ISTITUTO DI ANALISI DEI SISTEMI ED INFORMATICA
"Antonio Ruberti"
CONSIGLIO NAZIONALE DELLE RICERCHE

G. Fasano, G. Liuzzi, S. Lucidi, F. Rinaldi

A LINESEARCH-BASED DERIVATIVE-FREE APPROACH FOR NONSMOOTH OPTIMIZATION

R. 1, January 2013

Giovanni Fasano – Università Ca'Foscari Venezia, Dipartimento di Management, San Giobbe, Cannaregio 873, 30121 Venice, Italy. fasano@unive.it.

Giampaolo Liuzzi – Consiglio Nazionale delle Ricerche, Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti", Viale Manzoni 30, 00185 Rome, Italy. liuzzi@iasi.cnr.it.

Stefano Lucidi – "Sapienza" Università di Roma, Dipartimento di Ingegneria Informatica Automatica e Gestionale "A. Ruberti", Via Ariosto 25, 00185 Rome, Italy. lucidi@dis.uniroma1.it.

Francesco Rinaldi – Università di Padova, Dipartimento di Matematica, Via Trieste 63, 35121 Padua, Italy. rinaldi@math.unipd.it.

This work has been partially funded by the Italian national project RITMARE 2012-2016.

ISSN: 1128–3378

Collana dei Rapporti dell'Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti", CNR

viale Manzoni 30, 00185 ROMA, Italy

tel. ++39-06-77161
fax ++39-06-7716461
email: iasi@iasi.cnr.it
URL: http://www.iasi.cnr.it

Abstract

In this paper, we propose new linesearch-based methods for nonsmooth optimization problems when first-order information on the problem functions is not available. In the first part, we describe a general framework for bound-constrained problems and analyze its convergence towards stationary points, using the Clarke-Jahn directional derivative. In the second part, we consider inequality constrained optimization problems where both objective function and constraints can possibly be nonsmooth. In this case, we first split the constraints into two subsets: difficult general nonlinear constraints and simple bound constraints on the variables. Then, we use an exact penalty function to tackle the difficult constraints and we prove that the original problem can be reformulated as the bound-constrained minimization of the proposed exact penalty function. Finally, we use the framework developed for the bound-constrained case to solve the penalized problem, and we prove that every accumulation point of the generated sequence of points is a stationary points of the original constrained problem.
In the last part of the paper, we report extended numerical results on both bound-constrained and nonlinearly constrained problems, showing the effectiveness of our approach when compared to some state-of-the-art codes from the literature.

Key words: Derivative-free optimization, Lipschitz optimization, Exact penalty functions, Inequality constrained optimization, Stationarity conditions

1. Introduction

In this paper, we consider the optimization of a nonsmooth function $f : \mathbb{R}^n \to \mathbb{R}$ over a feasible set defined by lower and upper bounds on the variables and, possibly, by nonlinear and nonsmooth inequality constraints $g : \mathbb{R}^n \to \mathbb{R}^m$, namely

$$\begin{array}{ll} \min & f(x) \\ s.t. & g(x) \leq 0 \\ & l \leq x \leq u, \end{array}$$

where $l, u \in \mathbb{R}^n$. We assume that the problem functions (though non-smooth) are Lipschitz continuous and that first-order information is unavailable, or impractical to obtain (e.g. when problem functions are expensive to evaluate or somewhat noisy).
Such a kind of optimization problems encompasses many real-world problems arising in different fields like e.g. computational mathematics, physics and engineering, and presents a twofold difficulty. On the one hand, problem functions are typically of the black-box type, so that first order information is unavailable; on the other hand, the functions present a certain level of nonsmoothness (see e.g. [3], [10] and [18]).
In [4] and [5] the Mesh Adaptive Direct Search (MADS) class of algorithms is introduced, where an asymptotically dense set of directions is generated and combined with an extreme barrier approach, in order to provide a general and flexible framework for nonsmooth constrained problems. In [2] the use of a deterministic scheme for the generation of an asymptotically dense set of search directions is proposed, thus defining the ORTHOMADS method. A different way to handle the constraints within MADS-type algorithms is proposed in [6] where the authors combine a filter-based strategy [13] with a progressive barrier approach.

In [11] it is proved that the efficiency of direct search methods (like e.g. MADS), when applied to nonsmooth problems, can be improved by using simplex gradients to order poll directions.
In the first part of this paper, we describe a general framework for solving bound-constrained nonsmooth optimization problems. The approach, called DFN$_{simple}$, combines a projected line-search with the use of search directions which are asymptotically dense in the unit sphere. These two features make the algorithm very flexible as for the way to generate the asymptotically dense set of search directions, and allow us to prove convergence to stationary points of the problem in the Clarke-Jahn sense [16]. Then, we propose an improved version of the algorithm, namely CS-DFN, which further performs linesearches along the coordinate directions.
In the second part, we focus on nonlinearly constrained problems. We assume that two different classes of constraints exist, namely, difficult general nonlinear constraints ($g(x) \leq 0$) and simple bound constraints on the problem variables ($l \leq x \leq u$). The main idea is that of getting rid of the nonlinear constraints by means of an exact penalty approach. Therefore, we construct a merit function that penalizes the general nonlinear inequality constraints and we resort to the minimization of the penalty function subject to the simple bound constraints. In this way, using again the framework developed for the bound-constrained case, we define an algorithm (which is called DFN$_{con}$) to tackle nonlinearly constrained problems. We are able to prove that the new bound-constrained problem is to a large extent equivalent to the original problem and that the sequence generated by means of the described approach converges to stationary points of the original problem, in the sense that every accumulation point is stationary for the constrained problem.
In the last part of the paper, an extended numerical experience (on 142 bound-constrained and 296 nonlinearly constrained problems) is carried out. We first test two versions of then

DFN$_{simple}$ algorithm, obtained by embedding into the scheme two different pseudorandom sequences to generate the asymptotically dense set of search directions. In particular, we compare the Halton [14] and Sobol sequences [30, 8] within our method. Then we analyze the performances of both our methods DFN$_{simple}$ and CS-DFN, and we compare CS-DFN with two state-of-the-art solvers on a large set of 142 bound-constrained nonsmooth problems. Finally, we focus on nonlinearly constrained problems. In the latter case, we compare our code DFN$_{con}$ with two well-known codes on a large test set of 296 nonsmooth constrained problems. The codes DFN$_{simple}$, CS-DFN and DFN$_{con}$ are freely available for download at the url: `http://www.dis.uniroma1.it/~lucidi/DFL`.

The paper has the following structure: Section 2 contains some preliminaries and technical definitions used throughout the paper. In Section 3, we analyze the approach for the bound-constrained case. In Section 4, we extend the approach to nonlinearly constrained problems. The numerical results are reported in Section 5. We summarize our conclusions in Section 6, and an Appendix completes the paper, including auxiliary results.

As regards the notation used in this paper, given a vector $v \in \mathbb{R}^n$, a subscript will be used to denote either the ith of its entries v_i or the fact that it is an element of an infinite sequence of vectors $\{v_k\}$. In case of possible misunderstanding or ambiguities, the ith component of a vector will be denoted by $(v)_i$ or $\{v\}_i$. We denote by v^j the generic jth element of a finite set of vectors, and in particular $e^1, \dots, e^n$ represent the coordinate unit vectors. Given two vectors $a, b \in \mathbb{R}^n$, we indicate with $y = \max\{a, b\}$ the vector such that $y_i = \max\{a_i, b_i\}$, $i = 1, \dots, n$. Furthermore, given a vector $v \in \mathbb{R}^n$ we denote by $v^+ = \max\{0, v\}$. By $S(0,1)$ we indicate the unit sphere with center in the origin, i.e. $S(0,1) = \{x \in \mathbb{R}^n \ : \ \|x\|_2 = 1\}$, and given a subset $A \subseteq \mathbb{R}^n$, $Co(A)$ indicates its convex hull. Finally, $[\cdot]_+$ denotes the orthogonal projection on a compact convex subset of $\mathbb{R}^n$, and $\partial f(x)$ is the generalized gradient of f at x.

2. Preliminary results

In this section, we introduce some definitions and report some results that will be used throughout the paper. In particular, we first recall the definition of a dense subset of $\mathbb{R}^n$.

Definition 2.1. *Given two sets $A, B \subseteq \mathbb{R}^n$, we say that B is dense in A if, for all $a \in A$ and for all $\epsilon > 0$, an element $b \in B$ exists such that*

$$\|a - b\| \le \epsilon.$$

Then, we introduce the less intuitive definition of asymptotically dense sequence of sets (of directions).

Definition 2.2. *Let us consider the sequence of sets of directions $\{D_k\}_{k=0}^{\infty}$, with $D_k = \{d_0, \dots, d_k\}$. We say that the sequence is asymptotically dense in $S(0,1)$ if for any $\epsilon > 0$, there exists an index $\bar{k} \ge 0$ such that for all directions $\bar{d} \in S(0,1)$ and for all $k \ge \bar{k}$*

$$\|d_{i_k} - \bar{d}\| \le \epsilon, \qquad \text{for some } d_{i_k} \in D_k.$$

Using the latter definition we are able to introduce the following assumption and a related result, which will be crucial in our analysis.

Assumption 1. *There exists a map $\mathcal{M} : \mathbb{N} \mapsto \mathbb{R}^n$, such that the sequence of sets $\{D_k\}_{k=0}^{\infty}$, where $D_k = \{d_0, \dots, d_k\}$, with $d_k = \mathcal{M}(k)$, is asymptotically dense in $S(0,1)$.*

Proposition 2.3. *Let $A \subseteq \mathbb{R}^n$ and consider the sequence $\{x_k\} \subset A$ converging to $\bar{x} \in A$. Let Assumption 1 hold, being $\{D_k\}_{k=0}^{\infty}$ the sequence of search directions generated by map $\mathcal{M}$. Then, for all $\epsilon > 0$, an index $\bar{k}$ exists, such that for all $\bar{d} \in S(0,1)$ and for all $k \geq \bar{k}$,*

$$\begin{aligned} &\|x_k - \bar{x}\| \leq \epsilon, \\ &\|d_{i_k} - \bar{d}\| \leq \epsilon, \qquad \textit{for some } d_{i_k} \in D_k. \end{aligned} \tag{1}$$

Proof. Since $\{x_k\} \to \bar{x}$ by assumption, suppose by contradiction that for any $\bar{k}$,

$$\|d_{i_k} - \bar{d}\| > \epsilon, \qquad \text{for all } d_{i_k} \in D_k, \tag{2}$$

for some $k \geq \bar{k}$ and some $\epsilon > 0$. Let us consider the sequences $\{y_h\}$ and $\{p_h\}$, with $y_h = x_{h+\bar{k}}$ and $p_h = \mathcal{M}(h+\bar{k})$. Thus $\{y_h\} \to \bar{x}$; in addition, the sequence $\{P_h\}_{h=0}^{\infty}$, with $P_h = \{p_1, \ldots, p_h\}$, is asymptotically dense in the unit sphere $S(0,1)$ by Assumption 1. Then, the index $\bar{h}$ exists such that for all $h \geq \bar{h}$, there exist an index $(i_h \leq h)$ and a direction $p_{i_h} \in P_h$ such that

$$\|p_{i_h} - \bar{d}\| \leq \epsilon. \tag{3}$$

Now, recalling that $p_{i_h} = d_{i_h+\bar{k}}$, relation (3) yields a contradiction with (2). $\square$

From Proposition 2.3 we can prove the next result, which characterizes the limit points of the sequences generated in our framework.

