



A parameter-free unconstrained reformulation for nonsmooth problems with convex constraints

Giulio Galvan¹ · Marco Sciandrone¹ · Stefano Lucidi²

Received: 1 December 2020 / Accepted: 22 June 2021 / Published online: 30 June 2021
© The Author(s) 2021

Abstract

In the present paper we propose to rewrite a nonsmooth problem subjected to convex constraints as an unconstrained problem. We show that this novel formulation shares the same global and local minima with the original constrained problem. Moreover, the reformulation can be solved with standard nonsmooth optimization methods if we are able to make projections onto the feasible sets. Numerical evidence shows that the proposed formulation compares favorably against state-of-art approaches. Code can be found at <https://github.com/jth3galv/dfppm>.

Keywords Nonsmooth constrained optimization · Derivative-free

1 Introduction

In this paper, we consider the optimization of a nonsmooth function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ over a closed convex set, namely

$$\begin{aligned} \min f(x) \\ \text{s.t. } x \in X. \end{aligned} \quad (1)$$

We assume that f is locally Lipschitz continuous and that first order information is unavailable or impractical to obtain.

The aim of the optimization, for nonsmooth problems, is to find Clarke-stationary points [6, 9].

✉ Giulio Galvan
giulio.galvan@unifi.it

Marco Sciandrone
marco.sciandrone@unifi.it

Stefano Lucidi
lucidi@diag.uniroma1.it

¹ DINFO, Università di Firenze, Via di Santa Marta 3, 50139 Firenze, Italy

² DIAG, “Sapienza” Università di Roma, Via Ariosto 25, 00185 Roma, Italy

Literature on derivative-free methods for smooth constrained optimization (i.e. when f is differentiable even though derivatives are not available) is wide. Several approaches, based on the *pattern search* methods dating back to [13], have been developed for bound and linearly constrained problems in [16] and [17] and more general type of constraints in [18]. Other works stem from research presented in [11, 21] whose line-search approach has been extended for box and linearly constrained problems in [20] and [22] while more general constraints are covered in [19]. An interesting work in the field of global optimization is [8]. We refer the reader interested in derivative-based methods, which are not considered in this work, to [7].

The nonsmooth case has seen less development. One of the two major approaches that have emerged is represented by *Mesh adaptive direct search* (MADS) that dates back to [3, 4] and that has been later modified in [1, 5]. This method combines a dense search with an extreme barrier to deal with the constraints. A second main approach, proposed in [10], is instead based on an exact penalty function.

In this case, the feasible set is expressed by a possibly nonsmooth set of inequalities $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and the original problem is replaced by the penalized version, for a given $\epsilon > 0$,

$$\min_x f(x) + \frac{1}{\epsilon} \sum_{i=1}^m \max\{0, g_i(x)\}. \quad (2)$$

It can be shown (see Proposition 3.6 in [10]) that, under suitable assumptions, a value ϵ^* exists such that $\forall \epsilon \in (0, \epsilon^*)$ every Clarke-stationary point \bar{x} of (2) is also a stationary point of the original problem. Thus, any algorithm for nonsmooth unconstrained optimization can be applied. In [10], however, a linesearch based algorithm that employs a dense set of directions alongside the $2n$ coordinate directions (CS-DFN) is proposed. This latter reformulation combined with CS-DFN is shown to be favorably comparable against state-of-art MADS based software like NOMAD [15].

The value of ϵ^* is, however, in general unknown and choosing a proper value of ϵ can be a difficult task. Setting a wrong value of ϵ can be extremely harmful to the performance of the algorithm. For example, if f is unbounded outside the feasible set setting too high a value of ϵ can drive the algorithm towards minus infinity. On the other hand, too small a value ($< \epsilon^*$), even if theoretical convergence is assured, can yield extremely poor performances because the algorithms may be forced to take really small steps near the boundary of the feasible set.

In this work we propose a novel way of treating convex constraints that is not based on penalty functions. We assume that the feasible set X is a closed convex set and that a projection operator onto the feasible set is available. We do not require X to have an analytical expression nor make any other regularity assumptions. However, we make the assumption that the computational effort needed to compute the projection is negligible compared to the evaluation of the objective function. Indeed, the only computational cost we consider is the number of evaluations of the objective function.

The paper is organized as follows. In Sect. 2 we recall some necessary definitions and known results before introducing the proposed reformulation in Sect. 3. Then, in Sect. 4, we prove the equivalence between the novel formulation and the

original problem. Some numerical results are given in Sect. 5. In particular, we propose a comparison between the exact penalty approach of [10] and the proposed reformulation. We make use of the CS-DFN algorithm, used in [10], for both formulations to make a fair comparison. Finally, we give some concluding remarks in Sect. 6.

2 Preliminary background

We recall that optimality conditions for nonsmooth problems can be given in terms of the Clarke generalized directional derivative [9]. In particular, in the unconstrained case, we have that

Definition 1 (Clarke stationarity–unconstrained case) A point $\bar{x} \in X$ is Clarke stationary w.r.t. problem $\min_x f(x)$ if

$$f^\circ(\bar{x};d) \geq 0 \quad \forall d \in \mathbb{R}^n,$$

where

$$f^\circ(x;d) = \limsup_{y \rightarrow x, t \rightarrow 0} \frac{f(y + td) - f(y)}{t} \geq 0$$

is the Clarke generalized directional derivative.

We follow [6] for the treatment of Clarke stationarity for constrained problems. First, we define the cone of the *hyper-tangent* directions.

Definition 2 (Hyper-Tangent Cone) A vector $d \in \mathbb{R}^n$ is said to be a *hyper-tangent* vector to the set X at $\bar{x} \in X$ if there exists $\epsilon > 0$ such that

$$y + tw \in X \quad \forall y \in X \cap B(x, \epsilon), \quad w \in B(d, \epsilon), \quad t \in (0, \epsilon).$$

The set of all hyper-tangent vector is called the hyper-tangent cone to X at \bar{x} and is denoted $T_X^H(\bar{x})$. For a more detailed treatment we refer the reader to [6]. Figure 6.5 of [6] offers a graphical illustration of the hyper-tangent cone.

Then we can give a definition of Clarke-stationary points as expressed by the following.

Definition 3 (Clarke stationarity–constrained case) A point $\bar{x} \in X$ is Clarke stationary w.r.t. problem $\min_{x \in X} f(x)$ if

$$f^\circ(\bar{x};d) \geq 0 \quad \forall d \in T_X^H(\bar{x}),$$

where

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

We will make use of the following result which relates to the Clarke-derivative and classical directional derivative in the case of convex functions. For the proof we refer the reader to Theorem 3.42 in [14].

Theorem 1 *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a convex functional which is Lipschitz continuous at some $\bar{x} \in X$. Then the Clarke derivative f at \bar{x} coincides with the directional derivative of f at x that is*

$$f^\circ(\bar{x};d) = \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{f(y+td) - f(y)}{t} = \lim_{t \rightarrow 0} \frac{f(\bar{x}+td) - f(\bar{x})}{t} = f'(\bar{x};d).$$

3 A novel formulation

Generalizing a bit, all the approaches that have been proposed in the literature to deal with general constraints, try to steer the search towards the feasible set by adding (maybe in a sequential manner) to the objective function some kind of penalty ϕ , which, in its most general form, can be described by

$$\phi(x) = \begin{cases} 0 & \text{if } x \in X \\ > 0 & \text{otherwise.} \end{cases}$$

Such function can be a smooth quadratic or an exact nonsmooth penalty or, also, a hard barrier that takes $+\infty$ outside the feasible set. The problem is thus rewritten as

$$\min_x P(x;\epsilon) = f(x) + \epsilon\phi(x),$$

where $\epsilon > 0$ is a parameter that must be set.

