Escaping unknown discontinuous regions in blackbox optimization

Charles Audet, Alain Batailly, Solène Kojtych

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

LABORATOIRE D'ANALYSE VIBRATOIRE ET ACOUSTIQUE

Escaping unknown discontinuous regions in blackbox optimization

Charles Audet[1], Alain Batailly[2], Solène Kojtych[2]

Abstract

The design of key nonlinear systems often requires the use of expensive blackbox simulations presenting inherent discontinuities whose positions in the space of variables cannot be analytically predicted. Without further precautions, the solution of related optimization problems leads to design configurations which may be close to discontinuities of the blackbox output functions. These discontinuities may betray unsafe regions of the design space, such as nonlinear resonance regions. To account for possible changes of operating conditions, an acceptable solution must be away from unsafe regions of the space of variables. The objective of this work is to solve a constrained blackbox optimization problem with the additional constraint that the solution should be outside unknown zones of discontinuities or strong variations of the objective function or the constraints. The proposed approach is an extension of the Mesh Adaptive Direct Search (Mads) algorithm and aims at building a series of inner approximations of these zones. The algorithm, called DiscoMads, relies on two main mechanisms: revealing discontinuities and progressively escaping the surrounding zones. A convergence analysis supports the algorithm and preserves the optimality conditions of Mads. Numerical tests are conducted on analytical problems and on three engineering problems illustrating the following possible applications of the algorithm: the design of a simplified truss, the synthesis of a chemical component and the design of a turbomachine blade. The DiscoMads algorithm successfully solves these problems by providing a feasible solution away from discontinuous regions.

Keywords

blackbox optimization, discontinuous functions, mesh adaptive direct search

1 - GERAD and Département de Mathématiques et Génie industriel, Polytechnique Montréal, C.P. 6079, Succ. Centre-ville, Montréal (Québec), H3C 3A7, Canada (https://www.gerad.ca/Charles.Audet/).
2 - Départment de Génie Mécanique, Polytechnique Montréal, Montréal, (Québec) H3C 3A7, Canada.

Exclusion des zones de discontinuitées inconnues en optimisation de boîtes noires

Charles Audet[1], Alain Batailly[2], Solène Kojtych[2]

Résumé

La conception de systèmes non linéaires clés requiert généralement l'utilisation de simulations de type boîtes noires, coûteuses en temps de calculs. Ces simulations peuvent présenter des discontinuités inhérentes dont la position dans l'espace des variables ne peut être prédite analytiquement. Sans précautions supplémentaires, la solution des problèmes d'optimisation associés peut correspondre à une configuration proche des discontinuités des sorties de la boîte noire. Or, ces discontinuités peuvent révéler des régions critiques de l'espace de conception, telles que des résonances non linéaires. Afin de prendre en compte de possibles changements des conditions opératoires, une solution acceptable doit être située loin de ces régions critiques. L'objectif de ce travail est de résoudre un problème d'optimisation de boîte noire contraint avec une contrainte additionnelle imposant que la solution doit être en dehors des zones inconnues de discontinuités ou de fort accroissement de la fonction objectif ou des contraintes. L'approche proposée est une extension de l'algorithme de recherche directe sur treillis adaptatif (Mads, de l'anglais Mesh Adaptive Direct Search) et vise à construire une série d'approximations intérieures de ces zones. L'algorithme proposé, nommé DiscoMads, repose sur deux mécanismes principaux : la révélation des discontinuités et l'exclusion progressive des zones avoisinantes. L'algorithme est supporté par une analyse de convergence qui préserve les conditions d'optimalité de Mads. Des tests numériques sont effectués sur des problèmes analytiques et sur trois problèmes d'ingénierie : le design d'un treillis simplifié, la synthèse d'un composant chimique et la conception d'une aube de turbomachine. Ces problèmes sont résolus avec succès par l'algorithme DiscoMads qui fournit une solution réalisable loin des zones de discontinuités.

Mots-clés

optimisation de boîtes noires, fonctions discontinues, recherche directe sur maillage adaptatif

1 - GERAD et Département de Mathématiques et Génie industriel, Polytechnique Montréal, C.P. 6079, Succ. Centre-ville, Montréal (Québec), H3C 3A7, Canada (https://www.gerad.ca/Charles.Audet/).

2 - Départment de Génie Mécanique, Polytechnique Montréal, Montréal, (Québec) H3C 3A7, Canada.

1 Introduction

Motivation

In several engineering domains, such as the power, automative and aerospace industries, a highly competitive global economical context and stringent environmental requirements have forced engineers to push the limits in the design of a wide variety of complex mechanical systems. Nonlinear mechanical phenomena that have long been avoided, oftentimes by means of design solutions increasing the system's environmental footprint, must now be fully accounted for and prevented with appropriate design strategies. The intricacy of designing nonlinear mechanical systems is notably related to the fact that they may exhibit a variety of coexisting stable solutions, featuring very distinct magnitudes of vibration. In these regions, a small variation of operating conditions can induce a bifurcation of the system, *i.e.* a change from one configuration to another [38]. Depending on the nature of the nonlinear phenomena (cubic force, friction, contacts, etc.), bifurcations are characterized by various changes in the dynamics. For some systems involving contact interactions, such as blade-casing interactions in turbomachinery [31], regions of coexistent solutions may reveal the presence of a nonlinear resonance, with high amplitudes of vibration. These regions of the design space, called *nonlinear resonance regions*, must be detected and avoided to ensure safe mechanical designs.

From a mechanical standpoint, the prediction and understanding of nonlinear phenomena has long been the focus of researchers. Regarding numerical aspects, qualitative methods [27] are able to identify nonlinear resonance regions, but they are focused on steady state phenomena and their use is problematic in the analysis of large-scale systems. Indeed, numerical challenges prevent these methods from providing design-specific guidelines. Consequently, quantitative methods (such as time integration) are the standard in an industrial context for the design of nonlinear systems. As there exists no unified theoretical framework for comprehensively analyzing nonlinear mechanical systems, these methods are integrated into *ad hoc* industrial tools. Considering the high degree of maturity reached by these tools [34], an emerging challenge is integrating these methods into automated design processes. Indeed, despite their accuracy, it remains challenging to use them for the detection of non-linear resonance regions. As a consequence, it is essential for engineers to provide configurations accounting for a safe margin with respect to the position of nonlinear resonance regions. The present work deals with the search for an optimization strategy providing safe configurations away from nonlinear resonance regions, such as those detected by current industrial tools.

From a mathematical standpoint, two aspects need to be taken into account. First, nonlinear resonance regions are characterized by discontinuities of numerical quantities, because of abrupt changes between two coexistent solutions. These discontinuities are deterministic, their magnitudes are generally much larger than the floating point precision and their positions cannot be predicted *a priori*. Moreover, the position of the computed discontinuity does not accurately match the exact position of the nonlinear resonance region. For security reasons, a solution away from discontinuous regions is thus required. Second, related nonlinear numerical simulations often rely on time-consuming multiphysics models in which gradients of functions are nonexistent or difficult to estimate; they can thus be considered as blackboxes [10]. As these simulations go beyond the capabilities of commercial software packages, they are not integrated within automated design frameworks. Consequently, noninvasive optimization methods are required during the design process.

The following constrained optimization problem is considered, where the objective function and the constraints are returned by a blackbox :

$$\begin{aligned} \min_{x\in\Omega} \quad & f(x) \\ \text{s.t.} \quad & d(x) \leq 0, \end{aligned} \tag{1.1}$$

where $\Omega = \{x \in X : c(x) \leq 0\}$ and X is a subset of $\mathbb{R}^n$ defined by unrelaxable constraints [30]. The components of the vector $c : X \to \overline{\mathbb{R}}^m$, where $\overline{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$, of relaxable constraints are denoted by $c_j, j \in \{1, ..., m\}$ and the objective function $f : X \to \overline{\mathbb{R}}$ is denoted c_0 in some context. In addition, one constraint $d : X \to \mathbb{R}$ is introduced to ensure that x is away from discontinuities of the user-selected output functions from the subset of indices $J \subseteq \{0, 1, ..., m\}$.

Literature review

Generally, industrial design processes of complex nonlinear mechanical systems are confidential and unpublished. Moreover, as the accounting of nonlinear phenomena in these processes is an emerging challenge, to the best of our knowledge there exists no reference method for solving Problem (1.1) within an industrial context. This problem could be tackled as a robust optimization problem where the variability of the quantities of interest must be minimized, or as a blackbox optimization problem [10]. In both frameworks, additional developments would be required to solve the problem. However, the computation times associated with nonlinear simulation tools, as well as the need for noninvasive optimization methods, makes blackbox optimization particularly well suited to this case. For these reasons, problem (1.1) is tackled in the present work from a blackbox optimization standpoint.

Blackbox optimization methods include heuristics, such as genetic algorithms, widely used in an industrial context [16], and algorithms supported by a convergence analysis, such as direct search methods or trust region methods. These methods benefit from convergence guarantees with respect to the smoothness of the functions. Optimality conditions have also been derived for discontinuous functions [39] for some methods. Among them, the Mesh Adaptive Direct Search (Mads) [7] algorithm has been recently used in the design of a complex nonlinear engineering system [28]. Some methods dedicated to discontinuous optimization include the general penalty-based approach proposed by Birgin, Krejić, and Martínez [15] in which the discontinuous objective function is replaced by a sequence of continuous approximations, and the McCormick relaxation-based branch-and-bound global optimization approach of Wechsung and Barton [40] for factorable discontinuous objective functions that include IF-THEN-ELSE statements. However, both approaches do not give control on the distance to the discontinuity and do not treat discontinuities in the constraints.

An important difficulty of problem (1.1) lies in the treatment of the constraint d. Indeed, as the discontinuities in X are not analytically known, the value of d at a point x cannot be computed from a single evaluation of the blackbox. The constraint d requires an infinite number of evaluations to be deemed feasible. Infinite constraints are encountered in some fields of robust optimization such as reliability-based design optimization [33] where the solution must respect a limit probability of failure to be acceptable. Methodologies in this context rely on an approximation of the probability. To the best of our knowledge, the treatment of infinite constraints in blackbox robust optimization is limited to minimax problems, such as the iterative outer approximation approach proposed by Menickelly and Wild [32].

A straightforward strategy for accurately approximating d consists of performing a large number of evaluations in some neighborhood. With costly blackboxes, this is only possible if the evaluations are performed on surrogate models of f and c. Surrogates are widely used in contexts where the simulation cost is prohibitive, such as blackbox optimization or uncertainty quantification. A global approximation may be used as a surrogate, but the basis of interpolation functions must be carefully chosen [1, 36] to limit oscillations and control the accuracy in the vicinity of the discontinuity. A more common strategy consists of constructing local continuous approximations after an effective localization of the discontinuities [3, 19, 35, 37].

A suitable strategy in the studied context is to directly compute an approximation of d from a limited number of true evaluations of the blackbox. An accurate localization of the discontinuities is required in this case as well. Some methods build a parametric expression of the position of the discontinuity [2, 37] or an implicit characterization [21, 35] requiring the labeling of points close to the discontinuity. This labeling may be done from the values of the functions only [19], which requires a global sampling of the solutions. A local labeling of discontinuous regions is preferred when a limited number of samples is available. In this case, the labeling requires two parameters quantifying a neighborhood and a limit variation of the functions over this neighborhood [14, 16, 23]. Other methods also use some approximation coefficients, influenced by oscillations of models in the vicinity of discontinuities [3, 25]. In any cases, for sparse data, discontinuities and strong gradients cannot be distinguished numerically with this labeling. Sargsyan *et. al.* [37] define *weak discontinuities* as discontinuities or sharp gradient regions; this definition is used in the present work.

Modeling

Solving problem (1.1) in a blackbox optimization framework requires numerically quantifying discontinuities. The proposed approach detects weak discontinuities by using two parameters. An unsafe region D of weak discontinuities

is introduced and based on a limit rate of change $\tau > 0$ on an open ball of radius $r_{\mathrm{d}} \geq 0$,

$$D = \{y \in X \ : \ \exists j \in J, \ \exists z \in X \cap B_{r_{\mathrm{d}}}(y), \ |c_j(y) - c_j(z)| > \tau \|y - z\|\},$$

where $B_{r_{\mathrm{d}}}(y)$ is the open ball of radius r_{d} centered at y. In the situation where the output functions are piecewise differentiable and L is a Lipschitz constant for all the output functions indexed in J, suppose that $\tau > L$ and that a point y belongs to D (with a point z satisfying the inequality for that y). Then, there necessarily exists a discontinuity between y and z for one of the output functions of index $j \in J$. In other cases, D also contains areas without discontinuities but undergoing strong variations of one of the output functions indexed in J. Consequently, weak discontinuous regions of full dimension in X may be detected with the proposed modeling. An example for a region D around a single discontinuity is depicted in Figure 1.

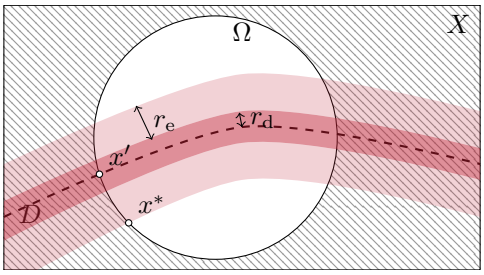


Figure 1. Example of problem (1.2) with a discontinuity (- - -). Regions D (■), M (■ ∪ ■), and $\hat{\Omega}$ (□) and infeasible domain for c (▨). Solutions x' and x^* of the problem, respectively without and with the constraint d.