Corollary 2.4. *Let $\{x_k\} \subset A$ be a sequence converging to $\bar{x} \in A$ and let Assumption 1 hold, with $\{d_k\}$ being the sequence of search directions generated by map $\mathcal{M}$. Given the direction $\bar{d} \in S(0,1)$ there exist subsequences $\{x_k\}_K \subseteq \{x_k\} \subset A$ and $\{d_k\}_K \subseteq \{d_k\}$ such that*

$$\begin{aligned} \lim_{k\to\infty, k\in K} x_k &= \bar{x}, \\ \lim_{k\to\infty, k\in K} d_k &= \bar{d}. \end{aligned}$$

3. The bound-constrained case

In this section we consider the bound-constrained problem

$$\begin{aligned} &\min\ f(x) \\ &s.t.\ x \in X, \end{aligned} \tag{4}$$

where we indicate by X the set of bound constraints on the variables, i.e.

$$X = \{x \in \mathbb{R}^n : l \leq x \leq u\},$$

and f is Lipschitz continuous. We observe that to our purposes, the boundedness assumption on the set X might be relaxed by suitable assumptions on the objective function of the problem. For points in the feasible set X we address also the definition of cone of feasible directions, as follows.

Definition 3.1 (Cone of feasible directions) *Given problem (4) and any point $x \in X$*

$$D(x) = \{d \in \mathbb{R}^n : d_i \geq 0 \ \textit{if}\ x_i = l_i,\ d_i \leq 0 \ \textit{if}\ x_i = u_i,\ i = 1, \ldots, n\}$$

is the cone of feasible directions at x with respect to X.

We also report a technical proposition whose proof can be found in [21].

Proposition 3.2. *Given problem* (4)*, let* $\{x_k\} \subset X$ *for all* k*, and* $\{x_k\} \to \bar{x}$ *for* $k \to \infty$*. Then, for* k *sufficiently large,*

$$D(\bar{x}) \subseteq D(x_k).$$

Moreover, on the guideline of Corollary 2.4 and considering the set $D(x_k)$ from the latter proposition, we can consider also the next result.

Corollary 3.3. *Given problem* (4)*, let* $\{x_k\} \subset X$ *be a sequence converging to* $\bar{x}$*. Let Assumption 1 hold, being* $\{d_k\}$ *the sequence of search directions generated by map* $\mathcal{M}$*, and* $d_k \in D(x_k)$ *(i.e.* d_k *is a feasible direction). Given a direction* $\bar{d} \in D(\bar{x}) \cap S(0,1)$ *there exist subsequences* $\{x_k\}_K \subseteq \{x_k\} \subset X$ *and* $\{d_k\}_K \subseteq \{d_k\}$ *such that*

$$\lim_{k\to\infty, k\in K} x_k = \bar{x},$$
$$\lim_{k\to\infty, k\in K} d_k = \bar{d}.$$

The necessary optimality conditions for problem (4) can be characterized in terms of the Clarke-Jahn generalized directional derivative of the objective function. In particular, we recall that given the point $x \in X$, the Clarke-Jahn directional derivative of the function f along the direction $d \in D(x)$ is given by (see [16]):

$$f^\circ(x; d) = \limsup_{\substack{y \to x, y \in X \\ t \downarrow 0, y + td \in X}} \frac{f(y + td) - f(y)}{t}. \tag{5}$$

From [16] we recall the every local minimum of problem (4) satisfies the following definition.

Definition 3.4. *Given problem (4),* $x^\star$ *is a stationary point if*

$$f^\circ(x^\star; d) \geq 0, \qquad \forall\, d \in D(x^\star). \tag{6}$$

We propose in the next sections two algorithms, having different performances on the nonsmooth bound-constrained problem (4).

3.1. A simple derivative-free algorithm

As discussed in the Introduction, even in the simpler case of bound constraints, since the objective function f is possibly not continuously differentiable on X, a finite number of search directions is not sufficient to investigate the local behavior of $f(x)$ on X [18]. Hence, recalling [5], we resort to the use of a set of search directions which is eventually dense in the unit sphere. We prove that the use of such a simple set of search directions is sufficient to enforce convergence to stationary points of problem (4).
On this purpose, here we propose a very simple Derivative-Free algorithm for solving the Nonsmooth problem (4), namely DFN$_{simple}$, where a map satisfying Assumption 1 is adopted, in order to generate a set of search directions dense in the unit sphere.
In this algorithm, apart from the initializations, at any iteration k we use the map $\mathcal{M}$ to generate the search direction d_k. Then, we investigate the behavior of the function $f(x)$ along the direction d_k, by means of the linesearch procedure *Projected Continuous Search*. Given the current

iterate x_k at step k, the latter procedure first evaluates the function at $[x_k \pm \tilde{\alpha}_k d_k]_+$. In case a sufficient reduction of the function value is obtained, then an extrapolation along the search direction is performed, so that a suitable step-length α_k is computed, and is used as a tentative step-length for the next iteration, i.e. $\tilde{\alpha}_{k+1} = \alpha_k$. On the other hand, if at $[x_k \pm \tilde{\alpha}_k d_k]_+$ we do not obtain a sufficient reduction of the function value, then the tentative step-length at the next iteration is suitably reduced by a scale factor, i.e. $\tilde{\alpha}_{k+1} = \theta\tilde{\alpha}_k$, $\theta \in (0, 1)$. More formally the resulting algorithm and the corresponding linesearch procedure adopted are summarized in the next schemes, where $[\cdot]_+$ denotes the projection on X.

Algorithm DFN$_{simple}$

Data. $\theta \in (0, 1)$, $x_0 \in X$, $\tilde{\alpha}_0 > 0$, the map $\mathcal{M} : \mathbb{N} \mapsto \mathbb{R}^n$ such that for $k \geq 0$, $d_k = \mathcal{M}(k)$ and

$$\|d_k\| = 1.$$

For $k = 0, 1, \ldots$

Set $d_k = \mathcal{M}(k)$.

Compute α_k by the *Projected Continuous Search*$(\tilde{\alpha}_k, x_k, d_k; \alpha_k, \tilde{d}_k)$.

If $(\alpha_k == 0)$ **then** $\tilde{\alpha}_{k+1} = \theta\tilde{\alpha}_k$ and $\tilde{x}_k = x_k$

else $\tilde{\alpha}_{k+1} = \alpha_k$ and $\tilde{x}_k = [x_k + \alpha_k \tilde{d}_k]_+$

Find $x_{k+1} \in X$ such that $f(x_{k+1}) \leq f(\tilde{x}_k)$.

End For

Projected Continuous Search $(\tilde{\alpha}, y, p; \alpha, p^+)$

Data. $\gamma > 0$, $\delta \in (0, 1)$.

Step 0. Set $\alpha = \tilde{\alpha}$.

Step 1. **If** $f([y + \alpha p]_+) \leq f(y) - \gamma\alpha^2$ **then** set $p^+ = \ \ p$ and go to Step 4.

Step 2. **If** $f([y - \alpha p]_+) \leq f(y) - \gamma\alpha^2$ **then** set $p^+ = -p$ and go to Step 4.

Step 3. Set $\alpha = 0$ and return.

Step 4. Let $\beta = \alpha/\delta$.

Step 5. **If** $f([y + \beta p^+]_+) > f(y) - \gamma\beta^2$ return.

Step 6. Set $\alpha = \beta$ and go to Step 4.

It is worth noting that in Algorithm DFN$_{simple}$ the next iterate x_{k+1} is required to satisfy $f(x_{k+1}) \leq f(\tilde{x}_k)$. This allows in principle to compute x_{k+1} by minimizing suitable approximating models of the objective function, thus possibly improving the efficiency of the overall scheme.
In the following results we analyze the global convergence properties of Algorithm DFN$_{simple}$. In particular, in the next proposition we prove that the procedure described in Algorithm DFN$_{simple}$ cannot cycle.

Proposition 3.5. *Algorithm DFN$_{simple}$ is well-defined.*

Proof. In order to show that Algorithm DFN$_{simple}$ is well-defined we prove that the *Projected Continuous Search* cannot infinitely cycle between Step 4 and Step 6. Let us consider the *Projected Continuous Search*, we proceed by contradiction assuming that an infinite monotonically increasing sequence of positive numbers $\{\beta_j\}$ exists such that

$$f([y + \beta_j p^+]_+) \le f(y) - \gamma\beta_j^2.$$

The above relation contradicts the fact that X is compact, by definition, and that function f is continuous, thus concluding the proof. □

Now, in the following proposition we prove that the stepsizes computed by the procedure *Projected Continuous Search* eventually go to zero.

Proposition 3.6. *Let $\{\alpha_k\}$, $\{\tilde{\alpha}_k\}$ be the sequences generated by Algorithm DFN$_{simple}$, then*

$$\lim_{k\to\infty} \max\{\alpha_k, \tilde{\alpha}_k\} = 0. \tag{7}$$

Proof. We split the iteration sequence $\{k\}$ into two sets K_1, K_2, with $K_1 \cup K_2 = \{k\}$ and $K_1 \cap K_2 = \emptyset$. We denote by

- K_1 the set of iterations when $\tilde{\alpha}_{k+1} = \alpha_k$;
- K_2 the set of iterations when $\tilde{\alpha}_{k+1} = \theta\tilde{\alpha}_k$ and $\alpha_k = 0$.

Note that K_1 and K_2 cannot be both finite. Let us first suppose that K_1 is infinite, then the instructions of the algorithm imply, for $k \in K_1$,

$$f(x_{k+1}) \le f([x_k + \alpha_k \tilde{d}_k]_+) \le f(x_k) - \gamma\alpha_k^2. \tag{8}$$

Taking into account the compactness of X and the continuity of f, we get from the above relation that $\{f(x_k)\}$ tends to a limit $\bar{f}$. Then, by (8), it follows

$$\lim_{k\to\infty, k\in K_1} \alpha_k = 0, \tag{9}$$

which also implies

$$\lim_{k\to\infty, k\in K_1} \tilde{\alpha}_k = 0. \tag{10}$$

Now, let us suppose that K_2 is infinite. Then, directly $\lim_{k\to\infty, k\in K_2} \alpha_k = 0$ and, by recalling that $\theta \in (0, 1)$,

$$\lim_{k\to\infty, k\in K_2} \tilde{\alpha}_k = 0. \tag{11}$$

Relations (9), (10) and (11) yield (7), thus concluding the proof. □

Using the latter result, along with the map $\mathcal{M}$ satisfying Assumption 1, we can provide the next technical lemma, which will be necessary to prove the main global convergence result for algorithm DFN$_{simple}$.

Lemma 3.7. *Let $\{x_k\}$, $\{d_k\}$ and $\{\alpha_k\}$ be the sequences generated by Algorithm DFN$_{simple}$, and let Assumption 1 hold. Then, for any accumulation point $\bar{x}$ of $\{x_k\}$ and for any direction $\bar{d} \in D(\bar{x})$, $\bar{d} \neq 0$, there exists a subsequence of indices K such that*

$$\lim_{k\to\infty, k\in K} x_k = \bar{x}, \tag{12}$$

$$\lim_{k\to\infty, k\in K} d_k = \bar{d}, \tag{13}$$

$$\lim_{k\to\infty, k\in K} \tilde{\alpha}_k = 0. \tag{14}$$

Moreover,

(i) for $k \in K$ sufficiently large, an $\alpha > 0$ exists such that

$$[x_k + \alpha d_k]_+ \neq x_k,$$

(ii) the following limit holds

$$\lim_{k\to\infty, k\in K} v_k = \bar{d},$$

where

$$v_k = \frac{[x_k + \tilde{\alpha}_k d_k]_+ - x_k}{\tilde{\alpha}_k}. \tag{15}$$

Proof. Relations (12) and (13) follow from Corollary 3.3; on the other hand, Proposition 3.6 yields relation (14).
Now, in order to prove items (i)-(ii), let us recall that for any $i = 1, \dots, n$

$$\{[x_k + \alpha d_k]_+\}_i = \max\{l_i, \min\{u_i, (x_k + \alpha d_k)_i\}\}.$$

Now we show that, for $k \in K$ sufficiently large, an $\alpha > 0$ exists such that

$$[x_k + \alpha d_k]_+ \neq x_k. \tag{16}$$

By contradiction, let us assume that for $k \in K$ sufficiently large, for all $\alpha > 0$, we have

$$[x_k + \alpha d_k]_+ = x_k. \tag{17}$$

Let us choose $\alpha = \tilde{\alpha}_k$ for all k and, since $\bar{d} \neq 0$ an index i exists such that one of the following three cases holds.

