



A new infeasible proximal bundle algorithm for nonsmooth nonconvex constrained optimization

Najmeh Hoseini Monjezi¹ · S. Nobakhtian^{1,2}

Received: 23 July 2018 / Published online: 15 June 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

Proximal bundle method has usually been presented for unconstrained convex optimization problems. In this paper, we develop an infeasible proximal bundle method for nonsmooth nonconvex constrained optimization problems. Using the improvement function we transform the problem into an unconstrained one and then we build a cutting plane model. The resulting algorithm allows effective control of the size of quadratic programming subproblems via the aggregation techniques. The novelty in our approach is that the objective and constraint functions can be any arbitrary (regular) locally Lipschitz functions. In addition the global convergence, starting from any point, is proved in the sense that every accumulation point of the iterative sequence is stationary for the improvement function. At the end, some encouraging numerical results with a MATLAB implementation are also reported.

Keywords Nonsmooth optimization · Nonconvex optimization · Constrained programming · Proximal bundle method · Improvement function · Infeasible algorithm

Mathematics Subject Classification 90C26 · 90C30 · 49J52 · 65K05

1 Introduction

We consider the following constrained optimization problem

✉ S. Nobakhtian
nobakht@math.ui.ac.ir

Najmeh Hoseini Monjezi
najmeh.hoseini@sci.ui.ac.ir

¹ Department of Mathematics, University of Isfahan, Isfahan, Iran

² Institute for Research in Fundamental Sciences (IPM), P.O. Box 19395-5746, Tehran, Iran

$$\begin{aligned} \min_{x \in X} \quad & f(x), \\ & g(x) \leq 0, \end{aligned} \tag{1}$$

where the objective and constraint functions $f, g: \mathbb{R}^n \rightarrow \mathbb{R}$ are locally Lipschitz but potentially nonsmooth and nonconvex. In addition $X \subseteq \mathbb{R}^n$ is a convex compact set.

Several numerical methods have been proposed for solving nonsmooth problems. Two classes of the most efficient methods are subgradient methods [17,34] and bundle methods. Bundle methods were first introduced by Lemaréchal [23] and have been developed over the years. Proximal bundle methods are currently considered among the most efficient methods for nonsmooth optimization problems, at least for convex problems [15,16,24]. Not long after works on the bundle methods for the convex case, the unconstrained counterpart of problem (1) was considered in [8,20,22] and more recently in [13,14,18]. The success of the proximal bundle method in unconstrained optimization (convex and nonconvex functions) greatly stimulated research to extend it to solve nonconvex constrained problems.

Extending Lemaréchal's algorithm to the nonconvex case, Mifflin [28] gives a bundle algorithm using the so-called downshift mechanism for problem (1). In [27], a proximal bundle method is presented for problem (1) with linear constraints by using the improvement function for handling the constraints. This method admits only the weak convergence in the sense that at least one of the accumulation points of the serious iterative sequence is stationary. We recall that, a new trial point is called a serious iterate if it is accepted as the next step, and a null iterate otherwise. Recently in [11] an algorithm is stated for problem (1) with the strong convergence, in the sense that every accumulation point is stationary, for classes of lower $-C^1$ functions (in the sense of [31]). In [7], an algorithm is proposed for problem (1) where f and g are lower $-C^1$ or upper $-C^1$ functions and the progress function is used for handling the constraint and showed that every accumulation point of the serious iterative sequence is stationary for the progress function. In addition, redistributed proximal bundle methods [13,14] are extended for solving problem (1) in [37] by using the penalty function, where the objective and constraint functions are lower $-C^2$. Moreover, these methods are studied in [25] by using the improvement function where the objective and constraint functions are lower $-C^1$. More recently, in [26] a semi-infinite programming (SIP) problem is studied, where the objective function is lower $-C^2$ and the constraint functions are twice continuously differentiable. By using a maximum function, the SIP problem is formulated as a nonsmooth constrained optimization problem. Then an infeasible bundle method based on the so-called improvement function is presented. By utilizing a special constraint qualification, the global convergence is obtained in the sense that the generated sequence converges to the KKT points of problem (1) as well as the KKT points of the SIP problem.

Some other methods for constrained nonsmooth nonconvex optimization problems are gradient sampling (GS) methods; see, for instance [5,36], where the penalty and improvement functions for handling the constraints are used, respectively. The global convergence of the sequential quadratic programming GS algorithm [36] and the penalty sequential quadratic programming GS algorithm [5] are established in the sense that, with probability one, every accumulation point of the iterative sequence is stationary, respectively, for the improvement function and the penalty function.

Techniques for bundle compression and aggregation, which are usual in the convex bundle algorithm, are not used in [26,28,37] and although mentioned in [7,11,25,27], are not explicitly addressed for updating formulas and the practical aspects. Without these features, we can not guarantee that the number of constraints in the subproblems can be kept smaller than a given desired bound. Therefore the method can not be guaranteed to be practical, then numerical difficulties may arise. In our method, we preserve the possibility of using aggregation techniques, which allow effective control to the size of subproblems and in addition we present updating formulas.

It is worth mentioning that in the most of the cited methods for constrained problems, e.g., [5,7,27,28,36] all the serious iterates and also the starting point, are required to be feasible. In those methods computing a feasible point is essential to start the algorithm. This preliminary general nonsmooth feasibility problem may be as difficult as to solve problem (1).

In this paper, we propose an infeasible proximal bundle method to solve problem (1). To handle nonconvexity, we extend the redistributed proximal bundle method [13,14] to the class of all regular locally Lipschitz functions, which is less restrictive than lower $-C^1$ and lower $-C^2$ functions. We use the improvement function to regularize the constraint, which is one of the most effective tools in this context; see [21,26,33]. At least part of the motivation for using the improvement function consists in avoiding the need to estimate a suitable value of the penalty parameter, which is often a delicate task. Then we obtain the convex cutting plane model. Different from the unconstrained optimization, our cutting plane model no longer approximates the objective function f or its local convexification as in [13,14]. In addition, unlike [26] our model does not approximate a maximum of the piecewise-linear models of the local convexification of the objective and constraint functions. However, we consider the augmented improvement function at the current step. Moreover, we compute the proximal bundle point of the model function to obtain new bundle elements and generate better minimizer estimate. The global convergence of the proposed algorithm is proved in the sense that, every accumulation point of the serious iterates sequence is stationary for the improvement function, when f and g are regular locally Lipschitz functions.

It is worth mentioning that in [26,37] the authors use the fact that when f and g are lower $-C^2$ and the convexity parameter is large enough then the local convexification of f and g can be obtained, however, in our model the regular functions are considered.

The proposed method is implemented in the MATLAB environment and applied to solve some test optimization problems. The numerical results show that this method can successfully be used to solve optimization problems. We also compare the proposed algorithm with the public software SolvOpt [34] and the proximal bundle method [25].

The rest of the paper is organized as follows. In Sect. 2 we review some basic definitions and results from nonsmooth analysis. In Sect. 3 we provide the details of our algorithm and its convergence presented in Sect. 4. Computational experience is reported in Sect. 5 and, finally, in Sect. 6 the conclusions are presented.

2 Preliminaries

Our notations are fairly standard. We denote by $\langle u, v \rangle = \sum_{i=1}^n u_i v_i$ the inner product of two vectors $u, v \in \mathbb{R}^n$. We use $\| \cdot \|$ to denote the standard Euclidean norm. For $x \in \mathbb{R}^n$ and $\varepsilon > 0$, $B_\varepsilon(x)$ is an open ball of the radius ε centered at x and $\overline{B}_\varepsilon(x)$ is a closed ball of radius ε centered at x .

First we recall some concepts and results from nonsmooth analysis. Most of the materials included here can be found in [3]. Then we recall the definition of the improvement function and state some of its properties.

A function $h: \mathbb{R}^n \rightarrow \mathbb{R}$ is said to be locally Lipschitz of rank $L > 0$ at $x \in \mathbb{R}^n$ if for some $\varepsilon > 0$, we have $|h(y) - h(z)| \leq L\|y - z\|$ for all $y, z \in B_\varepsilon(x)$. The classical directional derivative of h at x in the direction $d \in \mathbb{R}^n$ is defined as

$$h'(x; d) := \lim_{\alpha \rightarrow 0} \frac{h(x + \alpha d) - h(x)}{\alpha}.$$

The Clarke directional derivative of h at x in the direction d is defined by

$$h^\circ(x; d) := \limsup_{\substack{y \rightarrow x \\ \alpha \downarrow 0}} \frac{h(y + \alpha d) - h(y)}{\alpha}.$$

The Clarke subdifferential (generalized gradient) of h at x is defined as

$$\partial_c h(x) := \left\{ \xi \in \mathbb{R}^n \mid \xi^T d \leq h^\circ(x; d), \forall d \in \mathbb{R}^n \right\}.$$

Each element $\xi \in \partial_c h(x)$ is called a subgradient of h at x . It is well-known that $\partial_c h(x)$ is a nonempty convex compact set in \mathbb{R}^n . Also the Clarke subdifferential $\partial_c h(x)$ is upper semicontinuous for every $x \in \mathbb{R}^n$.

The function $h: \mathbb{R}^n \rightarrow \mathbb{R}$ is convex, if $h(\lambda x + (1 - \lambda)y) \leq \lambda h(x) + (1 - \lambda)h(y)$, for all $x, y \in \mathbb{R}^n$ and $\lambda \in [0, 1]$. The subdifferential of a convex function h at x is the set

$$\partial h(x) := \left\{ \xi \in \mathbb{R}^n \mid h(y) \geq h(x) + \langle \xi, y - x \rangle, \forall y \in \mathbb{R}^n \right\},$$

and it coincides with the Clarke subdifferential for every convex function.

Let $\varepsilon \geq 0$ the ε -subdifferential of a convex function h at x is defined as

$$\partial_\varepsilon h(x) := \left\{ \xi \in \mathbb{R}^n \mid h(y) \geq h(x) + \langle \xi, y - x \rangle - \varepsilon, \forall y \in \mathbb{R}^n \right\}.$$

Each element $\xi \in \partial_\varepsilon h(x)$ is called an ε -subgradient of h at x [27].

The function $h: \mathbb{R}^n \rightarrow \mathbb{R}$ is regular at x provided that h is locally Lipschitz at x and admits directional derivatives $h'(x; d)$ at x for all d , with $h'(x; d) = h^\circ(x; d)$.

Let S be a subset of \mathbb{R}^n . Its indicator function $I_S: \mathbb{R}^n \rightarrow \mathbb{R}_\infty$ is the function which has value 0 on S and $+\infty$ elsewhere.

The family of lower $-C^1$ functions was introduced in [35]. It constitutes a broad class of locally Lipschitz functions that contains lower $-C^2$ functions as a subfamily. Combining [6, Thm. 2 and Cor. 3] with [35, Prob. 2.4], when $h: \mathbb{R}^n \rightarrow \mathbb{R}$ is lower $-C^1$, then h is semismooth and regular. But the converse is not necessarily true. Spingarn in [35, P. 83] gives an example of a regular function in \mathbb{R}^2 whose Clarke subdifferentials is not strictly submonotone at some points and therefore the function is not lower $-C^1$.

A point $x^* \in \mathbb{R}^n$ is called a stationary point of $h: \mathbb{R}^n \rightarrow \mathbb{R}$ if $0 \in \partial_c h(x^*)$. If x^* is a local minimizer of h , then x^* is a stationary point of h . Also a point $x^* \in X$ is called a stationary point of h on X if $0 \in \partial_c h(x^*) + \partial I_X(x^*)$. When $x^* \in X$ is a local minimizer of h on X , then x^* is stationary for h on X .

Recall that $x^* \in X$ satisfies the Fritz John necessary optimality conditions for problem (1) if there exist $\lambda_0^* \geq 0, \lambda_1^* \geq 0$ with $\lambda_0^* + \lambda_1^* = 1$ such that $0 \in \lambda_0^* \partial_c f(x^*) + \lambda_1^* \partial_c g(x^*) + \partial I_X(x^*)$ and $\lambda_1^* g(x^*) = 0, g(x^*) \leq 0$. When $\lambda_0^* > 0$, then x^* satisfies the Karush–Kuhn–Tucker (KKT) conditions.

2.1 The improvement function

We introduce the improvement function $h: \mathbb{R}^n \rightarrow \mathbb{R}$ associated with problem (1). For a given point $x \in \mathbb{R}^n$, let $h_x(y) = h(y, x) := \max\{f(y) - f(x), g(y)\}$. We recall some useful properties of this function. Directly by definition of the improvement function and subdifferential calculus [3], we have

$$\partial_c h(y, x) \subseteq \begin{cases} \partial_c f(y) & f(y) - f(x) > g(y) \\ \text{conv}\{\partial_c f(y) \cup \partial_c g(y)\} & f(y) - f(x) = g(y) \\ \partial_c g(y) & f(y) - f(x) < g(y), \end{cases} \tag{2}$$

where ‘‘conv’’ denotes convex hull, and equality holds when f and g are regular functions. In addition $h(x, x) = g^+(x) = \max\{g(x), 0\}$, for all $x \in \mathbb{R}^n$.

Let $X \subseteq \mathbb{R}^n$ be a convex set. We have $I_X(\cdot)$ is a convex function, and $\partial I_X(\cdot)$ is the normal cone to X at x , i.e.,

$$\partial I_X(x) = N_X(x) := \{v \in \mathbb{R}^n \mid \langle v, y - x \rangle \leq 0, \forall y \in X\}.$$

First we recall the following result from [7, Lemma 2.1], which shows that the improvement function is suitable to serve as an alternative of the penalty function.

Lemma 1 *Suppose that f and g are locally Lipschitz functions, then the following hold:*

- (A) *If x^* is a local minimizer of problem (1), then it is local minimizer of $h(\cdot, x^*)$ on X . In addition we have $0 \in \partial_c h(x^*, x^*) + \partial I_X(x^*)$.*
- (B) *If $0 \in \partial_c h(x^*, x^*) + \partial I_X(x^*)$ for $x^* \in X$, then only one of the following occurs:*
 - (i) *if $g(x^*) > 0$, then $0 \in \partial_c g(x^*) + \partial I_X(x^*)$, i.e., x^* is a stationary point of g on X .*
 - (ii) *if $g(x^*) < 0$, then $0 \in \partial_c f(x^*) + \partial I_X(x^*)$, i.e., x^* is stationary for f on X and it is a KKT point of problem (1).*

(iii) finally $g(x^*) = 0$ implies that x^* satisfies the Fritz John necessary conditions for problem (1).

3 Description of the algorithm

In this section we describe the new algorithm. The main idea of the method is as follows. Given the last serious iterate x^k (the starting point x^0 is regarded as the first serious iterate). We generate candidate points by iterating with the proximal bundle method applied to a piecewise-linear model of the augmented improvement function, until the next serious iterate x^{k+1} is obtained. However, to decide whether a candidate point can be the next serious iterate, we use a usual descent condition. When the new serious iterate x^{k+1} has been found, the piecewise-linear model should be redefined, according to x^{k+1} , for the next iteration. In the sequel of this section we will discuss different elements of the algorithm, which itself will be presented at the end of this section.