Consider a local minimum of the original problem x^* . We have that, for some neighborhood $B(x^*, \delta)$,

$$f(x^*) \leq f(x) \quad \forall x \in B(x^*, \delta) \cap X.$$

The strategy of penalty-based approaches is making the penalty large enough (either by making ϵ large or by using a hard barrier) so that

$$f(x) + p(x) \geq f(x^*) + p(x^*) = f(x^*), \quad \forall x \in B(x^*, \delta)$$

and, hence, x^* is also a local minimum of the penalized problem.

The idea behind the proposed reformulation is, instead, to avoid penalties by “assigning” to a point x outside the feasible set the value of the objective function computed at its projection $\pi(x)$. In this way, we do not ever compute f outside the

feasible set and we do not need to “correct” f with a penalty for points outside the feasible set.

Let π be the projector operator over X defined as

$$\pi(x) \in \operatorname{argmin}_{z \in X} \|z - x\|.$$

Notice that since X is compact and convex the projection has a unique solution. A proof of the uniqueness of the projection for convex sets alongside other properties of the projection can be found in Proposition 2.1.3 of [7]. One property that will be extensively used in the following is the *non expansiveness* of the projection i.e. $|\pi(x) - \pi(y)| \leq |x - y| \forall x, y \in X$. Dealing with non convex sets would require a more complex treatment since the projection may not be unique. We leave the study of a possible extension to future work.

We can thus define the problem

$$\min_x f(\pi(x))$$

where each point outside the feasible set assumes the value of its projection. In the latter formulation it is guaranteed that no point outside the feasible set can take a value lower than some point in X . We have however that all the points that share the same projection (consider a ray perpendicular to the constraints) share the same function value. To overcome this issue is sufficient to add to the previous formulation a term that penalizes the distance of a point from its projection. We thus propose to replace the original problem by

$$\min_x \tilde{f}(x) = f(\pi(x)) + d_X(x), \tag{3}$$

where $d_X(x) = \|x - \pi(x)\|$ is the distance from x to the feasible set X .

We note also that since the projection operator is continuous we have that \tilde{f} is continuous. Moreover, if f is bounded from below on the feasible set X then \tilde{f} is bounded on \mathbb{R}^n since $\tilde{f}(x) \geq f(\pi(x)) \geq \inf_{x \in X} f(x)$. On the contrary in the penalty approach $f(x) + \frac{1}{\epsilon} \sum_{i=1}^m \max\{0, g_i(x)\}$ can be unbounded.

Consider, as a simple example, the problem HS224 from the test suite [26].

$$\begin{aligned} \min_x & 2x_1^2 + x_2^2 - 48x_1 - 40x_2 \\ \text{s.t.} & x_1 + 3x_2 \geq 0 \\ & 18 - x_1 - 3x_2 \geq 0 \\ & x_1 + x_2 \geq 0 \\ & 8 - x_1 - x_2 \geq 0 \\ & 0 \leq x \leq 6. \end{aligned}$$

The level curves of the original objective function f and of the modified problem \tilde{f} are shown in Fig. 1.

Notice how the solution to the problem $x^* = (4, 4)$ becomes an unconstrained global minimum in the proposed formulation.

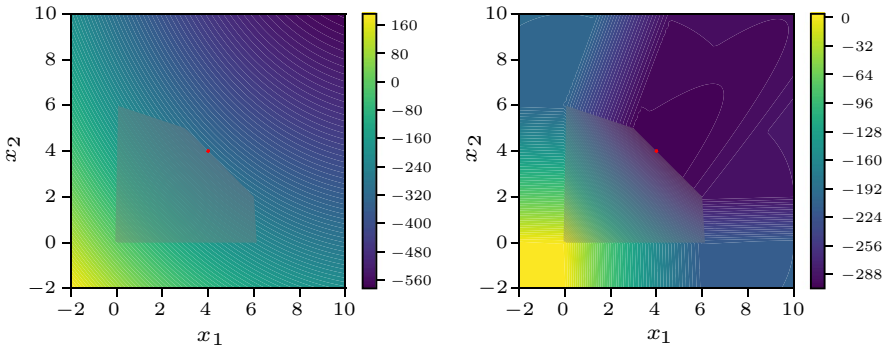


Fig. 1 Level curves of f (left) and \tilde{f} (right) for the two-variables problem HS224. The color bar indicates the objective function values. The feasible set is represented by the area shadowed in gray. The solution is indicated by the red dot

4 Equivalence of the formulations

In this section, we prove the equivalence between the original constrained problem (1) and the proposed formulation (3) in terms of both local/global minima and stationary points. We also show that by modifying the objective function we do not lose Lipschitz continuity so that if f is (locally) Lipschitz \tilde{f} is (locally) Lipschitz too. We start with the latter.

Lemma 1 *Let f be locally Lipschitz continuous. Then the modified function $\tilde{f} = f(\pi(x)) + \|x - \pi(x)\|$ is also locally Lipschitz.*

Proof Let $x_0 \in \mathbb{R}^n$. Since f is locally Lipschitz there exists L_0 and δ_0 so that

$$|f(x) - f(x_0)| \leq L_0 \|x - x_0\| \quad \forall x \in B(x_0, \delta_0).$$

Now consider \tilde{f} . For every $x \in B(x_0, \delta_0)$ we have

$$\begin{aligned} |\tilde{f}(x) - \tilde{f}(x_0)| &= |f(\pi(x)) + \|x - \pi(x)\| - f(\pi(x_0)) - \|x_0 - \pi(x_0)\|| \\ &\leq |f(\pi(x)) - f(\pi(x_0))| + |\|x - \pi(x)\| - \|x_0 - \pi(x_0)\|| \\ &\leq L_0 \|\pi(x) - \pi(x_0)\| + |\|x - \pi(x)\| - \|x_0 - \pi(x_0)\|| \\ &\leq L_0 \|x - x_0\| + |\|x - \pi(x)\| - \|x_0 - \pi(x_0)\||, \end{aligned}$$

where we have used the local Lipschitz continuity of f and the non expansiveness property of the projection operation. Now, by the triangular inequality we have $\|x_0 - \pi(x_0)\| \leq \|x_0 - x\| + \|x - \pi(x)\| + \|\pi(x) - \pi(x_0)\|$ so that

$$\begin{aligned} \|x_0 - \pi(x_0)\| - \|x - \pi(x)\| &\leq \|x_0 - x\| + \|\pi(x) - \pi(x_0)\| \\ &\leq 2\|x_0 - x\|. \end{aligned}$$

The same reasoning applies to the opposite sign $-\|x_0 - \pi(x_0)\| + \|x - \pi(x)\|$ so that $\| \|x - \pi(x)\| - \|x - \pi(x_0)\| \| \leq 2\|x_0 - x\|$ and we can conclude that

$$|\tilde{f}(x) - \tilde{f}(x_0)| \leq (2 + L_0)\|x - x_0\| \quad \forall x \in B(x_0, \delta_0).$$

□

We now consider the relationship between the global and local minimum of the two formulations. We first prove that each global (local) minimum of the original problem is also a global (local) minimum of the proposed formulation in Proposition 1 and 2.