1) $\bar{x}_i = l_i$ (which implies $\bar{d}_i > 0$): we can write

$$\{[x_k + \tilde{\alpha}_k d_k]_+\}_i = \max\{l_i, (x_k + \tilde{\alpha}_k d_k)_i\};$$

since x_k is feasible and (13) holds, for k sufficiently large we have

$$\max\{l_i, (x_k + \tilde{\alpha}_k d_k)_i\} \geq \max\left\{l_i, \left(x_k + \frac{\tilde{\alpha}_k}{2}\bar{d}\right)_i\right\},$$

and by (14) we get

$$\max\left\{l_i, \left(x_k + \frac{\tilde{\alpha}_k}{2}\bar{d}\right)_i\right\} = \left(x_k + \frac{\tilde{\alpha}_k}{2}\bar{d}\right)_i \neq (x_k)_i. \tag{18}$$

2) $\bar{x}_i = u_i$ (which implies $\bar{d}_i < 0$): we can write

$$\{[x_k + \tilde{\alpha}_k d_k]_+\}_i = \min\{u_i, (x_k + \tilde{\alpha}_k d_k)_i\};$$

since x_k is feasible and (13) holds, for k sufficiently large we have

$$\min\{u_i, (x_k + \tilde{\alpha}_k d_k)_i\} \leq \min\left\{u_i, \left(x_k + \frac{\tilde{\alpha}_k}{2}\bar{d}\right)_i\right\},$$

and by (14) we get

$$\min\left\{u_i, \left(x_k + \frac{\tilde{\alpha}_k}{2}\bar{d}\right)_i\right\} = \left(x_k + \frac{\tilde{\alpha}_k}{2}\bar{d}\right)_i \neq (x_k)_i. \tag{19}$$

3) $l_i < \bar{x}_i < u_i$ (which implies $\bar{d}_i \neq 0$): we can write

$$\{[x_k + \tilde{\alpha}_k d_k]_+\}_i = (x_k + \tilde{\alpha}_k d_k)_i;$$

since x_k is feasible and (13) holds, for k sufficiently large we have

$$(x_k + \tilde{\alpha}_k d_k)_i \neq (x_k)_i. \tag{20}$$

Then, by (18), (19) and (20) we have a contradiction with (17), which proves (i).
Now, we recall definition (15) and note that, by (16), the vector v_k is eventually nonzero. By the definition of the vector v_k, we have for its $i-$th entry

$$\begin{aligned}(v_k)_i &= \frac{\max\{l_i, \min\{u_i, (x_k + \tilde{\alpha}_k d_k)_i\}\} - (x_k)_i}{\tilde{\alpha}_k} && (21)\\ &= \frac{\min\{u_i, \max\{l_i, (x_k + \tilde{\alpha}_k d_k)_i\}\} - (x_k)_i}{\tilde{\alpha}_k}. && (22)\end{aligned}$$

Now, let us distinguish among the following three cases, for k sufficiently large and $k \in K$:

1) $\bar{x}_i = l_i$: then by (21) we have

$$(v_k)_i = \frac{\max\{l_i, (x_k + \tilde{\alpha}_k d_k)_i\} - (x_k)_i}{\tilde{\alpha}_k}$$

and recalling that whenever $\bar{x}_i = l_i$ it must be $\bar{d}_i \geq 0$, we distinguish two subcases:

a) when $\bar{d}_i > 0$, then $(v_k)_i = \max\left\{\frac{l_i - (x_k)_i}{\tilde{\alpha}_k}, (d_k)_i\right\} = (d_k)_i$;

b) when $\bar{d}_i = 0$, then

$$\lim_{k\to\infty, k\in K} (v_k)_i = \lim_{k\to\infty, k\in K} \max\left\{\frac{l_i - (x_k)_i}{\tilde{\alpha}_k}, (d_k)_i\right\} = 0 = (\bar{d})_i.$$

2) $\bar{x}_i = u_i$: then by (22) we have

$$(v_k)_i = \frac{\min\{u_i, (x_k + \tilde{\alpha}_k d_k)_i\} - (x_k)_i}{\tilde{\alpha}_k}$$

and recalling that whenever $\bar{x}_i = u_i$ it must be $\bar{d}_i \leq 0$, we distinguish two subcases:

a) when $\bar{d}_i < 0$, then $(v_k)_i = \min\left\{\frac{u_i-(x_k)_i}{\tilde{\alpha}_k}, (d_k)_i\right\} = (d_k)_i$;

b) when $\bar{d}_i = 0$, then

$$\lim_{k\to\infty, k\in K} (v_k)_i = \lim_{k\to\infty, k\in K} \min\left\{\frac{u_i-(x_k)_i}{\tilde{\alpha}_k}, (d_k)_i\right\} = 0 = (\bar{d})_i;$$

3) $l_i < \bar{x}_i < u_i$: then by (21) or (22) we have $(v_k)_i = (x_k + \tilde{\alpha}_k d_k - x_k)_i/\tilde{\alpha}_k = (d_k)_i$;

which imply that $\lim_{k\to\infty, k\in K} v_k = \bar{d}$, so that (ii) is proved. □

Finally, we are now ready to prove the main convergence result for Algorithm DFN$_{simple}$. We highlight that according to the following proposition, every limit point of the sequence of iterates $\{x_k\}$, generated by Algorithm DFN$_{simple}$, is a stationary point for problem (4).

Proposition 3.8. *Let $\{x_k\}$ and $\{d_k\}$ be the sequences generated by Algorithm DFN$_{simple}$. Then, every limit point of $\{x_k\}$ is stationary for problem (4).*

Proof. We recall that by Definition 3.4 we consider the stationarity condition at $\bar{x}$:

$$f^\circ(\bar{x}; \bar{d}) = \limsup_{\substack{y \to \bar{x}, y \in X \\ t \downarrow 0, y + t\bar{d} \in X}} \frac{f(y + t\bar{d}) - f(y)}{t} \geq 0, \qquad \forall \bar{d} \in D(\bar{x}). \tag{23}$$

Let $\bar{x}$ be a limit point of $\{x_k\}$ and $\tilde{K} \subseteq \{1, 2, \dots\}$ be a subset of indices such that

$$\lim_{k\to\infty, k\in\tilde{K}} x_k = \bar{x}.$$

We proceed by contradiction and assume that a direction $\bar{d} \in D(\bar{x}) \cap S(0, 1)$ exists such that

$$f^\circ(\bar{x}; \bar{d}) = \limsup_{\substack{x_k \to \bar{x}, x_k \in X, \\ t \downarrow 0, x_k + t\bar{d} \in X}} \frac{f(x_k + t\bar{d}) - f(x_k)}{t} < 0. \tag{24}$$

Now, let $K \subseteq \tilde{K}$ be the set of indices considered in Corollary 3.3 such that

$$\lim_{k\to\infty, k\in K} x_k = \bar{x}, \tag{25}$$

$$\lim_{k\to\infty, k\in K} d_k = \bar{d}, \tag{26}$$

$$\lim_{k\to\infty, k\in K} \tilde{\alpha}_k = 0. \tag{27}$$

Further, let v_k be defined as in Lemma 3.7 relation (15), and set $y_k = x_k$ along with, for every $k \in K$,

$$\eta_k = \begin{cases} \alpha_k/\delta & \text{if } \alpha_k > 0 \\ \tilde{\alpha}_k & \text{otherwise.} \end{cases}$$

Then, by the instructions of the Algorithm DFN$_{simple}$ and recalling the results of Lemma 3.7, we have for every $k \in K$ and sufficiently large,

$$f(x_k + \eta_k v_k) > f(x_k) - \gamma \eta_k^2,$$

that is

$$\frac{f(x_k + \eta_k v_k) - f(x_k)}{\eta_k} > -\gamma\eta_k. \tag{28}$$

Then, considering that $\eta_k \to 0$ by Proposition 3.6,

$$\begin{aligned}
&\limsup_{\substack{x_k \to \bar{x}, x_k \in X \\ t \downarrow 0, x_k + t\bar{d} \in X}} \frac{f(x_k + t\bar{d}) - f(x_k)}{t} \ge \limsup_{k\to\infty, k\in K} \frac{f(x_k + \eta_k\bar{d}) - f(x_k)}{\eta_k} = \\
&\limsup_{k\to\infty, k\in K} \frac{f(x_k + \eta_k\bar{d}) + f(x_k + \eta_k v_k) - f(x_k + \eta_k v_k) - f(x_k)}{\eta_k} \ge \\
&\limsup_{k\to\infty, k\in K} \frac{f(x_k + \eta_k v_k) - f(x_k)}{\eta_k} - L\|\bar{d} - v_k\|,
\end{aligned}$$

where L is the Lipschitz constant of f. By (25)–(28) and *(ii)* of Lemma 3.7 we get, from the latter relation,

$$\limsup_{\substack{x_k \to \bar{x}, x_k \in X \\ t \downarrow 0, x_k + t\bar{d} \in X}} \frac{f(x_k + t\bar{d}) - f(x_k)}{t} \ge 0$$

which contradicts (24) and concludes the proof. $\square$

3.2. Combining DFN$_{simple}$ with coordinate searches

A possible way to improve the efficiency of Algorithm DFN$_{simple}$ can be to take advantage of the experience in the smooth case. For example, we can draw inspiration from the paper [25] where the objective function is repeatedly investigated along the directions $\pm e^1, \dots, \pm e^n$ in order to capture the local behavior of the objective function. In fact, the use of a set of search directions, which is constant with iterations, allows to store the actual and tentative steplengths, i.e. α^i and $\tilde{\alpha}^i$, respectively, that roughly summarize the sensitivity of the function along those directions. Thus, when the function is further investigated along such search directions, we can exploit information gathered in the previous searches along them.

In the following, we propose a new algorithm, where the search along coordinate directions is performed until the steplengths α^i and $\tilde{\alpha}^i$ are greater than a given threshold $\eta > 0$. In particular, the sampling along the coordinate directions is performed by means of a Continuous Search procedure [25, 23].

Algorithm CS-DFN

Data. $\theta \in (0,1)$, $\eta > 0$, $x_0 \in X$, $\tilde\alpha_0 > 0$, $\tilde\alpha_0^i > 0$, $d_0^i = e^i$, for $i = 1,\dots,n$, the map $\mathcal{M} : \mathbb{N} \mapsto \mathbb{R}^n$ such that, for $k \geq 0$, $d_k = \mathcal{M}(k)$ and $\|d_k\| = 1$.

