}^*\|_2 \rightarrow 0$ for $k \rightarrow \infty$ and

$$\inf_{k \in \mathbb{N}_0} r_k > 0. \tag{6}$$

The assumption presented here is very similar to the main assumption made in Still (2001a). The difference is the assumption of non-vanishing radii (6) of the local minimizers. As we will show in Sect. 3.1 this is a necessary assumption.

For the construction of a feasible point a MFCQ-like condition for $\widetilde{\text{SIP}}$ is needed. Therefore, we ask under which conditions, the functions $D_1\tilde{g}_i, i \in I$, are continuous on $U_{\mathbf{x}^*} \times Z$, where $U_{\mathbf{x}^*}$ is a neighborhood of \mathbf{x}^* .

Lemma 3 *Let the functions $g_i(\mathbf{x}, \mathbf{y}), i \in I$, be differentiable w.r.t. (\mathbf{x}, \mathbf{y}) and $\mathbf{t}(\mathbf{x}, \mathbf{z})$ be differentiable with respect to its first argument \mathbf{x} . Let $U_{\mathbf{x}^*}$ be a neighborhood of \mathbf{x}^* . We assume that*

- (i) *the functions $Dg_i(\mathbf{x}, \mathbf{y}), i \in I$, are continuous on $U_{\mathbf{x}^*} \times \mathbb{R}^n$ and*
- (ii) *the function $D_1\mathbf{t}(\mathbf{x}, \mathbf{z})$ is continuous on $U_{\mathbf{x}^*} \times Z$.*

Then, the functions $\tilde{g}_i, i \in I$, are differentiable w.r.t. \mathbf{x} around \mathbf{x}^ and this derivative is given by:*

$$D_1\tilde{g}_i(\mathbf{x}, \mathbf{z}) = D_1g_i(\mathbf{x}, \mathbf{t}(\mathbf{x}, \mathbf{z})) + D_2g_i(\mathbf{x}, \mathbf{t}(\mathbf{x}, \mathbf{z})) \cdot D_1\mathbf{t}(\mathbf{x}, \mathbf{z}),$$

which is continuous on $U_{\mathbf{x}^} \times Z$.*

Before introducing Theorem 2, we collect the required conditions in an assumption.

Assumption 3 *Let the functions $g_i(\mathbf{x}, \mathbf{y}), i \in I$, be differentiable w.r.t. (\mathbf{x}, \mathbf{y}) and $\mathbf{t}(\mathbf{x}, \mathbf{z})$ be differentiable in its first argument \mathbf{x} . Assume, that there exists a neighborhood $U_{\mathbf{x}^*}$ of \mathbf{x}^* such that:*

- (i) *The functions $Dg_i(\mathbf{x}, \mathbf{y}), i \in I$, are continuous on $U_{\mathbf{x}^*} \times \mathbb{R}^n$.*
- (ii) *The function $D_1\mathbf{t}(\mathbf{x}, \mathbf{z})$ is continuous on $U_{\mathbf{x}^*} \times Z$.*
- (iii) *There exists a vector $\xi \in \mathbb{R}^m$ such that for all $i \in I$:*

$$D_1\tilde{g}_i(\mathbf{x}^*, \mathbf{z}) \cdot \xi \leq -1 \text{ for all } \mathbf{z} \in Z_0^i(\mathbf{x}^*),$$

where $Z_0^i(\mathbf{x}^*) = \{\mathbf{z} \in Z \mid \tilde{g}_i(\mathbf{x}^*, \mathbf{z}) = 0\}$ denotes the set of active indices.

For a sequence $\{\mathbf{x}^k\}_{k \in \mathbb{N}_0}$ calculated by Algorithm 1 we denote the current violation for $k \in \mathbb{N}_0$ by:

$$\alpha_k = \max_{i \in I} \max_{z \in Z} \tilde{g}_i(\mathbf{x}^k, \mathbf{z}).$$

Now, we can bound the difference of the current iterate to the limit point.

Theorem 2 *Let Assumptions 2 and 3 hold and the objective function f of GSIP be Lipschitz-continuous near \mathbf{x}^* .*

- (i) *There is a constant $c_1 > 0$ and a $k' \in \mathbb{N}_0$ such that*

$$0 \leq f(\mathbf{x}^*) - f(\mathbf{x}^k) \leq c_1\alpha_k \text{ for all } k \geq k'.$$

(ii) *There is a constant $c_2 > 0$ and a $k'' \in \mathbb{N}_0$ such that*

$$\|\mathbf{x}^* - \mathbf{x}^k\|_2 \leq c_2 \alpha_k^{1/\rho} \text{ for all } k \geq k''.$$

We have added Eq. 6 to the assumptions. In Sect. 3.1 we will show that this additional assumption is needed and the statement is wrong without this assumption. That’s why we give a complete proof and don’t reduce the statement to the statements in Still (2001a)

Proof Construction of a feasible point: In a first step we construct a feasible point $\bar{\mathbf{x}}^k$. To do so, we move the current iterate towards feasibility: For $k \in \mathbb{N}_0$ and $t \in [0, 1]$ let

$$\bar{\mathbf{x}}^k(t) = \mathbf{x}^k + t \cdot 2\alpha_k \boldsymbol{\xi},$$

where $\boldsymbol{\xi}$ is chosen according to Assumption 3.iii. We show next that the point $\bar{\mathbf{x}}^k(1)$ is feasible. Firstly, we consider indices $\mathbf{z} \in Z$ which are close to an active index.

For every $i \in I$ the function $D_1 \tilde{g}_i(\cdot, \cdot)$ is by Assumption 3 continuous and Z is compact. Thus, there is some ε such that for every $i \in I$ it holds

$$D_1 g_i(\mathbf{x}, \mathbf{z}) \cdot \boldsymbol{\xi} \leq -\frac{1}{2},$$

for $\mathbf{z} \in Z_0^{i,\varepsilon}(\mathbf{x}^*)$ and $\|\mathbf{x} - \mathbf{x}^*\|_2 < \varepsilon$, where

$$Z_0^{i,\varepsilon}(\mathbf{x}^*) := \{\mathbf{z} \in Z \mid \exists \mathbf{z}^0 \in Z_0^i(\mathbf{x}^*) \text{ with } \|\mathbf{z} - \mathbf{z}^0\|_2 < \varepsilon\}.$$

Choose k_1 large enough such that

$$\|\mathbf{x}^k - \mathbf{x}^*\|_2 + 2\alpha_k \|\boldsymbol{\xi}\|_2 \leq \varepsilon$$

for $k \geq k_1$. By the mean value theorem there is for every $i \in I$, $\mathbf{z} \in Z_0^{i,\varepsilon}(\mathbf{x}^*)$ and $k \geq k_1$ an $s \in [0, 1]$ such that

$$\tilde{g}_i(\bar{\mathbf{x}}^k(1), \mathbf{z}) = \tilde{g}_i(\mathbf{x}^k, \mathbf{z}) + D_1 \tilde{g}_i(\bar{\mathbf{x}}^k(s), \mathbf{z}) \cdot 2\alpha_k \boldsymbol{\xi} \leq \alpha_k - \frac{1}{2} \cdot 2\alpha_k = 0.$$

It remains to consider the indices which have a given distance to the active indices. As for $i \in I$, the set $Z \setminus Z_0^{i,\varepsilon}(\mathbf{x}^*)$ is compact and the functions $\tilde{g}_i(\cdot, \cdot)$ are continuous, there is a ε_2 such that

$$\tilde{g}_i(\mathbf{x}, \mathbf{z}) < 0,$$

for $\mathbf{z} \in Z \setminus Z_0^{i,\varepsilon}(\mathbf{x}^*)$ and $\|\mathbf{x} - \mathbf{x}^*\|_2 < \varepsilon_2$. Choose k_2 large enough such that

$$\|\mathbf{x}^k - \mathbf{x}^*\|_2 + 2\alpha_k \|\boldsymbol{\xi}\|_2 < \varepsilon_2$$

for $k \geq k_2$. Thus for $k \geq \max(k_1, k_2)$ the point $\bar{x}^k := \bar{x}^k(1)$ is feasible.