3.1 Model function

As usual in the bundle methods, the available information is used to define a piecewise-linear model of $h(\cdot, x^k)$. In our setting, we are working with possible nonconvexity of f, g and consequently of $h(\cdot, x^k)$. We define the linearization errors for f and g at x^k , corresponding with $y^j \in \mathbb{R}^n$, respectively by

$$e_{f_j}^k := f(x^k) - f(y^j) - \langle \xi_f^j, x^k - y^j \rangle, \tag{3a}$$

$$e_{g_j}^k := g(x^k) - g(y^j) - \langle \xi_g^j, x^k - y^j \rangle, \tag{3b}$$

where $\xi_f^j \in \partial_c f(y^j)$ and $\xi_g^j \in \partial_c g(y^j)$. For a convex function, linearization errors are always nonnegative. This feature is crucial to prove the convergence of the most of the bundle methods [4,16]. However, in the nonconvex case, it may yield negative linearization error. To overcome this drawback, we extend the approach introduced in [13,14] for nonconvex constrained optimization problems and work with the following augmented function

$$h_{\eta_l}(\cdot, x^k) = h(\cdot, x^k) + \frac{\eta_l}{2} \|\cdot - x^k\|^2, \tag{4}$$

where η_l is a certain parameter, that adjusted dynamically. Let l be the current iteration index and $\mathcal{L}_l \subseteq \{0, 1, \dots, l - 1\}$ is the index set at the l th iteration. We generate the convex piecewise-linear model as

$$\begin{aligned} H_l(y, x^k) &:= h(x^k, x^k) + \max_{j \in \mathcal{L}_l} \left\{ -c_{f_j}^k + \langle s_{f_j}^k, y - x^k \rangle, -c_{g_j}^k + \langle s_{g_j}^k, y - x^k \rangle \right\} \\ &= g^+(x^k) + \max_{j \in \mathcal{L}_l} \left\{ -c_{f_j}^k + \langle s_{f_j}^k, y - x^k \rangle, -c_{g_j}^k + \langle s_{g_j}^k, y - x^k \rangle \right\}, \end{aligned} \tag{5}$$

where

$$c_{f_j}^k := e_{f_j}^k + g^+(x^k) + \eta_l b_j^k, \quad c_{g_j}^k := e_{g_j}^k + g^+(x^k) - g(x^k) + \eta_l b_j^k, \quad (6a)$$

$$s_{f_j}^k := \xi_{f_j}^j + \eta_l d_j^k, \quad s_{g_j}^k := \xi_{g_j}^j + \eta_l d_j^k, \quad d_j^k := y^j - x^k, \quad b_j^k := \frac{\|y^j - x^k\|^2}{2}, \quad (6b)$$

and $e_{f_j}^k, e_{g_j}^k$ are stated in (3a), (3b) and the bundle of information is defined as follows

$$\mathcal{B}_l \subseteq \bigcup_{i \in \mathcal{L}_l} \{(f_i, g_i, \xi_{f_i}^i, \xi_{g_i}^i, e_{f_i}^k, e_{g_i}^k, b_i^k, d_i^k)\},$$

where $f_i := f(y^i)$ and $g_i := g(y^i)$. Clearly $H_l(\cdot, x^k)$ is a piecewise-linear model for the augmented improvement function (4). We call $m_f(\cdot, x^k) := g^+(x^k) - c_{f_j}^k + \langle s_{f_j}^k, \cdot - x^k \rangle$ and $m_g(\cdot, x^k) := g^+(x^k) - c_{g_j}^k + \langle s_{g_j}^k, \cdot - x^k \rangle$ as oracle planes. Any choice of the positive parameter $\eta_l \in \mathbb{R}$ that keeps $c_{f_j}^k$ and $c_{g_j}^k$ nonnegative is acceptable. In our setting we take

$$\eta_l \geq \max \left\{ \max_{\substack{j \in \mathcal{L}_l \\ y^j \neq x^k}} \left\{ \frac{-\left(e_{f_j}^k + g^+(x^k)\right)}{b_j^k}, \frac{-\left(e_{g_j}^k + g^+(x^k) - g(x^k)\right)}{b_j^k} \right\}, \omega \right\} + \omega, \quad (7)$$

where $\omega > 0$ is a positive constant.

Remark 1 In [13], the authors established the proximal bundle method for unconstrained problem when f is lower $-C^2$. Moreover in [37] a proximal bundle method and in [26] an infeasible proximal bundle algorithm were presented for constrained problems where f and g are lower $-C^2$. They can say nothing about what is obtained if the sequence $\{\eta_l\}$ remains too small. Since they used the fact that when η_l is large enough and f, g are lower $-C^2$, then the augmented function $h_{\eta_l}(\cdot, x^k)$ is convex. Indeed, in our approach we do not require any lower $-C^2$, lower $-C^1$ or prox-regular assumption on the functions. Therefore $h_{\eta_l}(\cdot, x^k)$ in (4) can be nonconvex in some iterations and only we choose η_l such that (7) is satisfied.

Lemma 2 *If η_l is selected according to Eq. (7), then for every $j \in \mathcal{L}_l$, we obtain $c_{f_j}^k \geq 0$ and $c_{g_j}^k \geq 0$.*

Proof By using (7) for all $j \in \mathcal{L}_l$ when $y^j \neq x^k$, we have

$$(\eta_l - \omega)b_j^k \geq -\left(e_{f_j}^k + g^+(x^k)\right) \quad \text{and} \quad (\eta_l - \omega)b_j^k \geq -\left(e_{g_j}^k + g^+(x^k) - g(x^k)\right),$$

and by (6a) we obtain $c_{f_j}^k \geq \omega b_j^k \geq 0$ and $c_{g_j}^k \geq \omega b_j^k \geq 0$, for all $j \in \mathcal{L}_l$ and $y^j \neq x^k$. When $y^j = x^k$ we have $c_{f_j}^k = g^+(x^k) \geq 0$ and $c_{g_j}^k = g^+(x^k) - g(x^k) \geq 0$. Hence for all $j \in \mathcal{L}_l$ we get $c_{f_j}^k \geq 0$ and $c_{g_j}^k \geq 0$. □

We assume the last serious iterate until now, will be one of the bundle points, i.e., $x^k \in \{y^j\}_{j \in \mathcal{L}_l}$. That is, there exists $l(k) \in \mathcal{L}_l$ such that $x^k = y^{l(k)}$. Hence $m_{f_l}^0(\cdot, x^k) = g^+(x^k) + \langle \xi_f^{l(k)}, \cdot - x^k \rangle$ and $m_{g_l}^0(\cdot, x^k) = g^+(x^k) - g(x^k) + \langle \xi_g^{l(k)}, \cdot - x^k \rangle$ are the cutting planes of $H_l(\cdot, x^k)$. Also $c_{f_l}^k = g^+(x^k)$ and $c_{g_l}^k = g^+(x^k) - g(x^k)$, therefore we have

$$H_l(x^k, x^k) = g^+(x^k) + \max_{j \in \mathcal{L}_l} \left\{ -c_{f_j}^k, -c_{g_j}^k \right\} = g^+(x^k).$$

The iterates $\{y^l\}$ include both the serious steps x^k and the null steps y^l (which are candidate points that were not serious). As we have mentioned above, for all k there exists $l(k)$ such that $x^k = y^{l(k)}$. Therefore we have $\{x^k\} \subseteq \{y^l\}$.

Given $\mu_l > 0$, a positive proximal parameter, the next iterate y^l is generated by solving the following convex problem

$$\min_{y \in X} H_l(y, x^k) + \frac{\mu_l}{2} \|y - x^k\|^2. \tag{8}$$

Clearly, y^l is unique. If y^l provides a significant decrease with respect to $h(x^k, x^k)$, we call it serious iterate, otherwise null iterate (we explain this case in the next subsection). From the optimality conditions of problem (8) we have $0 \in \partial H_l(y^l, x^k) + \mu_l(y^l - x^k) + \partial I_X(y^l)$. Therefore, there exist $\widehat{\xi}_l^k \in \partial H_l(y^l, x^k)$ and $v_l \in \partial I_X(y^l)$ such that $y^l = x^k - \frac{1}{\mu_l}(\widehat{\xi}_l^k + v_l)$ and $\mu_l(x^k - y^l) = (\widehat{\xi}_l^k + v_l) \in \partial H_l(y^l, x^k) + \partial I_X(y^l)$. Since the model (5) is piecewise-linear, there exist $\alpha^l, \beta^l \in \mathbb{R}^{|\mathcal{L}_l|}$ with $\alpha_j^l, \beta_j^l \geq 0$ and $\sum_{j=1}^{|\mathcal{L}_l|} (\alpha_j^l + \beta_j^l) = 1$ such that $\widehat{\xi}_l^k := \sum_{j=1}^{|\mathcal{L}_l|} (\alpha_j^l s_{f_j}^k + \beta_j^l s_{g_j}^k)$. We called $\widehat{\xi}_l^k$ the aggregate subgradient. The aggregate error is defined by

$$\widehat{e}_l^k := g^+(x^k) - H_l(y^l, x^k) - \langle \widehat{\xi}_l^k, x^k - y^l \rangle \geq 0, \tag{9}$$

where the inequality follows from $\widehat{\xi}_l^k \in \partial H_l(y^l, x^k)$. It is easy to verify that $\widehat{\xi}_l^k \in \partial_{\mathcal{E}_l^k} H_l(x^k, x^k)$. We define the aggregate plane by $m_l^*(\cdot, x^k) := g^+(x^k) - \widehat{e}_l^k + \langle \widehat{\xi}_l^k, \cdot - x^k \rangle$. To avoid overflow, when generating the new model $H_l(\cdot, x^k)$, we may replace all older planes corresponding to $\alpha_j^l > 0$ and $\beta_j^l > 0$ by the aggregate plane. Note that this aggregate plane can directly be defined from the aggregate couple $(\widehat{e}_l^k, \widehat{\xi}_l^k)$.

Accordingly, we have two kinds of planes in the bundle, namely: the oracle and aggregate planes. Therefore we shall write the index set in the form $\mathcal{L}_l := \mathcal{L}_l^{\text{oracle}} \cup \mathcal{L}_l^{\text{agg}}$ and their bundles in the form $\mathcal{B}_l := \mathcal{B}_l^{\text{oracle}} \cup \mathcal{B}_l^{\text{agg}}$,

$$\mathcal{B}_l^{\text{oracle}} \subseteq \bigcup_{i \in \mathcal{L}_l^{\text{oracle}}} \left\{ (f_i, g_i, \xi_f^i, \xi_g^i, e_{f_i}^k, e_{g_i}^k, b_i^k, d_i^k) \right\},$$

and $\mathcal{B}_l^{\text{agg}} \subseteq \bigcup_{i \in \mathcal{L}_l^{\text{agg}}} \left\{ (\widehat{e}_i^k, \widehat{\xi}_i^k \in \partial H_i(y^i, x^k)) \right\}.$

It is worth mentioning that for any iterate y^l (serious or null), we add one element in $\mathcal{B}_l^{\text{oracle}}$ that is $(f_l, g_l, \xi_f^l, \xi_g^l, e_{f_j}^k, e_{g_l}^k, b_l^k, d_l^k)$ (and also in $\mathcal{L}_l^{\text{oracle}}$). Corresponding to any element in this bundle we have two cutting planes in the model, for f and g . Like our method, in [19,33] for convex constrained problems one bundle for both f and g is considered and thus the implementation of the algorithm and updating the formulas are simple. However, unlike our method, in some bundle methods for constrained problems two bundles for f and g independently are considered; see e.g., [25,26]. Therefore the convex piecewise-linear model is as follows

$$H_l(y, x^k) = g^+(x^k) + \max \left\{ \max_{j \in \mathcal{L}_l^{\text{agg}}} \left\{ -\widehat{e}_j^k + \langle \widehat{\xi}_j^k, y - x^k \rangle \right\}, \right. \\ \left. \max_{j \in \mathcal{L}_l^{\text{oracle}}} \left\{ -c_{f_j}^k + \langle s_{f_j}^k, y - x^k \rangle, -c_{g_j}^k + \langle s_{g_j}^k, y - x^k \rangle \right\} \right\}. \tag{10}$$

3.2 Acceptance test

An iterate y^l is considered good enough to become the next serious iterate when $h(y^l, x^k)$ provides a significant decrease with respect to $h(x^k, x^k) = g^+(x^k)$. Let $m \in (0, 1)$ be a given parameter. Define the predicted decrease $\delta^l := \widehat{e}_l^k + \frac{1}{2\mu_l} \|\widehat{\xi}_l^k + v_l\|^2$. Note that since $\widehat{e}_l^k \geq 0$, it follows that $\delta^l \geq 0$. When y^l satisfies the descent test

$$h(y^l, x^k) \leq g^+(x^k) - m\delta^l, \tag{11}$$

a serious iterate is declared, i.e., $x^{k+1} := y^l$. Otherwise, y^l is a null iterate and x^k remains unchanged. The algorithm stops when δ^l is small enough. In this case, both \widehat{e}_l^k and $\|\widehat{\xi}_l^k + v_l\|$ are small and since $\widehat{\xi}_l^k \in \partial_{e_l^k} H_l(y^l, x^k)$ and $v_l \in \partial I_X(y^l)$, we deduce x^k is a stationary point (we will resume the discussion of convergence in Sect. 4).

Remark 2 Recalling the definition of $h(\cdot, x^k)$, we conclude that if the descent test is satisfied and a serious iterate is declared, we get

$$f(x^{k+1}) - f(x^k) \leq g^+(x^k) - m\delta^l \quad \text{and} \quad g(x^{k+1}) \leq g^+(x^k) - m\delta^l. \tag{12}$$

In particular, if x^k is feasible, i.e., $g^+(x^k) = 0$, it follows from (12) that $f(x^{k+1}) < f(x^k)$ and $g(x^{k+1}) \leq 0$. Thus x^{k+1} is feasible too and $\{f(x_k)\}$ is strictly monotone. Otherwise, when x^k is infeasible, it is possible $f(x_{k+1}) > f(x_k)$ due to $g^+(x^k) = g(x^k) > 0$. Therefore outside of the feasible region, the method is not monotone with respect to f . However outside of the feasible region we have $g(x^{k+1}) < g^+(x^k) = g(x^k)$. This seems intuitively reasonable: while it is natural to accept the increase in the objective function value in order to decrease infeasibility.

3.3 Updating the model

Suppose that y^l is a null iterate, we will augment the piecewise-linear model $H_l(\cdot, x^k)$ and generate $H_{l+1}(\cdot, x^k)$. Note that the cutting plane respect to the last serious iterate (i.e., x^k) should keep in the model. To make $H_{l+1}(\cdot, x^k)$ better than $H_l(\cdot, x^k)$, we should add two more elements: the cutting plane and the aggregate plane corresponding to y^l . It is enough to make sure $\{(\widehat{e}_l^k, \widehat{\xi}_l^k)\} \subseteq \mathcal{B}_{l+1}^{\text{agg}}$ and $\{(f_l, g_l, \xi_f^l, \xi_g^l, e_{f_l}^k, e_{g_l}^k, b_l^k, d_l^k)\} \subseteq \mathcal{B}_{l+1}^{\text{oracle}}$. Moreover we set $\mathcal{L}_{l+1}^{\text{oracle}} = \mathcal{L}_l^{\text{oracle}} \cup \{l\}$ and $\mathcal{L}_{l+1}^{\text{agg}} = \mathcal{L}_l^{\text{agg}} \cup \{l\}$. Therefore, when there is a null iterate the updating of the bundle and presenting of the model accomplish without any problem.

Remark 3 It is worth mentioning that the total number of the elements in \mathcal{B}_l can be kept as small as three, at every step of the method. It can be given by

$$\mathcal{B}_{l+1} = \left\{ \left(f_l, g_l, \xi_f^l, \xi_g^l, e_{f_l}^k, e_{g_l}^k, b_l^k, d_l^k \right), \left(\widehat{e}_l^k, \widehat{\xi}_l^k \right), \left(f_{l(k)}, g_{l(k)}, \xi_f^{l(k)}, \xi_g^{l(k)}, e_{f_{l(k)}}^k, e_{g_{l(k)}}^k, b_{l(k)}^k, d_{l(k)}^k \right) \right\}.$$

As usual in the convex proximal bundle method (see, for instance [19]) the total number of elements in the bundle can be kept as small as two, namely: the cutting plane at the last iterate y^l and the last aggregate plane. Let us emphasize that the situation in our setting is slightly different from methods for standard convex programming. These differences are actually natural, having in mind that convex and nonconvex cases require significantly different tools. Like our setting in [7] the cutting plane respect to the last serious iterate x^k is considered.

When a new serious iterate is declared, the model $H_l(\cdot, x^k)$ has to be properly revised and at the next iteration we start working with the new model $H_l(\cdot, x^{k+1})$. Therefore any element in $\mathcal{B}_l^{\text{agg}}$ and $\mathcal{B}_l^{\text{oracle}}$ that is depended on k should be redefine. On the other hand, since we did not store y^j and $H_j(y^j, x^k)$ for any $j \in \mathcal{L}_l$, we redefine the elements of $\mathcal{B}_l^{\text{agg}}$ and $\mathcal{B}_l^{\text{oracle}}$ by updating formulas. Incidentally, by updating formulas the computation of the new cutting model $H_l(\cdot, x^{k+1})$ is easy. We update the linearization errors and the other items for all $j \in \mathcal{L}_l^{\text{oracle}}$ according to the following relations

$$e_{f_j}^{k+1} = e_{f_j}^k + f(x^{k+1}) - f(x^k) - \langle \xi_f^j, x^{k+1} - x^k \rangle, \tag{13a}$$

$$e_{g_j}^{k+1} = e_{g_j}^k + g(x^{k+1}) - g(x^k) - \langle \xi_g^j, x^{k+1} - x^k \rangle, \tag{13b}$$

$$b_j^{k+1} = b_j^k + \frac{1}{2} \|x^{k+1} - x^k\|^2 - \langle d_j^k, x^{k+1} - x^k \rangle, \tag{13c}$$

$$d_j^{k+1} = d_j^k - (x^{k+1} - x^k). \tag{13d}$$

The update aims at satisfying (3a), (3b) and the two last relations (6b). The aggregate subgradients and the aggregate errors for all $j \in \mathcal{L}_l^{\text{agg}}$, respectively, are updated by

$$\widehat{\xi}_j^{k+1} = \widehat{\xi}_j^k + \eta_l(x^k - x^{k+1}), \tag{14a}$$

$$\begin{aligned} \widehat{e}_j^{k+1} &= \widehat{e}_j^k + g^+(x^{k+1}) - g^+(x^k) + \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 - \langle \widehat{\xi}_j^k, x^{k+1} - x^k \rangle \\ &\quad + (f(x^{k+1}) - f(x^k))^+. \end{aligned} \tag{14b}$$

The update aims at satisfying the key relation $\widehat{\xi}_j^k \in \partial_{\widehat{e}_j^k} H_l(x^k, x^k)$; see Lemma 4.