Proposition 1 *Every global minimum of problem (1) is also a global minimum for problem (3).*

Proof Let $x^* \in X$ be a global minimum for problem (1). Suppose by contradiction that there exists $\bar{x} \in \mathbb{R}^n$ such that $\tilde{f}(\bar{x}) < \tilde{f}(x^*)$. Then

$$\tilde{f}(\bar{x}) = f(\pi(\bar{x})) + \|\bar{x} - \pi(\bar{x})\| < \tilde{f}(x^*) = f(\pi(x^*)) + \|x^* - \pi(x^*)\| = f(x^*).$$

Thus, we have found a point $y = \pi(\bar{x}) \in X$ s.t.

$$f(y) < f(x^*) - \|\bar{x} - y\| < f(x^*),$$

which is a contradiction. □

Proposition 2 *Every local minimum of problem (1) is also a local minimum for problem (3).*

Proof Let $x^* \in X$ be a local minimum for problem (1). Then there exists a ball $B(x^*, \rho)$ with $\rho > 0$ s.t.

$$f(x^*) \leq f(x) \quad \forall x \in B(x^*, \rho) \cap X.$$

Let $\bar{x} \in B(x^*, \rho)$ and suppose by contradiction that $\tilde{f}(\bar{x}) < \tilde{f}(x^*)$. Thus we have

$$\tilde{f}(\bar{x}) = f(\pi(\bar{x})) + \|\bar{x} - \pi(\bar{x})\| < \tilde{f}(x^*) = f(\pi(x^*)) + \|x^* - \pi(x^*)\| = f(x^*).$$

Let $y = \pi(\bar{x}) \in X$. We have, by the properties of the projection operator that

$$\|y - \pi(x^*)\| = \|y - x^*\| \leq \|\bar{x} - x^*\| \leq \rho,$$

so that $y \in B(x^*, \rho) \cap X$. Moreover, it holds that

$$f(y) < f(x^*) - \|\bar{x} - y\| < f(x^*),$$

which is a contradiction. □

Furthermore, since f and \tilde{f} take the same values on X it holds that any global (local) minimum of the modified problem which belongs to the feasible set is also a global (local) minimum of the original problem.

Proposition 3 *Every global (local) minimum $x \in X$ for problem (3) is also a global (local) minimum of problem (1).*

We now show, in Lemma 2, that no minimal point does exist outside the feasible region X so that we have a perfect equivalence between global and local minima in the two formulations as remarked in Corollary 1

Lemma 2 *Suppose $\hat{x} \in \mathbb{R}^n \setminus X$ then \hat{x} is not a global or local minimum for problem (3). In particular, $d = \pi(\hat{x}) - \hat{x}$ is a descent direction for \tilde{f} at \hat{x} .*

Proof We will prove the thesis by showing that there exists a descent direction at \hat{x} and hence \hat{x} cannot be a minimum.

Consider the projection of \hat{x} onto the feasible set $\bar{x} = \pi(\hat{x})$. Let $d = \bar{x} - \hat{x}$ and consider a point $\hat{x} + \alpha d$ with $\alpha \in (0, 1]$. For every y we have

$$\begin{aligned} (\hat{x} + \alpha d - \bar{x})^T (y - \bar{x}) &= (\hat{x} + \alpha(\bar{x} - \hat{x}) - \bar{x})^T (y - \bar{x}) \\ &= (1 - \alpha)(\hat{x} - \bar{x})(y - \bar{x}) \leq 0, \end{aligned} \quad (4)$$

where the latter holds because $\bar{x} = \pi(\hat{x})$. Since the projection has a unique solution, from (4) we conclude that $\bar{x} = \pi(\hat{x}) = \pi(\hat{x} + \alpha d) \forall \alpha \in [0, 1]$. Thus we have that

$$\begin{aligned} \tilde{f}(\hat{x}) - \tilde{f}(\hat{x} + \alpha d) &= f(\bar{x}) + \|\hat{x} - \bar{x}\| - f(\pi(\hat{x} + \alpha d)) - \|\hat{x} + \alpha d - \pi(\hat{x} + \alpha d)\| \\ &= f(\bar{x}) + \|\hat{x} - \bar{x}\| - f(\bar{x}) - \|\hat{x} + \alpha(\bar{x} - \hat{x}) - \bar{x}\| \\ &= \|\hat{x} - \bar{x}\| - (1 - \alpha)\|\hat{x} - \bar{x}\| \\ &= \alpha\|\hat{x} - \bar{x}\| \\ &> 0, \end{aligned}$$

so that d is a descent direction at \hat{x} . □

By putting together Proposition 1, 2, 3 and Lemma 2 we establish the perfect equivalence of the two formulations in terms of local and global minima as expressed by the following.

Corollary 1 *Every global (local) minimum of problem (1) is also a global (local) minimum of problem (3) and reciprocally.*

To conclude the discussion we investigate on the relationship between stationary points. In particular, it is interesting to check if Clarke-stationary points of the modified problem are also Clarke-stationary for the original problem.

We are able to prove the latter under the following assumption.

Assumption 1 We assume that X is such that

$$\limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \left\| \frac{\pi(y + td) - \pi(y)}{t} - d \right\| = 0,$$

for every hyper-tangent direction $T_X^H(\bar{x})$ and every feasible point $\bar{x} \in X$.

We start by showing that any Clarke-stationary point of the modified problem must belong the feasible set.

Lemma 3 *Let \bar{x} be a Clarke-stationary point of problem (3). Then $\bar{x} \in X$.*

Proof Since \bar{x} is Clarke-stationary we have, by definition, that

$$\tilde{f}^\circ(\bar{x}; d) = \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{\tilde{f}(y + td) - \tilde{f}(y)}{t} \geq 0, \quad \forall d \in \mathbb{R}^n. \tag{5}$$

In particular the latter must hold also for direction $d = \pi(\bar{x}) - \bar{x}$.

Now, let us suppose by contradiction that $\bar{x} \notin X$. Letting $\hat{d} = \pi(y) - y$ we can write

$$\begin{aligned} \frac{\tilde{f}(y + td) - \tilde{f}(y)}{t} &= \frac{\tilde{f}(y + t\hat{d}) - \tilde{f}(y)}{t} + \frac{\tilde{f}(y + td) - \tilde{f}(y + t\hat{d})}{t} \\ &\leq -\|\pi(y) - y\| + \tilde{L} \|d - \hat{d}\|, \end{aligned}$$

where we have used, that \hat{d} is a descent direction for \tilde{f} at y (Lemma 2) for the first term and that \tilde{f} is Lipschitz continuous (Lemma 1) for the second one.