Now the two claims follow easily:

Proof of i): By Assumption 2 the radii r_k of the local minimizers \mathbf{x}^k don't converge to 0. Thus for large k the distance of the local solution \mathbf{x}^* to \mathbf{x}^k is smaller than r_k and we have

$$f(\mathbf{x}^*) \geq f(\mathbf{x}^k). \tag{7}$$

Again, for large k , the feasible point \bar{x}^k is contained in U . By the Lipschitz-continuity of f , there is a constant L such that for large k

$$0 \leq f(\mathbf{x}^*) - f(\mathbf{x}^k) \leq f(\bar{x}^k) - f(\mathbf{x}^k) \leq L \cdot \|\bar{x}^k - \mathbf{x}^k\|_2 = L \cdot 2\alpha_k \|\xi\|_2, \tag{8}$$

which shows the first assertion.

Proof of ii): By Assumption 2 the limit point \mathbf{x}^* is a local solution of order ρ . Thus for large k :

$$\begin{aligned} \|\mathbf{x}^* - \mathbf{x}^k\|_2 &\leq \|\mathbf{x}^* - \bar{x}^k\|_2 + \|\bar{x}^k - \mathbf{x}^k\|_2 \\ &\leq \left(\frac{f(\bar{x}^k) - f(\mathbf{x}^*)}{q} \right)^{\frac{1}{\rho}} + 2 \cdot \alpha_k \cdot \|\xi\|_2 \\ &\leq \left(\frac{f(\bar{x}^k) - f(\mathbf{x}^k)}{q} \right)^{\frac{1}{\rho}} + 2 \cdot \alpha_k \cdot \|\xi\|_2 \\ &\leq \left(\frac{L \cdot 2\alpha_k \|\xi\|_2}{q} \right)^{\frac{1}{\rho}} + 2 \cdot \alpha_k \cdot \|\xi\|_2 \\ &\leq \left(\left(\frac{L2\|\xi\|_2}{q} \right)^{\frac{1}{\rho}} + 2\|\xi\|_2 \right) \alpha_k^{\frac{1}{\rho}}, \end{aligned}$$

where we've used Eqs. (7) and (8) for the third and fourth inequality. □

3.1 Counterexample for rate of convergence

We have added the assumptions of non-vanishing radii (6) for the above convergence result. The following example shows that without this assumption the convergence rate may be arbitrarily bad.

We construct the example within two steps. First, we present a Lipschitz continuous function on $X = [-1, 1]^2$ with an infinite number of local solutions.

Therefore, let $\{\mathbf{c}^{(i)}\}_{i \in \mathbb{N}}$ with

$$\mathbf{c}^{(i)} = \left(\frac{1}{10^i}, \frac{1}{i} \right)^\top.$$

Around every $\mathbf{c}^{(i)}$ we choose a circle U_i with radius $r_i := \frac{1}{10^{i+1}}$:

$$U_i := \{\mathbf{x} \in X \mid \|\mathbf{x} - \mathbf{c}^{(i)}\|_2 < r_i\}.$$

Note that the circles U_i are disjoint. We introduce the objective function $f : X \rightarrow \mathbb{R}$ by:

$$f(\mathbf{x}) = \begin{cases} \|\mathbf{x}\|_2 + 2\|\mathbf{c}^{(i)} - \mathbf{x}\|_2 - 2r_i, & \text{for } \mathbf{x} \in U_i, i \in \mathbb{N}, \\ \|\mathbf{x}\|_2, & \text{otherwise.} \end{cases}$$

We summarize the properties of the objective function in the following lemma:

Lemma 4 *For the above construction, the following is true:*

- (i) *The function f is Lipschitz continuous.*
- (ii) *For every $i \in \mathbb{N}$ the center point \mathbf{c}_i is a strict local minimum of f with radius r_i .*

Proof (i) It is easy to see that we can write f for every $\mathbf{x} \in X$ as

$$f(\mathbf{x}) = \min\{\|\mathbf{x}\|_2, \min_{i \in \mathbb{N}} \|\mathbf{x}\|_2 + 2\|\mathbf{c}^{(i)} - \mathbf{x}\|_2 - 2r_i\}.$$

As every single function is Lipschitz continuous with Lipschitz constant 3, the minimum is again Lipschitz continuous.

- (ii) Consider for $i \in \mathbb{N}$ a point $\mathbf{x} \in U_i$ with $\mathbf{x} \neq \mathbf{c}^{(i)}$. The triangle inequality yields

$$\begin{aligned} f(\mathbf{x}) &= \|\mathbf{x}\|_2 + 2\|\mathbf{c}^{(i)} - \mathbf{x}\|_2 - 2r_i \\ &\geq \|\mathbf{c}^{(i)}\|_2 + \|\mathbf{c}^{(i)} - \mathbf{x}\|_2 - 2r_i \\ &> \|\mathbf{c}^{(i)}\|_2 - 2r_i \\ &= f(\mathbf{c}^{(i)}). \end{aligned}$$

□

We consider the following standard semi-infinite problem:

$$\begin{aligned} \text{SIP}_{\text{ex}} : \quad & \min_{\mathbf{x} \in M_{\text{ex}}} f(\mathbf{x}) \\ & \text{with } M_{\text{ex}} := \{\mathbf{x} \in \mathbb{R}^2 \mid g(\mathbf{x}, \mathbf{y}) := -(x_2 - y)^2 + x_1 \leq 0 \text{ for all } \mathbf{y} \in Y\} \\ & \text{and } Y := [-1, 1] \end{aligned}$$

As transformation we choose the identity. Then, we have $Z = Y$. We investigate the structure of the problem in the next lemma.

Lemma 5 *Problem SIP_{ex} fulfills the following properties:*

- (i) *The feasible set is given by*

$$M = \{\mathbf{x} \in X \mid x_1 \leq 0\}.$$

- (ii) The origin $\mathbf{x}^* = (0, 0)$ is a local minimizer of order 1.
- (iii) There exists a vector $\boldsymbol{\xi} \in \mathbb{R}^2$ such that

$$D_1 g(\mathbf{x}^*, y_0) \cdot \boldsymbol{\xi} \leq -1$$

for every $y_0 \in Y$ with $g(\mathbf{x}^*, y_0) = 0$.

Proof (i) Let $\mathbf{x} \in X$. The solution of the lower level is given by $y = x_2$. Thus,

$$\max_{y \in Y} g(\mathbf{x}, y) = x_1.$$

- (ii) On the feasible set M the objective function coincides with the norm, which clearly has a local minimum of order 1 in $\mathbf{x}^* = (0, 0)$.
- (iii) As by part (i) the only solution of the lower level for \mathbf{x}^* is given by $y_0 = 0$, we have with $\boldsymbol{\xi} := (-1, 0)^\top$

$$D_1 g(\mathbf{x}^*, y_0) \cdot \boldsymbol{\xi} = (1, -2x_2) \cdot \begin{pmatrix} -1 \\ 0 \end{pmatrix} = 1 \cdot -1 = -1.$$

□

If we take the strict local solution $\mathbf{c}^{(1)}$ as initial point and no point as initial discretization for Algorithm 1, the first point added to the discretization will be $\frac{1}{1}$. The next element of the sequence of strict local minimizers is still in $M(\dot{Z}_1)$ thus we can choose $\mathbf{c}^{(2)}$ as next iterate. Proceeding this way yields in the k -th iteration the local minima $\mathbf{c}^{(k+1)}$ as iterate $\mathbf{x}^{(k+1)}$ and the discretization $Z^{k+1} = \{\frac{1}{i} \mid 1 \leq i \leq k + 1\}$. The constraint violation after the k -th step is

$$\max_{y \in Y} g(\mathbf{x}^{(k+1)}, y) = \frac{1}{10^{k+1}},$$

but the distance to limiting solution is greater than $\frac{1}{k+1}$, which violates the convergence rate given in Theorem 2.

- Remark 2** (i) The second component of the local solutions $\frac{1}{i}$ can be replaced by an arbitrarily slow converging sequence. For this reason, there can be an arbitrarily bad rate of convergence.
- (ii) We presented this example in the context of an adaptively chosen discretization. The same example can also be used to construct a counterexample for uniformly chosen discretization points. Thus, this example shows that the assumption of non-vanishing radii (6) is missing in Still (2001a).