As the region D is defined by the two parameters r_{d} and τ, the exact position of discontinuities within D is unknown. To account for the remoteness of a solution to discontinuities, a safe margin region $M = X \cap (\cup_{x \in D} B_{r_{\mathrm{e}}}(x))$ is introduced, where $r_{\mathrm{e}} > 0$ is a user-defined remoteness parameter. The margin M contains the points of Ω which should be discarded in the resolution of problem (1.1).

Finally, the optimization problem can be stated as

$$\min_{x \in \hat{\Omega}} \quad f(x) \tag{1.2}$$

where $\hat{\Omega} = \Omega \setminus M$ is the ideal feasible region. An example for the layout of the regions is depicted in Figure 1. The solution of problem (1.2) is depicted by the point x^* whereas the point x' is the solution of the minimization problem of f on Ω. Problems (1.2) and (1.1) are equivalent if the function d is defined so that $d(x) \leq 0 \iff x \notin M$.

The proposed approach solves problem (1.2) in a blackbox optimization context by building a series of inner approximations of the margin M. The algorithm is called DiscoMads because it is based on the Mads algorithm, which is well suited to incorporating necessary detection mechanisms, and can escape weak Discontinuous regions. When the parameter r_{d} is fixed to 0 by the user, the region D is empty and the behavior of the algorithm is identical to Mads. The algorithm relies on two original mechanisms: a procedure for revealing weak discontinuities in the space of variables and a mechanism for excluding the revealed areas from the feasible region progressively. The main original contribution of this work is an algorithm with proven convergence, providing a solution that is safe from weak discontinuities of the blackbox output, whose positions in the space of variables cannot be described analytically.

Although the algorithm is originally designed to find a solution away from discontinuities, the modeling (1.2) and the flexibility in the choice of the parameters enable us to search for a solution lying in a region where variations of the functions are bounded. Consequently, numerical tests illustrate possible applications of the algorithm with different choices of parameters alongside a turbomachine design problem illustrating the primary motivation for the algorithm.

The rest of this work is organized as follows. The classical Mads algorithm is recalled in section 2, the DiscoMads algorithm is described in section 3, and convergence results are given in section 4. Numerical results are given in section 5 for analytical problems and on several engineering applications. Limitations of the approach as well as future research directions are discussed in section 6.

2 Mesh Adaptive Direct Search

Mads [5, 7] is an iterative algorithm which only relies on the values of the blackbox output functions and not the derivatives.

2.1 Mads for unconstrained problems

Starting from $x^0 \in X$, the algorithm generates a sequence of trial points at which the blackbox should be evaluated. Let x^k be the *incumbent solution* at iteration k, the feasible solution with the best f-value. The aim of an iteration is to find a new incumbent with a better value of f.

All points evaluated by the algorithm at iteration k must lie on a *mesh* defined by a mesh size parameter $\delta^k \in \mathbb{R}^n_+$:

$$M^k := \{x^k + \delta^k \mathbb{D} y : y \in \mathbb{N}^{2n}\},$$

where $\mathbb{D} = [I \ -I] \in \mathbb{R}^{n \times 2n}$ for the present work and $I \in \mathbb{N}^{n \times n}$ is the identity matrix.

Each Mads iteration is divided into two steps: an optional and flexible *search step*, followed by a local rigorously defined *poll step*. During the search step, the blackbox may be evaluated at a finite number of points $S^k \subset M^k$ arbitrarily chosen by the user, allowing for a more global exploration of the space of variables. If the search step fails, the poll step is conducted to evaluate points around the poll center x^k. The set of poll trial points $P^k \subset M^k$ (defined in [7]) contains points at a distance at most Δ^k of x^k, where $\Delta^k \in \mathbb{R}^n_+$ is the *poll size parameter* such as $\delta^k = \min(\Delta^k, (\Delta^k)^2)$.

If both the search and the poll steps fail to find a new incumbent, then the iteration is said to be *unsuccessful*, and the parameters δ^k and Δ^k are reduced [7]. Otherwise, the iteration is said to be *successful*, and these parameters are increased or kept unchanged, so that the next mesh is at least as coarse as the previous one. The next iterate x^{k+1} is the new incumbent. During an iteration, when evaluating successively the points of first S^k and then P^k, if a new incumbent is found, the iteration may stop opportunistically without evaluating the remaining points.

In Mads, the normalized poll directions grow dense in the unit sphere as the number of iterations increases. A hierarchy of optimality conditions [7] derived from Clarke nonsmooth calculus [17] is given for the point where the algorithm converges, depending on the smoothness of blackbox output functions.

2.2 Mads for constrained problems

Mads is able to treat constraints in optimization problems with two approaches: the extreme barrier [7], which rejects all infeasible points, and the progressive barrier [5], which relies on a threshold on the constraint violation to reject infeasible points. A distinction is made between the types of constraints: unrelaxable constraints [30] defining X should be treated with the extreme barrier, whereas relaxable constraints $c(x)$ can be treated with both approaches. As the developed algorithm requires the use of the progressive barrier for at least one of the constraints, more details are given about this approach.

In the progressive barrier approach, the violation of the relaxable constraints is quantified by the constraint aggregation function $h(x) : \mathbb{R}^n \to \overline{\mathbb{R}}_+$ [20] defined by:

$$h(x) = \begin{cases} \sum_{j=1}^{m} \max(c_j(x), 0)^2 & \text{if } x \in X, \\ \infty & \text{otherwise.} \end{cases}$$

If $h(x) = 0$, then x is Ω-feasible since it respects both relaxable and unrelaxable constraints. Otherwise, if $0 < h(x) < \infty$, then x respects the unrelaxable constraints but not all the relaxable constraints ($x \in X \setminus \Omega$). Based on f and h, an ordering of the points is defined thanks to the relation of *dominance* (Definition 12.2 in [10]):

Definition 2.1 (Dominated points in constrained optimization). *The feasible point $x \in \Omega$ is said to dominate $y \in \Omega$ when $f(x) < f(y)$. The infeasible point $x \in X \setminus \Omega$ is said to dominate $y \in X \setminus \Omega$ when $f(x) \leq f(y)$ and $h(x) \leq h(y)$ with at least one strict inequality. A point x in some set $S \subset X$ is said to be undominated if there are no $y \in S$ that dominate x.*

At the beginning of iteration k, two sets of incumbent solutions are built. Let $\mathbb{F}^k$ denote the set of feasible incumbent solutions at iteration k, and let $\mathbb{I}^k$ denote the set of infeasible incumbent solutions at iteration k,

$$\mathbb{F}^k = \underset{x \in V^k}{\arg\min}\{f(x) : h(x) = 0\}, \qquad \mathbb{I}^k = \underset{x \in \mathbb{U}^k}{\arg\min}\{f(x) : 0 < h(x) < h^k_{\max}\},$$

where V^k is the cache (the set of points previously evaluated by the algorithm), $\mathbb{U}^k$ is the set of infeasible undominated points at the beginning of evaluation k, and $h^k_{\max}$ is a rejection threshold updated at each iteration. The poll step is conducted around both a feasible point of $\mathbb{F}^k$ and an infeasible point of $\mathbb{I}^k$ when possible. Different decision rules are applied for the choice of the poll center [5].

Three iteration types are distinguished in Mads with the progressive barrier. The iteration is *dominating* if there exists a trial point dominating an incumbent belonging to $\mathbb{F}^k$ or $\mathbb{I}^k$. If the iteration is not dominating but there exists a trial point with a better f value than the infeasible incumbent, then the iteration is *improving*. Otherwise, the iteration is *unsuccessful*. Depending on the iteration type, the parameters δ^k and Δ^k and the barrier threshold h^k_{max} are updated [5].

3 Algorithmic approach

DiscoMads is directly based on the Mads algorithm but includes two new algorithmic mechanisms: a revelation mechanism used after each evaluation (subsection 3.1) and an exclusion mechanism (subsection 3.2) used when a weak discontinuity is revealed. To cover this case, a new type of iteration is introduced with specific barrier and mesh parameter updates (subsection 3.3). Finally, a revealing poll is included in the poll step in addition to the classical poll of Mads to ensure a systematic detection mechanism (subsection 3.4).

3.1 Mechanism for revealing discontinuities

The discontinuity revelation requires the following user-defined parameters: the detection radius $r_{\mathrm{d}} \geq 0$ and the bound $\tau > 0$ on the rate of change for the revealing output functions of indices $J \subseteq \{0, 1, ..., m\}$. The revelation phase is performed after each successful evaluation of the blackbox at a point $y \in X$. For each previous point $z \neq y$ successfully evaluated in $B_{r_{\mathrm{d}}}(y)$, the rate of change τ_j of each revealing output function $j \in J$ is computed as

$$\tau_j(y, z) = \frac{|c_j(y) - c_j(z)|}{\|y - z\|}.$$

If $\tau_j(y, z) > \tau$, a weak discontinuity is revealed for the output function c_j thanks to the points y and z. These points are marked as *revealing points*.

Definition 3.1 (Revealing point). *A cache point $y \in V^k$ is said to be* revealing *for the blackbox output function of index $j \in J$ if $y \in X$ and if there exists a point $z \in V^k \cap X \cap B_{r_d}(y)$ such that $|c_j(y) - c_j(z)| > \tau\|y - z\|$.*

Let the following D^k denote the set of revealing points at the start of iteration k, with respect to all the revealing blackbox output functions:

$$D^k = \left\{y \in V^k \cap X : \exists z \in V^k \cap X \cap B_{r_{\mathrm{d}}}(y), \exists j \in J, |c_j(y) - c_j(z)| > \tau\|y - z\|\right\},$$

where V^k is the cache at the start of iteration k. As revealing points indicate areas with a high rate of change, $D^k \subseteq D$. Moreover, as long as new revealing points are discovered, the set of revealing points is enriched, and thus: $D^k \subseteq D^{k+1}$.

3.2 Mechanism for circumventing revealed areas

From iteration k, any feasible solution $x \in X$ of problem (1.2) should satisfy, for all $y \in D^k$,

$$\|x - y\| \geq r_{\mathrm{e}} \iff r_{\mathrm{e}} - \|x - y\| \leq 0 \iff r_{\mathrm{e}} - \mathrm{dist}(x, D^k) \leq 0, \tag{3.1}$$

where $\text{dist}(x, D^k) = \inf_{y \in D^k}(\|x - y\|)$ is the distance from x to the set D^k. As a consequence, an additional constraint $d^k(x) : X \to \mathbb{R}$ is introduced to quantify the remoteness of a point x from the set D^k at iteration k:

$$d^k(x) = \begin{cases} 1 - \frac{\text{dist}(x, D^k)}{r_e} & \text{if} \quad D^k \cap B_{r_e}(x) \neq \emptyset, \\ 0 & \text{otherwise.} \end{cases} \tag{3.2}$$

On the one hand, if there are no revealing points in $B_{r_e}(x)$ at iteration k, then $d^k(x) = 0$; the point x is not considered in the vicinity of a weak discontinuity at this stage. On the other hand, if there exists a revealing point from D^k in $B_{r_e}(x)$, then x is in an unsafe region and is penalized: thus $d^k(x) > 0$. It is worth noting that only the point $y \in D^k \cap B_{r_e}(x)$ which is the nearest to x, influences the value $d^k(x)$. The denominator r_e is used to scale d^k between 0 and 1.

The problem considered by DiscoMads at iteration k may be written as

$$\min_{x \in \Omega^k} \quad f(x), \tag{3.3}$$

with $\Omega^k = \{x \in X : c(x) \leq 0, d^k(x) \leq 0\}$ the set of feasible solutions at iteration k. The constraint $d^k(x)$ is quantifiable and relaxable [30] and is treated with the progressive barrier approach developed for Mads [5]. In our context, the constraint violation aggregation function $h^k : \mathbb{R}^n \to \overline{\mathbb{R}}_+$ used for the progressive barrier and the dominance relation depends on the iteration k and is defined by:

$$h^k(x) = \begin{cases} \sum_{j=1}^{m} \max(c_j(x), 0)^2 + \max(d^k(x), 0)^2 & \text{if } x \in X, \\ \infty & \text{otherwise.} \end{cases} \tag{3.4}$$

3.3 Revealing iteration

After the evaluation of a point y at iteration k, if the revealing procedure of subsection 3.1 returns at least one new revealing point $z \notin D^k$, the iteration is declared *revealing*. The evaluations at iteration k are stopped opportunistically to avoid wasting evaluations in the unfavorable revealed area. Let R denote the set of new revealing points, and let Y be the set of points evaluated at iteration k. The set of revealing points and the cache for the next iteration are updated as

$$D^{k+1} \leftarrow D^k \cup R \quad \text{and} \quad V^{k+1} \leftarrow V^k \cup Y. \tag{3.5}$$

Then, the constraints d^{k+1} and h^{k+1} are computed for all cache points $V^{k+1} \cap X$ with (3.2) and (3.4). From a numerical standpoint, only the points whose distance to R is less than r_e need to be updated. In a blackbox optimization context, the computation time required for the revealing procedure and the potential update of constraints is negligible with respect to the running time of the blackbox. Additionally, as $D^k \subseteq D^{k+1}$, for all x in X the following equation is satisfied:

$$d^{k+1}(x) \geq d^k(x) \quad \Longrightarrow \quad h^{k+1}(x) \geq h^k(x).$$

In DiscoMads the constraint violation aggregation function h^k is used, the set of infeasible undominated points at iteration k may thus differ from the one used in Mads and is denoted U^k. As the value of h^k may increase with respect to k, the set U^{k+1} may be different from U^k, although no dominating point can be found in a revealing iteration. The set U^{k+1} is nonempty because it contains at least a revealing point of $R \neq \emptyset$. However, it is possible that all points of V^{k+1} exceed the barrier threshold $h^k_{\max}$. Let $I^k = \arg\min_{x \in U^k}\{f(x) : 0 < h^k(x) < h^k_{\max}\}$ denote the set of infeasible incumbent solutions (undominated points), and let $F^k = \arg\min_{x \in V^k}\{f(x) : h^k(x) = 0\}$ be the set of feasible incumbent solutions at iteration k. These sets are redefined with respect to Mads as they are based on the values of h^k, but they are used in the same way for the choice of the poll center. Due to a possible update of the

constraint d^k, F^{k+1} may be empty. It is essential to ensure that I^{k+1} is nonempty to continue the algorithm at iteration $k+1$. As a consequence, the threshold $h^{k+1}_{\max}$ must be chosen carefully.