Now for every $y \rightarrow \bar{x}$ we have that $\|d - \hat{d}\| = \|\pi(\bar{x}) - \bar{x} - \pi(y) + y\| \rightarrow 0$ and that $\|\pi(y) - y\| \rightarrow \|\pi(\bar{x}) - \bar{x}\| = M > 0$ since $\bar{x} \notin X$. Thus we have that

$$\frac{\tilde{f}(y + td) - \tilde{f}(y)}{t} \leq -M < 0,$$

which contradicts (5). □

Proposition 4 *Let \bar{x} be a Clarke-stationary point of problem (3). Then, under Assumption 1, \bar{x} is also a Clarke-stationary point of problem (1).*

Proof Let \bar{x} be a Clarke-stationary point for problem (3). By Lemma 3 it must be that $\bar{x} \in X$. Then from Definition 3 we have

$$\tilde{f}^\circ(\bar{x}; d) \geq 0,$$

for every hyper-tangent direction $d \in T_X^H(\bar{x})$.

We can calculate

$$\begin{aligned}
0 \leq \tilde{f}^\circ(\bar{x}; d) &= \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{\tilde{f}(y + td) - \tilde{f}(y)}{t} \\
&\leq \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{f(\pi(y + td)) - f(\pi(y))}{t} + \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{d_X(y + td) - d_X(y)}{t}
\end{aligned}$$

Since d_X is a convex function we have, by Theorem 1, that $d_X^\circ = d'_X$. Thus

$$\limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{d_X(y + td) - d_X(y)}{t} = \lim_{t \rightarrow 0} \frac{d_X(\bar{x} + td) - d_X(\bar{x})}{t} = 0.$$

Hence

$$\begin{aligned}
0 &\leq \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{f(\pi(y + td)) - f(\pi(y))}{t} \\
&\leq \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{f(\pi(y) + td) - f(\pi(y))}{t} + \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{f(\pi(y + td)) - f(\pi(y) + td)}{t} \\
&\leq \limsup_{y \rightarrow \bar{x}, t \rightarrow 0, y \in X} \frac{f(y + td) - f(y)}{t} + L \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \frac{\|\pi(y + td) - \pi(y) - td\|}{t} \\
&\leq f^\circ(\bar{x}; d) + L \limsup_{y \rightarrow \bar{x}, t \rightarrow 0} \left\| \frac{\pi(y + td) - \pi(y)}{t} - d \right\|.
\end{aligned}$$

Thus, because of Assumption 1, we conclude that

$$f^\circ(\bar{x}; d) \geq 0.$$

□

5 Numerical experiments

In the following, we propose some numerical experiments to investigate advantages of the proposed formulation comparing it against the exact penalty approach of [10]. To make a fair comparison we used the same algorithm to solve both formulations. In particular, we used the CS-DFN algorithm proposed in [10]. In the following, we call *solver* an algorithm applied to a particular formulation of a given problem. So we compare the *exact penalty* solver, i.e. the CS-DFN algorithm applied to the penalized formulation, and the *Projection-based Penalty Method* (PPM) solver, i.e. the CS-DFN algorithm applied to the proposed formulation.

Test problems We set up a benchmark composed of 28 problems belonging to different classes: general nonlinear functions subjected to (1) non-degenerate linear constraints from the collection [12, 26]; (2) degenerate linear constraints from [2]; (3) general convex constraints again from [12, 26]; and minmax programs under linear constraints from [23]. The problems are listed in Table 1.

Performance metric To compare the results we employ *data profiles*. Data profile for benchmarking derivative free algorithms have been proposed in [24]. They take into

Table 1 Benchmark problems details

| Problem | n | m | Type | |
|--------------|----|----|----------------------------------------------------|-------------------------------------|
| HS24 | 2 | 5 | Smooth with linear constraints from [12, 26] | |
| HS36 | 3 | 7 | | |
| HS37 | 3 | 8 | | |
| HS44 | 4 | 10 | | |
| HS86 | 5 | 15 | | |
| HS224 | 2 | 5 | | |
| HS231 | 2 | 2 | | |
| HS232 | 2 | 5 | | |
| HS250 | 3 | 8 | | |
| HS331 | 2 | 4 | | |
| AS6($n=6$) | 6 | 12 | Smooth with linear degenerate constraints from [2] | |
| AS6($n=7$) | 7 | 14 | | |
| AS6($n=8$) | 8 | 16 | | |
| AS7($n=6$) | 6 | 12 | | |
| AS7($n=7$) | 7 | 14 | | |
| AS7($n=8$) | 8 | 16 | Smooth with convex non linear from [12, 26] | |
| HS22 | 2 | 2 | | |
| HS29 | 3 | 1 | | |
| HS43 | 4 | 3 | | |
| HS65 | 3 | 7 | | |
| HS66 | 3 | 8 | | |
| HS270 | 5 | 5 | | |
| MAD1 | 2 | 1 | | Minmax with linear constraints [23] |
| MAD2 | 2 | 1 | | |
| MAD4 | 2 | 1 | | |
| MAD5 | 2 | 1 | | |
| PENTAGON | 6 | 15 | | |
| WONG2 | 10 | 3 | | |

account the case of unconstrained problems. Consider a test set of \mathcal{P} problems. We fix a tolerance parameter τ and we say that a problem has been solved by if

$$f(x) - f_L \leq \tau(f(x_0) - f_L), \quad (6)$$

where f_L is an accurate estimate of the minimum of the problem (in our benchmark we can set $f_L = f^*$ since f^* is available). Note that, here, function values are referred to the original problem even if the solver employs a different formulation.

Then for each solver s we define its data profile as

$$d_s(\alpha) = \frac{1}{|\mathcal{P}|} \left| \left\{ p \in \mathcal{P} \text{ s.t. } \frac{nf(\tau)}{n_p + 1} \leq \alpha \right\} \right|,$$

where n_p is the number of variables of problem p and $nf(\tau)$ is the number of function evaluations needed to satisfy the convergence criterion (6).

Data profiles are extracted for different values of τ to compare the solvers against different balances of speed versus accuracy.

In the constrained case, however, the proposed scheme is not readily applicable. We propose to modify condition (6) by considering also the constraint violation as follows:

$$(1 - \beta)(f(x) - f_L) + \beta \|g_+(x)\| \leq \tau(1 - \beta)(f(x_0) - f_L) + \tau\beta \|g_+(x_0)\|, \quad (7)$$

where $\beta \in (0, 1)$ is a new parameter which balances function value and constraint violation.

Note that, by violating the constraints, the function values can be lower than the f^* . Naturally, by choosing a high value for β this situation can be arbitrarily penalized as long as f does not go to $-\infty$. These cases must be removed before computing the profiles.