4 Convexity preserving transformations

Of course not all transformations destroy the convexity in the lower level problems. The topic of this section outlines under which conditions convexity properties can be maintained. We begin with the lower level problem.

The lower level problems $\tilde{Q}_i(\mathbf{x}), i \in I$, of the transformation-induced SIP problem are convex for all $\mathbf{x} \in X$, if the transformation-induced functions $\tilde{g}_i(\mathbf{x}, \mathbf{z}), i \in I$, are concave in \mathbf{z} for all \mathbf{x} and the set Z is convex.

A class of mappings, which map convex sets onto convex sets and preserve concavity, are the affine-linear ones. Therefore, it holds:

Lemma 6 For all $\mathbf{x} \in X$ let the functions $g_i(\mathbf{x}, \cdot), i \in I$, be concave and the set Z convex. Furthermore, let the transformation \mathbf{t} be affine-linear in its second argument, i.e.

$$\mathbf{t}(\mathbf{x}, \mathbf{z}) = \mathbf{A}(\mathbf{x}) \cdot \mathbf{z} + \mathbf{b}(\mathbf{x}),$$

where the mappings $\mathbf{A} : X \rightarrow \mathbb{R}^{n \times \tilde{n}}$ and $\mathbf{b} : X \rightarrow \mathbb{R}^n$ are at least continuous for all $\mathbf{x} \in X$. Then, the problems $\tilde{Q}_i(\mathbf{x}), i \in I$, are convex for all $\mathbf{x} \in X$.

Proof As the set Z is by definition convex, it only remains to prove the concavity of the objective function of the lower level problem. We show the concavity by checking the definition. Therefore fix $\mathbf{x} \in X$ and consider $\mathbf{z}_1, \mathbf{z}_2 \in Z, \lambda \in [0, 1]$. With the concavity of $g_i(\mathbf{x}, \cdot)$ it follows for $i \in I$:

$$\begin{aligned} \tilde{g}_i(\mathbf{x}, \lambda \mathbf{z}_1 + (1 - \lambda)\mathbf{z}_2) &= g_i(\mathbf{x}, \mathbf{A}(\mathbf{x})(\lambda \mathbf{z}_1 + (1 - \lambda)\mathbf{z}_2) + \mathbf{b}(\mathbf{x})) \\ &= g_i(\mathbf{x}, \lambda(\mathbf{A}(\mathbf{x})\mathbf{z}_1 + \mathbf{b}(\mathbf{x})) + (1 - \lambda)(\mathbf{A}(\mathbf{x})\mathbf{z}_2 + \mathbf{b}(\mathbf{x}))) \\ &\geq \lambda g_i(\mathbf{x}, \mathbf{A}(\mathbf{x})\mathbf{z}_1 + \mathbf{b}(\mathbf{x})) + (1 - \lambda)g_i(\mathbf{x}, \mathbf{A}(\mathbf{x})\mathbf{z}_2 + \mathbf{b}(\mathbf{x})) \\ &= \lambda \tilde{g}_i(\mathbf{x}, \mathbf{z}_1) + (1 - \lambda)\tilde{g}_i(\mathbf{x}, \mathbf{z}_2). \end{aligned}$$

□

After the lower level problems, we now consider the discretized problems solved in Step 5. For the computation of global solutions, it is of great value to have convex discretized problems, i. e., the functions $\tilde{g}_i(\mathbf{x}, \mathbf{z}), i \in I$, (and the function f) are convex in \mathbf{x} (for all \mathbf{z}). The following example illustrates that the transformation can destroy this property as well.

Example 2 Let the infinite index set be a semi-circle in arbitrary position with variable radius, which can be modeled as follows:

$$Y(\mathbf{x}) = \left\{ \mathbf{y} \in \mathbb{R}^2 \mid \begin{aligned} &(y_1 - x_1)^2 + (y_2 - x_2)^2 \leq x_3^2 \\ &(y_1 - x_1) \frac{x_4}{\sqrt{x_4^2 + x_5^2}} + (y_2 - x_2) \frac{x_5}{\sqrt{x_4^2 + x_5^2}} \leq 0 \end{aligned} \right\},$$

with $\mathbf{x} \in X := \mathbb{R}^2 \times \mathbb{R}_{>0}^3 \times \mathbb{R}$. The mapping, by which the semi-circle can be represented as image of the set $Z = [0, 1] \times [-1/2, 1/2]$ is

$$\mathbf{t}(\mathbf{x}, \mathbf{z}) := \begin{pmatrix} x_1 + z_1 x_3 \cos(\text{atan2}(-x_5, -x_4) + \pi z_2) \\ x_2 + z_1 x_3 \sin(\text{atan2}(-x_5, -x_4) + \pi z_2) \end{pmatrix},$$

with atan2 being the bivariate arctangent

$$\begin{aligned} \text{atan2} : \mathbb{R}^2 \setminus \{\mathbf{0}\} &\rightarrow \mathbb{R}, \\ (a, b) &\mapsto 2 \arctan \left(\frac{b}{\sqrt{a^2 + b^2} + a} \right). \end{aligned} \tag{9}$$

Obviously, the sets $Y(\mathbf{x})$, $\mathbf{x} \in X$, as well as the set Z are convex. We consider the function $g(\mathbf{x}, \mathbf{y}) = x_6^2 - y_1 + y_2$, which is convex in \mathbf{x} and \mathbf{y} . However, the function

$$\begin{aligned} \tilde{g}(\mathbf{x}, \mathbf{z}) &:= g(\mathbf{x}, \mathbf{t}(\mathbf{x}, \mathbf{z})) \\ &= x_6^2 - x_1 - z_1 x_3 \cos(\text{atan2}(-x_5, -x_4) + \pi z_2) \\ &\quad + x_2 + z_1 x_3 \sin(\text{atan2}(-x_5, -x_4) + \pi z_2) \end{aligned}$$

is not convex in \mathbf{x} for all $\mathbf{z} \in Z$. For example with $\mathbf{x}_1 = (0, 0, 1, 1, -1, 0)^T$, $\mathbf{x}_2 = (0, 0, 1, 1, 0, 0)^T$ and $\mathbf{z} = (1, 0)^T$ one gets:

$$\frac{1}{2} \tilde{g}(\mathbf{x}_1, \mathbf{z}) + \frac{1}{2} \tilde{g}(\mathbf{x}_2, \mathbf{z}) = \frac{1}{2}(\sqrt{2} + 0) < 1 = \tilde{g}\left(\frac{1}{2}\mathbf{x}_1 + \frac{1}{2}\mathbf{x}_2, \mathbf{z}\right).$$

The next lemma states, under which conditions on the functions g_i , $i \in I$, and the transformation \mathbf{t} , convexity in \mathbf{x} is preserved. It follows by the fact that the composition of a convex and a linear mapping is convex again.

Lemma 7 *Let the functions $g_i(\mathbf{x}, \mathbf{y})$, $i \in I$, be convex in (\mathbf{x}, \mathbf{y}) . Furthermore, let the transformation \mathbf{t} be affine-linear in its first argument, i.e.*

$$\mathbf{t}(\mathbf{x}, \mathbf{z}) = \mathbf{A}(\mathbf{z}) \cdot \mathbf{x} + \mathbf{b}(\mathbf{z}), \tag{10}$$

where $\mathbf{A} : \mathbb{R}^{\tilde{n}} \rightarrow \mathbb{R}^{n \times m}$ and $\mathbf{b} : \mathbb{R}^{\tilde{n}} \rightarrow \mathbb{R}^n$. Then the functions $\tilde{g}_i(\mathbf{x}, \mathbf{z})$, $i \in I$, are convex in \mathbf{x} for any $\mathbf{z} \in Z$.

Obviously, the rotation used in Example 2 does not satisfy condition (10). However, for translation, scaling, and shearing condition (10) is satisfied.