Define $N(k, h_{\lim})$ as the number of infeasible undominated points at iteration k whose value of h^k is less than the threshold $h_{\lim} > 0$:

$$N(k, h_{\lim}) \quad \text{defined as} \quad \#\left\{x \in U^k : h^k(x) \leq h_{\lim}\right\},$$

where $\#$ is the cardinality of the set. Let $\overline{x} \in U^{k+1}$ denote the unique infeasible undominated point such as

$$N(k+1, h(\overline{x})) = \begin{cases} \min\left(N(k, h^k_{\max}), \#\{U^{k+1}\}\right) & \text{if} \quad N(k, h^k_{\max}) \neq 0, \\ \#\left\{U^{k+1}\right\} & \text{otherwise.} \end{cases}$$

As $U^{k+1} \neq \emptyset$, there always exists a point $\overline{x}$ satisfying these equations and $N(k+1, h(\overline{x})) \neq 0$. The value $h^{k+1}_{\max}$ is chosen here to keep, when possible, the same number of infeasible undominated points below the barrier threshold before and after the detection of a revealing point. This value is computed with the following equation:

$$h^{k+1}_{\max} = h(\overline{x}). \tag{3.6}$$

Finally, at the end of a revealing iteration, the mesh size and frame size parameters are updated as follows:

$$\delta^{k+1} = \delta^k \quad \text{and} \quad \Delta^{k+1} = \Delta^k. \tag{3.7}$$

3.4 Revealing poll

The poll step of Mads is enriched to solve problem (1.2). Indeed, when DiscoMads converges toward a promising solution due to repeated unsuccessful iterations, the frame size parameter Δ^k is decreased. When $\Delta^k < r_{\mathrm{d}} + r_{\mathrm{e}}$, the classical poll of Mads can no longer detect revealing points at a sufficient distance to ensure the convergence results described in section 4.

An additional revealing poll is defined when $r_{\mathrm{d}} > 0$ (if $r_{\mathrm{d}} = 0$ then the forbidden area D is empty). Let C^k stand for a set of points defined by:

$$C^k = \begin{cases} \{x_F\} & \text{where} \quad x_F \in F^k & \text{if } F^k \neq \emptyset, \\ \{x_I\} & \text{where} \quad x_I \in I^k & \text{otherwise.} \end{cases}$$

To construct the set of additional trial points $P^k_+ \subset M^k$, $n_{\mathrm{rnd}} \geq 1$ point(s) are randomly generated in the ball of radius $r_{\mathrm{m}} > r_{\mathrm{d}} + r_{\mathrm{e}}$ centered at a point of C^k and then rounded onto the current mesh M^k. During an execution of DiscoMads, the points of P^k_+ are evaluated with an opportunistic strategy before the points P^k that are generated by the classical poll of Mads.

The complete DiscoMads algorithm is shown in Figure 2. The blackbox output functions $c(x)$ related to constraints (including revealing functions) can be handled with either the extreme or the progressive barrier approach. By default in Mads, the constraints treated with the extreme barrier approach contribute to the definition of X. If a point does not belong to X, it is rejected by DiscoMads and treated the same way as in the Mads algorithm. This treatment is not mentioned in Figure 2 and the reader is referred to the Mads algorithm [7] for further information. Consequently, such a point is not used to detect weak discontinuities and cannot be a revealing point.

4 Convergence analysis

The convergence analysis follows the same steps as the ones carried out for Mads [5, 7]. Under appropriate assumptions, it is first shown that the mesh size parameter gets infinitely fine. It follows that DiscoMads generates a refining point $\hat{x}$. Depending on the position of $\hat{x}$ in X, local optimality conditions are derived from Clarke nonsmooth calculus [17]. The original contribution of the analysis lies in Theorem 4.14 which covers the case where $\hat{x}$ belongs to the margin M.

DiscoMads: Revealing and escaping discontinuous regions

- initialization (given a starting point $x^0 \in X$ such as $f(x^0) < \infty$, $h(x^0) < \infty$):
  - define problem parameters: $r_\mathrm{d} \geq 0$, $\tau > 0$, $r_\mathrm{e} > 0$, and $J \subseteq \{0, 1, ..., m\}$;
  - define the revealing poll parameters $n_\mathrm{rnd} \geq 1$ and $r_\mathrm{m} > r_\mathrm{e} + r_\mathrm{d}$;
  - define the usual Mads parameters and the starting cache $V^0 \leftarrow \{x^0\}$;
  - let $D^0 = \emptyset$ and set the iteration counter $k \leftarrow 0$.
- preparation: generate a set S^k of user-defined points for the search (optional), generate the revealing poll set P_+^k (subsection 3.4) and the classical poll set P^k according to the Mads algorithm [5] and let the cache $V^{k+1} \leftarrow V^k$.
- evaluation: for each successfully evaluated point $x \in S^k \cup P_+^k \cup P^k$, if $x \in X$:
  - compute $d^k(x)$ according to (3.2) and $h^k(x)$ according to (3.4);
  - update the cache $V^{k+1} \leftarrow V^{k+1} \cup \{x\}$;
  - conduct the revelation procedure (subsection 3.1) with the cache V^{k+1} for the output functions $j \in J$,
  - if a new revealing point is detected, the iteration is revealing: go to update.
  - if x is improving or dominating, the evaluation may stop opportunistically: go to update.
- update:
  - if the iteration is revealing, compute the set of revealing points D^{k+1} according to (3.5); for all $v \in V^{k+1}$, update $d^{k+1}(v)$ and $h^{k+1}(v)$ according to (3.2) and (3.4); update the barrier threshold h_max^{k+1} as in (3.6) and the mesh parameters δ^{k+1} and Δ^{k+1} as in (3.7);
  - otherwise, classify the iteration and update the parameters as in Mads [5] considering the constraint violation function h^k;
  - if no stopping criterion is met, $k \leftarrow k + 1$ and go back to preparation.

Figure 2. DiscoMads for revealing and escaping discontinuous regions.

4.1 Preliminaries

In the studied context, the blackbox output is generated from deterministic functions of $\mathbb{R}^n$ in $\mathbb{R}$. The starting point does not need to be $\hat{\Omega}$-feasible and may be in the unsafe area M. It should only satisfy Assumption 4.1 required by the progressive barrier [7] to order points with the relation of dominance.

Assumption 4.1. There exists a user-provided initial point x_0 such that $x^0 \in X$ and that both $f(x^0)$, $h(x^0)$ are finite.

As the unsafe region is unknown at the beginning of the optimization ($D^0 = \emptyset$), then $h^0(x^0) = h(x^0)$ and Assumption 4.1 ensures that $h^0(x^0)$ is finite. More generally, for any points $x \in X$, if $h(x)$ is bounded, then $h^k(x)$ is bounded for all k as $d^k(x)$ is bounded by 1. As the iterates produced by the algorithm may be unbounded, Assumption 4.2 from Mads [5] is considered.

Assumption 4.2. All points considered by the algorithm lie in a bounded set.

In the case where the set X defined by unrelaxable constraints is bounded, then given Assumption 4.1, Assumption 4.2 is satisfied. Additionally, a bounded X may be easily defined for a large part of engineering problems in which the variables are necessarily bounded. An assumption on the continuity of the output functions is also necessary to account for the margin M: a tailored definition of piecewise continuity (Definition 4.3) is introduced for Assumption 4.4.

Definition 4.3 (Piecewise continuous function). *The function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be* piecewise continuous *if there exists a finite subset of indices $K \subseteq \mathbb{N}$ and a set of open sets $\{X_i\}_{i \in K}$ satisfying*

$$\begin{cases} X_i \cap X_j = \emptyset & \forall i \in K, \forall j \in K, j \neq i, \\ \cup_{i \in K} \overline{X}_i = \overline{X}, & \\ f|_{X_i} \in \mathscr{C}^0 & \forall i \in K, \end{cases}$$

and if, for all $y \in X$, there exists $i \in K$ such that $y \in \overline{X}_i$ and $f|_{X_i \cup \{y\}} \in \mathscr{C}^0$, where $f|_A$ denotes the restriction of f on a set A and $\overline{X}$ is the closure of X.

Assumption 4.4. The blackbox output functions are piecewise continuous on X.

An explicit expression for the remoteness constraint d of problem (1.1) is given at this point, mirroring the expression of d^k of problem (3.3) solved by DiscoMads at each iteration. The absolute constraint $d : X \to \mathbb{R}$ indicating if a point x belongs to the margin M is then defined by:

$$d(x) = \begin{cases} 1 - \frac{\text{dist}(x,D)}{r_e} & \text{if} \quad D \cap B_{r_e}(x) \neq \emptyset, \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

Thus $0 \leq d(x) \leq 1$ and $x \in M \iff d(x) > 0$. The corresponding absolute constraint violation aggregation function is defined by $\hat{h} : \mathbb{R}^n \to \overline{\mathbb{R}}_+$:

$$\hat{h}(x) = \begin{cases} \sum_{j=1}^{m} \max(c_j(x), 0)^2 + \max(d(x), 0)^2 & \text{if } x \in X, \\ \infty & \text{otherwise.} \end{cases} \tag{4.2}$$

Thus $x \in \hat{\Omega} \iff \hat{h}(x) = 0$. Additional properties on the functions d and $\hat{h}$ and the related functions d^k and h^k at iteration k are introduced.

Property 4.5 (Continuity of d, $\hat{h}$, d^k and h^k). *(i) d is continuous on X, and $\hat{h}$ is continuous on Ω. (ii) d^k is continuous on X, and h^k is continuous on Ω for all $k \in \mathbb{N}$.*

Proof. *i)* If D is an empty set, then d is constant and continuous according to Equation (4.1). If D is non-empty, then $\text{dist}(x, D)$ is well defined and the distance from a point to a set is continuous on $\mathbb{R}^n$. Thus d is continuous on $X \subseteq \mathbb{R}^n$. If $x \in \Omega$, then $\hat{h}(x) = \max(d(x), 0)^2$. The maximum of two continuous functions is a continuous function, so $\hat{h}$ is continuous on Ω.

ii) The proof is identical to *i)* by substituting D by D^k, d by d^k and $\hat{h}$ by h^k. ∎

By definition, the constraint d is locally $\mathcal{C}^1$ on the margin M. Assume that the constraint $d(x) \leq 0$ is replaced by $d(x) \leq \epsilon$, where $\epsilon > 0$ is small, and that this new constraint is active at the optimum. Consequently, the optimum x^* lies in the margin M and the Lagrange multiplier $\lambda \leq 0$ associated with the constraint exists. The effect of the change in constraint on the value $f(x^*)$ is about $\lambda\epsilon$. In a blackbox optimization context, no empirical deductions about the multiplier can be made. However, the change of optimal value with respect to the tuning parameter r_e may be analyzed. If r_e is replaced by $r_e - \Delta$ with $\Delta > 0$, then $d(x) \leq \frac{\Delta}{r_e}$. Hence, the variation of the value $f(x^*)$ is about $\lambda\frac{\Delta}{r_e} < 0$, a coherent decrease in the objective function.

Property 4.6 (Characterization of the sequence $\{d^k(x)\}$). *Given a point $x \in X$, the sequence $\{d^k(x)\}$ admits a finite limit when k tends to infinity.*

Proof. Let x be a point in X. If D^k is empty for all k, then $d^k(x)$ is constant and the sequence $\{d^k(x)\}$ is convergent. Otherwise, there exists a sufficiently large rank $\bar{k}$ such that $D^{\bar{k}}$ is nonempty. Let $k \geq \bar{k}$; by construction, $V^k \subseteq V^{k+1}$ and $D^k \subseteq D^{k+1}$. Consequently, $\text{dist}(x, D^{k+1}) \leq \text{dist}(x, D^k)$ and $d^{k+1}(x) \geq d^k(x)$. The sequence $\{d^k(x)\}$ increases monotonically from the rank $\bar{k}$ and is bounded above by 1 by definition. Thus the sequence $\{d^k(x)\}$ admits a finite limit. ∎

Under Assumptions 4.1 and 4.2, it is first shown that the mesh gets infinitely fine.