We extract different data profiles for different values of both τ and β . In particular we extract the curves for τ^{-k} with $k \in \{1, 3, 5, 7\}$ and $\beta \in \{0.9, 0.99\}$

Solvers details As already mentioned for both solvers we employ the CS-DFN proposed in [10]. We report the pseudo-code of the method in Algorithm 1.

```

1 def linesearch( $x, d, \tilde{\alpha}, \gamma, \delta$ ):
2   if  $f(x + \tilde{\alpha}d) > f(x) - \gamma\tilde{\alpha}^2$  then
3     return 0
4   else
5      $\alpha = \tilde{\alpha}$ 
6     while  $\frac{\alpha}{\delta} \leq \alpha_{max}$  and  $f(x + \frac{\alpha}{\delta}d) \leq f(x) - \gamma(\frac{\alpha}{\delta})^2$  do
7        $\alpha = \frac{\alpha}{\delta}$ 
8     return  $\alpha$ 

```

Data: $x_0 \in \mathbb{R}^n$, $\theta \in (0, 1)$, $\delta > 0$, $\gamma > 0$, $\eta > 0$,
 $\tilde{\alpha}_0^i > 0$, $d_0^i = e^i$ for $i = 1, \dots, n$.
 $\tilde{\alpha}_0$, a sequence $\{d_k\}$ of search directions s.t. $\|d_k\| = 1 \forall k$

```

1 for  $k = 0, 1, \dots$  do
2    $z_{k+1}^0 = x_k$ 
3   for  $i = 1, \dots, n$  do
4      $\alpha = \text{linesearch}(z_{k+1}^{i-1}, d_k^i, \tilde{\alpha}_k^i, \gamma, \delta)$ 
5     if  $\alpha > 0$  then
6        $\alpha_k^i = \alpha$ ,  $\tilde{\alpha}_{k+1}^i = \alpha_k^i$ ,  $d_{k+1}^i = d_k^i$ 
7     else
8        $\alpha = \text{linesearch}(z_{k+1}^{i-1}, -d_k^i, \tilde{\alpha}_k^i, \gamma, \delta)$ 
9       if  $\alpha > 0$  then
10         $\alpha_k^i = \alpha$ ,  $\tilde{\alpha}_{k+1}^i = \alpha_k^i$ ,  $d_{k+1}^i = -d_k^i$ 
11      else
12         $\alpha_k^i = 0$ ,  $\tilde{\alpha}_{k+1}^i = \theta\tilde{\alpha}_k^i$ 
13       $z_{k+1}^i = z_{k+1}^{i-1} + \alpha_k^i d_k^i$ 
14      if  $\max_{i=1, \dots, n} \{\alpha_{k+1}^i, \tilde{\alpha}_{k+1}^i\} \leq \eta$  then
15         $\alpha = \text{linesearch}(z_{k+1}^n, d_k, \tilde{\alpha}_k, \gamma, \delta)$ 
16        if  $\alpha > 0$  then
17           $\tilde{\alpha}_{k+1} = \alpha_k$ 
18        else
19           $\alpha_k = 0$ ,  $\tilde{\alpha}_{k+1} = \theta\tilde{\alpha}_k$ 
20         $z_{k+1}^{n+1} = z_k^n + \alpha_k d_k$ 
21      find  $x_{k+1}$  s.t.  $f(x_{k+1}) \leq f(z_{k+1}^{n+1})$ 

```

Algorithm 1: CS-DFN

We set $\delta = 0.5$, $\gamma = 10^{-6}$, $\eta = 10^{-6}$ for Algorithm 1. The dense sequence of direction $\{d_k\}$ was obtained by the implementation available at [25] of the Sobol quasi-random generation proposed in [27].

For the *exact penalty* solver we employ the adaptive strategy for tuning ϵ which is proposed in [10] and is deemed to be a better choice.

The implementation of the algorithms, alongside the code needed to reproduce all the following experiments is available as *python* code at <https://github.com/jth3g/alv/dfppm>.

5.1 A first comparison

We start by comparing the PPM solver against the *exact penalty*. We let both solvers run for up to 10^4 function evaluations and then we extract the data profiles, which are reported in Fig. 2. From the plots we can see that the PPM solver enjoys generally better performance both in terms of speed and robustness (number of problems eventually solved). We note however that neither solver manages to solve more than the 65% of the test problems within the budget of function evaluations when a relatively high precision ($\tau = 10^{-7}$) is required.

5.2 A parametrization

In this section, we introduce a scale factor ϵ that controls how much to penalize points outside the feasible region. Namely we modify our formulation as

$$\min_x f(\pi(x)) + \epsilon \|x - \pi(x)\|.$$

To understand the effect of ϵ we start with a qualitative analysis. In Fig. 3 we show the iterates of the algorithm when run on the same problem with different values of ϵ .

From Fig. 3 we can see that the iterates for greater values of ϵ stay closer to the feasible set while for small values of ϵ the algorithm is allowed to stay far from it.

To understand whether staying closer to the feasible set has a good or bad effect on the overall optimization process we measure the performance on the solver for different values of ϵ . Namely we try $\epsilon \in \{0.1, 1, 10, 100\}$. We use the same setup of the previous experiments. In Fig. 4 we give the data profiles for the different solvers (we include, for later reference, another configuration ($\epsilon = 10$, $\sigma = 2$) which is explained later in the manuscript).

From Fig. 4 we can see that choosing a high value of ϵ yields good performances when low precision is required although they quickly lessen for higher values of τ . For instance for $\tau = 10^{-7}$, $\beta = 0.99$ the algorithm with $\epsilon = 100$ manages to satisfy the convergence criterion only in roughly 40% of the test problems. The opposite is true for small values of ϵ . The algorithm is generally slower but can achieve very good solutions if a greater number of function evaluations is allowed.

It is, thus, natural to ask if employing an adaptive strategy for ϵ may be advantageous. For example, one could start with ϵ set to a large number and gradually decrease it to get to accurate solutions.

Coming up with a good schedule for ϵ that works well for all problems can be a hard task. However, the CS-DFN algorithm offers a good way to understand in what regime the algorithm is working by looking at the length of the steps that the algorithm takes at each iteration. We thus propose to set the value of ϵ as a function of the set length α_k . More precisely we set

$$\epsilon_{k+1} = \sigma \alpha_k. \quad (8)$$

In this way, at the beginning of the algorithm we can start with a large value of ϵ_0 and then let it decreasing as the algorithm takes smaller steps.