5 Numerical examples

In this section we apply Algorithm 1 to some native GSIP problems and two GSIP problems stemming from an important application of semi-infinite optimization, namely design centering. We illustrate how the arising problems are performantly solved by the proposed algorithm. A more detailed numerical analysis of the transformation-based discretization method as well as its application to a real world problem, the volume-maximal utilization of gemstones, can be found in Schwientek (2013). Note that the following numerical examples shouldn't be a full comparison to previously developed algorithms. Instead the examples illustrate that even complicated problems like the inscribing of two designs in a container can be solved in an easy way.

We implemented Algorithm 1 in MATLAB R2016a. In Step 1 we chose $\alpha = 10^{-6}$ as the termination tolerance. For the solution of the discretized SIP problems $\widetilde{\text{SIP}}(\dot{Z}_k), k \in \mathbb{N}_0$, (Step 5) as well as for the solution of the convex lower level problems $Q_i(\mathbf{x}^k), i \in I, k \in \mathbb{N}_0$, (Step 8) we used the SQP algorithm of the routine `fmincon` of the Optimization Toolbox V7.4 (R2016a) with default settings and using first derivatives. The computations have been performed on a 64-bit Windows machine with a Intel® Core™i7 5600U processor and 8 GB RAM. The time needed for the computation have been measured using the MATLAB function `timeit`.

5.1 Some native examples

In the first numerical example we consider the general semi-infinite optimization problem outlined in Lemonidis (2008), Example 1 (see also Jongen et al. 1998, Example 4-2). We added the constraint $x_1 \geq 0$, otherwise the infinite index set $Y(\mathbf{x})$ could be empty.

$$\begin{aligned} \text{GSIP}_1 : \quad & \min_{\mathbf{x} \in [-1, 1]^2} (x_1 - 1/4)^2 + x_2^2 \text{ s.t.} \\ & -y - x_2 \leq 0 \text{ for all } y \in Y(\mathbf{x}), \\ & x_1 \geq 0, \end{aligned}$$

with

$$Y(\mathbf{x}) = \{y \in [-1, 1] \mid y^2 - x_1 \leq 0\}.$$

The infinite index set $Y(\mathbf{x})$ can be explicitly calculated and is equal to $[-\sqrt{x_1}, \sqrt{x_1}]$. The (only) lower level problem

$$Q(\mathbf{x}) : \max_y -y - x_2 \text{ s.t. } y \in [-\sqrt{x_1}, \sqrt{x_1}]$$

is convex (in y), but the corresponding optimal value function $\varphi(\mathbf{x}) = \sqrt{x_1} - x_2$ isn't convex (in \mathbf{x}). The feasible set of GSIP_1 is

$$M = \{\mathbf{x} \in [-1, 1]^2 \mid x_1 \geq 0, \sqrt{x_1} \leq x_2\}$$

and its optimal solution $\mathbf{x}^* = (0, 0)$. By the simple transformation

$$t : [0, 1]^2 \times [0, 1] \rightarrow \mathbb{R} \text{ with } t(\mathbf{x}, z) = -z \cdot \sqrt{x_1} + (1 - z) \cdot \sqrt{x_1} = (1 - 2z)\sqrt{x_1}$$

GSIP_1 can be transferred into a standard semi-infinite optimization problem, whose discretized problems are convex.

We chose $\mathbf{x}^0 = (1, 1)$ as starting point ($f^0 = 25/16$) and the point 0.5 as initial discretization. After 2 iterations (0.12 s) Algorithm 1 stopped with $\mathbf{x}^* = (0, 0)$ and $f^* = 0.0625$. The number of finite constraints induced by discretization grew from 1 to 2.

The global solution of the lower level problem is given by the lower bound of the interval $[-\sqrt{x_1}, \sqrt{x_1}]$. In the first iteration the discretization point $z = 1$ is added. The chosen transformation maps this discretization point to the global solution of the lower level. This means that after the first iteration the discretized problem $\overline{\text{SIP}}(\dot{Z}_k)$ and the GSIP problem have the same feasible set.

We tested our new algorithm on the other 15 examples collected by Lemonidis (2008), too. This test set was also used in the recent publications (Mitsos and Tsoukalas 2015; Kirst and Stein 2019). However, to apply Algorithm 1 we need to ensure that the central Assumption 1 is satisfied. This means in particular that the infinite index set $Y(\mathbf{x})$ must be non-empty for every $\mathbf{x} \in X$. Unfortunately, most of the examples collected in Lemonidis (2008) do not satisfy this condition, namely Examples 1, 2, 4-9, 11, 13, and 16. However, in most of these examples (1, 2, 6-8, 11, and 13) it is possible to meet this condition by adding a single constraint. For the remaining examples (4, 5, 9, and 16) it is as follows. As shown by Lemonidis (2008), the infinite index set in Example 4 is either empty or the semi-infinite constraint can not be satisfied. For the Examples 5, 9, and 16 we could not easily derive a transformation. That's why we excluded these four examples from our numerical evaluations. The considered original and modified examples together with their transformation are listed in "Appendix 1".

In most of the taken examples the infinite index set is given by an interval. Then, a transformation can easily be constructed by convex combination of the interval ends. In Example 3 the infinite index set can be obtained by scaling the unit ball by the radius. A possible transformation for Example 6 is given by

$$\mathbf{y} = \mathbf{t}(\mathbf{x}, \mathbf{z}) := \begin{pmatrix} -4z_1 + x_1(1 - z_1) \\ \max\{-4, -4 - y_1\}z_2 + x_1(1 - z_2) \\ (y_1 + y_2)^2 z_3 + 16(1 - z_3) \end{pmatrix}$$

In Example 14 the infinite index set is a two-dimensional box. Thus, the Cartesian product of the convex combination of the lower and upper bounds is a suitable transformation.

In the 12 examples for which the transformation-based discretization method is applicable the algorithm stopped after two iterations. The time needed to solve the problems ranges from 0.08 to 0.14 s. The reasons for the small number of iterations are the same as described for the first example above.

In all examples, except Example 13, Algorithm 1 found the optimal solution reported in the literature. In Example 13 the solution can not be attained, because we removed it by adding the mentioned constraint to avoid an empty index set. The transformation-based discretization method then finds the point $\mathbf{x}^* = (0, 0.25, 0.25)$ with an objective value $f^* = 3.5681$.

To further demonstrate our new method, we turn to more complex examples with multiple semi-infinite constraints now.

5.2 Design centering

One important application of (general) semi-infinite programming is *design centering* (DC). There, the task is the following: given a set $C \subseteq \mathbb{R}^n$ - the so-called *container* - and a second, parametrized set $D(\mathbf{x}) \subseteq \mathbb{R}^n, \mathbf{x} \in X \subseteq \mathbb{R}^m$ - the so-called *design* - inscribe $D(\mathbf{x})$ into C such that some functional, commonly the volume, of $D(\mathbf{x})$ is maximized:

$$DC : \max_{\mathbf{x} \in X \subseteq \mathbb{R}^m} \text{Vol}(D(\mathbf{x})) \text{ s.t. } D(\mathbf{x}) \subseteq C.$$

Under the assumption that the sets C and $D(\mathbf{x})$ are given as solutions sets of some systems of inequalities

$$C = \{\mathbf{y} \in \mathbb{R}^n \mid g_i(\mathbf{y}) \leq 0, i \in I\}, \tag{11}$$

$$D(\mathbf{x}) = \{\mathbf{y} \in \mathbb{R}^n \mid u_j(\mathbf{x}, \mathbf{y}) \leq 0, j \in J\}, \tag{12}$$

Problem DC can be rewritten as a general semi-infinite optimization problem

$$\begin{aligned} \text{GSIP}_{DC} : \quad & \min_{\mathbf{x} \in X \subseteq \mathbb{R}^m} -\text{Vol}(D(\mathbf{x})) \\ \text{s.t.} \quad & g_i(\mathbf{y}) \leq 0 \text{ for all } \mathbf{y} \in D(\mathbf{x}), i \in I. \end{aligned}$$

For a detailed discussion of the reformulation of design centering problems as semi-infinite ones we refer to Stein (2006) and the references therein. Different solution techniques are discussed in Harwood and Barton (2017).