In the following lemma, we show the relationship between $h_{\eta_l}(\cdot, x^k)$ and $h_{\eta_l}(\cdot, x^{k+1})$, that these are the augmented functions in two tandem serious iterations. We also show that the same relationship holds for their cutting plane models.

Lemma 3 *Functions $h_{\eta_l}(\cdot, x^k)$ and $h_{\eta_l}(\cdot, x^{k+1})$ satisfy the following inequality*

$$\begin{aligned} h_{\eta_l}(y, x^{k+1}) &\geq h_{\eta_l}(y, x^k) - (f(x^{k+1}) - f(x^k))^+ + \eta_l \langle y - x^{k+1}, x^k - x^{k+1} \rangle \\ &\quad - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2, \end{aligned} \tag{15}$$

for all $y \in \mathbb{R}^n$. Furthermore, the above inequality is true for $H_l(\cdot, x^k)$ and $H_l(\cdot, x^{k+1})$ by replacing $h_{\eta_l}(\cdot, x^k)$ and $h_{\eta_l}(\cdot, x^{k+1})$, respectively.

Proof First, we show that

$$h(y, x^{k+1}) \geq h(y, x^k) - (f(x^{k+1}) - f(x^k))^+, \quad \forall y \in \mathbb{R}^n. \tag{16}$$

Consider $y \in \mathbb{R}^n$. If $f(y) - f(x^{k+1}) \geq g(y)$ and $f(y) - f(x^k) \geq g(y)$, then $h(y, x^{k+1}) - h(y, x^k) = f(x^k) - f(x^{k+1}) \geq -(f(x^{k+1}) - f(x^k))^+$. When $f(y) - f(x^{k+1}) < g(y)$ and $f(y) - f(x^k) < g(y)$, then $h(y, x^{k+1}) - h(y, x^k) = 0$ and (16) holds. Now suppose that $f(y) - f(x^{k+1}) < g(y)$ and $f(y) - f(x^k) \geq g(y)$, then $h(y, x^{k+1}) - h(y, x^k) = g(y) - f(y) + f(x^k) > f(y) - f(x^{k+1}) - f(y) + f(x^k) = f(x^k) - f(x^{k+1}) \geq -(f(x^{k+1}) - f(x^k))^+$. Finally if $f(y) - f(x^{k+1}) \geq g(y)$ and $f(y) - f(x^k) < g(y)$, then $h(y, x^{k+1}) - h(y, x^k) = f(y) - f(x^{k+1}) - g(y) \geq 0$ which again implies (16). Using (4) we get

$$h_{\eta_l}(y, x^{k+1}) \geq h_{\eta_l}(y, x^k) + \frac{\eta_l}{2} (\|y - x^{k+1}\|^2 - \|y - x^k\|^2) - (f(x^{k+1}) - f(x^k))^+.$$

We know that

$$\begin{aligned} \|y - x^k\|^2 &= \|y - x^{k+1} + x^{k+1} - x^k\|^2 \\ &= \|y - x^{k+1}\|^2 + \|x^{k+1} - x^k\|^2 + 2\langle y - x^{k+1}, x^{k+1} - x^k \rangle, \end{aligned}$$

which implies (15).

Now we show that (15) holds for $H_l(\cdot, x^k)$ and $H_l(\cdot, x^{k+1})$ by replacing $h_l(\cdot, x^k)$ and $h_l(\cdot, x^{k+1})$, respectively. For all $j \in \mathcal{L}_l^{\text{oracle}}$ we have

$$\begin{aligned} H_l(y, x^{k+1}) &\geq g^+(x^{k+1}) - c_{f_j}^{k+1} + \langle s_{f_j}^{k+1}, y - x^{k+1} \rangle \\ &= -e_{f_j}^{k+1} - \eta_l b_j^{k+1} + \langle \xi_j^j + \eta_l(y^j - x^{k+1}), y - x^{k+1} \rangle \end{aligned}$$

$$\begin{aligned}
 &= -\left(e_{f_j}^k + f(x^{k+1}) - f(x^k) - \langle \xi_f^j, x^{k+1} - x^k \rangle\right) - \eta_l \left(\frac{1}{2} \|x^{k+1} - x^k\|^2\right. \\
 &\quad \left.+ b_j^k - \langle y^j - x^k, x^{k+1} - x^k \rangle + \langle \xi_f^j + \eta_l(y^j - x^{k+1}), y - x^{k+1} \rangle\right) \\
 &= -(e_{f_j}^k + \eta_l b_j^k) + f(x^k) - f(x^{k+1}) - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 + \langle \xi_f^j, y - x^k \rangle \\
 &\quad + \eta_l \langle y^j - x^{k+1}, y - x^{k+1} \rangle + \eta_l \langle y^j - x^k, x^{k+1} - x^k \rangle \\
 &= -(e_{f_j}^k + \eta_l b_j^k) + f(x^k) - f(x^{k+1}) - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 + \langle \xi_f^j, y - x^k \rangle \\
 &\quad + \eta_l \langle y^j - x^k, y - x^k \rangle + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle \\
 &= -(e_{f_j}^k + \eta_l b_j^k) + f(x^k) - f(x^{k+1}) - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 + \langle s_{f_j}^k, y - x^k \rangle \\
 &\quad + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle \\
 &= g^+(x^k) - c_{f_j}^k + \langle s_{f_j}^k, y - x^k \rangle - (f(x^{k+1}) - f(x^k))^+ \\
 &\quad - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle, \tag{17}
 \end{aligned}$$

where the first inequality comes from the definition of $H_l(y, x^{k+1})$ in (10), the first equality holds by the first relation in (6a) and the first relation in (6b), the second equality is satisfied by (13a) and (13c) and the two last equalities follow from the first relation in (6a) and the first relation in (6b), respectively.

By the same discussion and adding $-(f(x^{k+1}) - f(x^k))^+$, for all $j \in \mathcal{L}_l^{\text{oracle}}$ we deduce

$$\begin{aligned}
 H_l(y, x^{k+1}) &\geq g^+(x^k) - c_{g_j}^k + \langle s_{g_j}^k, y - x^k \rangle - (f(x^{k+1}) - f(x^k))^+ \\
 &\quad - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle. \tag{18}
 \end{aligned}$$

Since (17) and (18) hold for all $j \in \mathcal{L}_l^{\text{oracle}}$, we get

$$\begin{aligned}
 H_l(y, x^{k+1}) &\geq g^+(x^k) + \max_{j \in \mathcal{L}_l^{\text{oracle}}} \{-c_{f_j}^k + \langle s_{f_j}^k, y - x^k \rangle, -c_{g_j}^k + \langle s_{g_j}^k, y - x^k \rangle\} \\
 &\quad - (f(x^{k+1}) - f(x^k))^+ - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle. \tag{19}
 \end{aligned}$$

By using the definition of $H_l(y, x^k)$, (14a), (14b), for all $j \in \mathcal{L}_l^{\text{agg}}$, we have

$$\begin{aligned}
 H_l(y, x^{k+1}) &\geq g^+(x^{k+1}) - \widehat{e}_j^{k+1} + \langle \widehat{\xi}_j^{k+1}, y - x^{k+1} \rangle \\
 &= g^+(x^{k+1}) - \left(\widehat{e}_j^k + g^+(x^{k+1}) - g^+(x^k) + \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 + (f(x^{k+1})\right. \\
 &\quad \left.- f(x^k))^+ - \langle \widehat{\xi}_j^k, x^{k+1} - x^k \rangle\right) + \langle \widehat{\xi}_j^k + \eta_l(x^k - x^{k+1}), y - x^{k+1} \rangle
 \end{aligned}$$

$$\begin{aligned}
 &= g^+(x^k) + (-\widehat{e}_j^k + \langle \widehat{\xi}_j^k, y - x^k \rangle) + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle \\
 &\quad - (f(x^{k+1}) - f(x^k))^+ - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2,
 \end{aligned}$$

therefore

$$\begin{aligned}
 H_l(y, x^{k+1}) &\geq g^+(x^k) + \max_{j \in \mathcal{L}_l^{\text{agg}}} \left\{ -\widehat{e}_j^k + \langle \widehat{\xi}_j^k, y - x^k \rangle \right\} - (f(x^{k+1}) - f(x^k))^+ \\
 &\quad - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle. \tag{20}
 \end{aligned}$$

Using (10), (19) and (20), we deduce

$$\begin{aligned}
 H_l(y, x^{k+1}) &\geq H_l(y, x^k) - (f(x^{k+1}) - f(x^k))^+ - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 \\
 &\quad + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle, \tag{21}
 \end{aligned}$$

and the proof is completed. □

Lemma 4 *Suppose that the associated y^l is declared as a serious iterate, i.e., $x^{k+1} = y^l$. Then the following hold:*

- (i) *If for each $j \in \mathcal{L}_l^{\text{oracle}}$ we update $e_{f_j}^{k+1}$ and $e_{g_j}^{k+1}$ according to relations (13a), (13b), then conditions (3a), (3b) are satisfied with $k := k + 1$.*
- (ii) *If for each $j \in \mathcal{L}_l^{\text{agg}}$ we update the aggregate errors and the aggregate subgradients according to relations (14a) and (14b), then $\widehat{e}_j^{k+1} \geq 0$ and $\widehat{\xi}_j^{k+1} \in \partial_{\widehat{e}_j^{k+1}} H_j(x^{k+1}, x^{k+1})$.*

Proof (i) Let $j \in \mathcal{L}_l^{\text{oracle}}$, we have

$$\begin{aligned}
 e_{f_j}^{k+1} &= e_{f_j}^k + f(x^{k+1}) - f(x^k) - \langle \xi_f^j, x^{k+1} - x^k \rangle \\
 &= f(x^k) - f(y^j) - \langle \xi_f^j, x^k - y^j \rangle + f(x^{k+1}) - f(x^k) - \langle \xi_f^j, x^{k+1} - x^k \rangle \\
 &= f(x^{k+1}) - f(y^j) - \langle \xi_f^j, x^{k+1} - y^j \rangle,
 \end{aligned}$$

and the same argument applies for $e_{g_j}^{k+1}$.

(ii) Let $j \in \mathcal{L}_l^{\text{agg}}$. Since $\widehat{\xi}_j^k \in \partial_{\widehat{e}_j^k} H_j(x^k, x^k)$ and $H_j(\cdot, x^k)$ is convex, it follows that $H_j(y, x^k) \geq H_j(x^k, x^k) + \langle \widehat{\xi}_j^k, y - x^k \rangle - \widehat{e}_j^k$ for all $y \in \mathbb{R}^n$. By using (14a), (14b) and (21), we obtain

$$\begin{aligned}
 H_j(y, x^{k+1}) &\geq H_j(y, x^k) - (f(x^{k+1}) - f(x^k))^+ + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle \\
 &\quad - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2 \\
 &\geq H_j(x^k, x^k) + \langle \widehat{\xi}_j^k, y - x^k \rangle - \widehat{e}_j^k - (f(x^{k+1}) - f(x^k))^+ \\
 &\quad + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle - \frac{\eta_l}{2} \|x^{k+1} - x^k\|^2
 \end{aligned}$$

$$\begin{aligned}
 &= \langle \widehat{\xi}_j^k, y - x^k \rangle + g^+(x^{k+1}) - \widehat{e}_j^{k+1} + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle \\
 &\quad - \langle \widehat{\xi}_j^k, x^{k+1} - x^k \rangle \\
 &= g^+(x^{k+1}) - \widehat{e}_j^{k+1} + \eta_l \langle x^k - x^{k+1}, y - x^{k+1} \rangle + \langle \widehat{\xi}_j^k, y - x^{k+1} \rangle \\
 &= g^+(x^{k+1}) + \langle \widehat{\xi}_j^k + \eta_l(x^k - x^{k+1}), y - x^{k+1} \rangle - \widehat{e}_j^{k+1}. \tag{22}
 \end{aligned}$$

By the convexity of $H_l(\cdot, x^{k+1})$, we get $\widehat{\xi}_j^{k+1} \in \partial_{\widehat{e}_j^{k+1}} H_j(x^{k+1}, x^{k+1})$. In particular, set $y := x^{k+1}$ in (22) and using this fact that $H_j(x^{k+1}, x^{k+1}) = g^+(x^{k+1})$ we deduce $\widehat{e}_j^{k+1} \geq 0$. □

Remark 4 In the following we assume that $\{\eta_l\}$ is bounded. From (7) it is clear that $\eta_l \geq 2\omega$ for all l , so we suppose that there exists $\bar{\eta}$ such that $2\omega \leq \eta_l \leq \bar{\eta}$ for all l . Since the analysis of the algorithm do not require the constant $\bar{\eta}$ to be known, this assumption is not restrictive on implementation of the algorithm. The boundedness of $\{\eta_l\}$ has been established in [14] for the unconstrained optimization problems with lower $-C^1$ objective functions and in [25] for the constrained optimization problems with lower $-C^1$, objective and constraint functions, respectively. However, in our convergence analysis we consider this assumption with regular functions which is weaker than lower $-C^1$ assumption.

3.4 Upper envelope model

The argument of this paper are actually two-fold. The first line is as follows. For nonconvex problem (1), we use the improvement function to get rid of the constraint and use the term $\frac{\eta}{2} \|\cdot - x\|^2$ to deal with potentially negative linearization errors. We do not require any lower $-C^2$, lower $-C^1$ or prox-regular assumption therefore the augmented function (4) can be convex or nonconvex. Then we state the convex cutting plane model $H_l(\cdot, x^k)$ for (4) and use the usual bundle argument for this model. The second line of argument deals with connecting the convex cutting plane model $H_l(\cdot, x^k)$ with the original nonconvex problem to analysis the convergence of the method. For this purpose, motivated by [30], we define the upper envelope model $M^\uparrow(y, x)$ associated with the cutting plane for constrained problems. It is worth mentioning that the upper envelope model in [30] was defined for the unconstrained optimization problems. Furthermore, the method of [30] employs the ‘‘downshift’’ mechanism, that is absolutely different from our mechanism for the definition of the cutting planes.

Suppose that $\bar{B}_R(x)$ is a fixed closed ball large enough such that contains all possible trial steps y^+ (this assumption was considered in [7,30]). Set $\mathcal{B}(x) := \{y \mid y \in \bar{B}_R(x), y \text{ is a trial point}\}$. We define the upper envelope model $M^\uparrow(\cdot, x): \mathbb{R}^n \rightarrow \mathbb{R}$, for a given point $x \in \mathbb{R}^n$, as

$$M^\uparrow(y, x) := g^+(x) + \sup_{\substack{2\omega \leq \eta \leq \bar{\eta}, y^+ \in \mathcal{B}(x) \\ \xi_f \in \partial_c f(y^+), \xi_g \in \partial_c g(y^+)}} \left\{ m_{y^+, \xi_f, \eta}^f(y, x), m_{y^+, \xi_g, \eta}^g(y, x) \right\},$$

where $2\omega \leq \eta \leq \bar{\eta}$ with $\bar{\eta}$ defined in Remark 4. The plane $m_{y^+, \xi_f, \eta}^f(y, x)$ is the cutting plane at the serious iterate x and the trial step y^+ as follows

$$\begin{aligned} m_{y^+, \xi_f, \eta}^f(y, x) &:= -c_{y^+} + \langle \xi_f + \eta(y^+ - x), y - x \rangle \\ &= -\frac{\omega}{2} \|y^+ - x\|^2 + \langle \xi_f + \eta(y^+ - x), y - x \rangle. \end{aligned} \tag{23}$$

Also $m_{y^+, \xi_g, \eta}^g(y, x)$ is the cutting plane at the serious iterate x and the trial step y^+ as

$$\begin{aligned} m_{y^+, \xi_g, \eta}^g(y, x) &:= -c_{y^+} + \langle \xi_g + \eta(y^+ - x), y - x \rangle \\ &= -\frac{\omega}{2} \|y^+ - x\|^2 + \langle \xi_g + \eta(y^+ - x), y - x \rangle. \end{aligned} \tag{24}$$

The boundedness of $\bar{B}_R(x)$ and the definition of η imply that $M^\uparrow(\cdot, x)$ is defined everywhere. In the following we state some properties of the upper envelope model $M^\uparrow(\cdot, x)$. These results can be found in [7, Lemma 6.1], however we state the proof since the definition of the upper envelope model $M^\uparrow(\cdot, x)$, its cutting planes and details of the proof are different from [7].