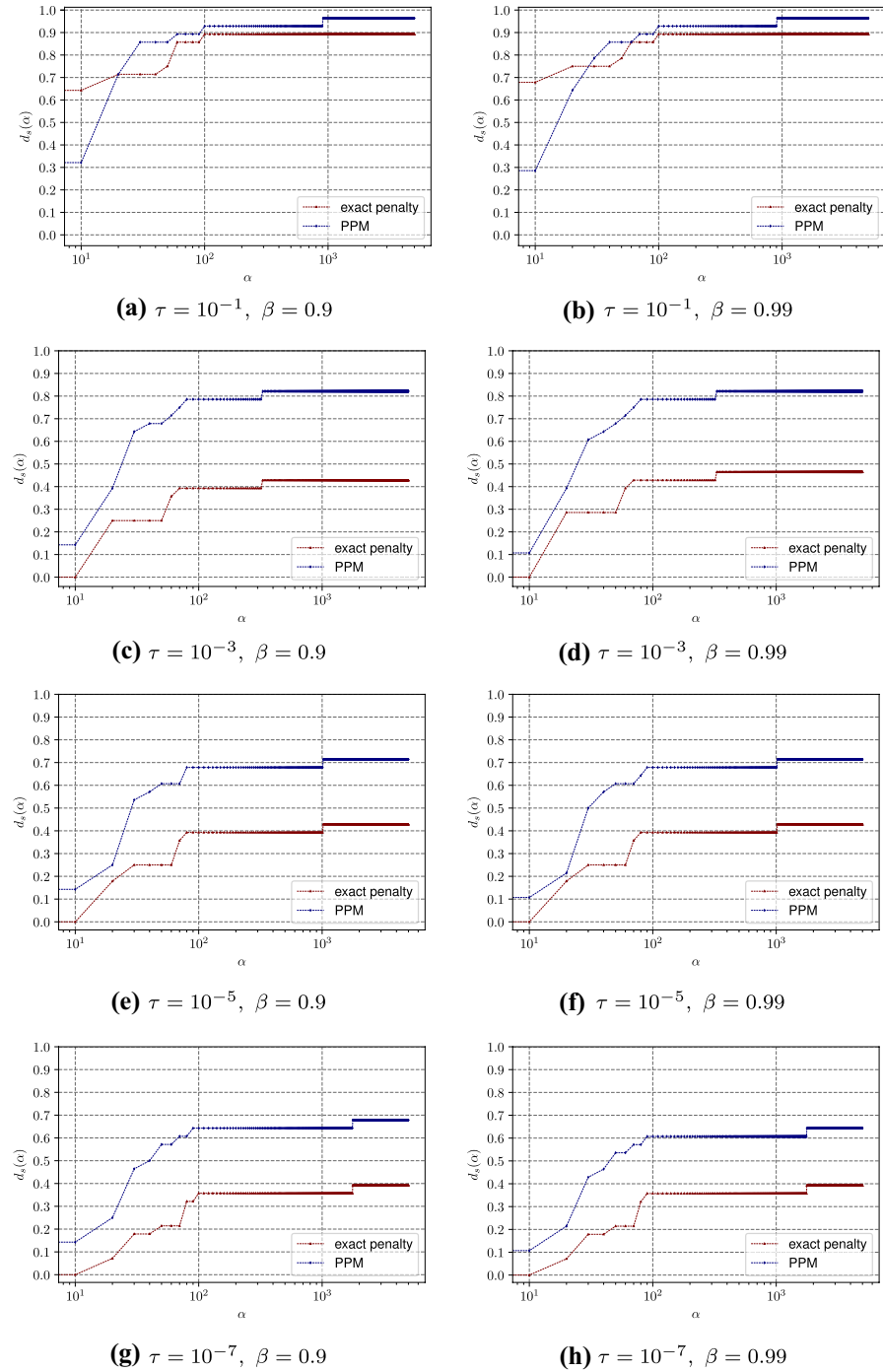


Fig. 2 Data profiles for the PPM solver and the *exact penalty*

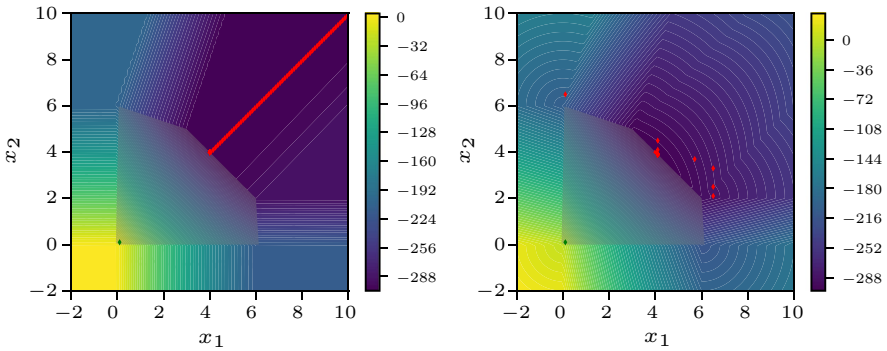


Fig. 3 Iterates (in red) of CS-DFN for $\epsilon = 0.1$ (left) and $\epsilon = 10$ (right) for problem HS224. The feasible set is in gray, the initial point in green. Each dot is obtained after a search along the $2n$ coordinate directions and possibly a direction from the dense sequence

We found, by manually tuning, that good performances can be obtained by setting $\epsilon_0 = 10$, $\sigma = 2$ although other configurations perform similarly well. As we can see, again in Fig. 4, this configuration performs almost equally well when low or high precision is required. Moreover, notice that when high precision is required we go from less than 70% of solved problems to more than 90%.

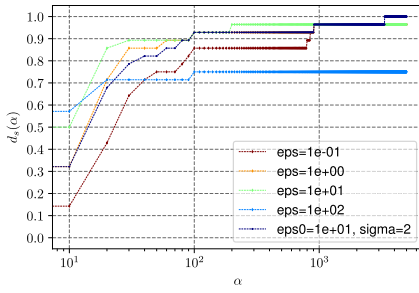
5.3 Final comparison

To end the discussion we propose a final comparison of the PPM solver, in its parameterized version, equipped with the adaptive strategy for tuning ϵ against the *exact penalty* approach. The results are reported in Fig. 5.

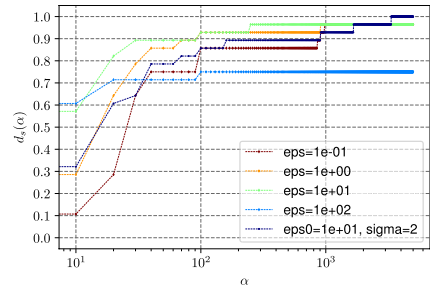
We can see that the PPM solver enjoys better performance for every threshold of accuracy although the *exact penalty* can be faster for some problems when a relatively low precision is required. We note, furthermore, that the exact penalty fails to reach accurate solutions for a large portion of the test problems whether, as already noticed, the PPM solver manages to reach more than the 90% of solved problems even when high precision is required. We also report, for completeness, in Table 2 the distance of the objective function from the optimum value and the constraint violation after the total budget of function evaluations has been used.

6 Conclusion

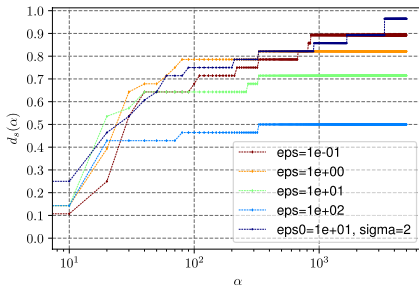
In this work we proposed to rewrite a nonsmooth optimization problem subjected to convex constraints as an unconstrained parameter-free problem. Such formulation is proven to be equivalent to the original problem, in terms of global and local minima. Furthermore we were able to prove, under suitable assumptions, that any Clarke-stationary point of the proposed formulation is also a Clarke-stationary point of the original problem. The formulation can be solved by any optimization algorithm for nonsmooth optimization.