One interesting application of design centering in the context of semi-infinite optimization is the maximal material usage in gemstone cutting. There, the task is to produce a set of precious gems from an irregularly shaped raw one in such a way that their total value is as high as possible. In the case of producing only one precious gem, the problem corresponds to a design centering, where the precious gem corresponds to the design, the raw one to the container and the volume of the precious shall be maximized. For a detailed introduction into the modeling and solution of this problem we refer to Winterfeld (2008); Küfer et al. (2008, 2015).

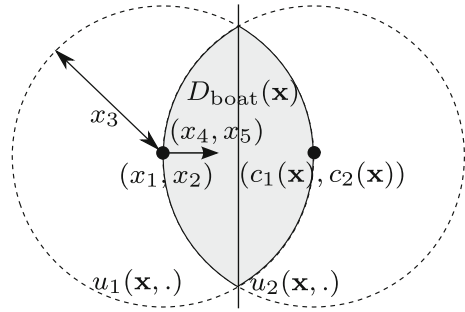
In a concrete case we consider a two-dimensional design centering problem, i.e. $n = 2$. Firstly, we introduce the container, called *concavified unit square* (see Fig. 3a for a graphical illustration),

$$C_{\text{cus}} := \{\mathbf{y} \in \mathbb{R}^2 \mid g_i(\mathbf{y}) \leq 0, i = 1, \dots, 6\}, \tag{13}$$

with

$$\begin{aligned} g_1(\mathbf{y}) &:= y_1 - 1, \quad g_2(\mathbf{y}) := -y_2, \quad g_3(\mathbf{y}) := -2(y_1 - 0.5)^2 + y_2 - 0.5, \\ g_4(\mathbf{y}) &:= -16y_1^2 - 4(y_2 - 0.5)^2 + 1, \quad g_5(\mathbf{y}) := y_2 - 1, \quad g_6(\mathbf{y}) := -y_1. \end{aligned}$$

Fig. 2 Functional description and parametrization of the boat design



The last two constraints aren't visible in Fig. 3, but are necessary to avoid unboundness of the container.

Now, we present the design. Remember that in addition to a functional description of the design (12) and a calculation rule for their area, we need for the transformation-based discretization method a representation of the design as an image of a compact set Z under a continuously differentiable mapping $\mathbf{t} : X \times Z \rightarrow \mathbb{R}^2$.

As design we consider the intersection of two variable, but equally sized circles. The center of each circle is a boundary point of the other one (see Fig. 2). We call this design *boat*. Thus, $X = \mathbb{R}^2 \times \mathbb{R}_{\geq 0} \times \mathbb{R}^2 \setminus \{\mathbf{0}\}$,

$$D_{\text{boat}}(\mathbf{x}) = \left\{ \mathbf{y} \in \mathbb{R}^2 \mid \begin{aligned} (y_1 - x_1)^2 + (y_2 - x_2)^2 - x_3^2 &\leq 0, \\ (y_1 - c_1(\mathbf{x}))^2 + (y_2 - c_2(\mathbf{x}))^2 - x_3^2 &\leq 0 \end{aligned} \right\}, \tag{14}$$

with $c_1(\mathbf{x}) = x_1 + \frac{x_3 x_4}{\sqrt{x_4^2 + x_5^2}}$ and $c_2(\mathbf{x}) = x_2 + \frac{x_3 x_5}{\sqrt{x_4^2 + x_5^2}}$.

The intersection points connecting line divides the boat into two equally sized circle segments of angle $2\pi/3$ and radius x_3 (see Fig. 2). Consequently,

$$\text{Vol}(D(\mathbf{x})) = \left(\frac{2\pi}{3} - \frac{\sqrt{3}}{2} \right) x_3^2.$$

Moreover, we have,

$$Z = [0, 1] \times [-1, 1],$$

and,

$$\mathbf{t}(\mathbf{x}, \mathbf{z}) = \begin{cases} \left(\begin{aligned} x_1 + z_1 x_3 \cos \left(\text{atan2}(x_5, x_4) + \frac{\pi}{3} + \frac{2\pi}{3} z_2 \right) \\ x_2 + z_1 x_3 \sin \left(\text{atan2}(x_5, x_4) + \frac{\pi}{3} + \frac{2\pi}{3} z_2 \right) \end{aligned} \right), & z_2 \leq 0 \\ \left(\begin{aligned} c_1(\mathbf{x}) + z_1 x_3 \cos \left(\text{atan2}(x_5, x_4) - \frac{4\pi}{3} + \frac{2\pi}{3} z_2 \right) \\ c_2(\mathbf{x}) + z_1 x_3 \sin \left(\text{atan2}(x_5, x_4) - \frac{4\pi}{3} + \frac{2\pi}{3} z_2 \right) \end{aligned} \right), & z_2 > 0 \end{cases},$$

where atan2 is the bivariate arctangent.

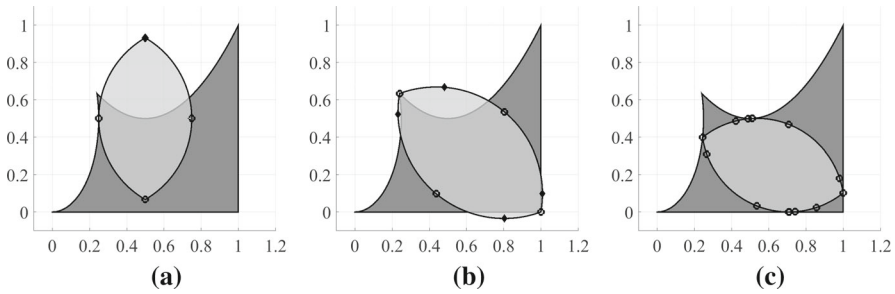


Fig. 3 Area-maximal inscribing of the design D_{boat} in the container C_{cus} [dark gray—container, light gray—design, circles—points of current discretization, which do not violate the container restrictions, diamonds—points of current discretization, which violate container restrictions and will be added in the next iteration]: **a** start solution with given discretization, **b** after solution of the first discretized SIP problem $\hat{\text{SIP}}(\dot{Z}_0)$, and **c** final solution (after 5 refinements) with final discretization

We chose $\mathbf{x}^0 = (0.25, 0.5, 0.5, 1, 0)$ as starting point with $f^0 = -0.3071$ (see Fig. 3a for a graphical illustration of this configuration). The initial discretization of the design consists of the four points $\{(1, -0.5), (1, 0.5), (1, 0), (1, 1)\}$ yielding 24 constraints (4 points for each of the 6 container constraints) in the initial discretized SIP problem. After 5 iterations/refinements (0.69 s) Algorithm 1 stopped with $\mathbf{x}^* = (0.5367, 0.0325, 0.4685, 0.4997, 1.2712)$ and $f^* = -0.2696$ (see Fig. 3c for a visualization of the final solution). The final discretization consisted of 14 points and thus 84 discretization-induced constraints. A extensive numerical study of the solution of DC problems as semi-infinite ones with different design-container-combinations can be found in Schwientek (2013).

The transformation-induced discretized SIP problems as well as the underlying GSIP problem stemming from a design centering problem are non-convex in general. For this reason, Algorithm 1 will terminate with either a local solution or a stationary point.

5.3 Inscribing multiple designs

A natural extension of the design centering task is to arrange two or more designs in a container yielding maximal total (design) volume. In practice, such problems arise as cutting or packing problems. In addition to the condition that the designs are completely located in the container, the designs are not allowed to overlap. In the following we will reformulate and evaluate such a problem as GSIP in the case of two designs. While we only consider the case of two designs, the extension of the approach to inscribe more than two designs is straightforward. For details we refer to Schwientek (2013), Küfer et al. (2015).