Theorem 4.7. *Suppose that Assumptions 4.1 and 4.2 hold, and let $\{\delta^k\}$ stand for the sequence of mesh size parameters generated by* DiscoMads *then*

$$\liminf_{k\to\infty} \delta^k = \liminf_{k\to\infty} \Delta^k = 0.$$

Proof. Suppose by way of contradiction that there exists the following lower bound $\delta_{\min}$ on the mesh size parameter: $0 < \delta_{\min} \leq \delta^k$ for all $k \geq 0$. By considering the closure of the bounded set of Assumption 4.2, it follows that all iterates belong to a compact set. It is shown in Proposition 3.4 of [6] that there is only a finite number of different iterates $n_{\text{it}} \in \mathbb{N}$, and one of them, denoted x, must be visited infinitely many times. Consequently, there exists a rank $\bar{k} \in \mathbb{N}$ beyond which all the iterates in $B_{r_{\text{d}}}(x)$ have been generated due to the revealing poll. Thus, for all the iterations $k \geq \bar{k}$, the evaluation of x cannot lead to a revealing iteration. As a consequence, x can be visited infinitely many times only if there is an infinite number of unsuccessful iterations. According to the results stated for Mads [7], the mesh size parameter is then reduced infinitely many times, which contradicts the hypothesis that $\delta_{\min}$ is a lower bound for δ^k. The end of the theorem follows from the equation $\delta^k = \min(\Delta^k, (\Delta^k)^2)$. ∎

The convergence analysis relies on the Clarke directional derivative of a function f in the direction p at point x denoted by $\hat{f}^\circ(x, p)$ [17]. The reader is referred to Jahn [24] for the definition of the hypertangent cone $T_Y^H(x)$ to a set Y at a point x. Finally, two key definitions from [10] are recalled. In the context of DiscoMads, the poll step refers to the revealing poll, followed by the classical poll of Mads.

Definition 4.8 (Mesh local optimizer). *The point x^k is called a* mesh local optimizer *if and only if both the search step and the poll step fail at iteration k.*

Definition 4.9 (Refining subsequence, refining point and refining direction). *A convergent subsequence of mesh local optimizers $\{x^k\}_{k\in K}$ (where K is an infinite set of indices) is said to be a* refining subsequence *if and only if $\lim_{k\in K} \delta^k = 0$. The limit of a refining subsequence is called its corresponding* refined point. *Given a refining subsequence $\{x^k\}_{k\in K}$ and its corresponding refined point $\hat{x}$, a direction p is said to be a* refining direction *if and only if there exists an infinite subset $L \subseteq K$ with poll directions $p^k \in \mathbb{D}$ such that $x^k + \delta^k p^k \in \Omega$ and $\lim_{k\in L} \frac{p^k}{\|p^k\|} = \frac{p}{\|p\|}$.*

4.2 Refining points analysis

With respect to Assumptions 4.1 and 4.2 and Theorem 4.7, it is shown that there exists at least one converging refining subsequence (Theorem 3.6 in [6]). Depending on the nature of the refining subsequence $\{x^k\}$ and the position of the refining point $\hat{x}$, different local optimality conditions are derived. As DiscoMads is an extension of Mads, it is chosen, when possible, to base a maximum on Mads optimality conditions without taking advantage of the revealing poll.

Two results follow directly from the convergence analysis of Mads [5]. First, a convergence result is given in subsection 4.2.1 for f in the case where a $\hat{\Omega}$-feasible refining subsequence converging to $\hat{x} \in \hat{\Omega}$ is generated. Second, the function $\hat{h}$ is analyzed in subsection 4.2.2 for the case where an infeasible refining subsequence in $X \setminus M$ converging to $\hat{x} \in X \setminus M$ is generated. For these two cases, the results of Mads are preserved with similar proofs because $h^k = \hat{h}$ outside the margin M. However, for refining sequences belonging to M, only h^k is known; the revealing poll mechanism, as well as Assumption 4.4, is required to state novel convergence results in subsection 4.2.3. These additional elements allow Theorem 4.14 to ensure stronger optimality conditions than those of Mads [5]. Additionally, this theorem may be applied as well to refining subsequences belonging to $X \setminus M$. Consequently, and to maintain coherence with former results, the theorems stated in the following subsections are kept as general as possible. An adequate use of the theorems for the different cases covered by the convergence analysis is presented in Table 4.1.

4.2.1 A feasible refining subsequence: Result on f based on Mads

Theorem 4.10. *Under Assumptions 4.1 and 4.2, suppose that the algorithm generates a refining subsequence $\{x^k\}_{k\in K}$, with $x^k \in \hat{\Omega}$ converging to a refined point $\hat{x} \in \hat{\Omega}$ near which f is Lipschitz. If $p \in T_{\hat{\Omega}}^H(\hat{x})$ is a refining direction for $\hat{x}$, then $f^\circ(\hat{x}, p) \geq 0$. Moreover, if the set of refining directions for $\hat{x}$ is dense in $T_{\hat{\Omega}}^H(\hat{x}) \neq \emptyset$, then $\hat{x}$ is a Clarke stationary point for (1.2).*

Proof. As the iterates $\{x^k\}_{k\in K}$ belong to $\hat{\Omega}$, then $d^k(x^k) = 0$ and $h^k(x^k) = 0$ for all $k \in K$. Consequently, Theorem 3.3 of [5] is valid for this refining subsequence. It follows that $f^\circ(\hat{x}, p) \geq 0$. The end of the theorem follows from Corollary 3.4 of [5]. ∎

Table 4.1. Cases covered by the convergence analysis of DiscoMads.

result on	$\{x^k\}$	$\hat{x}$	covered by	assumptions
f	$\hat{\Omega}$	$\hat{\Omega}$	Theorem 4.10	4.1, 4.2
$\hat{h}$	$X \setminus M$	$\hat{\Omega}$	subsection 4.2.2 (lines 1-2)	4.1, 4.2
		$X \setminus M$	Theorem 4.11	4.1, 4.2
			Theorem 4.14	4.1, 4.2, 4.4
		M	Theorem 4.14	4.1, 4.2, 4.4
	M	$\hat{\Omega}$	subsection 4.2.2 (lines 1-2)	4.1, 4.2
		$X \setminus M$	Theorem 4.14	4.1, 4.2, 4.4
		M	Theorem 4.14	4.1, 4.2, 4.4

It is worth noting that if all the blackbox output functions are revealing, then if $\hat{x} \in \hat{\Omega}$, the function f is necessarily Lipschitz near $\hat{x}$ by definition of $\hat{\Omega}$.

4.2.2 An infeasible refining subsequence: Result on $\hat{h}$ based on Mads

If $\hat{h}(\hat{x}) = 0$, then $\hat{x}$ is a global minimum of $\hat{h}$ on X. Otherwise, $\hat{x}$ satisfies some necessary conditions in order to be a local minimizer of $\hat{h}$.

THEOREM 4.11. *Under Assumptions 4.1 and 4.2, suppose that the algorithm generates a refining subsequence $\{x^k\}_{k\in K}$, with $x^k \in X \setminus M$ converging to a refined point $\hat{x} \in X$ near which $\hat{h}$ is Lipschitz. If $p \in T_X^H(\hat{x})$ is a refining direction for $\hat{x}$, then $\hat{h}^\circ(\hat{x}, p) \geq 0$. Moreover, if the set of refining directions for $\hat{x}$ is dense in $T_X^H(\hat{x}) \neq \emptyset$, then $\hat{x}$ is a Clarke stationary point for $\min_{x\in X} \hat{h}(x)$.*

Proof. For all the iterates of the refining subsequence $\{x^k\}_{k\in K}$, $d^k(x^k) = 0$ by definition of d^k in (3.2). Consequently, $h^k(x^k) = \hat{h}(x^k)$, and Theorem 3.5 of [5] can be applied without restriction to the refining subsequence. It follows that if p is a refining direction for $\hat{x}$, then $\hat{h}^\circ(\hat{x}, p) \geq 0$. The end of the proof is given by Corollary 3.6 of [5]. ∎

4.2.3 An infeasible refining subsequence: Result on $\hat{h}$ based on the revealing poll

With additional assumptions and by using the revealing poll mechanism, a stronger convergence result on $\hat{h}$ is derived in Theorem 4.14, relying on Lemmas 4.12 and 4.13. The quantity $\Delta r = r_{\mathrm{m}} - r_{\mathrm{e}} - r_{\mathrm{d}} > 0$ is introduced.

LEMMA 4.12. *Under Assumptions 4.1 and 4.2, if $\hat{x}$ is the refined point of a refining subsequence $\{x^k\}_{k\in K}$, then the revealing poll generates a dense set of trial points in $B_{r_m}(\hat{x})$.*

Proof. Let y belong to the open ball $B_{r_{\mathrm{m}}}(\hat{x})$, and let $\epsilon_1 > 0$; it suffices to show that the revealing poll around some iterate x^k generates a trial point $z^k \in B_{\epsilon_1}(y)$. Define $\epsilon = \min\left(\frac{r_{\mathrm{m}} - \|y-\hat{x}\|}{2}, \epsilon_1\right)$, a strictly positive quantity.

As $\{x^k\}_{k\in K}$ is a refining subsequence, it follows by definition that $\lim_{k\in K} \delta^k = 0$. Consequently, there exists a mesh size parameter $\delta' > 0$ such that

$$\mathrm{round}(u) \in B_\epsilon(y) \quad \forall u \in B_{\epsilon/2}(y), \forall \delta < \delta', \tag{4.3}$$

where $\mathrm{round}(u)$ is the rounding of u on the mesh of coarseness δ.

There exists a sufficiently large threshold t such that $\delta^k < \delta'$ and $x^k \in B_\epsilon(\hat{x})$ for all $k \in K$ with $k \geq t$. Thus for all $k \in K$, $k \geq t$ and $v \in B_\epsilon(y)$, the following equation holds:

$$\|v - x^k\| \leq \underbrace{\|v - y\|}_{<\epsilon} + \underbrace{\|y - \hat{x}\|}_{\substack{\leq r_{\mathrm{m}} - 2\epsilon \\ \text{by definition of } \epsilon}} + \underbrace{\|\hat{x} - x^k\|}_{<\epsilon} < r_{\mathrm{m}}.$$

Consequently, $B_\epsilon(y) \subset B_{r_\mathrm{m}}(x^k)$. The probability that the revealing poll around x^k generates a random point (before rounding) in $B_{\epsilon/2}(y)$ is given by the strictly positive constant:

$$\frac{\mathrm{vol}(B_{\epsilon/2}(y))}{\mathrm{vol}(B_{r_\mathrm{m}}(x^k))} > 0,$$

where vol(•) denotes the volume of the ball. Consequently, there exists an index $k \in K$ for which a random point u^k generated by the revealing poll around x^k belongs to $B_{\epsilon/2}(y)$. From (4.3), the corresponding trial point $z^k = \mathrm{round}(u^k)$ necessarily belongs to $B_\epsilon(y)$, which is contained in $B_{\epsilon_1}(y)$. ∎

LEMMA 4.13. *Under Assumptions 4.1, 4.2 and 4.4, suppose that the algorithm generates a refining subsequence* $\{x^k\}_{k\in K}$ *converging to a refined point* $\hat{x} \in X$*; if* $y \in X \cap B_{\Delta r}(\hat{x})$*, then* $\lim_{k\to\infty} h^k(y) = \hat{h}(y)$.

Proof. Let y be a point such that $y \in X \cap B_{\Delta r}(\hat{x})$. Three cases may occur: *i)* $y \in D$, *ii)* $y \in M \setminus D$, and *iii)* $y \in X \setminus M$. If $r_\mathrm{d} = 0$, then $D = M = \emptyset$ and only case *iii)* is possible.

i) Assume that $y \in D$; then there exists at least one blackbox output function c_j, $j \in J$, and a point $z \in X \cap B_{r_\mathrm{d}}(y)$ such that $\frac{|c_j(y)-c_j(z)|}{\|y-z\|} > \tau$. According to Assumption 4.4, there exists an open set $X_y \subseteq X$ such that $y \in \overline{X}_y$ and c_j is continuous on $X_y \cup \{y\}$. From Lemma 4.12, a dense set S of trial points is generated by the revealing poll in $B_{r_\mathrm{m}}(\hat{x})$. As $X_y \cap B_{r_\mathrm{m}}(\hat{x})$ is an open nonempty set in $B_{r_\mathrm{m}}(\hat{x})$, it is thus possible to build the following sequence of points $\{y^k\} \in S \cap X_y$ converging to y: $\lim_{k\in L} y^k = y$, where $L \subseteq K$.