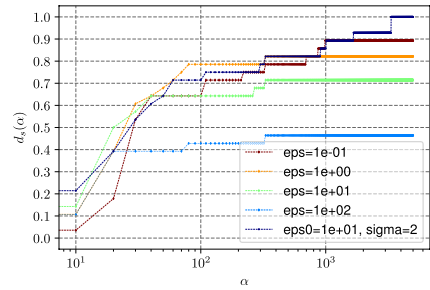
(a) $\tau = 10^{-1}, \beta = 0.9$



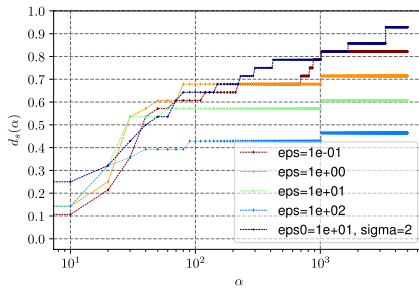
(b) $\tau = 10^{-1}, \beta = 0.99$



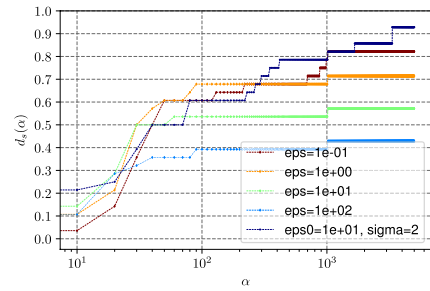
(c) $\tau = 10^{-3}, \beta = 0.9$



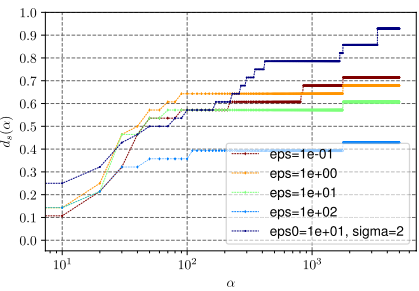
(d) $\tau = 10^{-3}, \beta = 0.99$



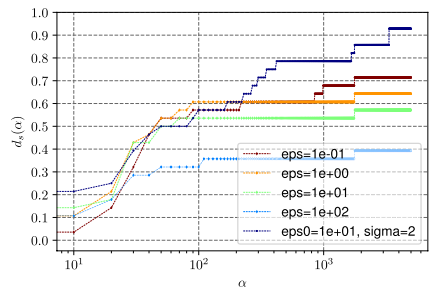
(e) $\tau = 10^{-5}, \beta = 0.9$



(f) $\tau = 10^{-5}, \beta = 0.99$



(g) $\tau = 10^{-7}, \beta = 0.9$



(h) $\tau = 10^{-7}, \beta = 0.99$

Fig. 4 Data profiles for different value of the parameter ϵ .

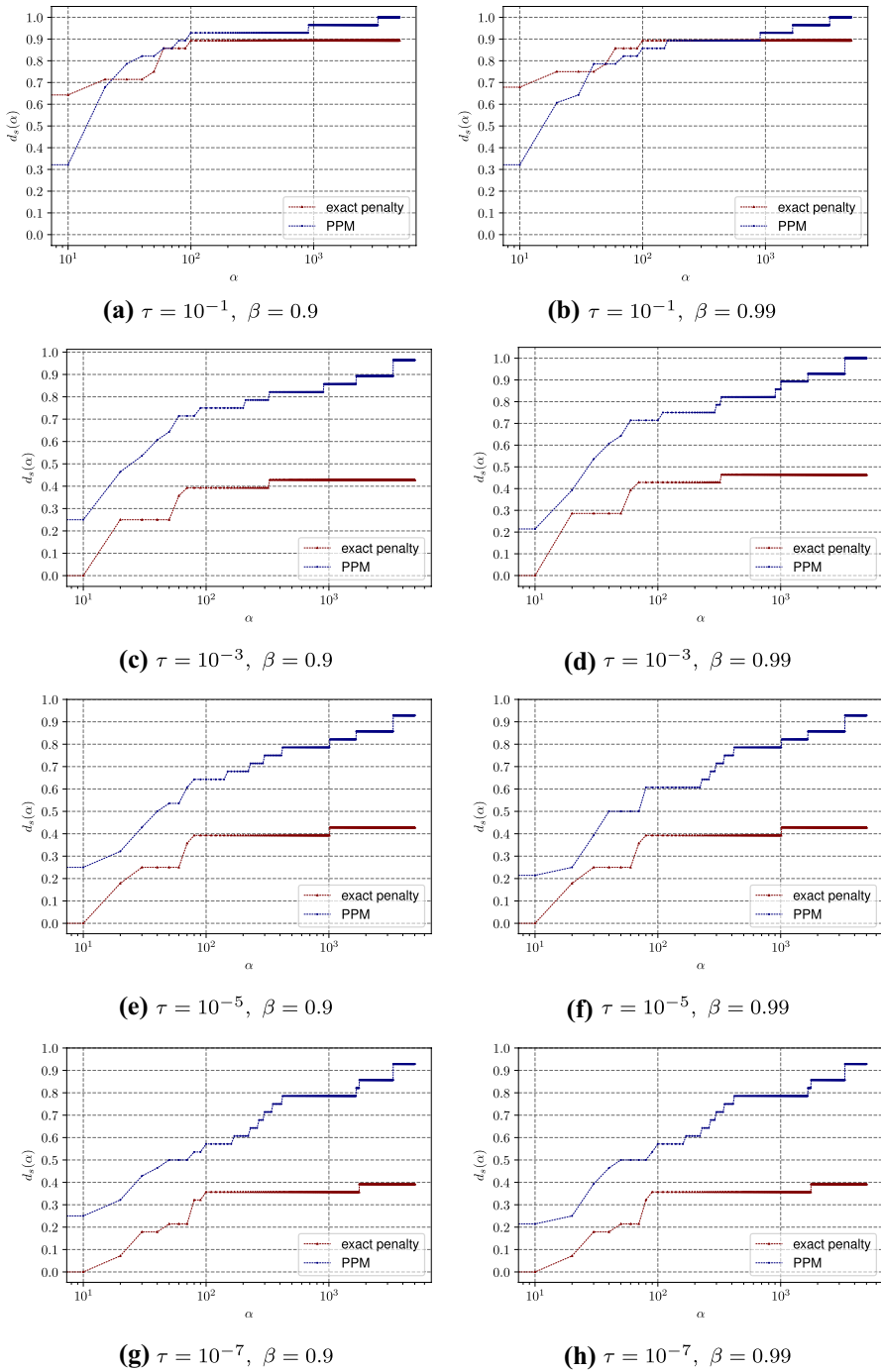


Fig. 5 Data profiles for the PPM and exact penalty solvers

Table 2 Details for the PPM and exact penalty solvers after the maximum budget of function evaluations have been used: Δf^* is the difference between the final objective function f and the minimum value f^* , $\|g_+\|$ is the norm of the constraints violations