Let the container C be as given in (11) and D_1, D_2 two (parametrized) designs as given in (12) with common parameter vector $\mathbf{x} \in \mathbb{R}^m$. Then, the task in multi-body design centering (in the case of two designs) is the following:

$$\begin{aligned}
 \text{MBDC}_2 : \quad & \max_{\mathbf{x} \in X \subseteq \mathbb{R}^m} \text{Vol}(D_1(\mathbf{x})) + \text{Vol}(D_2(\mathbf{x})) \text{ s.t.} \\
 & D_k(\mathbf{x}) \subseteq C, \quad k = 1, 2, \\
 & D_1(\mathbf{x}) \cap \text{int}(D_2(\mathbf{x})) = \emptyset.
 \end{aligned}
 \tag{15}$$

Under the assumption that the designs D_1 and D_2 are convex sets for each \mathbf{x} , a separation theorem can be applied, where $\boldsymbol{\eta} \in \mathbb{R}^n \setminus \mathbf{0}$ and β are the parameters of the separating hyperplane $H(\boldsymbol{\eta}, \beta) = \{\mathbf{y} \in \mathbb{R}^n \mid \boldsymbol{\eta}^T \mathbf{y} = \beta\}$, $\tilde{\mathbf{x}} = (\mathbf{x}, \boldsymbol{\eta}, \beta)$, and $\tilde{X} = \{\tilde{\mathbf{x}} \mid \boldsymbol{\eta} \neq \mathbf{0}\}$. Then, MBDC₂ can be rewritten as a GSIP in the following way:

$$\begin{aligned}
 \text{GSIP}_{\text{MBDC}_2} : \quad & \max_{\tilde{\mathbf{x}} \in \tilde{X} \subseteq \mathbb{R}^{m+n+1}} \text{Vol}(D_1(\mathbf{x})) + \text{Vol}(D_2(\mathbf{x})) \text{ s.t.} \\
 & g_i(\mathbf{y}) \leq 0 \text{ for all } \mathbf{y} \in D_k(\mathbf{x}), \quad i \in I, \quad k = 1, 2, \\
 & \boldsymbol{\eta}^T \mathbf{y} \leq \beta \text{ for all } \mathbf{y} \in D_1(\mathbf{x}), \\
 & \boldsymbol{\eta}^T \mathbf{y} \geq \beta \text{ for all } \mathbf{y} \in D_2(\mathbf{x}).
 \end{aligned}
 \tag{17}$$

The condition $\boldsymbol{\eta} \neq \mathbf{0}$ can be assured e.g. by requiring $\|\boldsymbol{\eta}\|_2^2 = 1$. Note that we have two index sets here, but extending Algorithm 1 to this situation is straightforward.

Using this approach, it is also possible to consider forbidden areas in the container when placing the designs and to avoid overlapping these with the designs. If one models the forbidden areas similar to the designs (but independent of \mathbf{x}), the separation approach above carries directly over. For more detailed explanations in this regard and further separation techniques we refer to Schwientek (2013).

For a numerical example of problem GSIP_{MBDC₂} we consider the container from (13) and take as the first design, D_1 , the *boat* from (14). As our second design, D_2 , we consider an ellipse with variable semi-axis lengths in arbitrary position. For such an ellipse, we have $m_2 = 5$,

$$D_{\text{ell}}(\mathbf{x}) = \{\mathbf{y} \in \mathbb{R}^2 \mid [\mathbf{y} - \mathbf{c}(\mathbf{x})]^T [\mathbf{A}(\mathbf{x})\mathbf{A}(\mathbf{x})^T]^{-1} [\mathbf{y} - \mathbf{c}(\mathbf{x})] - 1 \leq 0\}$$

with

$$\begin{aligned}
 \mathbf{c}(\mathbf{x}) &:= \begin{pmatrix} x_1^{(2)} \\ x_2^{(2)} \end{pmatrix} \text{ and } \mathbf{A}(\mathbf{x}) := \begin{pmatrix} x_3^{(2)} & x_5^{(2)} \\ 0 & x_4^{(2)} \end{pmatrix}, \text{ Vol}(D_{\text{ell}}(\mathbf{x})) = \pi |\det(\mathbf{A}(\mathbf{x}))|, \\
 Z &= [0, 1] \times [-\pi, \pi]
 \end{aligned}$$

and

$$\mathbf{t}(\mathbf{x}, \mathbf{z}) = \mathbf{A}(\mathbf{x}) \begin{pmatrix} z_1 \cos(z_2) \\ z_1 \sin(z_2) \end{pmatrix} + \mathbf{c}(\mathbf{x}).$$

This constellation results in 13 decision variables, 13 semi-infinite and 2 finite constraints. As starting point we chose the (infeasible) point $\tilde{\mathbf{x}}^0 = (\mathbf{x}_{\text{boat}}^0, \mathbf{x}_{\text{ellipse}}^0, \boldsymbol{\eta}^0)$,

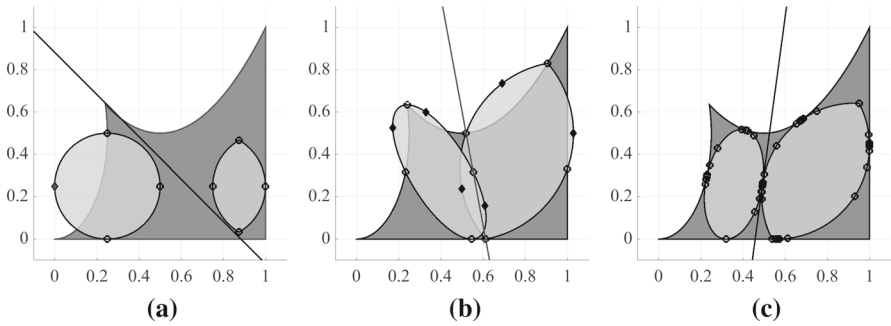


Fig. 4 Area-maximal inscribing of the design D_{boat} and D_{ell} in the container C_{cus} [dark gray—container, light gray—designs, circles—points of current discretization, which do not violate the container and separation restrictions, diamonds—points of current discretization, which violate container or separation restrictions and will be added in the next iteration, black straight line—line that separates both designs]: **a** start solution with given discretization, **b** after solution of the first discretized SIP problem $\tilde{\text{SIP}}(\tilde{Z}_0)$, and **c** final solution (after 12 refinements) with final discretization

β^0) with $\mathbf{x}_{\text{boat}}^0 = (0.75, 0.25, 0.25, 1, 0)$, $\mathbf{x}_{\text{ellipse}}^0 = (0.25, 0.25, 0.25, 0, 0.25)$, $\boldsymbol{\eta}^0 = (-\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2})$, and $\beta^0 = -0.625$ (see Fig. 4a). As initial discretizations we consider $\tilde{Z}_0^{\text{boat}} = \{(1, -0.5), (1, 0.5), (1, 0), (1, 1)\}$ and $\tilde{Z}_0^{\text{ellipse}} = \{(1, 0), (1, 0.5), (1, 1), (1, -0.5)\}$. These discretizations result in 56 finite constraints. After 12 iterations (simultaneous refinements of both infinite index sets) which takes 3.75 s Algorithm 1 terminates with solution $\tilde{\mathbf{x}}^* = (\mathbf{x}_{\text{boat}}^*, \mathbf{x}_{\text{ellipse}}^*, \boldsymbol{\eta}^*, \beta^*)$ with

$$\begin{aligned} \mathbf{x}_{\text{boat}}^* &= (0.5596, 0.4404, 0.4404, 0.9069, -0.5837), \\ \mathbf{x}_{\text{ellipse}}^* &= (0.3574, 0.2589, 0.1358, 0.0365, 0.2589), \\ (\boldsymbol{\eta}^*, \beta^*) &= (-0.9910, 0.1340, -0.4540), \end{aligned}$$

and $f^* = -0.3487$. The final discretization of the boat consists of 37 points and the one of the ellipse of 30 points (see Fig. 4c).

Further numerical evaluations, also with respect to other separation approaches and regarding a minimal distance between the designs, can be found in Schwientek (2013). Concerning the modeling and solution of the gemstone utilization task as semi-infinite optimization problem by means of the transformation-based discretization we refer to Schwientek (2013) and Küfer et al. (2015).