Lemma 5 *Suppose that f and g are regular functions, then the following hold:*

- (i) $M^\uparrow(\cdot, x)$ is a convex function.
- (ii) $H_l(\cdot, x) \leq M^\uparrow(\cdot, x)$ for all l .
- (iii) $M^\uparrow(x, x) = g^+(x)$.
- (iv) $\partial M^\uparrow(x, x) \subseteq \partial_c h(x, x)$.

Proof The proof of (i) is obvious. To prove (ii), from (6a), for all $j \in \mathcal{L}_l$, we have $c_{f_j}^k = e_{f_j}^k + g^+(x^k) + \eta_l b_j^k$. By using (7), we obtain

$$e_{f_j}^k + g^+(x^k) + \eta_l b_j^k \geq \frac{\omega}{2} \|y_j - x^k\|^2,$$

therefore for all $j \in \mathcal{L}_l$ we have $c_{f_j}^k \geq \frac{\omega}{2} \|y_j - x^k\|^2$. The same argument holds for g and we again get $c_{g_j}^k \geq \frac{\omega}{2} \|y_j - x^k\|^2$. Therefore we deduce $-c_{f_j}^k \leq -\frac{\omega}{2} \|y_j - x^k\|^2$ and $-c_{g_j}^k \leq -\frac{\omega}{2} \|y_j - x^k\|^2$. Compare the definition of $H_l(\cdot, x^k)$ in (5) with the definition of $M^\uparrow(\cdot, x)$ and their cutting planes [see relations (6a), (6b), (23) and (24)]. The second term in their cutting planes is similar, i.e., $s_{f_j}^k$ with $\xi_f + \eta(y^+ - x)$ and $s_{g_j}^k$ with $\xi_g + \eta(y^+ - x)$. Therefore

$$-c_{f_j}^k + \langle s_{f_j}^k, y - x^k \rangle \leq -\frac{\omega}{2} \|y_j - x^k\|^2 + \langle \xi_f^j + \eta_l(y_j - x^k), y - x^k \rangle,$$

and

$$-c_{g_j}^k + \langle s_{g_j}^k, y - x^k \rangle \leq -\frac{\omega}{2} \|y_j - x^k\|^2 + \langle \xi_g^j + \eta_l(y_j - x^k), y - x^k \rangle.$$

Since we assume $\overline{B}_R(x)$ is large enough such that it contains all possible trial steps, we get $H_l(\cdot, x) \leq M^\uparrow(\cdot, x)$ for all l and the proof of (ii) is completed.

We have $c_{y^+} = \frac{\omega}{2} \|y^+ - x\|^2 \geq 0$ and $c_x = 0$, therefore we obtain (iii)

$$M^\uparrow(x, x) = g^+(x) + \sup_{y^+ \in \overline{B}_R(x)} \{-c_{y^+}\} = g^+(x).$$

To prove (iv). Let $\overline{\xi} \in \partial M^\uparrow(x, x)$ and define $\overline{m}(\cdot, x) = M^\uparrow(x, x) + \langle \overline{\xi}, \cdot - x \rangle = g^+(x) + \langle \overline{\xi}, \cdot - x \rangle$, the tangent plane to the graph of $M^\uparrow(\cdot, x)$ at x associated with $\overline{\xi}$. By using convexity, we have $\overline{m}(\cdot, x) \leq M^\uparrow(\cdot, x)$. Fix a vector $d \in \mathbb{R}^n$ and consider the values $M^\uparrow(x + td, x)$ for $t > 0$. According to the definition of $M^\uparrow(\cdot, x)$ for any $t > 0$, there exist $2\omega \leq \eta_t \leq \overline{\eta}$, $y_t \in \overline{B}_R(x)$ and $\xi_t^f \in \partial_c f(y_t)$ or $\xi_t^g \in \partial_c g(y_t)$ such that $M^\uparrow(x + td, x) \leq g^+(x^k) + m_{y_t, \xi_t^f, \eta_t}^f(x + td, x) + t^2$ or $M^\uparrow(x + td, x) \leq g^+(x^k) + m_{y_t, \xi_t^g, \eta_t}^g(x + td, x) + t^2$.

We suppose that $M^\uparrow(x + td, x) \leq g^+(x^k) + m_{y_t, \xi_t^f, \eta_t}^f(x + td, x) + t^2$, therefore

$$\begin{aligned} \overline{m}(x + td, x) &\leq g^+(x^k) + m_{y_t, \xi_t^f, \eta_t}^f(x + td, x) + t^2 \\ &= g^+(x) - c_{y_t} + \langle \xi_t^f + \eta_t(y_t - x), td \rangle + t^2 \\ &= g^+(x) - \frac{\omega}{2} \|y_t - x\|^2 + \langle \xi_t^f + \eta_t(y_t - x), td \rangle + t^2. \end{aligned}$$

When $M^\uparrow(x + td, x) \leq g^+(x^k) + m_{y_t, \xi_t^g, \eta_t}^g(x + td, x) + t^2$, we again get the above relation with $\xi_t^g \in \partial_c g(y^t)$. Therefore, we can say

$$\overline{m}(x + td, x) \leq g^+(x) - \frac{\omega}{2} \|y_t - x\|^2 + \langle \xi_t + \eta_t(y_t - x), td \rangle + t^2, \tag{25}$$

where $\xi_t = \xi_t^f$ or ξ_t^g . Since $\xi_t^f \in \partial_c f(y_t)$, $\xi_t^g \in \partial_c g(y_t)$ and f, g are locally Lipschitz on $\overline{B}_R(x)$, hence $\{\xi_t\}$ is bounded. Therefore $\{\xi_t + \eta_t(y_t - x)\}$ is bounded and since d is an arbitrary fixed vector, then $\langle \xi_t + \eta_t(y_t - x), td \rangle \rightarrow 0$ as $t \rightarrow 0$. Passing the limit in (25), we get $\lim_{t \rightarrow 0} -\frac{\omega}{2} \|y_t - x\|^2 \geq 0$ which implies that $\lim_{t \rightarrow 0} \frac{\omega}{2} \|y_t - x\|^2 = 0$ and we deduce $y_t \rightarrow x$. Consider the sequence $\{\xi_t\}$, since f and g are regular, we have $\xi_t \in \partial_c h(y_t, x)$. As we have mentioned above, this sequence is bounded on $\overline{B}_R(x)$, passing to a subsequence if necessary, we may assume there exists $\widehat{\xi}$ such that $\xi_t \rightarrow \widehat{\xi}$. Since $y_t \rightarrow x$, it follows from the upper semicontinuity of the Clarke subdifferential that $\widehat{\xi} \in \partial_c h(x, x)$. From (25), we obtain $\langle \widehat{\xi}, d \rangle \leq \langle \xi_t + \eta_t(y_t - x), d \rangle + t$, let $t \rightarrow 0$ which implies that $\langle \widehat{\xi}, d \rangle \leq \langle \widehat{\xi}, d \rangle$. Thus

$$\langle \overline{\xi}, d \rangle \leq \langle \widehat{\xi}, d \rangle \leq \max\{\langle \xi, d \rangle, \xi \in \partial_c h(x, x)\} = (h(\cdot, x))^\circ(x, d).$$

Since d is an arbitrary vector, we get $\overline{\xi} \in \partial_c h(x, x)$ and the proof is completed. \square

3.5 Statement of the algorithm

Now we are in position to state our infeasible proximal bundle algorithm for solving problem (1).

Algorithm 1 (*Infeasible Constrained Proximal Bundle Method*)

Step 1 Choose parameters $m \in (0, 1)$, $\omega > 0$, $tol \geq 0$, $|\mathcal{L}|_{\max} \geq 3$. Choose $x^0 \in \mathbb{R}^n$, set $y^0 := x^0$, and compute $(f_0, g_0, \xi_f^0, \xi_g^0)$. Set $k := 0$, $l := 0$, $e_{f_0}^0 := 0$, $e_{g_0}^0 := 0$, $b_0^0 := 0$, $d_0^0 := 0$, $\mathcal{B}_l^{\text{oracle}} := \{(f_0, g_0, \xi_f^0, \xi_g^0, e_{f_0}^0, e_{g_0}^0, b_0^0, d_0^0)\}$, $\mathcal{B}_l^{\text{agg}} := \emptyset$, $\mathcal{L}_l^{\text{oracle}} := \{0\}$ and $\mathcal{L}_l^{\text{agg}} := \emptyset$.

Step 2 Select $\eta_l > 0$ as in (7) by replacing \mathcal{L}_l with $\mathcal{L}_l^{\text{oracle}}$. By using (10) set $H_l(y, x^k)$. Choose $\mu_l > 0$ and compute y^l as the solution to

$$\min_{y \in X} H_l(y, x^k) + \frac{\mu_l}{2} \|y - x^k\|^2.$$

Set $\widehat{\xi}_l^k + v_l := \mu_l(x^k - y^l)$, $\widehat{e}_l^k := g^+(x^k) - H_l(y^l, x^k) - \langle \widehat{\xi}_l^k, x^k - y^l \rangle$, $\delta^l := \widehat{e}_l^k + \frac{1}{2\mu_l} \|\widehat{\xi}_l^k + v_l\|^2$ and $(f_l, g_l, \xi_f^l, \xi_g^l, e_{f_l}^k, e_{g_l}^k, b_l^k, d_l^k)$.

Step 3 If $\delta^l \leq tol$, stop.

Step 4 Compute $h_l := h(y^l, x^k)$. If $h_l \leq g^+(x^k) - m\delta^l$, then y^l is a serious iterate. Otherwise, y^l is a null iterate.

Step 5 If $|\mathcal{L}_l^{\text{oracle}} \cup \mathcal{L}_l^{\text{agg}}| \geq |\mathcal{L}|_{\max}$, then delete at least two elements from $\mathcal{L}_l^{\text{oracle}} \cup \mathcal{L}_l^{\text{agg}}$ except $l(k)$, i.e., the element corresponding to x^k . In addition delete their corresponding elements from $\mathcal{B}_l^{\text{oracle}} \cup \mathcal{B}_l^{\text{agg}}$. Let $\mathcal{L}_{l+1}^{\text{oracle}} := \mathcal{L}_l^{\text{oracle}} \cup \{l\}$, $\mathcal{L}_{l+1}^{\text{agg}} := \mathcal{L}_l^{\text{agg}} \cup \{l\}$ and

$$\begin{aligned} \mathcal{B}_{l+1}^{\text{oracle}} &:= \mathcal{B}_l^{\text{oracle}} \cup \{(f_l, g_l, \xi_f^l, \xi_g^l, e_{f_l}^k, e_{g_l}^k, b_l^k, d_l^k)\}, \\ \mathcal{B}_{l+1}^{\text{agg}} &:= \mathcal{B}_l^{\text{agg}} \cup \{(\widehat{e}_l^k, \widehat{\xi}_l^k)\}. \end{aligned}$$

Step 6 If y^l is a serious iterate, set $x^{k+1} := y^l$, $f(x^{k+1}) := f_l$, $g(x^{k+1}) := g_l$. Update the bundle elements according to (13a)–(13d) for all $j \in \mathcal{L}_{l+1}^{\text{oracle}}$ and to (14a)–(14b) for all $j \in \mathcal{L}_{l+1}^{\text{agg}}$. Set $k = k + 1$.

Step 7 Set $l = l + 1$ and go to Step 2.

Remark 5 It is worth mentioning that contrary to some nonconvex constrained optimization algorithms that solve a quadratic programming subproblem and also use a line search method, e.g., [5,27,28,36], our setting does not employ any line search subroutine and it only solves a quadratic programming subproblem, without impairing its global convergence.

Remark 6 We require a usual rule for choosing the proximal parameter μ_l . The conditions that μ_l should be satisfied, for the global convergence, are very simple and we state these in the convergence results (see Sect. 4).

4 Convergence analysis

In this section, we establish the global convergence of Algorithm 1. We assume from now on that the stopping tolerance tol is set to zero. Note that, if the algorithm stops at Step 3, then $\delta^l = 0$. Indeed, $\delta^l = 0$ implies that $\|\xi_l^k + v_l\| = 0$ and $\widehat{e}_l^k = 0$. Since $\widehat{\xi}_l^k \in \partial_{\widehat{e}_l^k} H_l(x^k, x^k)$, we have

$$H_l(y, x^k) \geq H_l(x^k, x^k) + \langle \widehat{\xi}_l^k, y - x^k \rangle - \widehat{e}_l^k = g^+(x^k) + \langle \widehat{\xi}_l^k, y - x^k \rangle - \widehat{e}_l^k, \quad (26)$$

for all $y \in \mathbb{R}^n$. By $v_l \in \partial I_X(y^l)$, we get

$$\begin{aligned} I_X(y) &\geq I_X(y^l) + \langle v_l, y - y^l \rangle \\ &= \langle v_l, y - y^l \rangle + I_X(x^k) + \langle v_l, y - x^k \rangle + \langle v_l, x^k - y^l \rangle \\ &= I_X(x^k) + \langle v_l, y - x^k \rangle + \langle v_l, \frac{1}{\mu_l} (\widehat{\xi}_l^k + v_l) \rangle, \end{aligned} \quad (27)$$

for all $y \in \mathbb{R}^n$. By (26) and (27), we obtain

$$\begin{aligned} H_l(y, x^k) + I_X(y) &\geq g^+(x^k) + \langle \widehat{\xi}_l^k, y - x^k \rangle - \widehat{e}_l^k + I_X(x^k) + \langle v_l, y - x^k \rangle \\ &\quad + \left\langle v_l, \frac{1}{\mu_l} (\widehat{\xi}_l^k + v_l) \right\rangle \\ &= g^+(x^k) + I_X(x^k) + \langle \widehat{\xi}_l^k + v_l, y - x^k \rangle - \widehat{e}_l^k + \left\langle v_l, \frac{1}{\mu_l} (\widehat{\xi}_l^k + v_l) \right\rangle. \end{aligned} \quad (28)$$

We have $\|\widehat{\xi}_l^k + v_l\| = \widehat{e}_l^k = 0$ by using this fact in relation (28) we deduce

$$H_l(y, x^k) + I_X(y) \geq g^+(x^k) + I_X(x^k) + \langle 0, y - x^k \rangle.$$

Due to Lemma 5 (ii) and (iii), we get

$$M^\uparrow(y, x^k) + I_X(y) \geq M^\uparrow(x^k, x^k) + I_X(x^k) + \langle 0, y - x^k \rangle,$$

and by the convexity of $M^\uparrow(\cdot, x^k)$ and $I_X(\cdot)$, we deduce $0 \in \partial M^\uparrow(x^k, x^k) + \partial I_X(x^k)$. Using Lemma 5 (iv) we obtain $0 \in \partial_c h(x^k, x^k) + \partial I_X(x^k)$. Hence when Algorithm 1 stops finitely, the last serious iterate x^k is stationary for the improvement function $h(\cdot, x^k)$ on X .

From now on, we assume that Algorithm 1 does not stop and generates an infinite sequence of iterates. As customary in the bundle method, we consider two cases: the serious iterates sequence $\{x^k\}$ is either finite or infinite. Set $\mathcal{A} := \{l \mid l \text{ is a serious iteration}\}$ collects the indices of the serious iterates in the sequence $\{y^l\}$. We denote the feasible set of problem (1) by $\mathcal{D} := \{x \in X \mid g(x) \leq 0\}$.

Proposition 1 *Suppose that k_1 is any serious iteration, then for all $k \geq k_1$ we get $x^k \in \{x \in \mathbb{R}^n, g(x) \leq g^+(x^{k_1})\}$. Therefore, if there exists k_2 such that $x^{k_2} \in \mathcal{D}$, then $x^k \in \mathcal{D}$ for all $k \geq k_2$.*

Proof By using Remark 2 for any k we have $g(x^{k+1}) < g(x^k)$. □

From Proposition 1, we obtain immediately that if x^0 is a feasible point, then the sequence of the serious iterates $\{x^k\}$ is feasible too.

Lemma 6 *Suppose that f is bounded below on \mathcal{D} , and Algorithm 1 generates an infinite number of serious iterates. Then $\{\delta^l\}_{l \in \mathcal{A}} \rightarrow 0$ and $\{\widehat{e}_l^k\}_{l \in \mathcal{A}} \rightarrow 0$. Furthermore if there exists $\bar{\mu} > 0$ such that $\mu_l \leq \bar{\mu}$ for all $l \in \mathcal{A}$, then $\{\widehat{\xi}_l^k + v_l\}_{l \in \mathcal{A}} \rightarrow 0$.*

Proof By Proposition 1, we consider two cases: either $x^k \notin \mathcal{D}$ for all k or there exists an index k_3 such that $x^k \in \mathcal{D}$ for all $k \geq k_3$.