For $k = 0,1,\dots$
  Set $y_k^1 = x_k$
  For $i = 1,\dots,n$
    Compute α by the *Continuous Search*$(\tilde\alpha_k^i, y_k^i, d_k^i; \alpha, d_{k+1}^i)$.
    If $(\alpha == 0)$ **then** set $\alpha_k^i = 0$ and $\tilde\alpha_{k+1}^i = \theta\tilde\alpha_k^i$
    else set $\alpha_k^i = \alpha$ and $\tilde\alpha_{k+1}^i = \alpha$
    Set $y_k^{i+1} = y_k^i + \alpha_k^i d_{k+1}^i$.
  End For
  If $\Big(\max_{i=1,\dots,n}\{\alpha_k^i, \tilde\alpha_k^i\} \leq \eta\Big)$ **then**
    Set $d_k = \mathcal{M}(k)$.
    Compute α_k by the *Projected Continuous Search*$(\tilde\alpha_k, y_k^{n+1}, d_k; \alpha_k, \tilde d_k)$.
    If $(\alpha_k == 0)$ **then** $\tilde\alpha_{k+1} = \theta\tilde\alpha_k$ and $y_k^{n+2} = y_k^{n+1}$
    else $\tilde\alpha_{k+1} = \alpha_k$ and $y_k^{n+2} = [y_k^{n+1} + \alpha_k \tilde d_k]_+$
  else Set $\tilde\alpha_{k+1} = \tilde\alpha_k$ and $y_k^{n+2} = y_k^{n+1}$
  Find $x_{k+1} \in X$ such that $f(x_{k+1}) \leq f(y_k^{n+2})$.
End For

Continuous search $(\tilde\alpha, y, p; \alpha, p^+)$

Data. $\gamma > 0$, $\delta \in (0,1)$.
Step 1. Compute the largest $\bar\alpha$ such that $y + \bar\alpha p \in X$. Set $\alpha = \min\{\bar\alpha, \tilde\alpha\}$.
Step 2. If $\alpha > 0$ and $f(y + \alpha p) \leq f(y) - \gamma\alpha^2$ **then** set $p^+ = p$ and go to Step 6.
Step 3. Compute the largest $\bar\alpha$ such that $y - \bar\alpha p \in X$. Set $\alpha = \min\{\bar\alpha, \tilde\alpha\}$.
Step 4. If $\alpha > 0$ and $f(y - \alpha p) \leq f(y) - \gamma\alpha^2$ **then** set $p^+ = -p$ and go to Step 6.
Step 5. Set $\alpha = 0$ and return.
Step 6. Let $\beta = \min\{\bar\alpha, (\alpha/\delta)\}$.
Step 7. If $\alpha = \bar\alpha$ or $f(y + \beta p^+) > f(y) - \gamma\beta^2$ return.
Step 8. Set $\alpha = \beta$ and go to Step 6.

The following three propositions concern the convergence analysis of Algorithm CS-DFN. The third proof is omitted since it is very similar to the corresponding one for Algorithm DFN$_{simple}$.

Proposition 3.9. *Algorithm CS-DFN is well-defined.*

Proof. In order to show that Algorithm CS-DFN is well-defined we have to prove that both the *Continuous Search* and the *Projected Continuous Search* cannot infinitely cycle, respectively, between Step 6 and Step 8, and between Step 4 and Step 6. Let us first consider the *Continuous Search*. We proceed by contradiction and assume that an infinite monotonically increasing sequence of positive numbers $\{\beta_j\}$ exists such that

$$\beta_j < \bar{\alpha} \quad \text{and} \quad f(y + \beta_j p^+) \leq f(y) - \gamma \beta_j^2.$$

The above relation contradicts the fact that X is compact, by definition, and that function f in problem (4) is continuous. The rest of the proof trivially follows from Proposition 3.5. □

The proposition that follows concerns convergence to zero of the steplengths in Algorithm CS-DFN. In particular, since α_k^i and $\tilde{\alpha}_k^i$ tend to zero, it results that the search along the dense direction d_k is performed eventually infinitely many times.

Proposition 3.10. *Let $\{\alpha_k^i\}$, $\{\tilde{\alpha}_k^i\}$, $\{\alpha_k\}$ and $\{\tilde{\alpha}_k\}$ be the sequences generated by Algorithm CS-DFN, then*

$$\lim_{k\to\infty} \max\{\alpha_k^1, \tilde{\alpha}_k^1, \dots, \alpha_k^n, \tilde{\alpha}_k^n\} = 0, \tag{29}$$

$$\lim_{k\to\infty} \max\{\alpha_k, \tilde{\alpha}_k\} = 0. \tag{30}$$

Proof. Reasoning as in the proof of Proposition 1 in [25], we can prove (29).
Now we have to show (30). By virtue of (29), we know that an index $\bar{k}$ exists such that the dense direction d_k is investigated for all $k \geq \bar{k}$.
Then, without loss of generality, we split the iteration sequence $\{k\}$ into two sets K_1 and K_2, with $K_1 \cup K_2 = \{k\}$ and $K_1 \cap K_2 = \emptyset$. We denote by

- K_1 the set of iterations when $\tilde{\alpha}_{k+1} = \alpha_k$;
- K_2 the set of iterations when $\tilde{\alpha}_{k+1} = \theta\tilde{\alpha}_k$.

Hence, the proof follows by reasoning as in the proof of Proposition 3.6. □

Proposition 3.11. *Let $\{x_k\}$ and $\{d_k\}$ be the sequences generated by Algorithm CS-DFN. Then, every limit point of $\{x_k\}$ is stationary for problem (4).*

Proof. The proof trivially follows from Proposition 3.8. □

4. The nonsmooth nonlinearly constrained case

In this section, we consider Lipschitz-countinuous nonlinearly constrained problems of the following form:

$$\begin{aligned} &\min \ f(x) \\ &s.t. \ \ g(x) \leq 0, \\ &\qquad l \leq x \leq u, \end{aligned} \tag{31}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$ and $l, u \in \mathbb{R}^n$. The vectors l and u correspond respectively to lower and upper bounds on the variables $x \in \mathbb{R}^n$, and satisfy the additional condition $l < u$. We also assume throughout the paper that $f(x)$ and $g(x)$ are Lipschitz continuous functions, though they may be possibly *nondifferentiable*. Furthermore, $\mathcal{F}$ indicates the feasible set of problem (31), i.e.

$$\mathcal{F} = \{x \in X : g(x) \leq 0\}.$$

We highlight that, by definition, $X = \{x \in \mathbb{R}^n \ : \ l \leq x \leq u\}$ is a compact subset of $\mathbb{R}^n$.

4.1. Assumption and preliminary material

We further introduce some definitions and assumptions related to problem (31). First, in order to carry out the theoretical analysis, we use a version of the Mangasarian-Fromowitz Constraint Qualification (MFCQ) condition for nonsmooth problems.

Assumption 2 (MFCQ) *Given problem* (31)*, for any* $x \in X \setminus \mathcal{F}$ *a direction* $d \in D(x)$ *exists such that*

$$(\xi^{g_i})^\top d < 0, \tag{32}$$

for all $\xi^{g_i} \in \partial g_i(x)$*,* $i \in \{1, \dots, m : \ g_i(x) \geq 0\}$*.*

The nonlinearly constrained problem (31) can be handled partitioning the constraints in two different sets, the first one defined by general inequality constraints, and the second one consisting of simple bound constraints. Then, for this kind of problem, we can state necessary optimality conditions that explicitly take into account the presence of these two different sets of constraints.

Proposition 4.1 (Fritz John Optimality Conditions) *Let* $x^\star \in \mathcal{P}$ *be a local minimum of the problem* (31)*. Then, multipliers* $\lambda_0^\star, \lambda_1^\star, \dots, \lambda_m^\star \in \mathbb{R}$ *not all zero exist such that*

$$\max_{\substack{\xi^f \in \partial f(x^\star) \\ \xi^{g_i} \in \partial g_i(x^\star)}} \left\{ \left(\lambda_0^\star \xi^f + \sum_{i=1}^m \lambda_i^\star \xi^{g_i} \right)^\top d \right\} \geq 0, \qquad \forall \ d \in D(x^\star), \tag{33}$$

$$\lambda_i^\star \geq 0 \quad \textit{and} \quad \lambda_i^\star g_i(x^\star) = 0, \qquad \forall \ i = 1, \dots, m. \tag{34}$$

Proof. We refer the interested reader to the proof of Proposition A.1 in Appendix, where the more general case of problems with both equality and inequality constraints is considered. $\square$

Corollary 4.2 (KKT Necessary Optimality Conditions) *Let* $x^\star \in \mathcal{P}$ *be a local minimum of the problem* (31) *such that, for all* $\xi^{g_i} \in \partial g_i(x^\star)$*,* $i \in \{1, \dots, m : g_i(x^\star) = 0\}$*, a direction* $d \in D(x^\star)$ *exists such that:*

$$(\xi^{g_i})^\top d < 0, \qquad \forall \ i \in \{1, \dots, m : g_i(x^\star) = 0\}.$$

Then, multipliers $\lambda_1^\star, \dots, \lambda_m^\star \in \mathbb{R}$ *exist such that*

$$\max_{\substack{\xi^f \in \partial f(x^\star) \\ \xi^{g_i} \in \partial g_i(x^\star)}} \left\{ \left(\xi^f + \sum_{i=1}^m \lambda_i^\star \xi^{g_i} \right)^\top d \right\} \geq 0, \qquad \forall d \in D(x^\star), \tag{35}$$

$$\lambda_i^\star \geq 0 \quad \textit{and} \quad \lambda_i^\star g_i(x^\star) = 0, \qquad \forall \ i = 1, \dots, m.$$

Proof. The proof follows from Proposition A.2 in Appendix, where again the more general case of problems with both equality and inequality constraints is considered. $\square$

As regards the stationarity conditions for problem (31), taking into account the above propositions, we can now give the following definition.

Definition 4.3 (Stationary point) *Given the problem (31),* $\bar{x}$ *is a stationary point of (31) if the multipliers* $\bar{\lambda}_1, \dots, \bar{\lambda}_m$ *exist such that the following conditions hold:*

$$\max_{\substack{\xi^f \in \partial f(\bar{x}) \\ \xi^{g_i} \in \partial g_i(\bar{x})}} \left\{ (\xi^f + \sum_{i=1}^m \bar{\lambda}_i \xi^{g_i})^\top d \right\} \geq 0, \qquad \forall d \in D(\bar{x}), \tag{36}$$

$$\bar{\lambda}_i \geq 0 \qquad \text{and} \qquad \bar{\lambda}_i g_i(\bar{x}) = 0, \qquad \forall \ i = 1, \dots, m. \tag{37}$$

4.2. The penalty approach

Given problem (31), we introduce the following penalty function

$$Z_\varepsilon(x) = f(x) + \frac{1}{\varepsilon}\sum_{i=1}^{m} \max\{0, g_i(x)\}$$

and define the penalized problem

$$\begin{array}{ll} \min & Z_\varepsilon(x) \\ s.t. & x \in X. \end{array} \tag{38}$$

Remark 4.4. *Observe that, since f and g_i, $i = 1, \dots, m$, are Lipschitz continuous, with Lipschitz constants L_f and L_{g_i}, $i = 1, \dots, m$, the penalty function Z_ε is Lipschitz continuous too, with Lipschitz constant*

$$L = L_f + \frac{1}{\varepsilon}\sum_{i=1}^{m} L_{g_i}.$$

Remark 4.5. *Note that problem (38), for any $\varepsilon > 0$, has the same structure and properties of problem (4).*

We further note that our penalty approach differs form the ones previously proposed in the literature (see e.g. [12] and references therein), since only the general nonlinear constraints are penalized. The minimization of the penalty function is then carried out on the set defined by the bound constraints. We report in the following proposition the equivalence between the problem (38) and the nonlinearly constrained problem (31).

Proposition 4.6. *Let Assumption 2 hold. Given problem* (31) *and considering problem* (38)*, a threshold value $\varepsilon^\star > 0$ exists such that, for every $\varepsilon \in (0, \varepsilon^\star]$, every point $\bar{x} \in X$ such that*

$$\max_{\xi \in \partial Z_\varepsilon(\bar{x})} \xi^\top d \geq 0, \qquad \forall\, d \in D(\bar{x}), \tag{39}$$

is stationary for Problem (31).

The proof of the latter result is reported in Appendix B. We note that condition (39) is equivalent to the fact that the Clarke directional derivative of Z_ε at $\bar{x}$ is non-negative (see [9]).

4.3. A derivative-free algorithm

Now we report the algorithm adopted for solving problem (38), which is obtained from Algorithm CS-DFN by replacing f with Z_ε. For the sake of simplicity, we omit to report also the extension of Algorithm DFN$_{simple}$ to the general inequality constrained case, which requires trivial modifications.