6 Conclusions and future research

In the present paper we consider general semi-infinite optimization problems (GSIP), which have convex lower level problems and can be globally transformed into a standard semi-infinite optimization problem (SIP). For the numerical solution of such GSIPs we applied a discretization method to its SIP-reformulation. Because the convexity structure in the lower level can be lost through the transformation, but is essential

for refining the discretization, we solve in the refinement step the (convex) lower level problems of the underlying GSIP and transfer the global solutions into global solutions of the lower level problems of the induced SIP via the transformation. The convergence results for discretization methods for solving SIPs directly carry over in the case of global and local solutions. An open issue for future research is the convergence of stationary points. Finally, we have demonstrated the operation and performance of our method using three numerical examples.

Two interesting aspects for future investigations are the following:

- (1) As known from Still (2001a), the convergence rates can be improved for discretization methods for solving SIPs, when boundary points of the infinite index set are added in a consistent manner for its discretization and the index set satisfies additional conditions. Since it is required for the transformation only to be surjective, the boundary of the infinite index set Z of the induced SIP is generally not mapped to the boundary of the infinite index set $Y(\mathbf{x})$ and vice versa. This leads to the question whether it is sufficient to add boundary points of $Y(\mathbf{x})$ in the mentioned consistent manner to get improved convergence rates for the transformation-based discretization method.
- (2) To keep the dimensions of discretized SIP problems moderate, strategies have been developed to retain only α -active points in the discretization and remove the others during the process. Such algorithms are called exchange methods instead of discretization methods. An exploration of these deletion strategies applied to the transformation-based discretization method could prove interesting. Of special interest would be the impact this may have on the convergence of the method.

Acknowledgements Open Access funding provided by Projekt DEAL. We would like to thank the two anonymous reviewers and the Associate Editor for their many valuable comments, which have contributed to a significant improvement of the article.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Considered examples from Lemonidis' collection

The following examples were collected by Lemonidis (2008). We modified some of the examples in such a way that there exists a transformation as demanded in Assumption 1. We use the original numbering of the examples as in Lemonidis (2008).

Example 1 (modified):
$$\min_{\mathbf{x} \in [-1, 1]^2} (x_1 - 1/4)^2 + x_2^2 \quad \text{s.t.}$$

$$\begin{aligned} -y - x_2 &\leq 0 \text{ for all } y \in Y(\mathbf{x}), \\ x_1 &\geq 0, \end{aligned} \tag{18}$$

with

$$Y(\mathbf{x}) = \left\{ y \in [-1, 1] \mid y^2 - x_1 \leq 0 \right\}.$$

The constraint (18) is added to the original example. Then, a transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : [0, 1]^2 \times [0, 1] \rightarrow \mathbb{R} \text{ with } y = t(\mathbf{x}, z) := \sqrt{x_1}(1 - 2z).$$

Note that in the collection in Lemonidis (2008) the semi-infinite constraint is different. We use the version presented in the original source (Jongen et al. 1998).

Example 2 (modified):
$$\min_{\mathbf{x} \in [-1, 1]^2} x_2 \quad \text{s.t.}$$

$$\begin{aligned} -y^3 + x_2 &\leq 0 \text{ for all } y \in Y(\mathbf{x}), \\ 2x_2 - x_1^2 &\leq 0, \end{aligned} \tag{19}$$

with

$$Y(\mathbf{x}) = \left\{ y \in [-1, 0] \mid 2x_2 - y^3 - x_1^2 \leq 0 \right\}.$$

The constraint (19) is added to the original example. Then, a transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^2 \times [0, 1] \rightarrow \mathbb{R} \text{ with } y = t(\mathbf{x}, z) := \max \left\{ \sqrt[3]{2x_2 - x_1^2}, -1 \right\} z.$$

Example 3 (original):
$$\min_{\mathbf{x} \in [0, 1]^2} 1/2 x_1^4 + 2x_1 x_2 - 2x_1^2 \quad \text{s.t.}$$

$$y_1^2 + y_2^2 - x_1 + x_1^2 - x_2 \leq 0 \text{ for all } \mathbf{y} \in Y(\mathbf{x}),$$

with

$$Y(\mathbf{x}) = \left\{ \mathbf{y} \in [0, 1]^3 \mid y_1^2 + y_2^2 + y_3^2 - x_1 \leq 0 \right\}.$$

A transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^2 \times \left\{ \mathbf{z} \in [0, 1]^3 \mid z_1^2 + z_2^2 + z_3^2 \leq 1 \right\} \rightarrow \mathbb{R}^3 \text{ with } \mathbf{y} = t(\mathbf{x}, \mathbf{z}) := x_1 \mathbf{z}.$$

Example 6 (modified): $\min_{\mathbf{x} \in [-3, 2]^2} 4x_1^2 - x_2 - x_2^2 \quad \text{s.t.}$

$$\begin{aligned} x_2 - y_3 &\leq 0 \text{ for all } \mathbf{y} \in Y(\mathbf{x}), \\ x_1 &\geq -2, \end{aligned} \quad (20)$$

with

$$\begin{aligned} Y(\mathbf{x}) = \{ \mathbf{y} \in [-4, 4]^2 \times [0, 16] \mid & y_1 - x_1 \leq 0, \\ & y_2 - x_1 \leq 0, \\ & (y_1 + y_2)^2 - y_3 \leq 0 \}. \end{aligned}$$

The constraint (20) is added to the original example. Then, a transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$\begin{aligned} t : \mathbb{R}^2 \times [0, 1]^3 &\rightarrow \mathbb{R}^3 \text{ with} \\ \mathbf{y} = t(\mathbf{x}, \mathbf{z}) &:= \begin{pmatrix} -4z_1 + x_1(1 - z_1) \\ \max\{-4, -4 - y_1\}z_2 + x_1(1 - z_2) \\ (y_1 + y_2)^2 z_3 + 16(1 - z_3) \end{pmatrix}. \end{aligned}$$

Note that the semi-infinite constraint in the collection in Lemonidis (2008) contains a typing error. We use the version presented in the original source (Rückmann and Shapiro 2001).

Example 7 (modified): $\min_{\mathbf{x} \in [0, 1]^2} -x_1 \quad \text{s.t.}$

$$\begin{aligned} 3x_2^2 - y^5 &\leq 0 \text{ for all } \mathbf{y} \in Y(\mathbf{x}), \\ -4x_1^2 - x_2^2 + 1 &\leq 0, \end{aligned} \quad (21)$$

with

$$Y(\mathbf{x}) = \left\{ \mathbf{y} \in [-2, 0] \mid -y^5 - 4x_1^2 - x_2^2 + 1 \leq 0 \right\}.$$

The constraint (21) is added to the original example. Then, a transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^2 \times [0, 1] \rightarrow \mathbb{R} \text{ with } y = t(\mathbf{x}, z) := (-4x_1^2 - x_2^2 + 1)^{\frac{1}{5}} z.$$

Example 8 (modified): $\min_{\mathbf{x} \in [-1, 1]^2} -x_1 \quad \text{s.t.}$

$$\begin{aligned} -x_2 y &\leq 0 \text{ for all } \mathbf{y} \in Y(\mathbf{x}), \\ x_1 &\geq 0, \end{aligned} \quad (22)$$

with

$$Y(\mathbf{x}) = \left\{ y \in [-1, 1] \mid -y^2 + x_1 \leq 0 \right\}.$$

The constraint (22) is added to the original example. Then, a transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^2 \times [0, 1] \rightarrow \mathbb{R} \text{ with } y = t(\mathbf{x}, z) := \sqrt{x_1}(1 - 2z).$$

Note that the constraint describing the infinite index set is different in Lemonidis (2008). We use the version given in the original source (Guerra Vázquez and Rückmann 2005).