- (i) If $x^k \notin \mathcal{D}$ for all k , by using (12) we have $m\delta^{l(k+1)} \leq g(x^k) - g(x^{k+1})$ for all $k \geq 0$. Therefore $\{g(x^k)\}$ is decreasing and $g(x^k) > 0$. Thus this sequence is decreasing and bounded below which implies that, there exists \bar{g} such that $g_{l(k)} \rightarrow \bar{g}$ as $l(k) \in \mathcal{A}$ and $k \rightarrow \infty$. Therefore $\sum_{l \in \mathcal{A}} \delta^l \leq \frac{g(x_0) - \bar{g}}{m} < \infty$.
- (ii) Now suppose that, there exists k_3 such that if $k < k_3$, then $x^k \notin \mathcal{D}$ and for all $k \geq k_3$ we have $x^k \in \mathcal{D}$. By using (12) we have $m\delta^{l(k+1)} \leq f(x^k) - f(x^{k+1})$ for all $k \geq k_3$. Hence $\{f(x^k)\}$ is decreasing for $k \geq k_3$ and since f is bounded below on \mathcal{D} there exists \bar{f} such that $f(x^k) \rightarrow \bar{f}$ as $l(k) \in \mathcal{A}$ and $k \rightarrow \infty$. This implies

$$\sum_{l \in \mathcal{A}} \delta^l = \sum_{l \in \mathcal{A}, l < l(k_3)} \delta^l + \sum_{l \in \mathcal{A}, l \geq l(k_3)} \delta^l \leq \frac{1}{m}(g(x^0) - g(x^{k_3}) + f(x^{k_3}) - \bar{f}) < \infty.$$

Hence in each case we have $\sum_{l \in \mathcal{A}} \delta^l < \infty$, which implies that $\{\delta^l\}_{l \in \mathcal{A}} \rightarrow 0$. By the definition of δ^l we have $\widehat{e}_l^k \leq \delta^l$ and $\frac{1}{2\mu_l} \|\widehat{\xi}_l^k + v_l\|^2 \leq \delta^l$, it immediately follows that $\{\widehat{e}_l^k\}_{l \in \mathcal{A}} \rightarrow 0$ and by $\mu_l \leq \bar{\mu}$, we have $\{\widehat{\xi}_l^k + v_l\}_{l \in \mathcal{A}} \rightarrow 0$. Hence the proof is completed. □

In what follows, we assume that Algorithm 1 generates an infinite sequence of serious iterates and show that every accumulation point of this sequence is a stationary point of the improvement function. Here, motivated by Lemma 1, we call x^* is a stationary point of the constraint violation if $0 \in \partial_c g(x^*)$ along with $g(x^*) > 0$.

Theorem 1 *Suppose that f and g are regular functions, f is bounded below on \mathcal{D} and $\{\eta_l\}$ is bounded above. Assume that Algorithm 1 generates an infinite sequence of serious iterates. If there exists $\bar{\mu} > 0$ such that $\mu_l \leq \bar{\mu}$, then for every accumulation point of $\{x^k\}$, i.e., x^* we have $0 \in \partial_c h(x^*, x^*) + \partial I_X(x^*)$, that is x^* is a stationary point of the improvement function on X . In particular, x^* is either a stationary point of the constraint violation or a Fritz John point or a KKT point of problem (1).*

Proof By using Lemma 6 we have

$$\{\delta^l\}_{l \in \mathcal{A}} \rightarrow 0, \quad \{\widehat{e}_l^k\}_{l \in \mathcal{A}} \rightarrow 0 \quad \text{and} \quad \{\widehat{\xi}_l^k + v_l\}_{l \in \mathcal{A}} \rightarrow 0. \tag{29}$$

Suppose that $l \in \mathcal{A}$, since $\widehat{\xi}_l^k + v_l \in \partial H_l(y^l, x^k) + \partial I_X(y^l)$ and by the convexity of $H_l(\cdot, x^k) + I_X(\cdot)$ for all $y \in \mathbb{R}^n$, we obtain

$$\begin{aligned} M^\uparrow(y, x^k) + I_X(y) &\geq H_l(y, x^k) + I_X(y) \\ &\geq H_l(y^l, x^k) + I_X(y^l) + \langle \widehat{\xi}_l^k + v_l, y - y^l \rangle \\ &= g^+(x^k) - \widehat{e}_l^k + \langle \widehat{\xi}_l^k, y^l - x^k \rangle + I_X(y^l) + \langle \widehat{\xi}_l^k + v_l, y - y^l \rangle \\ &= g^+(x^k) - \widehat{e}_l^k + \langle \widehat{\xi}_l^k + v_l, y - x^k \rangle + \langle v_l, x^k - y^l \rangle \\ &= g^+(x^k) - \widehat{e}_l^k + \langle \widehat{\xi}_l^k + v_l, y - x^k \rangle + \left\langle v_l, \frac{1}{\mu_l} (\widehat{\xi}_l^k + v_l) \right\rangle, \end{aligned} \tag{30}$$

where the first inequality comes from Lemma 5 (ii), the first equality follows from (9) and the last equality from $x^k - y^l = \frac{1}{\mu_l} (\widehat{\xi}_l^k + v_l)$.

Since $\{x^k\}$ is in the compact set X , it has a subsequence $x_{k_l} \rightarrow_{k \in \mathcal{K}} x^*$ with $x^* \in X$. We know $l \in \mathcal{A}$ so that $l = l(k)$. Since $x^k \rightarrow x^*$ and $x^k - y^l = \frac{1}{\mu_l} (\widehat{\xi}_l^k + v_l) \rightarrow 0$, therefore $y^l \rightarrow x^*$ as $k \in \mathcal{K}$, $l(k) \rightarrow \infty$. On the other hand $v_l \in \partial I_X(y^l)$ and X is a convex compact set hence there exists $\bar{\varepsilon}$ such that $\{v_l\} \subseteq \partial I_X(x^*) + \overline{B_{\bar{\varepsilon}}}(0)$. Consider (30), (29) for all $k \in \mathcal{K}$, passing to the limit and by the boundedness of $\{v_l\}$, we obtain $\lim_{k \rightarrow \infty, k \in \mathcal{K}} M^\uparrow(y, x^k) + I_X(y) \geq \lim_{k \rightarrow \infty, k \in \mathcal{K}} g^+(x^k) + \langle 0, y - x^k \rangle$ and hence

$$M^\uparrow(y, x^*) + I_X(y) \geq g^+(x^*) + \langle 0, y - x^* \rangle = M^\uparrow(x^*, x^*) + I_X(x^*) + \langle 0, y - x^* \rangle.$$

Using the convexity of $M^\uparrow(\cdot, x^*) + I_X(\cdot)$, we deduce $0 \in \partial M^\uparrow(x^*, x^*) + \partial I_X(x^*)$ and by Lemma 5 (iv) we have $0 \in \partial_c h(x^*, x^*) + \partial I_X(x^*)$.

When $g(x^*) > 0$, then $0 \in \partial_c g(x^*) + \partial I_X(x^*)$ and, we get x^* is a stationary point of the constraint violation. If $g(x^*) = 0$, then x^* is a Fritz John point. Otherwise, x^* is a KKT point of problem (1). □

We conclude by considering the case when the number of serious iterates is finite, i.e., there exists $l^* := \max\{l \mid l \in \mathcal{A}\}$. We denote the corresponding last serious iterate by $x^{end} := x^{k(l^*)} = y^{l^*}$. Then for all $l \geq l^*$ the iterate x^{end} is fixed. We show that x^{end} is a stationary point of the improvement function on X .

Theorem 2 *Assume that f and g are regular functions and Algorithm 1 takes a finite number of serious iterates. If $\{\eta_l\}$ is bounded above and $\mu_l \leq \mu_{l+1} \leq \bar{\mu}$ for all $l \geq l^*$, then $0 \in \partial_c h(x^{end}, x^{end}) + \partial I_X(x^{end})$, i.e., x^{end} is a stationary point of the improvement function on X . In particular, x^* is either a stationary point of the constraint violation or a Fritz John point or a KKT point of problem (1).*

Proof Suppose that l is large enough, such that $l \geq l^*$. Let y^l be the optimal solution of (8). We show that the sequence $\{H_l(y^l, x^{end}) + \frac{\mu_l}{2} \|y^l - x^{end}\|^2\}$ is bounded above

and nondecreasing. Since $\widehat{\xi}_l^{end} \in \partial H_l(y^l, x^{end})$ and $H_l(\cdot, x^{end})$ is convex, we have $H_l(y, x^{end}) \geq H_l(y^l, x^{end}) + \langle \widehat{\xi}_l^{end}, y - y^l \rangle$ for every $y \in \mathbb{R}^n$. By using the definition of the aggregate plane and (9) we have $m_l^*(y, x^{end}) = H_l(y^l, x^{end}) + \langle \widehat{\xi}_l^{end}, y - y^l \rangle$. Substituting $y = x^{end}$, we get $H_l(y^l, x^{end}) = m_l^*(x^{end}, x^{end}) - \langle \widehat{\xi}_l^{end}, x^{end} - y^l \rangle$. Then we obtain

$$\begin{aligned} &H_l(y^l, x^{end}) + \frac{\mu_l}{2} \|y^l - x^{end}\|^2 \\ &\leq H_l(y^l, x^{end}) + \mu_l \|y^l - x^{end}\|^2 \\ &\leq H_l(y^l, x^{end}) + \mu_l \langle y^l - x^{end}, y^l - x^{end} \rangle - \langle v_l, x^{end} - y^l \rangle \\ &= H_l(y^l, x^{end}) + \langle \widehat{\xi}_l^{end}, x^{end} - y^l \rangle \\ &= m_l^*(x^{end}, x^{end}) \leq H_{l+1}(x^{end}, x^{end}) = g^+(x^{end}), \end{aligned}$$

where the first inequality is clear, and the second inequality follows from $v_l \in \partial I_X(y^l)$, the first equality holds by the definition of $\widehat{\xi}_l^{end}$, the second equality is satisfied by the definition of $m_l^*(\cdot, x^{end})$ and the last inequality by the definition of $H_{l+1}(\cdot, x^{end})$, hence the sequence is bounded above. Next let us prove that this sequence is nondecreasing

$$\begin{aligned} &H_{l+1}(y^{l+1}, x^{end}) + \frac{\mu_{l+1}}{2} \|y^{l+1} - x^{end}\|^2 \\ &\geq m_l^*(y^{l+1}, x^{end}) + \frac{\mu_{l+1}}{2} \|y^{l+1} - x^{end}\|^2 \\ &\geq m_l^*(y^{l+1}, x^{end}) + \frac{\mu_l}{2} \|y^{l+1} - x^{end}\|^2 \\ &= H_l(y^l, x^{end}) + \langle \widehat{\xi}_l^{end}, y^{l+1} - y^l \rangle + \frac{\mu_l}{2} \|y^{l+1} - x^{end}\|^2 \\ &= H_l(y^l, x^{end}) + \langle \mu_l(x^{end} - y^l) - v_l, y^{l+1} - y^l \rangle + \frac{\mu_l}{2} \|y^{l+1} - x^{end}\|^2 \\ &\geq H_l(y^l, x^{end}) + \langle \mu_l(x^{end} - y^l), y^{l+1} - y^l \rangle + \frac{\mu_l}{2} \|y^{l+1} - x^{end}\|^2 \\ &= H_l(y^l, x^{end}) + \frac{\mu_l}{2} (\|y^l - x^{end}\|^2 + \|y^{l+1} - y^l\|^2) \\ &\geq H_l(y^l, x^{end}) + \frac{\mu_l}{2} \|y^l - x^{end}\|^2. \end{aligned}$$

Where the first inequality holds by the definition of $H_{l+1}(\cdot, x^{end})$, the second inequality by $\mu_{l+1} \geq \mu_l$ and the first equality by the definition $m_l^*(\cdot, x^{end})$ and the third inequality follows from $v_l \in \partial I_X(y^l)$. Therefore the sequence $\{H_l(y^l, x^{end}) + \frac{\mu_l}{2} \|y^l - x^{end}\|^2\}$ is nondecreasing and by its boundedness, we obtain this sequence is convergent. Passing to the limit in the above inequality we have $\lim_{l \rightarrow \infty} \frac{\mu_l}{2} \|y^{l+1} - y^l\|^2 \leq 0$. By assumption that $\mu_l \leq \mu_{l+1} \leq \bar{\mu}$, we obtain $\mu_l \geq \mu_l^*$ that is $\{\mu_l\}$ is bounded below by μ_l^* . This implies that $y^{l+1} - y^l \rightarrow 0$, therefore $\{y^l\}$ is bounded.

By using the definition of δ^l we have

$$\begin{aligned}
 \delta^l &= \widehat{e}_l^{end} + \frac{1}{2\mu_l} \|\widehat{\xi}_l^{end} + v_l\|^2 \\
 &= g^+(x^{end}) - H_l(y^l, x^{end}) - \langle \widehat{\xi}_l^{end}, x^{end} - y^l \rangle + \frac{1}{2\mu_l} \|\widehat{\xi}_l^{end} + v_l\|^2 \\
 &\leq g^+(x^{end}) - H_l(y^l, x^{end}) - \langle \widehat{\xi}_l^{end}, x^{end} - y^l \rangle + \frac{1}{\mu_l} \|\widehat{\xi}_l^{end} + v_l\|^2 \\
 &= g^+(x^{end}) - H_l(y^l, x^{end}) - \langle \widehat{\xi}_l^{end}, x^{end} - y^l \rangle + \left\langle \frac{1}{\mu_l} (\widehat{\xi}_l^{end} + v_l), \widehat{\xi}_l^{end} + v_l \right\rangle \\
 &= g^+(x^{end}) - H_l(y^l, x^{end}) - \langle \widehat{\xi}_l^{end}, x^{end} - y^l \rangle + \langle x^{end} - y^l, \widehat{\xi}_l^{end} + v_l \rangle \\
 &\leq g^+(x^{end}) - H_l(y^l, x^{end}) + \langle x^{end} - y^l, v_l \rangle \\
 &\leq g^+(x^{end}) - H_l(y^l, x^{end}),
 \end{aligned}$$

where the first equality holds by the definition of \widehat{e}_l^{end} in (9), the last inequality follows from $v_l \in \partial I_X(y^l)$ and $\langle v_l, x^{end} - y^l \rangle \leq 0$ and the other relations are obvious. By considering above inequality for $l + 1$, we obtain

$$\delta^{l+1} \leq g^+(x^{end}) - H_{l+1}(y^{l+1}, x^{end}). \tag{31}$$

Since l is a null iteration, it follows that (11) is not satisfied. We observe that either $f(y) - f(x^{end}) > g^+(x^{end}) - m\delta^l$ or $g(y) > g^+(x^{end}) - m\delta^l$.

(I) Without loss of generality we suppose that $f(y) - f(x^{end}) > g^+(x^{end}) - m\delta^l$. Therefore we get

$$\begin{aligned}
 -c_{f_l}^{end} + \langle s_{f_l}^{end}, y^l - x^{end} \rangle &= -\left(e_{f_l}^{end} + g^+(x^{end}) + \frac{\eta_l}{2} \|y^l - x^{end}\|^2 \right) \\
 &\quad + \langle \xi_f^l + \eta_l(y^l - x^{end}), y^l - x^{end} \rangle \\
 &= -f(x^{end}) + f(y^l) + \langle \xi_f^l, x^{end} - y^l \rangle - g^+(x^{end}) \\
 &\quad + \langle \xi_f^l, y^l - x^{end} \rangle - \frac{\eta_l}{2} \|y^l - x^{end}\|^2 + \eta_l \|y^l - x^{end}\|^2 \\
 &= f(y^l) - f(x^{end}) - g^+(x^{end}) + \frac{\eta_l}{2} \|y^l - x^{end}\|^2 \\
 &> g^+(x^{end}) - m\delta^l - g^+(x^{end}) + \frac{\eta_l}{2} \|y^l - x^{end}\|^2 \\
 &\geq -m\delta^l,
 \end{aligned} \tag{32}$$

where the first equality is due to the first relation on (6a) and relations on (6b) and the second equality holds by (3a). By the definition of $H_{l+1}(\cdot, x^{end})$ in (10), we obtain

$$H_{l+1}(y, x^{end}) \geq g^+(x^{end}) - c_{f_l}^{end} + \langle s_{f_l}^{end}, y - x^{end} \rangle, \quad \forall y \in \mathbb{R}^n.$$

Substituting $y := y^{l+1}$, then we get

$$-H_{l+1}(y^{l+1}, x^{end}) \leq -g^+(x^{end}) + c_{f_l}^{end} - \langle s_{f_l}^{end}, y^{l+1} - x^{end} \rangle. \tag{33}$$

Thus (32) and (33) imply that $-m\delta^l - H_{l+1}(y^{l+1}, x^{end}) \leq -g^+(x^{end}) + \langle s_{f_l}^{end}, y^l - y^{l+1} \rangle$, and by (31) we obtain $0 \leq \delta^{l+1} \leq m\delta^l + \langle s_{f_l}^{end}, y^l - y^{l+1} \rangle$.