Algorithm DFN$_{con}$

Data. $\theta \in (0,1)$, $x_0 \in X$, $\varepsilon > 0$, $\tilde\alpha_0 > 0$, $\tilde\alpha_0^i > 0$, $d_0^i = e^i$, for $i = 1,\dots,n$, the map $\mathcal{M} : \mathbb{N} \mapsto \mathbb{R}^n$ such that $d_k = \mathcal{M}(k)$ and $\|d_k\| = 1$.

For $k = 0, 1, \dots$
    Set $y_k^1 = x_k$
    For $i = 1,\dots,n$
        Compute α by the *Continuous Search*$(\tilde\alpha_k^i, y_k^i, d_k^i; \alpha, d_{k+1}^i)$.
        If $(\alpha == 0)$ **then** set $\alpha_k^i = 0$ and $\tilde\alpha_{k+1}^i = \theta\tilde\alpha_k^i$
        else set $\alpha_k^i = \alpha$ and $\tilde\alpha_{k+1}^i = \alpha$
        Set $y_k^{i+1} = y_k^i + \alpha_k^i d_{k+1}^i$.
    End For
    If $(\max_{i=1,\dots,n}\{\alpha_k^i, \tilde\alpha_k^i\} \le \eta)$ **then** $(\max_{i=1,\dots,n}\{\alpha_k^i, \tilde\alpha_k^i, \tilde\alpha_k\} \le \delta_k)$
            Set $d_k = \mathcal{M}(k)$.
            Compute α_k by the *Projected Continuous Search*$(\tilde\alpha_k, y_k^{n+1}, d_k; \alpha_k, \tilde d_k)$.
            If $(\alpha_k == 0)$ **then** $\tilde\alpha_{k+1} = \theta\tilde\alpha_k$ and $y_k^{n+2} = y_k^{n+1}$
            else $\tilde\alpha_{k+1} = \alpha_k$ and $y_k^{n+2} = [y_k^{n+1} + \alpha_k\tilde d_k]_+$
    else set $\tilde\alpha_{k+1} = \tilde\alpha_k$, $y_k^{n+2} = y_k^{n+1}$
    Find $x_{k+1} \in X$ such that $Z_\varepsilon(x_{k+1}) \le Z_\varepsilon(y_k^{n+2})$.
End For

We observe that Algorithm DFN$_{\texttt{con}}$ can be used to solve the constrained problem (31) provided that the penalty parameter ε is sufficiently small, as the following proposition states.

Proposition 4.7. *Let $\varepsilon \in (0, \varepsilon^*]$ where ε^* is defined in Proposition 4.6; let $\{x_k\}$ and $\{d_k\}$ be the sequences generated by Algorithm DFN$_{\texttt{con}}$. Then, every limit point of $\{x_k\}$ is stationary for problem (31).*

Proof. By Proposition 3.11, every limit point $\bar x$ of $\{x_k\}$ is stationary for problem (38), namely

$$Z_\varepsilon^\circ(\bar x; d) \ge 0, \qquad \forall\, d \in D(\bar x).$$

Recalling (5), the latter result implies

$$\limsup_{y \to \bar x,\, t \downarrow 0} \frac{Z_\varepsilon(y + td) - Z_\varepsilon(y)}{t} \ge Z_\varepsilon^\circ(\bar x; d) \ge 0, \quad \forall\, d \in D(\bar x).$$

Since the limit in the above relation is the Clarke directional derivative, and by [9]

$$\limsup_{y \to \bar x,\, t \downarrow 0} \frac{Z_\varepsilon(y + td) - Z_\varepsilon(y)}{t} = \max_{\xi \in \partial Z_\varepsilon(\bar x)} \xi^\top d \ge 0, \quad \forall\, d \in D(\bar x),$$

then, by Proposition 4.6, we have that $\bar x$ is also stationary for problem (31). $\square$

5. Implementation details and numerical results

This section is devoted to investigate the numerical issues related to the implementation of the proposed algorithms. We first report the numerical experience related to bound-constrained problems, then we analyze the computational results related to the nonlinearly constrained case. All the experiments, have been conducted allowing for a maximum number of 20000 function evaluations (which is quite reasonable considered the dimensions of our test problems).

For the parameters included in the proposed algorithms (DFN_{simple}, CS-DFN, DFN_{con}) we considered the following setting: $\theta = 0.5$, $\gamma = 10^{-6}$, $\delta = 0.5$, $\eta = 10^{-3}$,

$$\begin{aligned} \tilde{\alpha}_0^i &= \max\left\{10^{-3}, \min\{1, |(x_0)_i|\}\right\}, \qquad i = 1, \dots, n, \\ \tilde{\alpha}_0 &= \frac{1}{n}\sum_{i=1}^{n} \tilde{\alpha}_0^i. \end{aligned}$$

Regarding the choice of the new iterate x_{k+1}, we remark that:

- in Algorithm DFN_{simple}, x_{k+1} is computed starting from $\tilde{x}_k$ and performing Projected continuous searches along a set of $n-1$ directions which define an orthonormal basis in $\mathbb{R}^n$ along with d_k;

- in Algorithms CS-DFN and DFN_{con}, if $(\max_{i=1,\dots,n}\{\alpha_k^i, \tilde{\alpha}_k^i\} \le \eta)$ then x_{k+1} is computed as above but starting from y_k^{n+2}. Otherwise, we set $x_{k+1} = y_k^{n+2}$.

In the implementation of Algorithm DFN_{con} we used a vector of penalty parameters $\varepsilon \in \mathbb{R}^m$ and considered the penalty function

$$Z_\varepsilon(x) = f(x) + \sum_{i=1}^{m} \frac{1}{\varepsilon_i} \max\{0, g_i(x)\},$$

which trivially preserves all the theoretical results proved in Section 4. The vector of penalty parameters is iteratively updated during progress of the algorithm and, in particular, we chose

$$(\varepsilon_0)_i = \begin{cases} 10^{-3} & \text{if } \max\{0, g_i(x_0)\} < 1 \\ 10^{-1} & \text{otherwise,} \end{cases} \qquad i = 1, \dots, m,$$

and adopted the updating rule

$$(\varepsilon_{k+1})_i = \begin{cases} 10^{-2}(\varepsilon_k)_i & \text{if } (\varepsilon_k)_i g_i(x_k) > \max\{\alpha_k, \tilde{\alpha}_k\} \\ (\varepsilon_k)_i & \text{otherwise,} \end{cases} \qquad i = 1, \dots, m.$$

The above updating rule is applied right before computation of the new iterate x_{k+1}. We notice that the rule described above takes inspiration from derivative-based exact penalty approaches (see e.g. [22], [27]) where the updating rule for the penalty parameter is based on the (scaled) comparison between the stationarity measure of the point and the constraint violation. In a derivative-free context, the stationarity measure can be approximated by means of the steplengths selected along the search directions, as showed in [19].

As a final note, in the implementation of our algorithms, we used as termination condition

$$\max\{\alpha_k, \tilde{\alpha}_k\} \le 10^{-13}. \tag{40}$$

However, we highlight that the algorithms are compared by means of performance and data profiles [26], that is by using a normalized convergence test on the function values. Thus, we adopted the tight convergence test (40) in order to provide enough information on all the codes compared.

The codes DFN$_{simple}$, CS-DFN and DFN$_{con}$ are freely available for download at the url:

`http://www.dis.uniroma1.it/∼lucidi/DFL`

5.1. Box constrained problems

The first part of the numerical experience has been carried out on a set of 142 bound-constrained problems from [31], [24] and [26], with a number of variables in the range $[1, 200]$ (see Table 1).

n	1	2	3	4	5	6	7	8	9	10	11	12	15	20	50	100	200
# of problems	4	34	12	17	10	8	6	6	6	10	6	6	1	8	4	2	2

Table 1: Distribution of problem dimensions for the bound-constrained case

As showed in the theoretical analysis of the different algorithms, our linesearch-based approach is able to guarantee convergence towards stationary points of the nonsmooth problem, by using search directions generated by a mapping $\mathcal{M}$ satisfying Assumption 1. This assumption is not particularly strong since it allows a wide choice of different mappings. In particular, we can adopt the mapping based on the Halton sequence [14], which is the one implemented in the NOMAD package [1, 20, 2]. But, unlike NOMAD, further mappings can be easily embedded into our algorithms, since we are not committed to use a modified Halton sequence in order to generate points on a mesh (see e.g. [14]). For instance, we implemented a mapping based on the Sobol sequence [30, 8], which is a pseudorandom sequence widely used in practice.

In order to show the behavior of the above pseudorandom sequences, we preliminarily compared two versions of the Algorithm DFL$_{simple}$, which respectively use the Halton and the Sobol sequence, on the test set of bound-constrained problems described above. The resulting experience is reported in Figure 1 using data and performance profiles [26].

As we can see, the Sobol pseudorandom sequence outperforms the Halton one for all precision levels, both in terms of efficiency and robustness. Then, we compared both our Algorithms DFN$_{simple}$ and CS-DFN and reported the results in Figure 2 in terms of performance and data profiles.

As we can see, the combination of coordinate and dense directions can improve the performance of the algorithm.

Finally, we compared CS-DFN with two state-of-the-art derivative free optimization codes, namely NOMAD [1, 20, 2] and BOBYQA [29]. In order to obtain a fair comparison, we ran NOMAD by using the default parameters, except for the `MODEL_SEARCH` which was disabled. Furthermore, BOBYQA was run by setting $\texttt{RHOBEG} = 1$ and $\texttt{RHOEND} = 10^{-13}$. The latter choice is motivated by the fact that the main focus of this paper is not on the effectiveness of models, but rather on the potentialities of the linesearch approach when using a dense set of directions. By taking a look at Figure 3, we can notice that BOBYQA is outperformed both by NOMAD and CS-DFN. This shows on the one hand that as expected BOBYQA, which is devised for continuously differentiable problems, suffers on nondifferentiable problems. On the other hand, this suggests that the chosen test set is composed of hard nonsmooth problems. Further, from Figure 3 we can infer that CS-DFN and NOMAD are quite comparable, with a slight preference

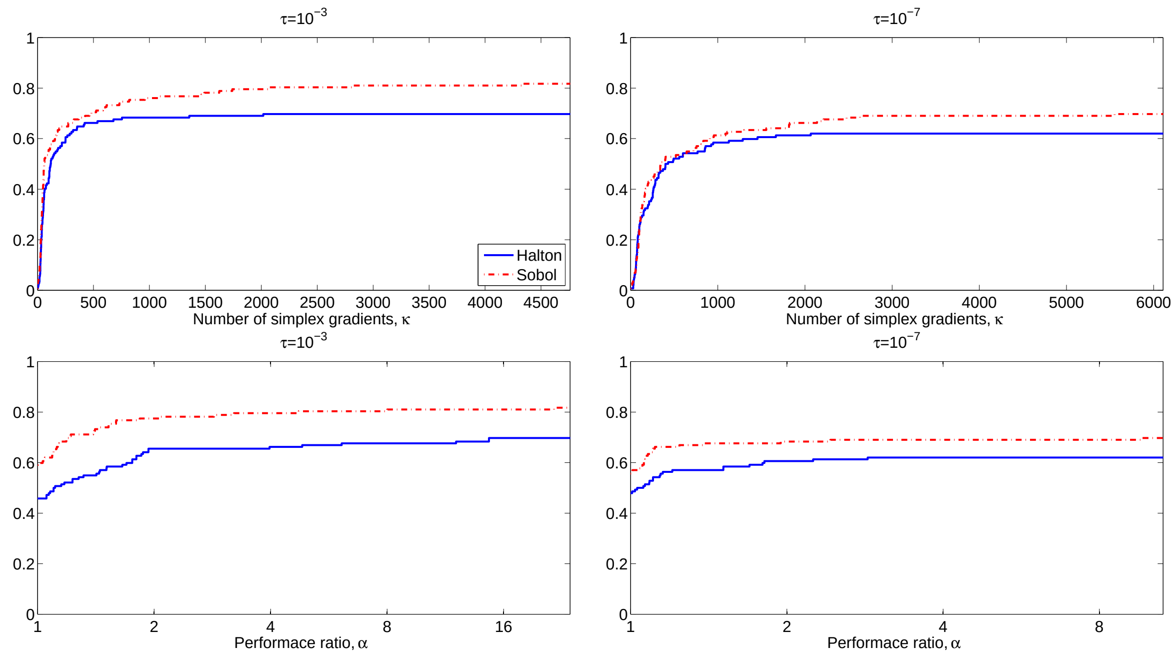

Figure 1: Data (top) and performance (bottom) profiles for the 142 bound-constrained problems. Comparison between Sobol and Halton pseudorandom sequences within DFN_{simple}