From Assumption 4.4, there also exists an open set $X_z \subseteq X$ such that $z \in \overline{X}_z$ and c_j is continuous on $X_z \cup \{z\}$. Moreover, $\|\hat{x} - y\| < \Delta r$ and $\|z - y\| < r_\mathrm{d}$, which results in $z \in B_{r_\mathrm{m}}(\hat{x})$. As the set S is dense on $B_{r_\mathrm{m}}(\hat{x})$, it is possible to build the following sequence of points $\{z^k\} \in S \cap X_z$ converging to z: $\lim_{k\in L'} z^k = z$, where $L' \subseteq L$. By continuity of c_j on X_y and X_z, the following equation holds:

$$\lim_{k\in L'} \frac{|c_j(y^k) - c_j(z^k)|}{\|y^k - z^k\|} = \frac{|c_j(y) - c_j(z)|}{\|y - z\|} > \tau.$$

Consequently, there exists a rank $k \in L'$ above which the pair of points (y^k, z^k) is revealing for the output function c_j. It follows that $d^k(y^k) = 1$ for a sufficiently large rank $k \in L'$. As d^k is continuous on X (Property 4.5), it follows that $\lim_{k\in L'} d^k(y^k) = \lim_{k\in L'} d^k(y)$. Since the sequence $\{d^k(y)\}_{k\in\mathbb{N}}$ has a finite limit (Property 4.6), each subsequence has the same limit, hence $\lim_{k\in L'} d^k(y) = \lim_{k\to\infty} d^k(y)$. It follows that $\lim_{k\to\infty} d^k(y) = \lim_{k\in L'} d^k(y^k) = 1 = d(y)$.

ii) Assume that $y \in M \setminus D$; then there exists a point $z \in D$ such that $\mathrm{dist}(y, D) = \|y - z\|$ and $\|y - z\| < r_\mathrm{e}$. There also exists a point $z' \in X \cup B_{r_\mathrm{d}}(z)$ and a blackbox output function c_j such that the pair of points (z, z') is revealing for c_j. As the inequalities

$$\|\hat{x} - z\| \le \underbrace{\|\hat{x} - y\|}_{<\Delta r} + \underbrace{\|y - z\|}_{<r_\mathrm{e}} < r_\mathrm{m} \quad \text{and} \quad \|\hat{x} - z'\| \le \underbrace{\|\hat{x} - z\|}_{<\Delta r + r_e} + \underbrace{\|z - z'\|}_{<r_\mathrm{d}} < r_\mathrm{m},$$

are satisfied, then z and z' belong to $B_{r_\mathrm{m}(\hat{x})}$. Consequently, two sequences $\{z^k\}_{k\in L}$ and $\{z'^k\}_{k\in L'\subseteq L}$ converging, respectively, to z and z' can be constructed in the same way as in *i)*. There exists a sufficiently large rank $k \in L'$ such that the pair (z^k, z'^k) is revealing for the output function c_j. From such a rank, D^k is nonempty and the distance $\mathrm{dist}(y, D^k)$ is well defined. As $D^k \subset D$ and $z^k \in D^k$, the following equation holds for all $k \in L'$:

$$\mathrm{dist}(y, D) \le \mathrm{dist}(y, D^k) \le \|y - z^k\|.$$

As the distance from a point to a set is continuous, the limit of this inequality can be considered. Remembering that z^k converges to z and that $\|y - z\| = \mathrm{dist}(y, D)$, it follows that $\lim_{k\in L'} \mathrm{dist}(y, D^k) = \mathrm{dist}(y, D)$. Hence, $\lim_{k\in L'} d^k(y) = d(y)$. Each subsequence of a converging sequence has the same limit, and thus $\lim_{k\to\infty} d^k(y) = d(y)$.

iii) Assume that $y \in X \setminus M$; then $d(y) = 0$. For all rank k and for all $x \in X$, $0 \le d^k(x) \le d(x)$, and then $d^k(y) = 0$. Consequently, $\lim_{k\to\infty} d^k(y) = d(y)$.

In each case, $\lim_{k\to\infty} d^k(y) = d(y)$. From (3.4), it follows that

$$\lim_{k\to\infty} h^k(y) = \sum_{j=1}^{m} \max(c_j(y),0)^2 + \max(d(y),0)^2 = \hat{h}(y).$$

□

The following theorem considers the case where $\hat{x}$ belongs to M and requires that $\hat{h}$ is piecewise continuous near $\hat{x}$. Note that this requirement is satisfied for all $x \in \Omega$ according to Property 4.5. The theorem states the convergence of the algorithm to a local minimizer of $\hat{h}$ on X, which is a stronger optimality condition than those derived for Mads involving contingent KKT stationary points [5]. This is made possible by taking advantage of the revealing poll mechanism as well as Assumption 4.4. A supporting graphic of the proof is given in Figure 3.

THEOREM 4.14. *Under Assumptions 4.1, 4.2 and 4.4 suppose that the algorithm generates a refining subsequence $\{x^k\}_{k\in K}$ converging to $\hat{x} \in X$ near which $\hat{h}$ is piecewise continuous; then $\hat{x}$ is a local minimizer of $\hat{h}$ on X.*

Proof. Let $\{x^k\}_{k\in K}$ be a refining subsequence converging to $\hat{x} \in X$. By way of contradiction, suppose that $\hat{x}$ is not a local minimizer of $\hat{h}$ on X; then there exists a point z near $\hat{x}$ such that $z \in B_{\Delta r}(\hat{x}) \cap X$ and

$$\hat{h}(\hat{x}) > \hat{h}(z). \tag{4.4}$$

As $\hat{h}$ is piecewise continuous near $\hat{x}$, there exist two open sets $X_{\hat{x}}$ and X_z such that $\hat{x} \in \overline{X}_{\hat{x}}$, $z \in \overline{X}_z$, and $\hat{h}$ is continuous on $\{\hat{x}\} \cup \overline{X}_{\hat{x}}$ and on $\{z\} \cup \overline{X}_z$. By the piecewise continuity of $\hat{h}$ and according to (4.4), there exists a pair $\epsilon > 0$, $\delta > 0$ such that

$$\hat{h}(a) > \hat{h}(b) + \delta \quad \forall a \in B_\epsilon(\hat{x}) \cap X_{\hat{x}}, \forall b \in B_\epsilon(z) \cap X_z. \tag{4.5}$$

As $\{x^k\}_{k\in K}$ converges to $\hat{x}$, according to Lemma 4.12, a dense set S of trial points is generated by the revealing poll in $B_{r_m}(\hat{x})$. Since the set $X_{\hat{x}} \cap B_{r_m}(\hat{x})$ is an open subset of $B_{r_m}(\hat{x})$, then infinitely many points are generated by the algorithm in $X_{\hat{x}}$. Consequently, there exists a refining subsequence $\{x^k\}_{k\in L}$ in $S \cap X_{\hat{x}}$ converging to $\hat{x}$: $\lim_{k\in L} x^k = \hat{x}$.

As $z \in (B_{\Delta r}(\hat{x}) \cap X) \subset (B_{r_m}(\hat{x}) \cap X)$, it is possible to build in the same way a subsequence $\{z^k\}_{k\in L'\subseteq L}$ in $S \cap X_z$ converging to z: $\lim_{k\in L'} z^k = z$. The sequences $\{x^k\}_{k\in L}$, $\{z^k\}_{k\in L'\subseteq L}$ and the pieces $X_{\hat{x}}$, X_z are depicted in Figure 3.

There exists a rank k sufficiently large such that $x^k \in B_\epsilon(\hat{x}) \cap X_{\hat{x}}$, $z^k \in B_\epsilon(z) \cap X_z$ and from Lemma 4.13 and by definition of the limit, the following inequalities hold:

$$\hat{h}(x^k) - \frac{\delta}{2} \quad < \quad h^k(x^k) \quad < \quad \hat{h}(x^k) + \frac{\delta}{2} \tag{4.6}$$

$$\hat{h}(z^k) - \frac{\delta}{2} \quad < \quad h^k(z^k) \quad < \quad \hat{h}(z^k) + \frac{\delta}{2}. \tag{4.7}$$

For such a rank k, it follows that

$$h^k(x^k) \underbrace{>}_{\text{from (4.6)}} \hat{h}(x^k) - \frac{\delta}{2} \underbrace{>}_{\text{from (4.5)}} \hat{h}(z^k) + \delta - \frac{\delta}{2} \underbrace{>}_{\text{from (4.7)}} h^k(z^k).$$

Thus $h^k(x^k) > h^k(z^k)$, and x^k is not a mesh local optimizer, which contradicts the fact that $x^k \in \{x^k\}_{k\in L}$ is a refining subsequence. Finally, $\hat{x}$ must be a local minimizer of $\hat{h}$ on X. □

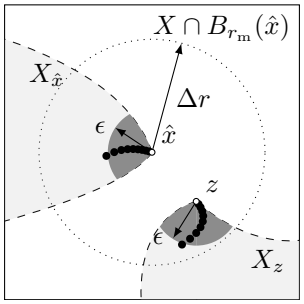


Figure 3. Schematic related to Theorem 4.14: pieces $X_{\hat{x}}$ and X_z (▭), and sequences of mesh points $\{x^k\}_{k\in L}$ and $\{z^k\}_{k\in L'}$ (•) converging, respectively, to $\hat{x}$ and z.

5 Numerical results

The parameters r_d, τ, and r_e required for DiscoMads are considered with different statuses. The quantities r_d and τ numerically characterize the unsafe region D and result from modeling choices. They are considered fixed by the user from insights on the underlying system dynamics, similarly to what is done in the literature [14, 16]. However, some guidelines for the choices of r_d and τ are given below considering a single revealing output c_j, and examples of values are given in numerical tests. In the case where a solution away from discontinuities is sought, when an estimate of the Lipschitz constant is known, it may be used for τ alongside a value of r_d large enough for early detection of revealing points during a run. When only an approximate magnitude of discontinuities "jumps" Δ_{c_j} is known, then τ may be chosen as $\frac{\Delta_{c_j}}{r_d}$. Otherwise, it is assumed that the user has insights on the magnitude of c_j. A default value may be chosen for r_d (*e.g.* $r_d = 1$ when x is scaled between 0 and 100) and τ may be chosen as the maximal safe variation allowed for c_j considering a variation of r_d in x. In the case where a solution outside strong variation regions is sought, r_d and τ derive directly from the sharp gradient modeling chosen by the user. On the contrary, the r_e parameter defining the margin M is considered as a flexible tuning parameter to obtain a solution far from the unsafe region D when the evaluation budget is limited. Indeed, in this case, the region D may not be sufficiently well approximated. The influence of r_e is analyzed in the numerical tests.

DiscoMads is implemented with NOMAD 3.9.1 [29], the open-source implementation of Mads. For the revealing poll, the convergence analysis only prescribes for the sampling radius that $r_m > r_e + r_d$ and for the number of sampled points that $n_{rnd} > 0$. For all the numerical tests, the radius r_m is arbitrarily fixed to $1.01(r_e + r_d)$ and n_{rnd} to n, the dimension of the problem. Unless otherwise stated, the default options of NOMAD are used. Particularly, the stopping criteria are the evaluation budget and the minimal mesh size; the first criterion met terminates the execution. The anisotropic mesh is disabled with DiscoMads to ease distance computations. The quadratic models are considered ill suited to representing discontinuous functions and are also disabled. For the search step, only the speculative search [7], activated by default in NOMAD 3.9.1, is used. This is a single-point search run after a dominating or improving iteration, in the corresponding previous successful direction. The opportunistic strategy is employed for both the search and poll steps.

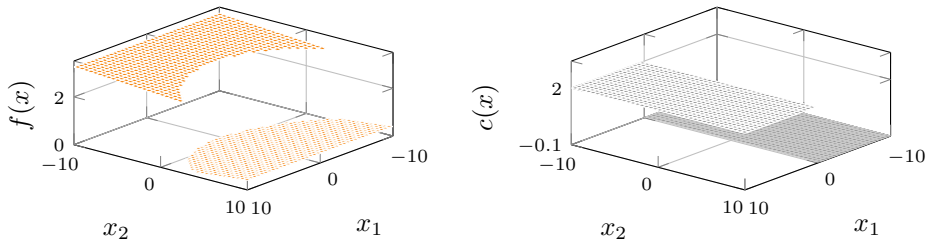
The behavior of DiscoMads is validated on several 2-dimensional test problems with closed-form expressions for f and c, among which only one is presented in the following section. For this problem, the region D is explicitly known for a pair of parameters (r_d, τ), and the low number of variables allows for comprehensive graphs. The performance and flexibility of the algorithm is then assessed on three design problems of nonlinear systems, illustrating three possible applications of the algorithm. These problems are representative of the blackbox optimization framework, as a numerical simulation is required to compute the objective function and the constraints. First, the design of a two-bar truss is considered, which leads to a 2-dimensional problem with a single discontinuity. This simplified problem is representative of stability problems occurring, for example, in crash simulations [16]. Then, a styrene production process is optimized and demonstrates an original use of DiscoMads to find a solution away from hidden constraint regions. To the best of our knowledge, this is the first time that this requirement is specifically treated. Finally, a turbomachine blade design problem exhibiting discontinuities similar to those of industrial numerical

simulations is considered. This problem is representative of emerging industrial challenges, which primarily motivates the development of DiscoMads. Consequently, an in-depth mechanical analysis of the solutions computed by the algorithm is provided.

5.1 Validation on analytical problems

Results of a typical run

Problem (1.2) is solved with $n = 2$ on $X = [-10; 10] \times [-10; 10]$, with the functions f and c depicted in Figure 4. The two functions are revealing for DiscoMads, and the region D is defined by $r_d = 0.25$ and $\tau = 0.3$. The margin radius is fixed to $r_e = 0.25$. The discontinuities and the regions are presented in Figure 5a with the optimal solution x^*. Up to 2000 evaluations are allowed, the minimal mesh size is fixed to 10^{-9}, and the starting point is $x^0 = (-5, -5)$ (Ω-feasible).