(II) If $g(y) > g^+(x^{end}) - m\delta^l$, then by the same argument we again conclude that

$$0 \leq \delta^{l+1} \leq m\delta^l + \langle s_{g_l}^{end}, y^l - y^{l+1} \rangle.$$

Since $\{y^l\}$ is bounded, the functions f and g are locally Lipschitz and X is compact, it follows that $\{\xi_f^l\}$ and $\{\xi_g^l\}$ are bounded on that set. Therefore by the boundedness of $\{\eta_l\}$, we get $s_{f_l}^{end} = \xi_f^l + \eta_l(y^l - x^{end})$ and $s_{g_l}^{end} = \xi_g^l + \eta_l(y^l - x^{end})$ are bounded. Since $y^l - y^{l+1} \rightarrow 0$ and $m \in (0, 1)$, we deduce $\delta^l \rightarrow 0$. By the definition of δ^l we also have $\widehat{e}_l^{end} \rightarrow 0$ and $\frac{1}{2\mu_l} \|\widehat{\xi}_l^{end} + v_l\|^2 \rightarrow 0$, on the other hand we have $\mu_l \leq \bar{\mu}$, therefore we get $\widehat{\xi}_l^{end} + v_l \rightarrow 0$. Which implies that $y^l \rightarrow x^{end}$. By the convexity of $H_l(\cdot, x^{end})$ and $\widehat{\xi}_l^{end} + v_l \in \partial H_l(y^l, x^{end}) + \partial I_X(y^l)$, it follows from Lemma 5 (ii) that

$$\begin{aligned} M^\uparrow(y, x^{end}) + I_X(y) &\geq H_l(y, x^{end}) + I_X(y) \\ &\geq H_l(y^l, x^{end}) + I_X(y^l) + \langle \widehat{\xi}_l^{end} + v_l, y - y^l \rangle \\ &= g^+(x^{end}) - \widehat{e}_l^{end} + \langle \widehat{\xi}_l^{end}, y^l - x^{end} \rangle + \langle \widehat{\xi}_l^{end} + v_l, y - y^l \rangle \\ &= g^+(x^{end}) - \widehat{e}_l^{end} + \langle \widehat{\xi}_l^{end} + v_l, y - x^{end} \rangle + \langle v_l, x^{end} - y^l \rangle \\ &= g^+(x^{end}) - \widehat{e}_l^{end} + \langle \widehat{\xi}_l^{end} + v_l, y - x^{end} \rangle + \left\langle v_l, \frac{1}{\mu_l} (\widehat{\xi}_l^{end} + v_l) \right\rangle, \end{aligned}$$

where the first equality follows from (9) and the last equality by $x^{end} - y^l = \frac{1}{\mu_l} (\widehat{\xi}_l^{end} + v_l)$. Passing to the limit in above inequality and by the boundedness of $\{v_l\}$ (since there exists $\bar{\varepsilon}$ such that $\{v_l\} \subseteq \partial I_X(x^{end}) + \bar{B}_{\bar{\varepsilon}}(0)$), we obtain

$$\begin{aligned} M^\uparrow(y, x^{end}) + I_X(y) &\geq g^+(x^{end}) + \langle 0, y - x^{end} \rangle \\ &= M^\uparrow(x^{end}, x^{end}) + I_X(x^{end}) + \langle 0, y - x^{end} \rangle. \end{aligned}$$

By the convexity of $M^\uparrow(\cdot, x^{end})$ and $I_X(\cdot)$, we get $0 \in \partial M^\uparrow(x^{end}, x^{end}) + \partial I_X(x^{end})$ and using Lemma 5 (iv) we deduce $0 \in \partial_c h(x^{end}, x^{end}) + \partial I_X(x^{end})$. The remainder of the proof is the same as the proof of Theorem 1. □

5 Numerical experiments

In this section, we report the results of numerical experiments for the proposed algorithm. To provide a brief comparison of Algorithm 1, we used the public software SolvOpt [34], which is one of the most efficient codes available for nonsmooth optimization, and the proximal bundle method (PBM) [25], which was proposed for constrained nonsmooth nonconvex optimization with inexact information. But in this experiment we consider the PBM method without any noise.

All algorithms are coded in MATLAB R2012a and run on a PC Intel Core I5 with CPU 2.5 GHz and 4 GB of RAM.

Two classes of test problems are used to measure the efficiency of the considered algorithms. The first class is taken from [1,32] with constant number of variables and the second class is from [10] such that these problems can be formulated with any number of variables. Therefore we state numerical results in two experiments.

We set the algorithm parameters as follows $tol := 10^{-8}$, $m := 0.01$ or 0.25 (for $n = 20, 50$ we used $m = 0.25$ and for the others we set $m = 0.01$), $\omega := 1.2$ and $|\mathcal{L}|_{max} := 100$. As our convex compact feasible set, we use $X := \overline{B}_{10}(0)$. The Quadratic programming solver is `quadprog.m`, which is available in the MATLAB optimization toolbox. It is worth mentioning that, there is not any sensitivity to `quadprog.m` and any other quadratic programming solver can replace it. For each run, and for all of the algorithms, we give the total number of the evaluations (Note that evaluations include function evaluations and subgradient evaluations). We did not report the computational times, since for most of the problems (i.e., for $n = 2, 3, 4, 5$) the time is near zero and for others there were not a very big difference in the computational times of the different solvers. To check the precision, we will report the objective function value and the constraint function value in the last serious iteration.

We consider two stopping criteria for all algorithms: the first one is “the algorithms stop when the optimality condition is satisfied” and the second one is “the algorithms stop when the number of iterations exceeded 10,000”. Nevertheless, stoppage due to the maximum number of function evaluations was never invoked in these experiments. In addition Algorithm 1 and PBM terminated when one of the following criteria is satisfied, if $\delta^l \leq 10^{-8}$ or the value of δ^l did not decrease in 150 consecutive iterations.

In order to obtain a better comparison of the considered algorithms, we analyze the results using the performance profile and data profile introduced in [2,9,29]. Next we give a brief descriptions of the performance and data profiles. Given a set of solvers S and a set of problems P . We are interested in using the number of function evaluations as an efficiency measure. For each problem $p \in P$ and solver $s \in S$, define

$t_{p,s}$ = number of function evaluations required to solve problem p by solver s .

In the performance profile, we use a ratio to compare the implementation of each solver s on problem p with the best solver on this problem. This ratio is defined as follow

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} | s \in S\}}.$$

We assume that a parameter $r_M \geq r_{p,s}$ for all p, s are chosen, and $r_{p,s} = r_M$ if and only if solver s does not solve problem p . The performance profile $\rho_s(\tau)$ is defined as

$$\rho_s(\tau) = \frac{1}{n_p} \text{size}\{p \in P | r_{p,s} \leq \tau\}.$$

Here n_p is the number of problems in P . It is clear that $\tau \in [1, r_M]$. The value of $\rho_s(1)$ gives the percentage of test problems for which the corresponding algorithm is the best (with respect to the number of function evaluations). Since the ratio for some problems and solvers is large, we scale τ using the natural logarithm. Therefore, in the performance profiles, the value of $\rho_s(\tau)$ at $\log(\tau) = 0$ (not at $\tau = 1$) gives the percentage of test problems for which the corresponding algorithm is the best.

Data profile was first introduced in [29]. Data profiles try to answer the question: what percentage of problems can be solved within the budget of k function evaluations? The authors in [29] assume that the required number of function evaluations to satisfy the convergence test is likely to grow as the number of variables increases. The data profile of an optimization algorithm s is defined using

$$d_s(\alpha) = \frac{1}{n_p} \text{size} \left\{ p \in P \mid \frac{t_{p,s}}{n+1} \leq \alpha \right\},$$

where n is the number of variables in the problem $p \in P$ and $d_s(\alpha)$ is the percentage of problems that can be solved with $\alpha(n+1)$ function evaluations.

5.1 Experiment 1

We recall that in the convergence analysis, we assume that f and g are regular locally Lipschitz functions which are less restrictive than the lower $-C^1$ assumption. Now our aim is to design some examples which are not lower $-C^1$. Spingarn in [35] gives an example of a regular function in \mathbb{R}^2 which is not lower $-C^1$, as follows:

$$f(x_1, x_2) = \begin{cases} |x_2| & \text{if } x_1 \leq 0 \\ |x_2| - x_1^2 & \text{if } x_1 \geq 0, |x_2| \geq x_1^2 \\ \frac{x_1^4 - x_2^2}{2x_1^2} & \text{if } x_1 \geq 0, |x_2| \leq x_1^2. \end{cases}$$

We consider this function as objective function and using some lower $-C^1$ constraints from [32], we design three problems as follows:

- Problem 1: $\min_{x \in X} f(x_1, x_2)$
 $g(x) = \max\{-x_1^4 - 2x_2^2 - 1, 2x_1^2 - x_2^2 - 2.5\} \leq 0.$
- Problem 2: $\min_{x \in X} f(x_1, x_2)$
 $g(x) = \max\{x_1^2 - x_2^2, -2x_1^2 - x_2^2\} \leq 0.$
- Problem 3: $\min_{x \in X} f(x_1, x_2)$
 $g(x) = \max\{100x_1^2 + x_2^2 - 101, 80x_1^2 - x_2^2 - 79\} \leq 0.$

We observe that $f(x_1, x_2)$ is bounded below by 0. We also have $f(0, 0) = 0$, so $0 = \min_{(x_1, x_2) \in \mathbb{R}^n} f(x_1, x_2)$. On the other hand $(0, 0)$ is a feasible point for Problems

Table 1 The description of test problems

Problem	Problem type	n	f_{opt}	References
1	Nonsmooth nonconvex	2	0	
2	Nonsmooth nonconvex	2	0	
3	Nonsmooth nonconvex	2	0	
4 (Beale)	Nonsmooth convex	3	0.1111	[1]
5 (Charalambous and Bandler)	Nonsmooth nonconvex	2	1.9522	[32]
6 (Charalambous and Bandler)	Nonsmooth nonconvex	2	2.25	[32]
7 (Rosen and Suzuki)	Nonsmooth convex	4	-44	[1]
8 (Fiacco and McCormick)	Nonsmooth nonconvex	2	2.6667	[1]

1–3. Therefore $x^* = (0, 0)$ is an optimal solution for these problems and the optimal value is $f^* = 0$.

In addition, we consider some problems from [1,32]. We consider two types of problems, with nonsmooth convex and nonsmooth nonconvex functions. A brief description of each test problem is given in Table 1, where the following notations are used. We denote n for the number of variables, f_{opt} for the optimal value and Ref. for the reference of each problem. For each problem we used ten randomly generated starting points (by using `rand.m` and `randn.m` functions in MATLAB) and the starting points are the same for all algorithms. Some of the starting points are feasible and the others are infeasible. To better organize the numerical results, these randomly generated starting points are not reported in the tables. The interested reader can find more details about these points by sending email to the authors. The numerical results are listed in Tables 2, 3, 4, 5, 6, 7, 8 and 9, where the following notations are used:

- x_0 : starting point;
- F/InF: feasible/infeasible;
- NE: number of evaluations;
- f_{final} : final objective value;
- g_{final} : final constraint value.

The results of Tables 2, 3, 4, 5, 6, 7, 8 and 9 show a reasonable performance of Algorithm 1. One can observe that Algorithm 1 performs well and provides an optimal solution to each example. We note that the accuracy obtained by Algorithm 1 was similar for both feasible and infeasible starting points, that means the algorithm is not sensitive with feasible or infeasible points. As comparing with PBM and SolvOpt, in this part we run every algorithm 80 times, for 51 examples we observe that Algorithm 1 needs the least number of function evaluations than SolvOpt and PBM. For 28 examples SolvOpt method uses the least number of evaluations than others and for one example PBM needs the least number of function evaluations. For more details and carefully comparison see Tables 2, 3, 4, 5, 6, 7, 8 and 9.

Figure 1 presents the performance profiles for Experiment 1. Algorithm 1 has the best performance for 63% of the problems; meaning that Algorithm 1 is able to solve 63% of the problems with the least number of function evaluations than the other solvers. The SolvOpt and PBM solve roughly 39% and 3% of the problems

Table 2 Comparison between Algorithm 1, PBM and SolvOpt for Problem 1

No.	F/InF	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
1	F	92\0.0000\ - 1.0000	200\0.0000\ - 1.0000	248\0.0000\ - 1.0370
2	F	36\0.0000\ - 1.0000	96\0.0000\ - 1.0000	309\0.0000\ - 1.0000
3	F	156\0.0000\ - 1.1009	404\0.0000\ - 1.7476	192\0.0000\ - 2.6218
4	InF	344\0.0000\ - 1.0157	320\0.0000\ - 1.0004	352\0.0000\ - 1.3819
5	F	56\0.0000\ - 1.0001	316\0.0000\ - 1.0000	219\0.0000\ - 1.7804
6	InF	144\0.0000\ - 1.7722	268\0.0000\ - 1.5289	198\0.0000\ - 1.0352
7	F	56\0.0000\ - 1.0000	76\0.0000\ - 1.0000	247\0.0000\ - 1.5200
8	F	68\0.0000\ - 1.0000	124\0.0000\ - 1.0000	324\0.0000\ - 1.0000
9	F	88\0.0000\ - 1.0000	168\0.0000\ - 1.0000	178\0.0000\ - 1.0000
10	F	78\0.0000\ - 1.3068	692\0.9127\ - 1.1044	189\0.0000\ - 1.0522

Table 3 Comparison between Algorithm 1, PBM and SolvOpt for Problem 2

No.	F/InF	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
1	F	368\0.0000\ - 0.3662	804\0.0000\ - 0.0080	222\0.0000\ - 5.6551
2	InF	68\0.0000\ - 0.0000	244\0.0000\ - 0.0000	1458\0.0000\0.0000
3	InF	160\0.0000\ - 2.6929	492\0.0000\ - 0.2112	213\0.0000\ - 1.1238
4	InF	596\0.0000\ - 2.3098	806\0.0000\ - 0.8563	1511\0.0000\ - 0.0146
5	F	304\0.0000\ - 0.8774	1952\0.0000\ - 0.2451	230\0.0000\ - 4.4388
6	InF	128\0.0000\ - 0.1838	808\0.0000\ - 0.0037	375\0.0000\ - 0.1428
7	F	348\0.0000\ - 0.6142	420\0.0000\ - 0.2416	203\0.0000\ - 4.7954
8	InF	128\0.0000\ - 0.9227	425\0.0000\ - 0.8067	1369\0.0000\0.0000
9	F	48\0.0000\ - 0.0449	484\0.0000\0.0000	330\0.0000\ - 0.2070
10	InF	124\0.0000\ - 6.9328	248\0.0000\ - 0.2321	1233\0.0000\0.0000

Table 4 Comparison between Algorithm 1, PBM and SolvOpt for Problem 3

No.	F/InF	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
1	F	272\0.0000\ - 77.7359	404\0.0001\ - 78.9512	171\0.0000\ - 78.9827
2	InF	168\0.0000\ - 75.6739	425\0.0000\ - 78.9848	214\0.0000\ - 52.4033
3	InF	196\0.0000\ - 77.8558	803\0.0368\ - 47.3978	256\0.0000\ - 0.7503
4	InF	208\0.0000\ - 72.0699	680\0.0000\ - 78.7087	192\0.0000\ - 19.1845
5	InF	180\0.0000\ - 78.7087	502\0.1384\ - 77.2705	218\0.0000\ - 17.0941
6	InF	282\0.0000\ - 78.9753	692\0.0829\ - 20.7627	287\0.0000\ - 4.8255
7	InF	232\0.0000\ - 77.5943	520\0.0000\ - 78.9923	385\0.0000\ - 73.7245
8	F	108\0.0000\ - 78.9509	120\0.0000\ - 78.9993	171\0.0000\ - 63.9593
9	InF	312\0.0000\ - 70.0585	456\1.6416\ - 12.1302	270\0.0000\ - 52.0973
10	InF	204\0.0000\ - 64.1682	652\1.8050\0.3988	222\0.0000\ - 11.5426