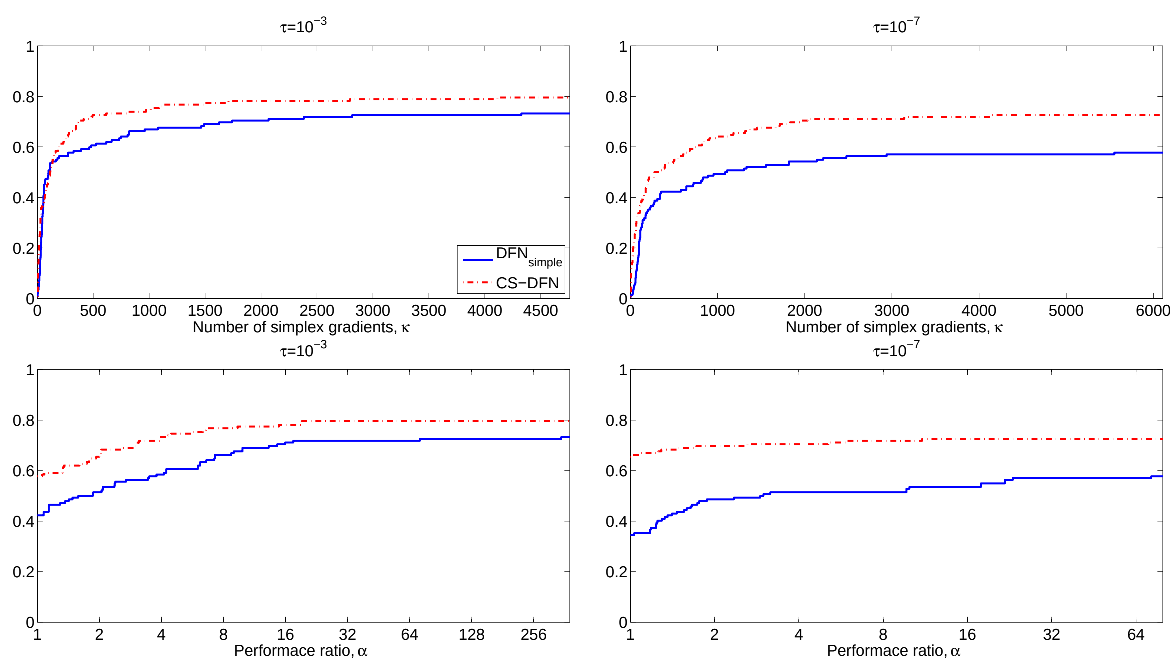

Figure 2: Data (top) and performance (bottom) profiles for the 142 bound-constrained problems. Comparison between DFN_{simple} and CS-DFN

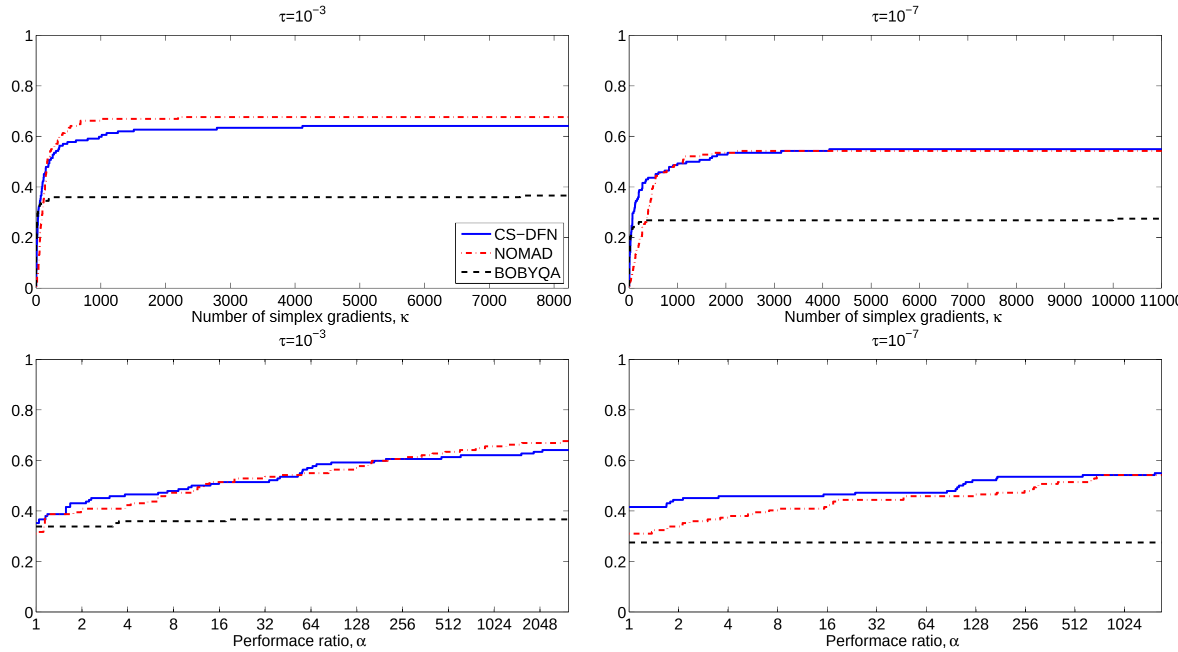

Figure 3: Data (top) and performance (bottom) profiles for the 142 bound-constrained problems. Comparison among CS-DFN, NOMAD and BOBYQA

for CS-DFN (at least for high precision levels).

5.2. Nonlinearly constrained problems

In the second part of our numerical experience, we defined a set of hard nonsmooth nonlinearly constrained test problems, by pairing the objective functions of the collection [24] with the constraint families proposed in [17], thus obtaining 296 problems. The problems in this collection have a number of constraints m in the range $[1, 199]$ and a number of variables n in the range $[1, 200]$ (see Table 2). We note that 205 out of 296 problems have a starting point x_0 which is not feasible, that is

$$h(x_0) > 10^{-6}, \qquad \text{with } h(x) = \max\left\{0, \max_{i=1,\dots,m}\{g_i(x)\}\right\}. \tag{41}$$

In order to adapt the procedure for constructing performance and data profiles, as proposed in [26], to the nonlinearly constrained case, we considered the convergence test

$$\tilde{f}_0 - f(x) \geq (1-\tau)(\tilde{f}_0 - f_L),$$

where $\tilde{f}_0$ is the objective function value of the *worst feasible point* determined by all the solvers (note that in the bound-constrained case, $\tilde{f}_0 = f(x_0)$), $\tau > 0$ is a tolerance, and f_L is computed for each problem as the smallest value of f (at a feasible point) obtained by any solver within 20000 function evaluations. We notice that when a point is not feasible (i.e. $h(x) > 10^{-6}$) we set $f(x) = +\infty$.
In Figure 4, we report the comparison among CS-DFN, NOMAD and COBYLA [28]. NOMAD

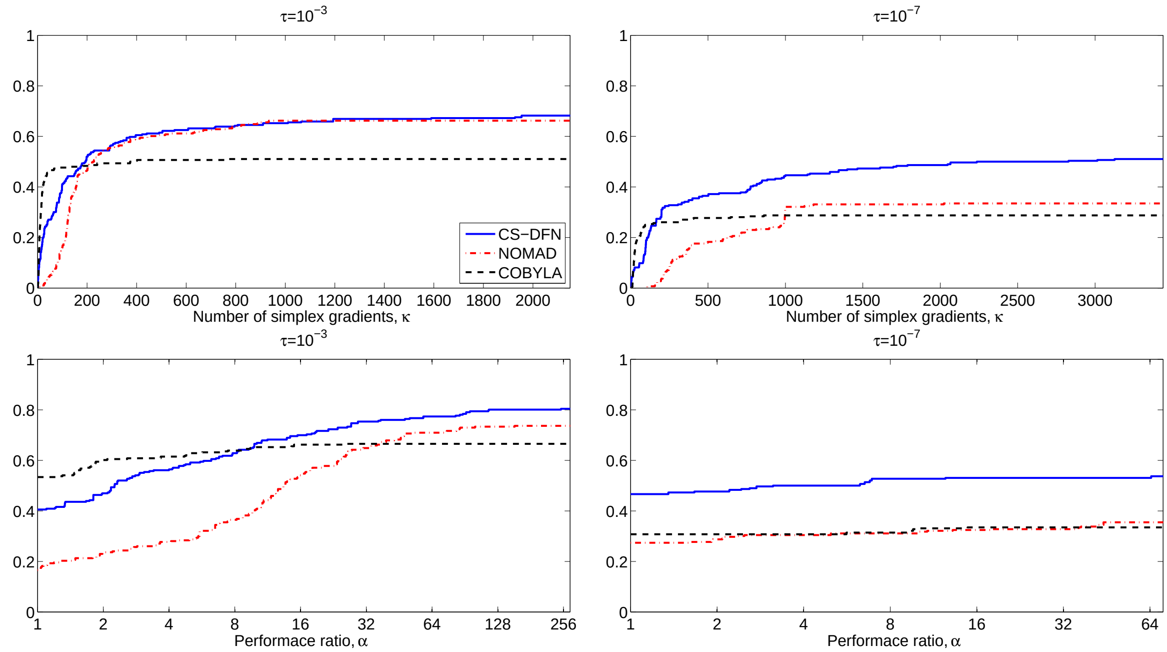

Figure 4: Data (top) and performance (bottom) profiles for the 296 constrained problems. Comparison among CS-DFN, NOMAD and COBYLA

was run by setting the constraints type to `PEB`, so that constraints are treated first with the progressive barrier, and once satisfied, with the extreme barrier approach. COBYLA was run by setting `RHOBEG` $= 1$ and `RHOEND` $= 10^{-13}$. We note that, when relatively small precision is required, COBYLA has an initial fast progress, but is not as robust as the other two codes. When high precision is required, CS-DFN outperforms both NOMAD and COBYLA.

n	2	3	4	5	6	10	20	50	100	200
# of problems	96	30	40	10	10	20	40	10	20	20

m	1	2	3	4	5	6	8	9	11	12
# of problems	151	39	17	24	9	1	3	2	3	2
m	18	19	22	23	48	49	98	99	198	199
# of problems	9	6	3	2	3	2	6	4	6	4

Table 2: Distribution of problem dimensions (n number of variables, m number of constraints) for the nonlinearly constrained test set

6. Conclusions

In this paper, we described new methods for dealing with nonsmooth optimization problems when no first-order information is available. We adopted a projected linesearch approach and we combined it with an asymptotically dense set of search directions. First, we considered problems with only bound constraints on the variables and we proposed two different algorithms

for their solution. Then, when dealing with nonlinear inequality constraints, we introduced the use of an exact penalty function to transform the given nonlinearly constrained problem into a bound-constrained problem, which is solved by adapting the method proposed for the bound-constrained case.
The numerical results reported in the paper show that the use of linesearches gives a large freedom in the choice of the way the set of directions is generated. Furthermore our analysis highlights the fact that coordinate searches can somehow improve the performance of the proposed algorithms. Finally, the comparison with other state-of-the-art codes on two large test sets of bound-constrained and nonlinearly-constrained nonsmooth problems points out that the proposed framework is efficient and sufficiently robust.