(a)

$$f(x) = \begin{cases} -0.025x_2 + 3 & \text{if} \quad \|x - \omega_f\| > 12, \\ 0.04x_2 & \text{otherwise.} \end{cases}$$

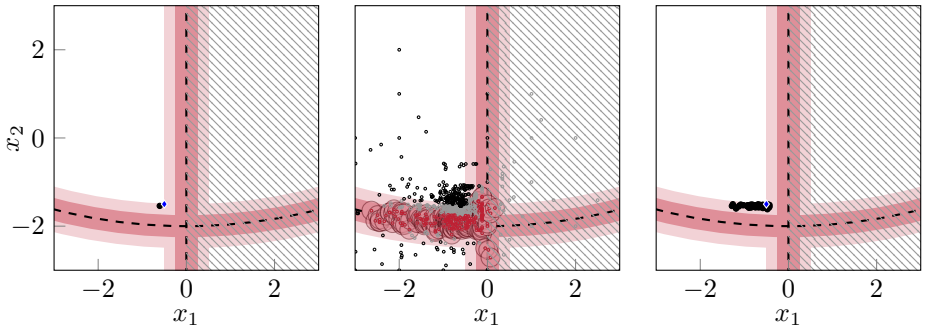
with $\omega_f = (0, 10)$

(b)

$$c(x) = \begin{cases} 2 & \text{if} \quad x_1 > 0, \\ -0.1 & \text{otherwise.} \end{cases}$$

Figure 4. Discontinuous functions with closed-form expressions.

The algorithm is stopped by the minimal mesh size criterion, which is consistent with the fact that the algorithm converges; the solution depicted in X in Figure 5a is outside the unsafe area D, as expected, but within the margin M. This does not contradict the convergence results based on an infinite number of iterations. For this run, 152 iterations out of 299 are revealing. All points evaluated during the run are presented in Figure 5b, and the exclusion areas are limited by circles of radii r_e. The superposition of these circles on the margin M shows that the algorithm is able to reveal the region D and to escape it.



(a) DiscoMads solution (●) and optimal solution x^* (⧫).

(b) Cache points: revealing (○), other Ω^k-infeasible (○), and Ω^k-feasible (○) points.

(c) DiscoMads solutions (●) and optimal solution x^* (⧫).

Figure 5. Results of DiscoMads on problem (1.2) with $X = [-10; 10] \times [-10; 10]$ and closed-form functions f and c. Single run (a),(b) and runs with 100 seeds (c).

Let $\{x_c^k\}_{k\in K}$ be the sequence of best feasible incumbents with $x_c^k \in F^k$, and K is the subset of iteration indices for which F^k is nonempty. Contrary to the usual behavior of Mads, $\{f(x_c^k)\}_{k\in K}$ is not decreasing; in DiscoMads, a revealing iteration leads to an update of the additional constraint d^k of some cache points, and the best current point may become infeasible due to such an update.

As the revealing poll is based on a random sampling, the runs must be repeated with different random seeds to validate the robustness of the algorithm. Results of 100 instances of DiscoMads executed with 100 different random seeds are presented in Figure 5c. The returned solutions are close to the optimal solution x^* and outside the unsafe region D. The robustness of the algorithm with regard to the random character of the revealing poll is illustrated on the analytical problems.

Discussion of algorithmic parameters

The parameters r_d and τ characterize the region D. However, locally, the border of the region is usually limited by only one of these parameters. Consequently, obtaining an explicit expression for the border of D is not trivial, even for low-dimensional problems. The border of D may be approximated in the graphs of this work, particularly in the vicinity of the X borders.

The parameter r_e can be tuned by the user to obtain solutions away from D. To show this flexibility, the algorithm is run 100 times with 100 distinct radii r_e varying from 0.001 to 0.5. The runs are executed with a single random seed. For each value of r_e, the distance of the solution from the closest discontinuity is computed and depicted in Figure 6, alongside the distances of the borders of regions D and M from the closest discontinuity. A general trend can be observed: when r_e increases, the distance to the closest discontinuity of the returned solution tends equally to increase. The solution tends to get closer to the margin border. The gaps are caused by the premature stop of the algorithm which can occur after a revealing iteration questions the feasibility status of the evaluated points.

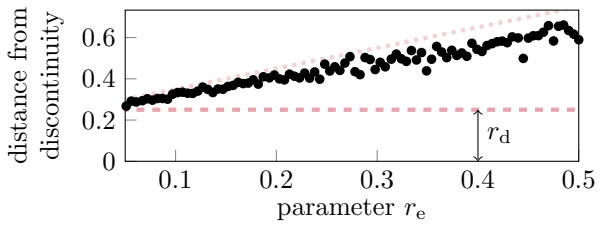


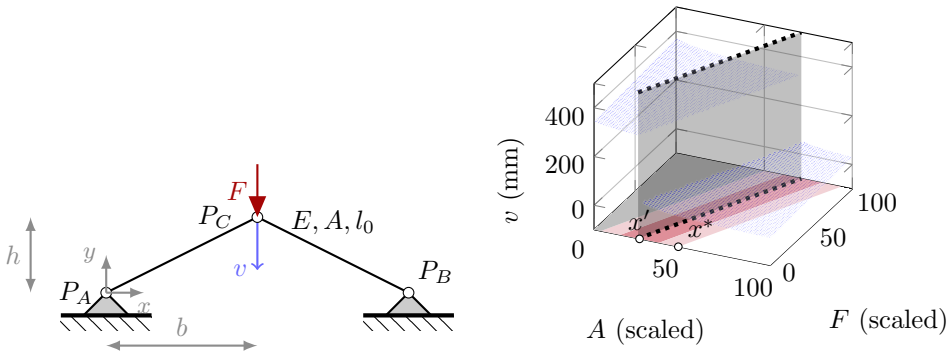
Figure 6. Distances of solutions (●) from the closest discontinuity with respect to r_e. Distances from the border of the regions D (- - -) and M (·····).

5.2 Design of a two-bar truss

The model consists of two identical bars of Young's modulus E, cross-sectional area A and free length l_0 (Figure 7a). The bars are constrained by pivot links to the frame at points P_A and P_B and are linked together by another pivot at the point P_C, whose coordinates at rest are denoted (b,h). The system parameters are fixed to $h = 0.5$ m, $b = 1$ m, and $E = 70{,}000$ MPa. A vertical load F is applied at point P_C, the bars are compressed, and buckling failure of the structure may occur. Consequently, the vertical displacement v of point P_C is discontinuous with respect to F and A. This quantity is depicted in Figure 7b, where F and A are scaled between 0 and 100, corresponding to the ranges $A \in [50; 300]$ mm^2 and $F \in [5; 10]$ kN.

The computation of v[1] results from a quasi-static analysis; the nonlinear equilibrium equation is $g(v) = 0$ [18, pp. 4-5]. It is solved iteratively with the Newton–Raphson method. The computation time of v for a given pair of variables (A, F) is about 0.5 ms on a personal computer.

[1] A blackbox for the buckling of a 2-bar truss is available online at https://gitlab.lava.polymtl.ca/depots_publics/codes/blackbox_buckling



(a) Schematic representation. **(b)** Displacement v.

Figure 7. Characteristics of the nonlinear two-bar truss.

Optimization problem

The two-bar truss design problem can be stated as a mass minimization problem (5.1) for a given load with respect to the two variables A and F scaled between 0 and 100. To avoid buckling, a constraint $c(x)$ limits the displacement v to $v_{\max} = 200$ mm, and the sought configuration must be away from the discontinuity of this constraint for safety reasons. In Figure 7b, a numerical estimation of the position of the discontinuity is depicted by the dashed line in the space of variables (A, F). The problem can be written as

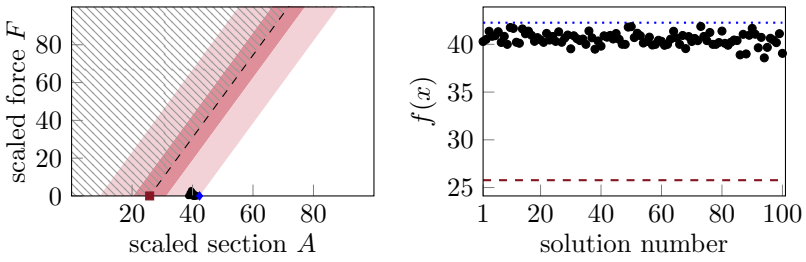
$$\begin{aligned} \min_{(A,F)\in X} \quad & f(A,F) \\ \text{subject to} \quad & c(A,F) \leq 0 \\ \text{and} \quad & d(A,F) \leq 0. \end{aligned} \tag{5.1}$$

with $X = \{(A,F) : 0 \leq A \leq 100, 0 \leq F \leq 100\}$, $f(A,F) = A$, and $c(A,F) = v(A,F) - v_{max}$. The constraint d refers to the remoteness constraint of problem (1.1).

To solve problem (5.1) with DiscoMads, the only revealing output function is the displacement constraint $c(x)$, and the unsafe area D (dark-colored in the plane (A,F) in Figure 7b) is described by the detection radius $r_{\text{d}} = 5$ and the limit rate of change $\tau = 0.02$. The parameter r_{e} is fixed at 10. Two starting points are considered: an Ω-feasible point $x_A^0 = (80, 80)$ describing a configuration without buckling, and an Ω-infeasible point $x_B^0 = (0, 100)$. This last point corresponding to a buckling configuration is not realistic and only used to validate the behavior of the algorithm in this case. Each problem instance is solved for 100 different random seeds.

Results

For each of the 100 runs from the starting point x_A^0, the solution returned by DiscoMads is depicted in the space of variables in Figure 8a. In all cases, the algorithm stopped when reaching the minimal allowed mesh size of 10^{-9}. Both the optimal solution x^* of problem (5.1) and the solution x' of this problem, with the constraint d relaxed, are also depicted in Figure 8a. The DiscoMads solutions are located in the same area and are very close to x^*, which attests to the robustness of the algorithm with respect to the random component of the revealing poll. The solutions are within the margin M and are not in the unsafe area D. This behavior is as expected considering the evaluation budget and the relatively simple topology of the problem, which makes it easier for the revealing poll to accurately detect the area D. From a mechanical standpoint, the solutions lead to reliable designs with no buckling effect, even for small variations of the variables around the solutions. The objective function values for these solutions are depicted in Figure 8b and compared to the values $f(x^*)$ and $f(x')$. As expected, the f values of the 100 solutions of problem (5.1) are higher than $f(x')$. The results from the problem instance with starting point x_B^0 are very similar and are not presented here. This demonstrates the robustness of the algorithm on this problem with an Ω-infeasible starting point.

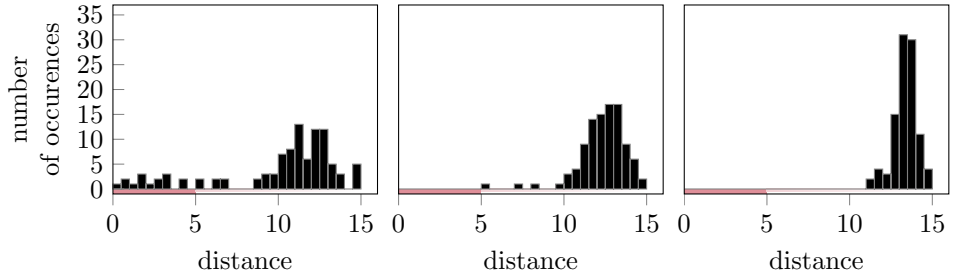


(a) Position of the solutions in X: DiscoMads runs (●), x^* (♦), and x' (■)

(b) f values for solutions returned by DiscoMads (●), x^* (·····), and x' (- - -).

Figure 8. Characteristics of the solutions returned by DiscoMads for the 100 runs from the Ω-feasible starting point $x_A^0 = (80, 80)$.

Finally, the influence of the evaluation budget on the quality of the returned solutions is investigated. A histogram representing the distance to the discontinuity of each solution is depicted in Figure 9 for three budgets of 100, 200, and 2000 evaluations. When the budget is low, the distance to the discontinuity of the solutions strongly varies—some solutions even belong to the unsafe area D. On the contrary, with the 2000-evaluation budget, all solutions are away from D and close to the border of $\hat{\Omega}$. This is explained by the fact that a higher evaluation budget allows for a more intensive sampling during the revealing poll.



(a) 100-evaluation budget. **(b)** 200-evaluation budget. **(c)** 2000-evaluation budget.

Figure 9. Histograms of the distances of the solutions from the discontinuity for different budgets for the runs with the starting point $x_A^0 = (80, 80) \in \hat{\Omega}$.

5.3 Design of a styrene production process

Optimization problem

The aim is to optimize a chemical process for styrene production by minimizing the production costs while satisfying environmental and industrial constraints [4]. The problem has 8 variables scaled between 0 and 100 and 11 constraints: 7 relaxable constraints and 4 binary constraints. This process is simulated in the open-source blackbox STYRENE[2] which incorporates several numerical methods, such as Newton or Runge–Kutta. The computation time of the blackbox for a single evaluation is about 600 ms on a personal computer.

This problem has been studied in many works [4, 8, 9, 22] and is well known to exhibit hidden constraints: for some points x, the simulation fails and the blackbox output cannot be computed [30], even if x is Ω-feasible. According to Gramacy and Le Digabel [22], about 20% of the blackbox evaluations violate some hidden constraints, and the returned solution may be close to points violating such constraints.

[2] https://github.com/bbopt/styrene

Solution with DiscoMads

The DiscoMads algorithm is applied to this 8-variable problem to obtain a solution away from the unsafe area where hidden constraints are violated. To achieve this, the blackbox is slightly modified to return an artificially high value of the objective and the relaxable constraints when a hidden constraint is violated. The violation of hidden constraints thus leads to discontinuities of the objective function, which can be revealed by DiscoMads.