| Problem | PPM ($\epsilon_0 = 10, \sigma = 2$) | | Exact penalty | |
|--------------|---------------------------------------|-----------|---------------|-----------|
| | Δf^* | $\ g_+\ $ | Δf^* | $\ g_+\ $ |
| HS36 | 4.03e-06 | 0.00e+00 | 1.32e+02 | 0.00e+00 |
| HS37 | 1.02e-08 | 0.00e+00 | 2.40e+01 | 0.00e+00 |
| HS44 | 2.88e-12 | 0.00e+00 | 0.00e+00 | 0.00e+00 |
| HS86 | 8.27e-08 | 0.00e+00 | 3.58e+00 | 0.00e+00 |
| HS224 | 7.87e-10 | 0.00e+00 | 1.47e+00 | 0.00e+00 |
| HS231 | 5.52e-11 | 0.00e+00 | 5.52e-11 | 0.00e+00 |
| HS232 | 9.23e-13 | 0.00e+00 | 4.44e-16 | 0.00e+00 |
| HS250 | 3.99e-10 | 0.00e+00 | 1.32e+02 | 0.00e+00 |
| HS331 | -1.44e-05 | 0.00e+00 | -1.44e-05 | 0.00e+00 |
| AS6($n=6$) | 9.62e-29 | 0.00e+00 | 1.37e-28 | 0.00e+00 |
| AS6($n=7$) | 7.18e-29 | 0.00e+00 | 1.04e-28 | 0.00e+00 |
| AS6($n=8$) | 1.32e-28 | 0.00e+00 | 1.10e-28 | 0.00e+00 |
| AS7($n=6$) | 1.49e-20 | 0.00e+00 | 4.86e-29 | 0.00e+00 |
| AS7($n=7$) | 2.95e-21 | 0.00e+00 | 7.00e-29 | 0.00e+00 |
| AS7($n=8$) | 5.89e-21 | 0.00e+00 | 2.54e-29 | 0.00e+00 |
| HS22 | 4.71e-14 | 0.00e+00 | 7.06e-09 | 6.66e-15 |
| HS29 | 6.22e-13 | 0.00e+00 | 5.84e+00 | 0.00e+00 |
| HS43 | 1.08e-12 | 1.22e-15 | 2.13e+01 | 0.00e+00 |
| HS65 | -3.53e-07 | 0.00e+00 | 2.77e-01 | 0.00e+00 |
| HS66 | 1.04e-10 | 0.00e+00 | 1.43e-02 | 0.00e+00 |
| HS270 | -3.55e-15 | 0.00e+00 | 1.00e+00 | 0.00e+00 |
| MAD1 | 3.90e-09 | 7.22e-16 | 1.38e-01 | 0.00e+00 |
| MAD2 | -2.99e-09 | 2.41e-10 | 6.51e-02 | 0.00e+00 |
| MAD4 | 3.89e-09 | 0.00e+00 | 7.30e-02 | 0.00e+00 |
| MAD5 | -4.62e-09 | 0.00e+00 | 2.27e-02 | 0.00e+00 |
| PENTAGON | 4.53e-02 | 0.00e+00 | 1.88e-01 | 0.00e+00 |
| WONG2 | 6.27e-01 | 1.78e-15 | 4.00e+00 | 0.00e+00 |

We compared the proposed formulation against a state-of-art approach for constrained nonsmooth optimization. In particular we compared it against the exact penalty method. We used the same solver that is shown to deliver state-of-art performances for the penalized problem to solve the proposed formulation. The results clearly show the advantages of the proposed formulation.

Future work will be devoted to (1) handle a mix convex and non-convex constraints by combining the proposed formulation with an exact penalty to deal with the non convex part of the constraints and (2) to study cases where the projection operation is expensive so that a truncated projection is to be employed.

Funding Open access funding provided by Università degli Studi di Firenze within the CRUI-CARE Agreement.

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

References

1. Abramson, M., Audet, C., Dennis, J., Digabel, S.: Orthomads: a deterministic mads instance with orthogonal directions. *SIAM J. Optim.* **20**(2), 948–966 (2009)
2. Abramson, M.A., Brezhneva, O.A., Jr., Pingel: Pattern search in the presence of degenerate linear constraints. *Optim. Methods Softw.* **23**(3), 297–319 (2008)
3. Audet, C., Custódio, A., Dennis, J.: Erratum: mesh adaptive direct search algorithms for constrained optimization. *SIAM J. Optim.* **18**(4), 1501–1503 (2008)
4. Audet, C., Dennis, J.: Mesh adaptive direct search algorithms for constrained optimization. *SIAM J. Optim.* **17**(1), 188–217 (2006)
5. Audet, C., Dennis, J.: A progressive barrier for derivative-free nonlinear programming. *SIAM J. Optim.* **20**(1), 445–472 (2009)
6. Audet, C., Hare, W.: *Derivative-Free and Blackbox Optimization*. Springer Series in Operations Research and Financial Engineering. Springer International Publishing, Cham, Switzerland (2017)
7. Bertsekas, D.P.: *Nonlinear Programming*. Athena Scientific, Belmont (1995)
8. Beyhaghi, P., Bewley, T.R.: Delaunay-based derivative-free optimization via global surrogates, part ii: convex constraints. *J. Global Optim.* **66**(3), 383–415 (2016)
9. Clarke, F.: *Optimization and Nonsmooth Analysis*. Society for Industrial and Applied Mathematics (1990)
10. Fasano, G., Liuzzi, G., Lucidi, S., Rinaldi, F.: A linesearch-based derivative-free approach for nonsmooth constrained optimization. *SIAM J. Optim.* **24**(3), 959–992 (2014)
11. Grippo, L., Lampariello, F., Lucidi, S.: Global convergence and stabilization of unconstrained minimization methods without derivatives. *J. Optim. Theory Appl.* **56**(3), 385–406 (1988)
12. Hock, W., Schittkowski, K.: *Test Examples for Nonlinear Programming Codes*. Springer-Verlag, Berlin, Heidelberg (1981)
13. Hooke, R., Jeeves, T.A.: “direct search” solution of numerical and statistical problems. *J. ACM* **8**(2), 212–229 (1961)
14. Jahn, J.: *Introduction to the Theory of Nonlinear Optimization*. Springer, Berlin Heidelberg (1996)
15. Le Digabel, S.: Algorithm 909: NOMAD: nonlinear optimization with the MADS algorithm. *ACM Trans. Math. Softw.* **37**(4), 1–15 (2011)
16. Lewis, R., Torczon, V.: Pattern search algorithms for bound constrained minimization. *SIAM J. Optim.* **9**(4), 1082–1099 (1999)
17. Lewis, R., Torczon, V.: Pattern search methods for linearly constrained minimization. *SIAM J. Optim.* **10**(3), 917–941 (2000)
18. Lewis, R., Torczon, V.: A globally convergent augmented lagrangian pattern search algorithm for optimization with general constraints and simple bounds. *SIAM J. Optim.* **12**(4), 1075–1089 (2002)
19. Liuzzi, G., Lucidi, S., Sciandrone, M.: Sequential penalty derivative-free methods for nonlinear constrained optimization. *SIAM J. Optim.* **20**(5), 2614–2635 (2010)
20. Lucidi, S., Sciandrone, M.: A derivative-free algorithm for bound constrained optimization. *Comput. Optim. Appl.* **21**(2), 119–142 (2002)
21. Lucidi, S., Sciandrone, M.: On the global convergence of derivative-free methods for unconstrained optimization. *SIAM J. Optim.* **13**(1), 97–116 (2002)
22. Lucidi, S., Sciandrone, M., Tseng, P.: Objective-derivative-free methods for constrained optimization. *Math. Program.* **92**(1), 37–59 (2002)
23. Lukšan, L., Vlček, J.: Test problems for nonsmooth unconstrained and linearly constrained optimization. *Technická zpráva* **798**, (2000)
24. Moré, J., Wild, S.: Benchmarking derivative-free optimization algorithms. *SIAM J. Optim.* **20**(1), 172–191 (2009)

25. naught101: Sobol random generation code. Software available at https://github.com/naught101/sobol_seq (2017)
26. Schittkowski, K. (ed.): More Test Examples for Nonlinear Programming Codes. Springer-Verlag, Berlin, Heidelberg (1987)
27. Sobol, I.: Uniformly distributed sequences with an additional uniform property. USSR Comput. Math. Math. Phys. **16**(5), 236–242 (1976)

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