A. Necessary optimality conditions

In this section we consider the following nonlinear programming problem featuring both equality and inequality nonlinear constraints beside the bound constraints on the variables:

$$\begin{array}{ll} \min & f(x) \\ s.t. & h(x) = 0, \\ & g(x) \leq 0, \\ & x \in X, \end{array} \tag{42}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}^p$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ are Lipschitz continuous functions. In the following, we denote

$$\mathcal{P} = \{x \in \mathbb{R}^n : g(x) \leq 0,\ h(x) = 0\} \cap X.$$

The following propositions extend the results in [15] to the case where both equality and inequality constraints are present beside an additional convex set of constraints. The proofs follow by combining results reported in [7] and [15].

Proposition A.1 (Fritz John Optimality Conditions) *Let $x^\star \in \mathcal{P}$ be a local minimum of the problem (42). Then, multipliers $\lambda_0^\star, \lambda_1^\star, \dots, \lambda_m^\star, \mu_1^\star, \dots, \mu_p^\star \in \mathbb{R}$ not all zero exist such that*

$$\max_{\substack{\xi^f \in \partial f(x^\star) \\ \xi^{g_i} \in \partial g_i(x^\star) \\ \xi^{h_j} \in \partial h_j(x^\star)}} \left\{ \left(\lambda_0^\star \xi^f + \sum_{j=1}^p \mu_j^\star \xi^{h_j} + \sum_{i=1}^m \lambda_i^\star \xi^{g_i} \right)^\top d \right\} \geq 0, \qquad \forall\ d \in D(x^\star) \tag{43}$$

$$\lambda_i^\star \geq 0 \quad \textit{and} \quad \lambda_i^\star g_i(x^\star) = 0 \qquad \forall\ i = 1, \dots, m. \tag{44}$$

Proof. Let

$$B = \{y \in \mathbb{R}^n : \|y - x^\star\| \leq \epsilon\}.$$

For each $k = 1, 2, \dots$, let us consider the penalized problem

$$\begin{array}{l} \min\ F_k(x) = f(x) + \dfrac{k}{2} \displaystyle\sum_{j=1}^p [h_j(x)]^2 + \dfrac{k}{2} \sum_{i=1}^m [g_i^+(x)]^2 + \dfrac{1}{2}\|x - x^\star\|^2 \\ s.t.\ \ x \in B \cap X, \end{array} \tag{45}$$

with $\epsilon > 0$ such that $f(x^\star) \le f(x)$, for all $x \in B \cap X$. Let x_k be a solution of problem (45). Then $\{x_k\}$ admits limit points. Since

$$F_k(x_k) = f(x_k) + \frac{k}{2}\sum_{j=1}^{p}[h_j(x_k)]^2 + \frac{k}{2}\sum_{i=1}^{m}[g_i^+(x_k)]^2 + \frac{1}{2}\|x_k - x^\star\|^2 \le F_k(x^\star) = f(x^\star) \tag{46}$$

we get, for every limit point $\bar{x}$, dividing the above relation by k and taking the limit

$$\sum_{j=1}^{p}[h_j(\bar{x})]^2 + \sum_{i=1}^{m}[g_i^+(\bar{x})]^2 = 0.$$

Hence, every limit point $\bar{x}$ is feasible. Furthermore, relation (46) yields $f(x_k) + \frac{1}{2}\|x_k - x^\star\|^2 \le f(x^\star)$ for all k; thus by taking the limit for $k \to \infty$ we get

$$f(\bar{x}) + \frac{1}{2}\|\bar{x} - x^\star\| \le f(x^\star). \tag{47}$$

Since $\bar{x} \in B \cap X$ and $\bar{x}$ is feasible, we have $f(x^\star) \le f(\bar{x})$; combining the latter inequality with (47) yields $\|\bar{x} - x^\star\| = 0$, so that $\bar{x} = x^\star$. Thus, we can conclude that the entire sequence $\{x_k\}$ converges to $x^\star$ and it follows that, for sufficiently large k, x_k is an interior point of the closed ball B.
Let us now define the following quantities:

$$\begin{aligned}
N_k &= \left\{1 + \sum_{j=1}^{p}[kh_j(x_k)]^2 + \sum_{i=1}^{m}\left[kg_i^+(x_k)\right]^2\right\}^{1/2}, \\
\lambda_0^k &= 1/N_k, \\
\lambda_i^k &= kg_i^+(x_k)/N_k, \\
\mu_j^k &= kh_j(x_k)/N_k.
\end{aligned}$$

Then, it results that $(\lambda_0^k, \lambda_1^k, \dots, \lambda_m^k, \mu_1^k, \dots, \mu_p^k)$ is a unit and nonnegative vector so that the sequence $\{(\lambda_0^k, \lambda_1^k, \dots, \lambda_m^k,\ \mu_1^k, \dots, \mu_p^k)\}$ admits limit points. Let us consider a subsequence of $\{x_k\}$, which we relabel $\{x_k\}$ again, for which the subsequence $\{(\lambda_0^k, \lambda_1^k, \dots, \lambda_m^k, \mu_1^k, \dots, \mu_p^k)\}$ converges to $(\lambda_0^\star, \lambda_1^\star, \dots, \lambda_m^\star,\ \mu_1^\star, \dots, \mu_p^\star)$. Note that, by definition, $\lambda_i^\star = 0$ for all i such that $g_i(x^\star) < 0$.
Now we proceed by contradiction and assume that a direction $\bar{d} \in D(x^\star)$ exists such that

$$\max_{\substack{\xi^f \in \partial f(x^\star) \\ \xi^{g_i} \in \partial g_i(x^\star) \\ \xi^{h_j} \in \partial h_j(x^\star)}} \left\{\left(\lambda_0^\star \xi^f + \sum_{j=1}^{p}\mu_j^\star \xi^{h_j} + \sum_{i=1}^{m}\lambda_i^\star \xi^{g_i}\right)^\top \bar{d}\right\} < 0, \qquad \bar{d} \in D(x^\star). \tag{48}$$

Since, for all k, x_k is a solution of problem (45), by (6), we can write

$$F_k^\circ(x_k; d) = \max_{\xi \in \partial F_k(x_k)} \xi^\top d \ge 0, \qquad \forall\, d \in D(x_k),$$

which is satisfied if

$$\left(\xi_k^f + \sum_{j=1}^{p} kh_j(x_k)\xi_k^{h_j} + \sum_{i=1}^{m} kg_i^+(x_k)\xi_k^{g_i} + (x_k - x^\star)\right)^\top d \ge 0, \qquad \forall\, d \in D(x_k),$$

where $\xi_k^f \in \partial f(x_k)$, $\xi_k^{h_j} \in \partial h_j(x_k)$ and $\xi_k^{g_i} \in \partial g_i(x_k)$. Recalling Proposition 3.2, for k sufficiently large $D(x^\star) \subseteq D(x_k)$, and we have

$$\left(\xi_k^f + \sum_{j=1}^{p} k h_j(x_k)\xi_k^{h_j} + \sum_{i=1}^{m} k g_i^+(x_k)\xi_k^{g_i} + (x_k - x^\star)\right)^\top d \geq 0, \qquad \forall\, d \in D(x^\star).$$

Dividing the above relation by N_k and taking the limit for $k \to \infty$, we obtain

$$\left(\lambda_0^\star \bar{\xi}^f + \sum_{j=1}^{p} \mu_j^\star \bar{\xi}^{h_i} + \sum_{i=1}^{m} \lambda_i^\star \bar{\xi}^{g_i}\right)^\top d \geq 0, \qquad \forall\, d \in D(x^\star),$$

where $\bar{\xi}^f \in \partial f(x^\star)$, $\bar{\xi}^{h_j} \in \partial h_j(x^\star)$ and $\bar{\xi}^{g_i} \in \partial g_i(x^\star)$. The above relation contradicts (48). □

Corollary A.2 (KKT Necessary Optimality Conditions) *Let $x^\star \in \mathcal{P}$ be a local minimum of the problem (42) such that, for all $\xi^{h_j} \in \partial h_j(x^\star)$, $j = 1, \dots, p$, $\xi^{g_i} \in \partial g_i(x^\star)$, $i \in \{1, \dots, m : g_i(x^\star) = 0\}$, a direction $d \in D(x^\star)$ exists such that:*

$$(\xi^{h_j})^\top d = 0, \qquad \forall\, j = 1, \dots, p, \tag{49}$$

$$(\xi^{g_i})^\top d < 0, \qquad \forall\, i \in \{1, \dots, m : g_i(x^\star) = 0\}, \tag{50}$$

and, there does not exist scalars $\mu_1^\star, \dots, \mu_p^\star$ not all zero such that

$$\left(\sum_{j=1}^{p} \mu_j^\star \xi^{h_j}\right)^\top d \geq 0, \qquad \forall\, d \in D(x^\star). \tag{51}$$

Then, multipliers $\lambda_1^\star, \dots, \lambda_m^\star, \mu_1^\star, \dots, \mu_p^\star \in \mathbb{R}$ exist such that

$$\max_{\substack{\xi^f \in \partial f(x^\star) \\ \xi^{g_i} \in \partial g_i(x^\star) \\ \xi^{h_j} \in \partial h_j(x^\star)}} \left\{\left(\xi^f + \sum_{j=1}^{p} \mu_j^\star \xi^{h_j} + \sum_{i=1}^{m} \lambda_i^\star \xi^{g_i}\right)^\top d\right\} \geq 0, \qquad \forall d \in D(x^\star). \tag{52}$$

$$\lambda_i^\star \geq 0 \quad \text{and} \quad \lambda_i^\star g_i(x^\star) = 0 \qquad \forall\, i = 1, \dots, m.$$

Proof. By Proposition A.1, we know that multipliers $\lambda_0^\star$, $\lambda_i^\star$, $i = 1, \dots, m$, $\mu_j^\star$, $j = 1, \dots, p$, exist such that (43) and (44) hold. Now, we proceed by contradiction and assume that $\lambda_0^\star = 0$. Reasoning as in the proof of Proposition A.1, we can obtain

$$\left(\sum_{j=1}^{p} \mu_j^\star \bar{\xi}^{h_j} + \sum_{i=1}^{m} \lambda_i^\star \bar{\xi}^{g_i}\right)^\top d \geq 0, \qquad \forall\, d \in D(x^\star), \tag{53}$$

where $\bar{\xi}^{h_j} \in \partial h_j(x^\star)$ and $\bar{\xi}^{g_i} \in \partial g_i(x^\star)$. If $\lambda_i^\star$ were all zero, then a contradiction would arise between (53) and (51). Otherwise, a contradiction would arise considering (53), (49) and (50), which concludes the proof. □

B. Exactness properties of $Z_\varepsilon(x)$

We assume throughout this section that the Assumption 2 holds.