Table 5 Comparison between Algorithm 1, PBM and SolvOpt for Problem 4

No.	F/InF	Algorithm 1 $NE \setminus f_{final} \setminus g_{final}$	PBM $NE \setminus f_{final} \setminus g_{final}$	SolvOpt $NE \setminus f_{final} \setminus g_{final}$
1	F	456\0.1112\ - 0.0000	625\0.1112\ - 0.0002	269\0.1111\0.0000
2	F	116\0.1111\ - 0.0048	741\0.1111\ - 0.0000	366\0.1111\0.0001
3	InF	364\0.1111\ - 0.0001	521\0.1809\ - 0.3021	365\0.1111\0.0001
4	InF	136\0.1111\ - 0.0079	426\0.1113\ - 0.0001	417\0.1111\0.0000
5	InF	136\0.1111\ - 0.0011	254\0.1112\ - 0.0000	441\0.1111\0.0000
6	InF	132\0.1113\ - 0.0002	348\0.1112\ - 0.0000	478\0.1111\0.0000
7	InF	640\0.1111\ - 0.0000	421\0.1111\0.0000	390\0.1111\0.0001
8	InF	188\0.1112\ - 0.0007	506\0.1447\ - 0.0365	258\0.1111\0.0000
9	InF	160\0.1112\ - 0.0024	532\0.1111\ - 0.0002	344\0.1111\0.0000
10	InF	284\0.1111\ - 0.0004	487\0.8481\ - 0.7561	1036\0.1111\0.0000

Table 6 Comparison between Algorithm 1, PBM and SolvOpt for Problem 5

No.	F/InF	Algorithm 1 $NE \setminus f_{final} \setminus g_{final}$	PBM $NE \setminus f_{final} \setminus g_{final}$	SolvOpt $NE \setminus f_{final} \setminus g_{final}$
1	InF	300\1.9522\ - 0.4734	804\1.9522\ - 0.3688	198\1.9522\ - 0.3530
2	InF	180\1.9522\ - 0.3159	524\1.9522\ - 0.3452	260\1.9522\ - 0.3530
3	F	192\1.9522\ - 0.3093	540\1.9524\ - 0.3678	196\1.9522\ - 0.3530
4	InF	288\1.9522\ - 0.3573	325\1.9522\ - 0.3652	247\1.9522\ - 0.3530
5	F	216\1.9522\ - 0.3540	415\1.9522\ - 0.3459	192\1.9522\ - 0.3632
6	F	200\1.9522\ - 0.3610	684\1.9523\ - 0.3540	241\1.9522\ - 0.3540
7	F	212\1.9522\ - 0.3266	1284\1.9525\ - 0.3850	222\1.9522\ - 0.3540
8	F	240\1.9522\ - 0.4827	1204\1.9522\ - 0.3382	232\1.9522\ - 0.3540
9	F	160\1.9522\ - 0.4016	389\1.9523\ - 0.3525	204\1.9522\ - 0.3540
10	F	172\1.9522\ - 0.3370	625\1.9522\ - 0.3520	211\1.9522\ - 0.3549

Table 7 Comparison between Algorithm 1, PBM and SolvOpt for Problem 6

No.	F/InF	Algorithm 1 $NE \setminus f_{final} \setminus g_{final}$	PBM $NE \setminus f_{final} \setminus g_{final}$	SolvOpt $NE \setminus f_{final} \setminus g_{final}$
1	InF	420\2.2500\ - 0.0000	489\2.1982\0.0341	381\2.2500\0.0000
2	InF	441\2.2500\0.0000	524\2.2502\ - 0.0001	257\2.2500\0.0000
3	InF	400\2.2500\0.0000	596\2.2500\0.0000	402\2.2500\0.0000
4	InF	528\2.2500\ - 0.0000	624\2.2500\0.0000	436\2.2500\0.0000
5	F	632\2.2500\0.0000	826\2.2500\ - 0.0000	423\2.2500\0.0000
6	InF	384\2.2500\0.0000	528\2.2500\ - 0.0000	325\2.2500\0.0000
7	InF	308\2.2500\0.0000	478\2.8353\0.0304	417\2.2500\0.0000
8	InF	444\2.2500\ - 0.0000	498\2.2500\ - 0.0000	405\2.2500\0.0000
9	F	292\2.2500\ - 0.0000	752\2.2500\ - 0.0002	314\2.2500\0.0000
10	InF	293\2.2500\ - 0.0000	631\2.0753\0.2897	367\2.2500\ - 0.0000

Table 8 Comparison between Algorithm 1, PBM and SolvOpt for Problem 7

No.	F/InF	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
1	F	568 \ - 44 \ - 0.0000	745 \ - 44.0000 \ - 0.0000	464 \ - 44 \ - 0.0001
2	InF	884 \ - 43.9999 \ - 0.0001	532 \ - 44.0000 \ - 4.1799	2242 \ - 44 \ 0.0000
3	InF	784 \ - 44.0001 \ - 0.0000	578 \ - 41.9821 \ - 5.5826	517 \ - 44.0000 \ 0.0000
4	InF	800 \ - 44 \ - 0.0000	792 \ - 43.8921 \ - 3.3509	401 \ - 44.0001 \ - 0.0000
5	InF	680 \ - 44.0001 \ - 0.0001	1256 \ - 44 \ - 0.0002	367 \ - 44 \ 0.0002
6	F	576 \ - 44 \ - 0.0000	652 \ - 44 \ 0.0005	371 \ - 44 \ 0.0004
7	F	500 \ - 43.9999 \ - 0.0000	745 \ - 44 \ 0.0003	365 \ - 44 \ 0.0004
8	F	300 \ - 44 \ - 0.0004	364 \ - 44 \ 0.0000	388 \ - 44 \ 0.0000
9	InF	428 \ - 44 \ - 0.0000	589 \ - 44 \ - 0.0000	388 \ - 43.9999 \ - 0.0001
10	InF	820 \ - 44.0001 \ - 0.0003	859 \ - 43.8987 \ 0.1204	399 \ - 44 \ - 0.0001

Table 9 Comparison between Algorithm 1, PBM and SolvOpt for Problem 8

No.	F/InF	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
1	InF	300 \ 2.6667 \ - 0.0000	406 \ 2.6667 \ - 0.0000	283 \ 2.6667 \ 0.0000
2	InF	288 \ 2.6667 \ 0.0000	336 \ 2.6667 \ - 0.0000	323 \ 2.6667 \ 0.0000
3	InF	168 \ 2.6667 \ - 0.0000	384 \ 2.6667 \ 0.0000	306 \ 2.6667 \ 0.0000
4	InF	280 \ 2.6667 \ - 0.0000	542 \ 2.6667 \ - 0.0000	349 \ 2.6667 \ 0.0000
5	InF	228 \ 2.6667 \ - 0.0000	386 \ 2.6668 \ - 0.0001	320 \ 2.6667 \ 0.0000
6	F	312 \ 2.6667 \ - 0.0000	488 \ 2.6667 \ 0.0000	339 \ 2.6667 \ 0.0000
7	F	384 \ 2.6667 \ - 0.0000	566 \ 2.6667 \ - 0.0000	351 \ 2.6667 \ 0.0000
8	InF	376 \ 2.6667 \ - 0.0000	558 \ 2.6667 \ 0.0000	334 \ 2.6667 \ 0.0000
9	F	320 \ 2.6667 \ - 0.0000	504 \ 2.6667 \ - 0.0000	337 \ 2.6667 \ 0.0000
10	InF	308 \ 2.6667 \ 0.0000	488 \ 2.6667 \ - 0.0000	381 \ 2.6667 \ 0.0000

with the least number of function evaluations. If we choose being within a factor of $\log(\tau) \simeq 0.4$ of the best solver as the scope of our interest, then Algorithm 1 is the best solver for all problems while SolvOpt and PBM are the best solvers for 78% and 50% of the problems.

Figure 2 illustrates the data profiles for Experiment 1. Suppose the user has a budget limit of 100 number of function evaluations; according to Fig. 2, with this budget Algorithm 1 has the best performance, solving roughly 68% of the problems; while SolvOpt and PBM solve 52% and 18% of the problems, respectively. Moreover, if the user has a budget limit of 200 number of function evaluations, then Algorithm 1 solves 100% of the problems and it has the best performance among all the solvers. While PBM has the worst performance among all the solvers since with this budget it solves roughly 64% of the problems.

Moreover it is clear from Figs. 1 and 2 that Algorithm 1 and PBM can solve 100% and 97% of the problems, while SolvOpt solves 80% of the problems.

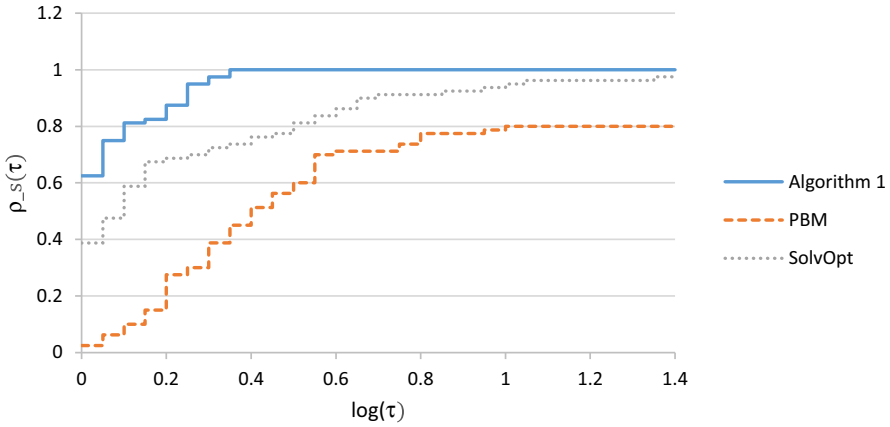


Fig. 1 Performance profile for Experiment 1

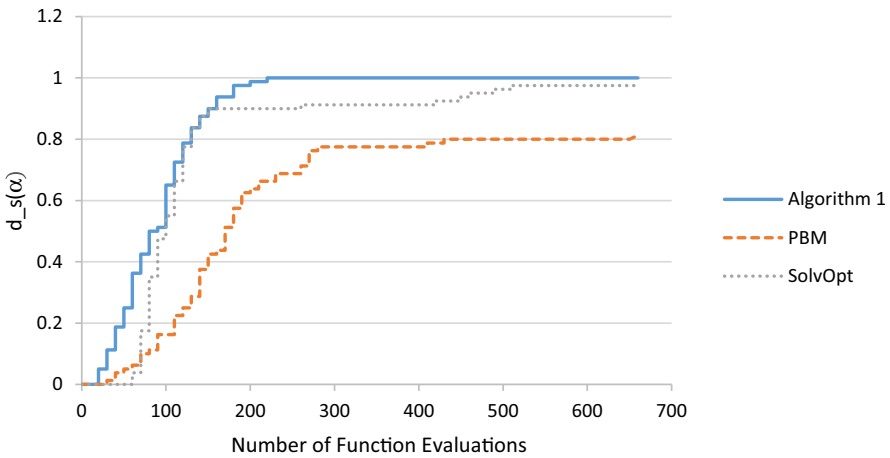


Fig. 2 Data profile for Experiment 1

5.2 Experiment 2

In this subsection, we consider a series of polynomial functions developed in [10], which were used in [13,14,25,37] and these problems can be formulated with any number of variables. For each $i = 1, 2, \dots, n$, the function $h_i: \mathbb{R}^n \rightarrow \mathbb{R}$ is defined by $h_i(x) := (ix_i^2 - 2x_i) + \sum_{j=1}^n x_j$. There are five classes of test functions defined by h_i in [10] as follows:

$$f_1(x) := \sum_{i=1}^n |h_i(x)|,$$

$$f_2(x) := \max_{1 \leq i \leq n} |h_i(x)|,$$

$$\begin{aligned}
 f_3(x) &:= \sum_{i=1}^n |h_i(x)| + \frac{1}{2} \|x\|^2, \\
 f_4(x) &:= \sum_{i=1}^n |h_i(x)| + \frac{1}{2} \|x\|, \\
 f_5(x) &:= \sum_{i=1}^n (h_i(x))^2.
 \end{aligned}$$

It has been proved in [13] that these test functions are nonconvex and globally lower $-C^1$ in \mathbb{R}^2 and they are nonsmooth except f_5 . These properties carry to higher dimensions as well. We also have $0 = \min_{x \in \mathbb{R}^n} f_k(x)$ and $\{0\} \in \operatorname{argmin}_{x \in \mathbb{R}^n} f_k(x)$ for $k = 1, 2, 3, 4, 5$. For constraint functions, we consider the pointwise maximum of a finite collection of quadratic functions from [12], as follows

$$g(x) = \max_{1 \leq i \leq N} \{ \langle x, A_i x \rangle + \langle b_i, x \rangle + c_i \},$$

where A_i are $n \times n$ matrices, $b_i \in \mathbb{R}^n$ and $c_i \in \mathbb{R}$ for $i = 1, 2, \dots, N$. The terms of A_i and b_i and the coefficients c_i were chosen randomly as uniformly distributed numbers in $[-5, 5]$. We consider the following test problems

$$\begin{aligned}
 \min_{x \in X} f_i(x) \\
 g(x) \leq 0,
 \end{aligned}$$

where $X \subseteq \mathbb{R}^n$ for $i = 1, 2, 3, 4, 5$ and $n = 2, 3, 4, 5, 10, 20, 50$. We use two starting points as follows $x_0 = [1, 1, \dots, 1]$ and $x_0 = [1, 1/2, \dots, 1/n]$ for each problem, hence we have 70 examples.

Our results for deterministic tests are summarized in Tables 10, 11, 12, 13 and 14, which show a reasonable performance of Algorithm 1. Where the notations are the same as the previous subsection. We analyse the results in more details as follows:

- For the objective function f_1 , we tested 14 nonconvex examples. Table 10 shows that values of f_{final} obtained by Algorithm 1 and SolvOpt are quite correct whereas PBM can not calculate the correct solution for $n = 10, 20, 50$. Algorithm 1 needs the least number of function evaluations for $n = 2, 3, 4$, whereas SolvOpt uses the least function evaluations for $n = 5, 10, 20, 50$.
- For the objective function f_2 , we tested 14 examples. Table 11 demonstrates that f_{final} obtained by Algorithm 1 is quite correct for all examples. Whereas, SolvOpt can not find the correct solution for $n = 5$ with $x_0 = [1, 1, \dots, 1]$ and PBM can not calculate the correct solutions for $n = 5, 20, 50$ with $x_0 = [1, 1, \dots, 1]$ and for $n = 50$ with $x_0 = [1, 1/2, \dots, 1/n]$. Which means that Algorithm 1 has the most accuracy for the objective function f_2 . Moreover, Algorithm 1 needs the least number of function evaluations for 10 examples, SolvOpt uses the least number of function evaluations for 3 examples and PBM needs the least number of evaluations for one example.

- The corresponding results for the 14 examples of function f_3 are reported in Table 12. These results show that f_{final} obtained by Algorithm 1 is quite correct for all examples. While, PBM fails to find the correct solution for $n = 20, 50$ with $x_0 = [1, 1, \dots, 1]$ and for $n = 10$ with $x_0 = [1, 1/2, \dots, 1/n]$. Also the generated solutions by SolvOpt are false for $n = 10, 50$ with $x_0 = [1, 1, \dots, 1]$. Moreover Algorithm 1 needs the least number of function evaluations for 8 examples and SolvOpt uses the least number of function evaluations for 6 examples.
- The results for function f_4 are proposed in Table 13. These results show that f_{final} obtained by Algorithm 1 is quite correct for all examples except $n = 20$ with $x_0 = [1, 1, \dots, 1]$ and $n = 50$ with $x_0 = [1, 1/2, \dots, 1/n]$. Whereas, PBM can not calculate the correct solutions for $n = 3, 10, 20, 50$ with $x_0 = [1, 1, \dots, 1]$ and for $n = 10, 50$ with $x_0 = [1, 1/2, \dots, 1/n]$ and SolvOpt fails to find the correct solutions for $n = 10, 20, 50$ with $x_0 = [1, 1, \dots, 1]$. Moreover Algorithm 1 uses the least number of evaluations for 6 examples, SolvOpt needs the least number of function evaluations for 7 examples and PBM needs the least number of function evaluations for one example.
- The generated results for function f_5 are stated in Table 14. The results illustrate that Algorithm 1 finds the correct solutions for all examples. While, PBM fails to find the correct solutions for $n = 3, 5, 50$ with $x_0 = [1, 1, \dots, 1]$ and for $n = 5, 10, 20$ with $x_0 = [1, 1/2, \dots, 1/n]$ and SolvOpt can not calculate the correct solution for $n = 20$ with $x_0 = [1, 1, \dots, 1]$. Moreover SolvOpt uses the least number of function evaluations for 12 examples and Algorithm 1 needs the least number of function evaluations for 2 examples. Which means that SolvOpt uses the least number of evaluations for function f_5 , that these results are acceptable since f_5 is a smooth function.