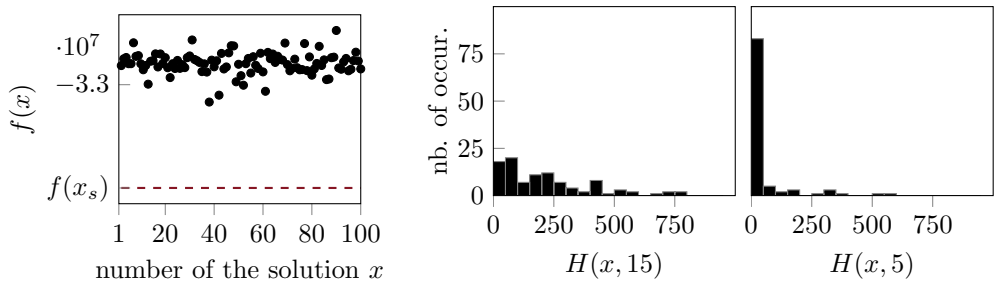
An indicator $H(x, \sigma)$ is introduced to quantify the quality of a solution and it is defined as the number of points violating hidden constraints among 1000 points randomly sampled in $X \cap B_\sigma(x)$. At the time when the text was written,, one of the best feasible solutions of the STYRENE problem is a point x_s such that $f(x_s) = -33\,709\,000$. As $H(x_s, 15) = 435$, this point is close to at least 435 points violating hidden constraints.

The starting point x_s is used for DiscoMads, and 100 instances of the problem are run with 100 random seeds. Only the objective function is used to reveal discontinuities, the detection procedure is driven by the parameters $r_\mathrm{d} = 5$ and $\tau = 10^{15}$, and the exclusion radius is fixed to $r_\mathrm{e} = 10$. The other parameters are based on previous publications related to STYRENE. In particular, a budget of 1000 evaluations is allowed, and the minimal mesh size is fixed to 10^{-7}. The binary constraints are treated with the extreme barrier approach and the relaxable constraints with the progressive barrier.

Results

For each seed, the f-value of the solution returned by the algorithm is depicted in Figure 10a alongside the value $f(x_s)$. The algorithm is able to return an Ω-feasible solution for each run. The solutions returned by DiscoMads have similar f-values and are not very sensitive to the random sampling for this case. Moreover, these values are higher than $f(x_s)$: as expected, the algorithm moved from the starting point x_s, which is revealed close to points violating hidden constraints.

For each solution x, the indicator $H(x, \sigma)$ is calculated for two values of σ, and its distributions are depicted in Figure 10. If $\sigma = r_\mathrm{e} + r_\mathrm{d} = 15$, most of the solutions have an $H(x, \sigma)$ value less than 400, but the dispersion of values is large. Note that in this case, the indicator quantifies the number of revealing points of D at distance less than $r_\mathrm{e} + r_\mathrm{d}$ from the returned solution. This indicator is severe because in most cases the returned solution lies in the region $M \setminus D$, according to previous numerical results. Consequently, it is difficult to draw conclusions about the performance of the algorithm from this isolated measure. If $\sigma = r_\mathrm{d} = 5$, most of the $H(x, \sigma)$ values are low and below $H(x_s, 5) = 120$, which indicates that few revealing points have been detected in balls of radius σ around the solutions. The algorithm is thus able to return solutions away from areas where hidden constraints are violated for this 8-variable problem.



(a) Values of f. **(b)** Histogram of $H(x, 15)$. **(c)** Histogram of $H(x, 5)$.

Figure 10. Characteristics of the solutions for the STYRENE problem.

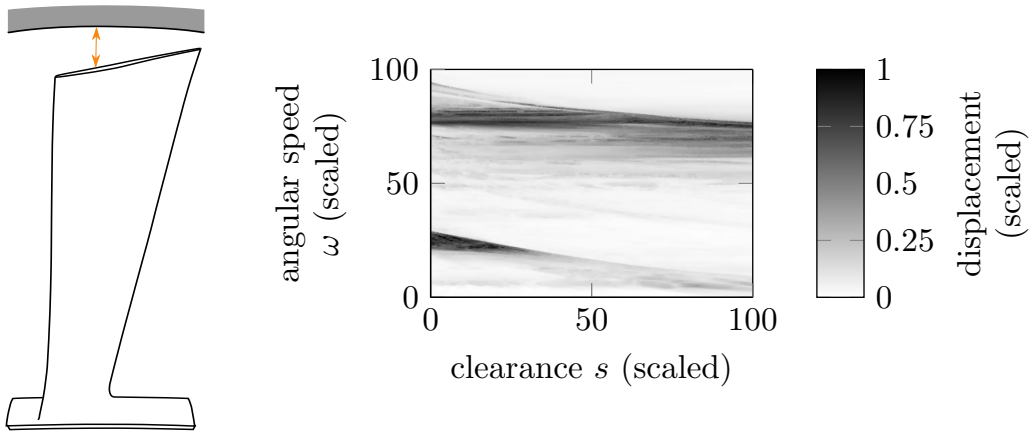
5.4 Design of a turbomachine blade

Over the past few years, more stringent economic and environmental constraints pushed engine manufacturers to reduce aircraft engine fuel consumption. A preferred approach consists of reducing clearances between rotating blades and the surrounding casing components to limit aerodynamic leakage and thus increase the overall engine

efficiency. However, due to vibrations in operation [31], clearance reductions yield more frequent structural contacts that must now be accounted for in blade design. These contacts lead to unsafe nonlinear resonance regions where the simulated quantities are discontinuous. These areas must be detected and escaped from in the optimization process.

Simulation

A simplified problem is considered in this section. The maximal radial displacement at the tip of the blade is analyzed with respect to two variables: the blade rotational speed ω and the tip clearance s (depicted in Figure 11a), assumed constant from the leading edge to the trailing edge. The radial displacement $u(s, \omega)$ is computed with a numerical strategy [31] relying on the industrial finite element model of a low-pressure compressor blade. For a pair of variables (ω, s), 20 revolutions of the blade at constant speed ω are simulated accounting for centrifugal effects [13].



(a) Blade (□), casing (■), tip clearance ↔.

(b) Interpolation of the maximal radial displacement at the tip of the blade computed from 45 000 simulation points (s, ω). Scaled displacement.

Figure 11. Presentation of the turbomachine blade design problem.

Let s_0 denote a typical tip clearance for compressor blades and let ω_0 denote the nominal blade rotational speed; s is varied between $0.8s_0$ and $6.5s_0$ and ω between $0.85\omega_0$ and $1.15\omega_0$. The displacements computed for 45 000 points (s, ω) are scaled between 0 and 1 and depicted in Figure 11b with respect to s and ω, both scaled between 0 and 100. Strong local variations of the displacement with respect to both speed and clearance are visible. Optimal configurations should lead to reduced displacements while minimizing the tip clearance for a specific range of speeds. As the system is strongly nonlinear, optimal configurations may be very close to unsafe configurations.

The computation time of a single simulation is about 1 min 40 s on a standard processor. Consequently, 52 days are required to run all the simulations on a single processor. From a practical standpoint, blackbox optimization is required to identify meaningful cost-effective configurations. A simplified blade design problem is considered in the next section as a proof of concept for the resolution of such industrial design problems with DiscoMads.

Optimization problem

The design problem is representative of a reverse engineering problem and aims at minimizing the tip clearance to reduce aerodynamic loss for a prescribed speed range. Two variables are considered: the tip clearance s and the rotational speed ω. To account for structural constraints, the displacement $u(s, \omega)$ is limited to the value $u_{\max}$, corresponding to a scaled displacement of 0.37 in Figure 11b. To ensure the reliability of the configuration, the displacement should not vary widely for small variations of the variables around this configuration. For a speed

range $[\omega_{\min};\omega_{\max}] \subseteq [0,100]$ the problem is stated as

$$\begin{aligned} \min_{(s,\omega)\in X} \quad & f(s,\omega) \\ \text{subject to} \quad & u(s,\omega) - u_{max} \leq 0 \\ \text{and} \quad & (s,\omega) \text{ away from discontinuities of } u, \end{aligned} \tag{5.2}$$

with $X = \{(s,\omega) : 0 \leq s \leq 100, \omega_{\min} \leq \omega \leq \omega_{\max}\}$ and $f(s,\omega) = s$. For development purposes, the displacement $u(s,\omega)$ is computed from a piecewise linear interpolation of the displacement on X (Figure 11b) built from the 45,000 simulation runs. The whole procedure can be applied with the original numerical strategy [31] in an engineering context.

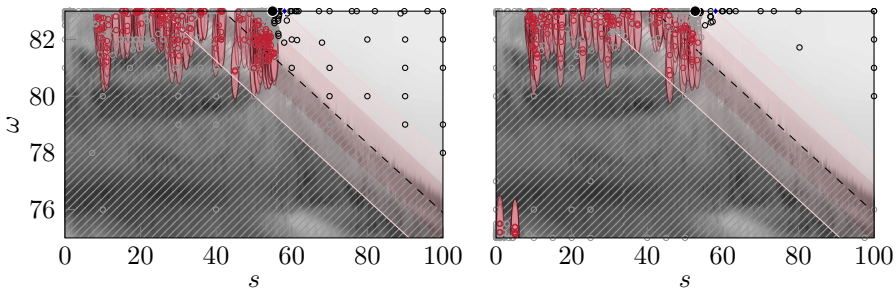
Four instances of problem (5.2), detailed in Table 5.1, are solved with DiscoMads with a single random seed. Only the displacement constraint is revealing, the unsafe region is fixed by the parameters $r_{\text{d}} = 1$ and $\tau = 0.4$ and the margin is fixed by $r_{\text{e}} = 1$. The minimal mesh size is set to 10^{-9}, and 2000 evaluations are allowed.

Table 5.1. Instances of problem (5.2) related to the design of turbomachine blades.

instance	1	2	3	4
range $[\omega_{\min}, \omega_{\max}]$	$[22, 28]$	$[22, 28]$	$[75, 83]$	$[75, 83]$
starting point (s^0, ω^0)	$(40, 25) \in \hat{\Omega}$	$(5, 25) \notin \Omega$	$(90, 80) \in \hat{\Omega}$	$(25, 76) \notin \Omega$

Optimization results and analysis of solutions

The cache points of instances 3 and 4 are depicted in the plane (s,ω) in Figure 12 alongside the returned solution. Exclusion balls are limited by vertically stretched circles. The solutions returned by DiscoMads are close to the optimal solution and outside the unsafe area, even with an Ω-infeasible starting point. In this case, problem (5.2) is more difficult because the algorithm must escape from the unsafe region D while moving to the feasible Ω domain at the same time. Instances 1 and 2 are solved successfully with DiscoMads as well [11].

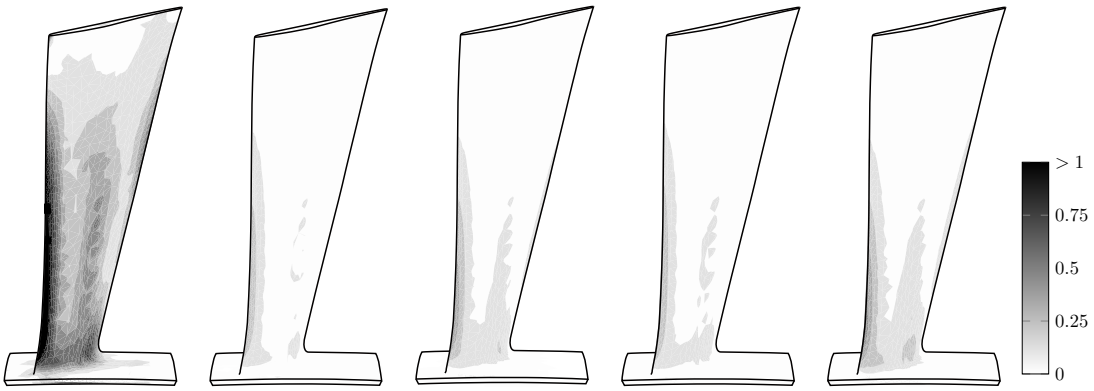


(a) instance 3 ($x^0 \in \hat{\Omega}$). **(b)** instance 4 ($x^0 \notin \Omega$).

Figure 12. Instances of problem (5.2): DiscoMads solution (●), optimal solution (♦), and cache points: revealing (○), other Ω^k-infeasible (○), and Ω^k-feasible (○) points.

A stress analysis is performed for the simulation point $(s,\omega) = (5, 25)$ for which the displacement is large. The stresses are computed at the time step of maximal radial displacement at the tip of the blade. The resulting scaled stress field is depicted in Figure 13a where 1 corresponds to the yield stress of the material. Some areas located along the leading edge undergo stress as high as 1.6 times the yield stress. Consequently, the Ω-infeasible domain of problem (5.2) contains critical design points. The same stress analysis is conducted for the solutions y^1 and y^3 returned by DiscoMads respectively for instances 1 and 3. The corresponding scaled stress fields, depicted in Figure 13b and Figure 13c, are very similar for the two points. Contrary to Figure 13a, stress levels are far below the yield stress, and higher stress areas are located on the leading edge in both cases. Finally, the stress fields are

computed for two additional points y'^1 and y'^3 located at distance r_e from y^1 and y^3 respectively, in the direction of the nearest discontinuity (Figure 13d and Figure 13e). Although the maximal stress for the point y'^1 is higher than for the point y^1, it is far below the yield stress and thus remains acceptable. The same observation holds for the points y'^3 and y^3. It is worth noticing that the points y'^1 and y'^3 close to solutions provided by DiscoMads lead to safe mechanical designs.