Proposition B.1. *Given problem* (31) *and considering problem* (38)*, a threshold value* $\varepsilon^\star > 0$ *exists such that, for every* $\varepsilon \in (0, \varepsilon^\star]$*, the function* $Z_\varepsilon(x)$ *has no stationary points in* $X \setminus \mathcal{F}$*.*

Proof. We proceed by contradiction and assume that for any integer k an $\varepsilon_k \leq 1/k$ and a point $x_k \in X \setminus \mathcal{F}$ exist such that

$$Z_\varepsilon^\circ(x_k; d) \geq 0, \qquad \forall\ d \in D(x_k). \tag{54}$$

Since, by definition $Z_\varepsilon^\circ(x; d) = \max_{\xi \in \partial Z_\varepsilon(x)} \xi^\top d$,

$$\partial Z_\varepsilon(x) \subseteq \partial f(x) + \frac{1}{\varepsilon} \sum_{i=1}^m \partial \big(\max\{0, g_i(x)\} \big)$$

and (see Proposition 2.3.12 in [9])

$$\partial(\max\{0, g_i(x)\}) \begin{cases} = \partial g_i(x), & \text{if } g_i(x) > 0, \\[2ex] \subseteq Co\{\partial g_i(x) \cup \{0\}\}, & \text{if } g_i(x) = 0, \\[2ex] = \{0\}, & \text{if } g_i(x) < 0. \end{cases}$$

Then (54) can be written as

$$\left(\xi_k^f + \frac{1}{\varepsilon_k} \sum_{i=1}^m \beta_k^i \xi_k^{g_i} \right)^\top d \geq 0, \qquad \forall\ d \in D(x_k), \tag{55}$$

$$0 \leq \beta_k^i \leq 1 \quad \text{and} \quad \beta_k^i = 0,\ \forall\ i\ :\ g_i(x_k) < 0$$

with $\xi_k^f \in \partial f(x_k)$, $\xi_k^{g_i} \in \partial g_i(x_k)$. Let us consider a limit point $\bar{x} \in X \setminus \mathcal{F}$ of $\{x_k\}$ and let us relabel the corresponding subsequence again $\{x_k\}$.
Further, by the fact that the generalized gradient of a Lipschitz continuous function is bounded on bounded sets, we get that

$$\lim_{k\to\infty} \xi_k^f = \bar{\xi}^f,$$
$$\lim_{k\to\infty} \xi_k^{g_i} = \bar{\xi}^{g_i},$$
$$\lim_{k\to\infty} \beta_k^i = \bar{\beta}^i.$$

Since ∂f and ∂g_i, $i = 1, \ldots, m$ are upper semicontinuous at $\bar{x}$ (see Proposition 2.1.5 in [9]) $\bar{\xi}^f \in \partial f(\bar{x})$, $\bar{\xi}^{g_i} \in \partial g_i(\bar{x})$, $i = 1, \ldots, m$. As $\bar{x} \notin \mathcal{F}$, then by Assumption 2 we know that a direction $\bar{d} \in D(\bar{x})$ exists such that

$$\left(\sum_{i: g_i(\bar{x}) \geq 0} \bar{\beta}^i \bar{\xi}^{g_i} \right)^\top \bar{d} < 0. \tag{56}$$

Multiplying (55) by ε_k we have

$$\left(\varepsilon_k \xi_k^f + \sum_{i=1}^m \beta_k^i \xi_k^{g_i} \right)^\top d \geq 0, \qquad \forall\ d \in D(x_k), \tag{57}$$

when k is sufficiently large, we have $\{i : g_i(x_k) \geq 0\} \supseteq \{i : g_i(\bar{x}) \geq 0\}$, then

$$\left(\varepsilon_k \xi_k^f + \sum_{i:g_i(\bar{x})\geq 0} \beta_k^i \xi_k^{g_i} \right)^\top d \geq \left(\varepsilon_k \xi_k^f + \sum_{i:g_i(x_k)\geq 0} \beta_k^i \xi_k^{g_i} \right)^\top d \geq 0, \qquad \forall\ d \in D(x_k). \tag{58}$$

Then recalling [21], $D(\bar{x}) \subseteq D(x_k)$ we can write

$$\left(\varepsilon_k \xi_k^f + \sum_{i:g_i(\bar{x})\geq 0} \beta_k^i \xi_k^{g_i} \right)^\top \bar{d} \geq 0,$$

which yields a contradiction with (56) since $\lim_{k\to\infty} \varepsilon_k = 0$. □

Now we report three further results concerning the exactness of $Z_\varepsilon(x)$ from reference [12].

Proposition B.2. *A threshold value $\varepsilon^\star > 0$ exists such that, for any $\varepsilon \in (0, \varepsilon^\star]$, every local minimum point of problem (38) is also a local minimum point of Problem (31).*

Proposition B.3. *A threshold value $\varepsilon^\star > 0$ exists such that, for any $\varepsilon \in (0, \varepsilon^\star]$, every global minimum point of problem (38) is also a global minimum point of Problem (31), and conversely.*

In order to give stationarity results for problem (38), we have the following proposition.

Proposition B.4. *For any $\varepsilon > 0$, every stationary point $\bar{x}$ of problem (38), such that $\bar{x} \in \mathcal{F}$, is also a stationary point of problem (31).*

Proof. Since $\bar{x}$ is, by assumption, a stationary point of problem (38), then reasoning as in the proof of Proposition B.1, we can write

$$\left(\xi^f + \frac{1}{\varepsilon} \sum_{i=1}^m \beta^i \xi^{g_i} \right)^\top d \geq 0, \qquad \forall\ d \in D(\bar{x}), \tag{59}$$

$$0 \leq \beta^i \leq 1 \quad \text{and} \quad \beta^i = 0, \qquad \forall\ i\ :\ g_i(\bar{x}) < 0$$

with $\xi^f \in \partial f(\bar{x})$, $\xi^{g_i} \in \partial g_i(\bar{x})$. Now, defining

$$\bar{\lambda}_i = \frac{\beta^i}{\varepsilon}, \qquad i = 1, \ldots, m$$

then, by (59), we have

$$\max_{\substack{\xi^f \in \partial f(\bar{x}) \\ \xi^{g_i} \in \partial g_i(\bar{x})}} \left\{ \left(\xi^f + \sum_{i=1}^m \bar{\lambda}_i \xi^{g_i} \right)^\top d \right\} \geq \left(\xi^f + \sum_{i=1}^m \bar{\lambda}_i \xi^{g_i} \right)^\top d \geq 0, \qquad \forall d \in D(\bar{x}). \tag{60}$$

Moreover, by definition of β^i and $\bar{\lambda}_i$ it is easy to see that condition (37) holds, so that $\bar{x}$ is a stationary point of (31). □

Finally we can prove Proposition 4.6.
Proof of Proposition 4.6. The proof easily follows by considering Propositions B.1 and B.4. □

Observe that the extension of the previous result to the case with equality constraints $h_j(x)$, $j = 1, \dots, p$, needs stronger assumptions on the differentiability of h_j [12].

References

[1] M. Abramson, C. Audet, G. Couture, J. Dennis Jr., S. Le Digabel, and C. Tribes, "The NOMAD project." Software available at url `http://www.gerad.ca/nomad`.

[2] M. Abramson, C. Audet, J. D. Jr., and S. Le Digabel, "Orthomads: A deterministic mads instance with orthogonal directions," *SIAM Journal on Optimization*, vol. 20, no. 2, pp. 948–966, 2009.

[3] N. Alexandrov and M. Hussaini, *Multidisciplinary Design Optimization - state of the art.* SIAM, Philadelphia: SIAM Proceedings Series, 1997.

[4] C. Audet, A. Custodio, and J. D. Jr., "Erratum: Mesh adaptive direct search algorithms for constrained optimization," *SIAM Journal on Optimization*, vol. 18, pp. 1501–1503, 2008.

[5] C. Audet and J. D. Jr., "Mesh adaptive direct search algorithms for constrained optimization," *SIAM Journal on Optimization*, vol. 17, no. 1, pp. 188–217, 2006.

[6] C. Audet and J. D. Jr., "A progressive barrier for derivative-free nonlinear programming," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 445–472, 2009.

[7] D. Bertsekas, *Nonlinear Programming 2nd Edition.* Belmont, MA: Athena Scientific, 1999.

[8] P. Bratley and B. Fox, "Algorithm 659: Implementing Sobol's quasirandom sequence generator," *ACM Transactions on Mathematical Software*, vol. 14, no. 1, pp. 88–100, 1988.

[9] F. Clarke, *Optimization and Nonsmooth Analysis.* New York: John Wiley & Sons, 1983.

[10] A. Conn, K. Scheinberg, and L. Vicente, *Introduction to Derivative-Free Optimization.* SIAM, Philadelphia: MPS/SIAM Ser. Optim., 2009.

[11] A. L. Custodio, J. D. Jr., and L. Vicente, "Using simplex gradients of nonsmooth functions in direct search methods," *IMA Journal of Numerical Analysis*, vol. 28, pp. 770–784, 2008.

[12] G. Di Pillo and F. Facchinei, "Exact barrier function methods for Lipschitz programs," *Applied Mathematics and Optimization*, vol. 32, pp. 1–31, 1995.

[13] R. Fletcher and S. Leyffer, "Nonlinear programming without a penalty function," *Mathematical Programming*, vol. 91, pp. 239–269, 2002.

[14] J. Halton, "On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals," *Numerische Mathematik*, vol. 2, pp. 84–90, 1960.

[15] J. Hiriart-Urruty, "On optimality conditions in nondifferentiable programming," *Mathematical Programming*, vol. 14, pp. 73–86, 1978.

[16] J. Jahn, *Introduction to the theory of nonlinear optimization.* Berlin: Springer-Verlag, 1996.

[17] N. Karmitsa, "Test problems for large-scale nonsmooth minimization," Tech. Rep. No. B. 4/2007, Department of Mathematical Information Technology, University of Jyväskylä, 2007.

[18] T. Kolda, R. Lewis, and V. Torczon, "Optimization by direct search: new perspectives on some classical and modern methods," *SIAM Review*, vol. 45, pp. 385–482, 2003.

[19] T. Kolda, R. Lewis, and V. Torczon, "Stationarity results for generating set search for linearly constrained optimization," *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 943–968, 2007.

[20] S. Le Digabel, "Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm," *ACM Transactions on Mathematical Software*, vol. 37, no. 4, pp. 1–15, 2011.

[21] C. Lin, S. Lucidi, L. Palagi, A. Risi, and M. Sciandrone, "Decomposition algorithm model for singly linearly-constrained problems subject to lower and upper bounds," *Journal of Optimization Theory and Applications*, vol. 141, pp. 107–126, 2009.

[22] G. Liuzzi and S. Lucidi, "New results on a continuously differentiable exact penalty function," *SIAM Journal on Optimization*, vol. 2, no. 4, pp. 558–574, 1992.

[23] G. Liuzzi, S. Lucidi, and F. Rinaldi, "Derivative-free methods for bound constrained mixed-integer optimization," *Computational Optimization and Applications*, vol. 53, pp. 505–526, 2012.

[24] G. Liuzzi, S. Lucidi, and M. Sciandrone, "A derivative-free algorithm for linearly constrained finite minimax problems," *SIAM Journal on Optimization*, vol. 16, no. 4, pp. 1054–1075, 2006.

[25] S. Lucidi and M. Sciandrone, "A derivative-free algorithm for bound constrained optimization," *Computational Optimization and Applications*, vol. 21, no. 2, pp. 119–142, 2002.

[26] J. More' and S. Wild, "Benchmarking derivative-free optimization algorithms," *SIAM Journal on Optimization*, vol. 20, no. 1, pp. 172–191, 2009.

[27] G. D. Pillo and S. Lucidi, "An augmented lagrangian function with improved exactness properties," *SIAM Journal on Optimization*, vol. 12, no. 2, pp. 376–406, 2001.

[28] M. Powell, "A direct search optimization method that models the objective and constraint functions by linear interpolation," in *Advances in Optimization and Numerical Analysis* (S. Gomez and J.-P. Hennart, eds.), pp. 51–67, Dordrecht: Kluwer Academic, 1994.

[29] M. Powell, "The bobyqa algorithm for bound constrained optimization without derivatives," tech. rep., Department of Applied Mathematics and Theoretical Physics, University of Cambridge, 2009.

[30] I. Sobol, "Uniformly distributed sequences with an additional uniform property," *USSR Computational Mathematics and Mathematical Physics*, vol. 16, pp. 236–242, 1977.

30.

[31] J. V. V. Lukšan, "Test problems for nonsmooth unconstrained and linearly constrained optimization," *Technical report, Institute of Computer Science, Academy of Sciences of the Czech Republic*, 2000.