Example 10 (original): $\min_{\mathbf{x} \in [-1, 1]^2} x_1 + x_2$ s.t.

$$-y \leq 0 \text{ for all } y \in Y(\mathbf{x}),$$

with

$$Y(\mathbf{x}) = \{y \in [0, 1] \mid x_1 - y \leq 0, x_2 - y \leq 0\}.$$

A transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^2 \times [0, \infty) \rightarrow \mathbb{R} \text{ with } y = t(\mathbf{x}, z) := \begin{cases} x_1 + z, & \text{if } x_1 \geq x_2, \\ x_2 + z, & \text{otherwise.} \end{cases}$$

Example 11 (modified): $\min_{\mathbf{x} \in [-5, 5]^3} x_1^2 + x_2^2 + x_3^2$ s.t.

$$\begin{aligned} x_1 + x_2 \exp(x_3 y) + \exp(2y) - 2 \sin(4y) &\leq 0 \text{ for all } y \in Y(\mathbf{x}), \\ x_2 &\geq -1, \end{aligned} \tag{23}$$

with

$$Y(\mathbf{x}) = \{y \in [0, 1] \mid 2y - x_2 - 1 \leq 0\}.$$

The constraint (23) is added to the original example. Then, a transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^2 \times [0, 1] \rightarrow \mathbb{R} \text{ with } y = t(\mathbf{x}, z) := \frac{x_2 + 1}{2}(1 - z).$$

Example 12 (original): $\min_{\mathbf{x} \in [-1, 1]} x^2$ s.t.

$$1/2 y^3 - x^2 \leq 0 \text{ for all } y \in Y(\mathbf{x}),$$

with

$$Y(\mathbf{x}) = \left\{ y \in [0, 1] \mid x^2 - y^2 \leq 0 \right\}.$$

A transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R} \times [0, 1] \rightarrow \mathbb{R} \text{ with } y = t(x, z) := |x|(1 - z) + z.$$

Example 13 (modified): $\min_{\mathbf{x} \in [-1, 1]^3} \exp(x_1) + \exp(x_2) + \exp(x_3)$ s.t.

$$\begin{aligned} \frac{1}{1 + y^2} - x_1 - x_2 y - x_3 y^2 \leq 0 \text{ for all } y \in Y(\mathbf{x}), \\ x_2 + x_3 \leq 1/2, \end{aligned} \quad (24)$$

with

$$Y(\mathbf{x}) = \{y \in [0, 1] \mid -1/2 y + x_2 + x_3 \leq 0\}.$$

The constraint (24) is added to the original example. Then, a transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^3 \times [0, 1] \rightarrow \mathbb{R} \text{ with } y = t(\mathbf{x}, z) := \max\{0, 2(x_2 + x_3)\}.$$

Example 14 (original): $\min_{\mathbf{x} \in [-1, 0]^3} x_1^2 + x_2^2 + x_3^2$ s.t.

$$x_1 (y_1 + y_2^2 + 1) + x_2 (y_1 y_2 - y_2^2) + x_3 (y_1 y_2 + y_2^2 + y_2) + 1 \leq 0 \text{ for all } \mathbf{y} \in Y(\mathbf{x}),$$

with

$$Y(\mathbf{x}) = \left\{ \mathbf{y} \in [0, 1]^2 \mid x_1^2 - y_1^2 \leq 0 \right\}.$$

A transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^3 \times [0, 1]^2 \rightarrow \mathbb{R}^2 \text{ with } \mathbf{y} = t(\mathbf{x}, \mathbf{z}) := \begin{pmatrix} z_1 + |x_1|(1 - z_1) \\ z_2 \end{pmatrix}.$$

Example 15 (original): $\min_{\mathbf{x} \in [0, 2]^2} x_2^2 - 4x_2$ s.t.

$$x_1 \cos(y) + x_2 \sin(y) - 1 \leq 0 \text{ for all } y \in Y(\mathbf{x}),$$

with

$$Y(\mathbf{x}) = \left\{ y \in [0, \pi] \mid -y^2 - 7/4 x_2 + 23/4 \leq 0 \right\}.$$

A transformation of a fixed index set Z to the variable index set $Y(\mathbf{x})$ is

$$t : \mathbb{R}^2 \times [0, 1] \rightarrow \mathbb{R} \text{ with } y = t(\mathbf{x}, z) := \sqrt{(23/4 - 7/4 x_2)z + \pi(1 - z)}.$$

References

- Bhattacharjee B, Lemonidis P, Green WH Jr, Barton PI (2005) Global solution of semi-infinite programs. *Math Program* 103(2):283–307
- Blankenship JW, Falk JE (1976) Infinitely constrained optimization problems. *J Optim Theory Appl* 19(2):261–281
- Djelassi H, Glass M, Mitsos A (2019) Discretization-based algorithms for generalized semi-infinite and bilevel programs with coupling equality constraints. *J Global Optim* 75(2):341–392
- Floudas CA, Stein O (2007) The adaptive convexification algorithm: A feasible point method for semi-infinite programming. *SIAM J Optim* 18(4):1187–1208
- Guerra Vázquez F, Rückmann JJ (2005) Extensions of the Kuhn-Tucker constraint qualification to generalized semi-infinite programming. *SIAM J Optim* 15(3):926–937
- Harwood SM, Barton PI (2017) How to solve a design centering problem. *Math Methods Oper Res* 86(1):215–254
- Jaklič A, Leonardi A, Solina F (2000) Segmentation and recovery of superquadrics, computational imaging and vision, vol 20. Kluwer Academic Publishers, Dordrecht
- Jongen HT, Rückmann JJ, Stein O (1998) Generalized semi-infinite optimization: a first order optimality condition and examples. *Math Program* 83:145–158
- Kaplan A, Tichatschke R (1997) On the numerical treatment of a class of terminal problems. *Optimization* 41:1–36
- Kirst P, Stein O (2019) Global optimization of generalized semi-infinite programs using disjunctive programming. *J Global Optim* 73(1):1–25
- Küfer KH, Maag V, Schwientek J (2015) Maximal material yield in gemstone cutting. In: Neunzert H, Prätzel-Wolters D (eds) *Currents in industrial mathematics: from concepts to research to education*. Springer, Berlin, pp 229–290
- Küfer KH, Stein O, Winterfeld A (2008) Semi-infinite optimization meets industry: a deterministic approach to gemstone cutting. *SIAM News* 41(8):
- Lemonidis P (2008) Global optimization algorithms for semi-infinite and generalized semi-infinite programs. Ph.D. thesis, Massachusetts Institute of Technology
- López M, Still G (2007) Semi-infinite programming. *Eur J Oper Res* 180(2):491–518
- Mitsos A, Tsoukalas A (2015) Global optimization of generalized semi-infinite programs via restriction of the right hand side. *J Global Optim* 61(1):1–17
- Reemtsen R (1991) Discretization methods for the solution of semi-infinite programming problems. *J Optim Theory Appl* 71(1):85–103
- Reemtsen R (1994) Some outer approximation methods for semi-infinite optimization problems. *J Comput Appl Math* 53(1):87–108
- Reemtsen R, Goerner S (1998) Numerical methods for semi-infinite programming: a survey. In: Reemtsen R, Rückmann JJ (eds) *Semi-infinite programming, nonconvex optimization and its applications*, vol 25. Kluwer Academic Publishers, Boston, pp 195–275
- Rückmann JJ, Shapiro A (2001) Second-order optimality conditions in generalized semi-infinite programming. *Set-Valued Anal* 9(1–2):169–186
- Schwientek J (2013) Modellierung und Lösung parametrischer Packungsprobleme mittels semi-infiniten Optimierung—Angewandt auf die Verwertung von Edelsteinen. Ph.D. thesis, TU Kaiserslautern
- Stein O (2006) A semi-infinite approach to design centering. In: Dempe S, Kalashnikov V (eds) *Optimization with multivalued mappings: theory, applications and algorithms*, springer optimization and its applications, vol 2. Springer, New York, pp 209–228

- Steuer mann HP (2011) Adaptive algorithms for semi-infinite programming with arbitrary index sets. Ph.D. thesis, Karlsruher Institut für Technologie (KIT)
- Still G (1999) Generalized semi-infinite programming: theory and methods. *Eur J Oper Res* 119:301–313
- Still G (2001) Discretization in semi-infinite programming: the rate of convergence. *Math Program* 91:53–69
- Still G (2001) Generalized semi-infinite programming: numerical aspects. *Optimization* 49:223–242
- Weber GW (1999) Generalized semi-infinite optimization: on some foundations. *J Comput Sci Technol* 4:41–61
- Winterfeld A (2008) Application of general semi-infinite programming to lapidary cutting problems. *Eur J Oper Res* 191:838–854

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.