(a) $(s, \omega) = (5, 25)$. **(b)** $y^1 \simeq (12.8, 28.0)$. **(c)** $y^3 \simeq (54.9, 83.0)$. **(d)** $y'^1 \simeq (12.5, 27.0)$. **(e)** $y'^3 \simeq (54.8, 82.0)$.

Figure 13. Stress fields for an unsafe configuration (a), DiscoMads solutions (b), (c) and adjoining solutions (d), (e). Values scaled by the yield stress.

6 Discussion

This work proposes an original approach for solving a constrained blackbox optimization problem with the additional constraint that the solution must be away from unsafe regions of weak discontinuities. The approach is based on the Mads algorithm and builds successive inner approximations of the safe margin M by revealing the unsafe region and escaping it as the algorithm is deployed.

The Mads convergence results are preserved and stronger optimality conditions are proved by using the revelation mechanism, under assumptions of piecewise continuity of functions. The developed approach is validated on analytical problems and numerical results on three engineering problems demonstrate the relevance of the algorithm while preserving the evaluation budget. These problems illustrate possible applications of the algorithm, *e.g.*, to successfully escape an unsafe region of hidden constraints.

Although DiscoMads was originally developed for a turbomachine blade design problem, it may be used for the design of other nonlinear systems where discontinuities reveal unsafe regions, such as resonance zones or bifurcations. Moreover, the algorithm returns information on the position of the unsafe region and can provide a better understanding of the underlying system dynamics. In a different context, the algorithm may be used for problems frequently exhibiting hidden constraints, in order to find a solution away from these regions. Finally, taking advantage of the flexible modeling proposed in this work, the algorithm may also be used for problems where sharp gradient regions should be detected and escaped.

Future efforts may focus on extending the application field of DiscoMads by relaxing some restrictive hypotheses. The mandatory scaling of the input may be avoided by adapting the anisotropic mesh option of Mads [12] or by using scaled detection regions. A distinct limit rate of change could also be considered for each output function, leading to a more complex convergence analysis. Another research direction may be to accelerate the numerical efficiency of DiscoMads. A more sophisticated revealing mechanism may be used, as well as the inclusion of a reliability index inspired from reliability-based design optimization [26] to prematurely stop the algorithm. Lastly, although the Mads surrogate framework is not used in this work for clarity, it would certainly accelerate the convergence of the algorithm.

From a mathematical standpoint, the developed approach is a proof of concept for the treatment of specific infinite constraints in blackbox optimization and may be generalized to other types of unsafe regions or other direct search methods. From a practical standpoint, it reinforces the possibilities of using blackbox optimization methods for the design of systems with strong variations in the quantities of interest. The development of such *ad hoc* algorithms contributes to the more systematic use of rigorous methods with proven optimality conditions for engineering problems.

Acknowledgments

We would like to thank Sébastien Le Digabel for his suggestion on using DiscoMads to detect hidden constraints in the STYRENE problem.

This research was undertaken thanks to funding from the NSERC's discovery grant RGPIN-2020-04448 (first and third author), the Canada Research Chairs Program (second and third author) and the PBEEE scholarship from FRQNT (third author).

References

[1] R. Ahlfeld, F. Montomoli, M. Carnevale, and S. Salvadori. Autonomous uncertainty quantification for discontinuous models using multivariate Padé approximations. J. Turbomach. Vol. 140, No. 4 (2018), 041004. DOI: 10.1115/1.4038826.

[2] J. M. Anderson. Modelling Step Discontinuous Functions Using Bayesian Emulation. PhD thesis. Auckland University of Technology, 2017. OAI: hdl.handle.net/10292/10543.

[3] R. Archibald, A. Gelb, R. Saxena, and D. Xiu. Discontinuity detection in multivariate space for stochastic simulations. J. Comput. Phys. Vol. 228, No. 7 (2009), 2676–2689. DOI: 10.1016/j.jcp.2009.01.001.

[4] C. Audet, V. Béchard, and S. Le Digabel. Nonsmooth optimization through mesh adaptive direct search and variable neighborhood search. J. Global Optim. Vol. 41, No. 2 (2008), 299–318. DOI: 10.1007/s10898-007-9234-1. eprint: https://www.gerad.ca/en/papers/G-2006-61.

[5] C. Audet and J. E. Dennis Jr. A Progressive Barrier for Derivative-Free Nonlinear Programming. SIAM J. Optim. Vol. 20, No. 1 (2009), 445–472. DOI: 10.1137/070692662.

[6] C. Audet and J. E. Dennis Jr. Analysis of Generalized Pattern Searches. SIAM J. Optim. Vol. 13, No. 3 (2003), 889–903. DOI: 10.1137/S1052623400378742.

[7] C. Audet and J. E. Dennis Jr. Mesh adaptive direct search algorithms for constrained optimization. SIAM J. Optim. Vol. 17, No. 1 (2006), 188–217. DOI: 10.1137/040603371.

[8] C. Audet, J. E. Dennis Jr., and S. Le Digabel. Globalization strategies for mesh adaptive direct search. Comput. Optim. Appl. Vol. 46, No. 2 (2010), 193–215. DOI: 10.1007/s10589-009-9266-1. eprint: https://www.gerad.ca/en/papers/G-2008-74.

[9] C. Audet, J. E. Dennis Jr., and S. Le Digabel. Trade-off studies in blackbox optimization. Optim. Method. Softw. Vol. 27, No. 4–5 (2012), 613–624. DOI: 10.1080/10556788.2011.571687.

[10] C. Audet and W. Hare. Derivative-Free and Blackbox Optimization. Springer, 2017. DOI: 10.1007/978-3-319-68913-5.

[11] C. Audet, A. Batailly, and S. Kojtych. *Escaping Unknown Discontinuous Regions in Blackbox Optimization*. Tech. rep. G-202-46. Groupe d'études et de recherche en analyse des décisions, 2020. eprint: https://www.gerad.ca/fr/papers/G-2020-46. published.

[12] C. Audet, S. Le Digabel, and C. Tribes. Dynamic scaling in the mesh adaptive direct search algorithm for blackbox optimization. Optim. Eng. Vol. 17, No. 2 (2016), 333–358. DOI: 10.1007/s11081-015-9283-0. eprint: https://www.gerad.ca/en/papers/G-2014-16.

[13] A. Batailly, M. Legrand, A. Millecamps, and F. Garcin. Numerical-experimental comparison in the simulation of rotor/stator interaction through blade-tip/abradable coating contact. J. Eng. Gas Turbines Power Vol. 134, No. 8 (2012), 082504. DOI: 10.1115/1.4006446. OAI: hal.archives-ouvertes.fr/hal-00746632.

[14] A. Bhosekar and M. Ierapetritou. A discontinuous derivative-free optimization framework for multi-enterprise supply chain. Optim. Lett. Vol. 14 (2020), 959–988. DOI: 10.1007/s11590-019-01446-5.

[15] E. G. Birgin, N. Krejić, and J. M. Martínez. On the minimization of possibly discontinuous functions by means of pointwise approximations. Optim. Lett. Vol. 11, No. 8 (2017), 1623–1637. DOI: 10.1007/s11590-016-1068-7.

[16] C. Boursier Niutta, E. J. Wehrle, F. Duddeck, and G. Belingardi. Surrogate modeling in design optimization of structures with discontinuous responses. Struct. Multidiscip. Optim. Vol. 57, No. 5 (2018), 1857–1869. DOI: 10.1007/s00158-018-1958-7.

[17] F. H. Clarke. Optimization and Nonsmooth Analysis. John Wiley & Sons, 1983. DOI: 10.1137/1.9781611971309.

[18] M. A. Crisfield. Non-linear finite element analysis of solids and structures. Vol. 1. John Wiley & Sons, 1993, pp. 4–5. URL: https://www.wiley.com/en-us/NonlinearFiniteElementAnalysisofSolidsandStructures,2ndEdition-p-9780470666449.

[19] S. De Marchi, W. Erb, F. Marchetti, E. Perracchione, and M. Rossini. Shape-Driven Interpolation With Discontinuous Kernels: Error Analysis, Edge Extraction, and Applications in Magnetic Particle Imaging. SIAM J. Sci. Comput. Vol. 42 (2020), B472–B491. DOI: 10.1137/19M1248777.

[20] R. Fletcher and S. Leyffer. Nonlinear Programming Without a Penalty Function. Math. Program. Vol. 91 (2002), 239–269. DOI: 10.1007/s101070100244.

[21] A. Gorodetsky and Y. Marzouk. Efficient localization of discontinuities in complex computational simulations. SIAM J. Sci. Comput. Vol. 36, No. 6 (2014), A2584–A2610. DOI: 10.1137/140953137.

[22] R. Gramacy and S. Le Digabel. The mesh adaptive direct search algorithm with treed Gaussian process surrogates. Pac. J. Optim Vol. 11, No. 3 (2015), 419–447. eprint: https://www.gerad.ca/en/papers/G-2011-37.

[23] Y. V. Halder, B. Sanderse, and B. Koren. An Adaptive Minimum Spanning Tree Multielement Method for Uncertainty Quantification of Smooth and Discontinuous Responses. SIAM J. Sci. Comput. Vol. 41, No. 6 (2019), A3624–A3648. DOI: 10.1137/18M1219643.

[24] J. Jahn. Introduction to the Theory of Nonlinear Optimization. 3rd ed. Berlin: Springer, 2007. DOI: 10.1007/978-3-540-49379-2.

[25] J.-H. Jung and V. R. Durante. An iterative adaptive multiquadric radial basis function method for the detection of local jump discontinuities. Appl. Numer. Math. Vol. 59, No. 7 (2009), 1449–1466. DOI: 10.1016/j.apnum.2008.09.002.

[26] Z. Kang, Y. Luo, and A. Li. On non-probabilistic reliability-based design optimization of structures with uncertain-but-bounded parameters. Struct. Saf. Vol. 33, No. 3 (2011), 196–205. DOI: 10.1016/j.strusafe.2011.03.002.

[27] M. Krack and J. Gross. Harmonic balance for nonlinear vibration problems. Vol. 1. Math. Eng. Springer, 2019. DOI: 10.1007/978-3-030-14023-6.

[28] J. Lainé, E. Piollet, F. Nyssen, and A. Batailly. Blackbox Optimization for Aircraft Engine Blades With Contact Interfaces. J. Eng. Gas Turbines Power Vol. 141, No. 6 (2019), 061016. DOI: 10.1115/1.4042808. OAI: hal.inria.fr/hal-02059582v1.

[29] S. Le Digabel. Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm. ACM Trans. Math. Vol. 37, No. 4 (2011). DOI: 10.1145/1916461.1916468.

[30] S. Le Digabel and S. Wild. *A Taxonomy of Constraints in Simulation-Based Optimization*. Tech. rep. G-2015-57. Les cahiers du GERAD, 2015. eprint: https://www.gerad.ca/en/papers/G-2015-57.

[31] M. Legrand, A. Batailly, B. Magnain, P. Cartraud, and C. Pierre. Full three-dimensional investigation of structural contact interactions in turbomachines. J. Sound Vib. Vol. 331, No. 11 (2012), 2578–2601. DOI: 10.1016/j.jsv.2012.01.017. OAI: hal.archives-ouvertes.fr/hal-03444757.

[32] M. Menickelly and S. M. Wild. Derivative-free robust optimization by outer approximations. Math. Program. Vol. 179, No. 1 (2020), 157–193. DOI: 10.1007/s10107-018-1326-9.

[33] M. Moustapha and B. Sudret. Surrogate-assisted reliability-based design optimization: a survey and a unified modular framework. Struct. Multidiscip. Optim. Vol. 60, No. 5 (2019), 2157–2176. DOI: 10.1007/s00158-019-02290-y.

[34] E. P. Petrov and D. J. Ewins. Generic Friction Models for Time-Domain Vibration Analysis of Bladed Disks. J. Turbomachinery Vol. 126, No. 1 (2004), 184–192. DOI: 10.1115/1.1644557.

[35] P. Pettersson, A. Doostan, and J. Nordström. Level set methods for stochastic discontinuity detection in nonlinear problems. J. Comput. Phys. Vol. 392 (2019), 511–531. DOI: 10.1016/j.jcp.2019.04.053. OAI: liu.diva-portal.org/smash/record.jsf?pid=diva2%3A1315457&dswid=1378.

[36] L. Romani, M. Rossini, and D. Schenone. Edge detection methods based on RBF interpolation. J. Comput. Appl. Math. Vol. 349 (2019), 532–547. DOI: 10.1016/j.cam.2018.08.006.

[37] K. Sargsyan, C. Safta, B. Debusschere, and H. Najm. Uncertainty Quantification given Discontinuous Model Response and a Limited Number of Model Runs. SIAM J. Sci. Comput. Vol. 34, No. 1 (2012), B44–B64. DOI: 10.1137/100817899.

[38] R. Seydel. Practical Bifurcation and Stability Analysis. 3rd ed. Vol. 5. Interdiscip. Appl. Math. Springer, 2010. DOI: 10.1007/978-1-4419-1740-9.

[39] L. N. Vicente and A. L. Custódio. Analysis of direct searches for discontinuous functions. Math. Program. Vol. 133, No. 1 (2012), 299–325. DOI: 10.1007/s10107-010-0429-8.

[40] A. Wechsung and P. I. Barton. Global optimization of bounded factorable functions with discontinuities. J.Global Optim. Vol. 58 (2014), 1–30. DOI: 10.1007/s10898-013-0060-3. URL: https://dspace.mit.edu/handle/1721.1/103613.