Figure 3 presents the performance profiles for Experiment 2. SolvOpt and Algorithm 1 have the best performance for 51% and 47% of the problems with the least number of function evaluations. While PBM has the best performance for 5% of the problems. The performances of Algorithm 1 and SolvOpt are more competitive than the performance of PBM method for these problems.

Figure 4 states the data profiles for Experiment 2. Suppose that the user has a budget limit of 100 number of function evaluations; according to Fig. 4, with this budget SolvOpt and Algorithm 1 solves roughly 53% and 52% of the problems, respectively; while PBM solves about 18% of the problems. Moreover, if the user has a budget limit of 300 number of function evaluations, then Algorithm 1 and SolvOpt solve about 93% and 90% of the problems; while PBM solves 58% of the problems.

Moreover the results in Figs. 3 and 4 are shown that Algorithm 1 and PBM can solve 97% and 94% of the problems, respectively; while PBM solves roughly 70% of the problems.

5.3 The behaviour of parameter η_l

At the end, we are interested in exploring the assumption that the parameter η_l remains bounded. In [14] the authors report that η_l was less than $2n + 2$ when solving 73 of the

Table 10 Comparison between Algorithm 1, PBM and SolvOpt for f_1

n	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
$x_0 = [1, 1, \dots, 1]$			
2	228\0.0000\ - 13.1440	440\0.0000\ - 13.0022	394\0.0000\ - 13.0000
3	684\0.0000\ - 8.0001	740\0.0000\ - 8.0000	693\0.0000\ - 8.0000
4	556\0.0000\ - 5.0032	612\0.0000\ - 5.0035	809\0.0000\ - 5.0000
5	3428\0.0000\ - 1.0002	2484\0.0000\ - 0.0000	585\0.0000\ - 0.0000
10	6160\0.0000\ - 4.0575	4884\0.0052\ - 5.7132	700\0.0000\ - 5.0515
20	5556\0.0000\ - 7.9350	4140\0.0061\ - 8.5914	885\0.0000\ - 0.3540
50	4884\0.0000\ - 4.7010	5188\0.0068\ - 4.7139	1315\0.0000\ - 3.6234
$x_0 = [1, 1/2, \dots, 1/n]$			
2	100\0.0000\ - 13.0021	404\0.0000\ - 13.0029	449\0.0000\ - 13.0145
3	152\0.0000\ - 8.0059	1008\0.0000\ - 8.0031	718\0.0000\ - 8.0239
4	1328\0.0000\ - 5.0008	492\0.0000\ - 5.0011	575\0.0000\ - 5.0011
5	1552\0.0000\ - 4.1016	1580\0.0000\ - 4.1041	611\0.0000\ - 4.1012
10	944\0.0000\ - 41.0000	920\0.0032\ - 35.6420	584\0.0000\ - 41.0000
20	2624\0.0000\ - 7.1000	2328\0.0015\ - 6.5421	593\0.0000\ - 7.1000
50	2500\0.0000\ - 50.9987	2884\0.0041\ - 40.5231	1719\0.0000\ - 51.5825

Table 11 Comparison between Algorithm 1, PBM and SolvOpt for f_2

n	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
$x_0 = [1, 1, \dots, 1]$			
2	120\0.0000\ - 5.1203	500\0.0000\0.0000	251\0.0000\0.0000
3	784\0.0000\ - 8.0031	740\0.0000\ - 8.0129	693\0.0000\ - 8.0015
4	556\0.0000\ - 5.0029	612\0.0000\ - 5.0035	809\0.0000\ - 5.0181
5	1708\0.0000\ - 4.1002	1436\0.0031\ - 3.0035	503\0.1640\ - 3.1258
10	1344\0.0000\ - 41.0071	2004\0.0000\ - 41.0089	1441\0.0000\ - 36.0887
20	2952\0.0000\ - 11.1556	3164\0.0049\ - 5.1549	3283\0.0000\ - 6.1300
50	5364\0.0000\ - 4.9401	8044\0.0068\ - 50.0729	7270\0.0000\ - 55.0339
$x_0 = [1, 1/2, \dots, 1/n]$			
2	192\0.0000\ - 7.99811	60\0.0000\ - 13.0278	286\0.0000\ - 13.0000
3	268\0.0000\ - 8.0131	1004\0.0000\ - 7.9985	677\0.0000\ - 8.4212
4	596\0.0000\ - 5.0000	631\0.0000\ - 5.0000	692\0.0000\ - 5.0000
5	224\0.0000\ - 4.0998	659\0.0000\ - 4.0085	707\0.0000\ - 4.1000
10	520\0.0000\ - 41.0001	1208\0.0000\ - 41.0005	952\0.0000\ - 41.0000
20	6080\0.0000\ - 7.1578	6448\0.0000\ - 6.5565	4204\0.0000\ - 6.5589
50	7284\0.0000\ - 47.8721	8244\0.0069\ - 46.7210	8277\0.0000\ - 48.8494

Table 12 Comparison between Algorithm 1, PBM and SolvOpt for f_3

n	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
$x_0 = [1, 1, \dots, 1]$			
2	152 \ 0.0000 \ - 14.8587	248 \ 0.0000 \ - 15.0863	250 \ 0.0000 \ - 15.0000
3	484 \ 0.0000 \ - 14.9983	896 \ 0.0000 \ - 15.0001	546 \ 0.0000 \ - 15.0000
4	588 \ 0.0000 \ - 10.9995	1032 \ 0.0000 \ - 10.9998	601 \ 0.0000 \ - 11.0000
5	332 \ 0.0000 \ - 5.0000	780 \ 0.0000 \ - 5.0000	593 \ 0.0000 \ - 5.0000
10	1092 \ 0.0000 \ - 3.0002	1404 \ 0.0000 \ - 3.0000	535 \ 0.0915 \ - 12.5151
20	3140 \ 0.0000 \ - 2.0999	2676 \ 0.0314 \ - 2.1000	1064 \ 0.0000 \ - 2.1000
50	1684 \ 0.0000 \ - 2.1621	3412 \ 0.1079 \ - 3.1312	1343 \ 0.0514 \ - 6.1794
$x_0 = [1, 1/2, \dots, 1/n]$			
2	164 \ 0.0000 \ - 15.8553	636 \ 0.0000 \ - 14.9981	308 \ 0.0000 \ - 15.0000
3	156 \ 0.0000 \ - 15.0000	756 \ 0.0000 \ - 15.0005	656 \ 0.0000 \ - 15.0000
4	800 \ 0.0000 \ - 11.0000	812 \ 0.0000 \ - 11.0000	573 \ 0.0000 \ - 11.0000
5	352 \ 0.0000 \ - 5.0000	756 \ 0.0000 \ - 5.0003	550 \ 0.0000 \ - 5.0000
10	448 \ 0.0000 \ - 3.0000	636 \ 0.0123 \ - 3.0121	577 \ 0.0000 \ - 3.1000
20	2724 \ 0.0000 \ - 2.1000	1836 \ 0.0000 \ - 2.0999	532 \ 0.0000 \ - 2.1000
50	1516 \ 0.0000 \ - 2.1721	2892 \ 0.0000 \ - 2.1617	612 \ 0.0000 \ - 2.1512

Table 13 Comparison between Algorithm 1, PBM and SolvOpt for f_4

n	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
$x_0 = [1, 1, \dots, 1]$			
2	272 \ 0.0000 \ - 15.0041	380 \ 0.0000 \ - 15.0003	389 \ 0.0000 \ - 15.0000
3	940 \ 0.0000 \ - 14.9950	1132 \ 0.0012 \ - 17.9989	509 \ 0.0000 \ - 15.0000
4	296 \ 0.0000 \ - 11.0000	452 \ 0.0000 \ - 11.0000	537 \ 0.0000 \ - 11.0000
5	384 \ 0.0000 \ - 4.9998	1156 \ 0.0000 \ - 5.0002	557 \ 0.0000 \ - 5.0000
10	1140 \ 0.0000 \ - 2.9999	1488 \ 0.0409 \ - 4.9955	587 \ 0.2139 \ - 12.5151
20	2824 \ 0.0241 \ - 3.1519	4084 \ 0.3151 \ - 4.1981	688 \ 0.1016 \ - 2.3618
50	1556 \ 0.0000 \ - 2.1515	2956 \ 0.0721 \ - 8.2014	987 \ 0.1685 \ - 4.3566
$x_0 = [1, 1/2, \dots, 1/n]$			
2	332 \ 0.0000 \ - 15.0036	176 \ 0.0000 \ - 15.0000	423 \ 0.0000 \ - 15.0000
3	144 \ 0.0000 \ - 14.9979	1208 \ 0.0000 \ - 14.9851	385 \ 0.0000 \ - 15.0000
4	404 \ 0.0000 \ - 10.9994	596 \ 0.0000 \ - 10.9885	477 \ 0.0000 \ - 11.0000
5	388 \ 0.0000 \ - 5.0000	1084 \ 0.0000 \ - 4.9985	580 \ 0.0000 \ - 5.0000
10	716 \ 0.0000 \ - 3.0000	1176 \ 0.1317 \ - 4.1321	613 \ 0.0000 \ - 3.0000
20	1248 \ 0.0000 \ - 2.1000	2056 \ 0.0000 \ - 2.1000	1138 \ 0.0000 \ - 2.1000
50	844 \ 0.0204 \ - 2.1701	3316 \ 0.1251 \ - 2.1712	712 \ 0.0000 \ - 2.1510

Table 14 Comparison between Algorithm 1, PBM and SolvOpt for f_5

n	Algorithm 1 $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	PBM $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$	SolvOpt $NE \setminus f_{\text{final}} \setminus g_{\text{final}}$
$x_0 = [1, 1, \dots, 1]$			
2	172 \ 0.0000 \ - 16.8166	388 \ 0.0000 \ - 17.1210	173 \ 0.0000 \ - 15.0117
3	348 \ 0.0000 \ - 14.8979	148 \ 0.0012 \ - 14.8951	206 \ 0.0000 \ - 15.0000
4	720 \ 0.0000 \ - 10.9980	804 \ 0.0000 \ - 10.9989	244 \ 0.0000 \ - 11.0000
5	1092 \ 0.0000 \ - 4.9244	1752 \ 0.0235 \ - 17.9985	264 \ 0.0000 \ - 5.0000
10	4348 \ 0.0000 \ - 2.9973	4804 \ 0.0000 \ - 2.9959	332 \ 0.0000 \ - 3.0000
20	2404 \ 0.0000 \ - 6.7112	3004 \ 0.0000 \ - 4.1312	448 \ 0.1016 \ - 2.9628
50	1612 \ 0.0000 \ - 2.8131	2052 \ 0.1091 \ - 2.1510	698 \ 0.0000 \ - 2.1392
$x_0 = [1, 1/2, \dots, 1/n]$			
2	104 \ 0.0000 \ - 15.0004	220 \ 0.0000 \ - 15.0010	172 \ 0.0000 \ - 15.0071
3	240 \ 0.0000 \ - 14.9981	300 \ 0.0000 \ - 14.8999	213 \ 0.0000 \ - 15.0000
4	468 \ 0.0000 \ - 11.0734	996 \ 0.0000 \ - 11.0000	233 \ 0.0000 \ - 11.0000
5	744 \ 0.0000 \ - 4.9998	1022 \ 0.0045 \ - 10.9597	230 \ 0.0000 \ - 5.0000
10	2920 \ 0.0000 \ - 3.0000	3284 \ 0.1317 \ - 3.0000	312 \ 0.0000 \ - 3.0000
20	1316 \ 0.0000 \ - 2.1008	1924 \ 0.1561 \ - 2.1010	491 \ 0.0000 \ - 2.4112
50	1276 \ 0.0000 \ - 2.1590	1724 \ 0.0000 \ - 2.2131	560 \ 0.0000 \ - 2.1731

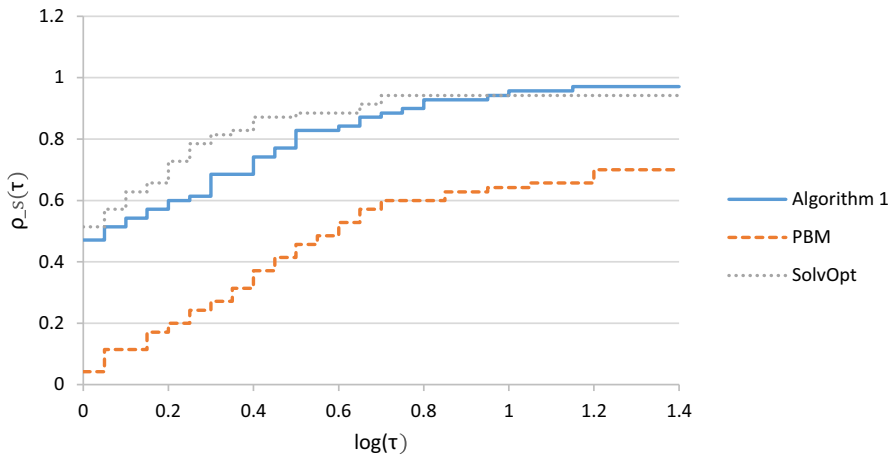


Fig. 3 Performance profile for Experiment 2

75 exact unconstrained optimization problems; for one test problem η_l was between $2n + 2$ and $25n$ and for the remaining one η_l exceeded $25n$.

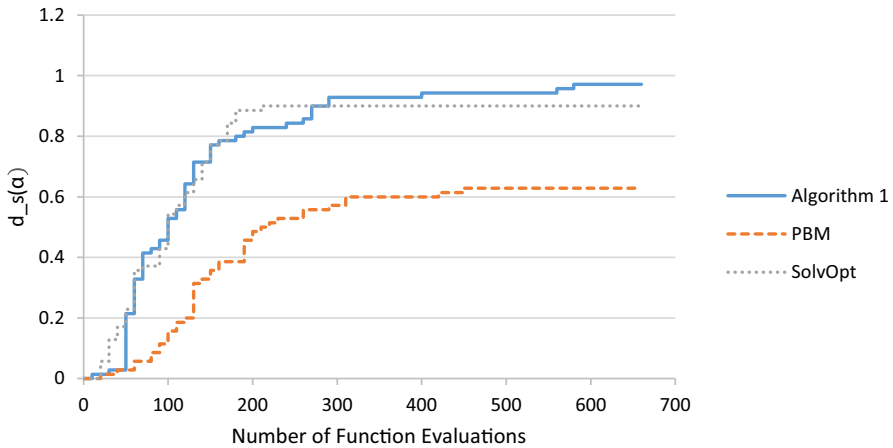


Fig. 4 Data profile for Experiment 2

In our results, to select η_l (in Step 2) we use (7) with equality, i.e.,

$$\eta_l = \max \left\{ \max_{\substack{j \in \mathcal{L}_l^{\text{oracle}} \\ y^j \neq x^k}} \left\{ \frac{-\left(e_{f_j}^k + g^+(x^k)\right)}{b_j^k}, \frac{-\left(e_{g_j}^k + g^+(x^k) - g(x^k)\right)}{b_j^k} \right\}, \omega \right\} + \omega.$$

For solving the 80 problems of Experiment 1 described in Sect. 5.1, $\eta_l = 2\omega$ in 60 problems, η_l is bounded by $2n$ in 16 problems and it is between $2n$ and $25n$ for 4 test problems. On the other hand, for solving the 70 problems of Experiment 2 considered in Sect. 5.1, $\eta_l = 2\omega$ in 25 problems, η_l is bounded by $2n$ in 39 problems and it is between $2n$ and $25n$ for 6 test problems. Overall, the experiments support that the assumption on the sequence $\{\eta_l\}$ is quite reasonable and this sequence is bounded.

6 Conclusions

We have presented an infeasible proximal bundle method using the improvement function and the aggregate technique which is adapted for nonsmooth nonconvex constrained optimization problems with regular functions. The global convergence of the proposed algorithm was proved in the sense that every accumulation point of the sequence of serious iterates is stationary for the improvement function, with an arbitrary starting point. We present results of numerical experiments using some test problems. Our limited computational experiments suggest the good performance of the proposed method and we can say that our new solver is comparable with the existing solvers for nonsmooth nonconvex optimization problems.

Acknowledgements The second-named author was partially supported by a Grant from IPM (No. 98900417). The authors would like to extend gratitude toward the anonymous referees whose suggestions helped to improve the presentation of this paper.

References

1. Asaadi, J.: A computational comparison of some non-linear programs. *Math. Program.* **4**, 144–154 (1973)
2. Beiranvand, V., Hare, W., Lucet, Y.: Best practices for comparing optimization algorithms. *Optim. Eng.* **18**, 815–848 (2017)
3. Clarke, F.H., Ledyaev, Y.S., Stern, R.J., Wolenski, P.R.: *Nonsmooth Analysis and Control Theory*. Springer, New York (1998)
4. Correa, R., Lemaréchal, C.: Convergence of some algorithms for convex minimization. *Math. Program.* **62**, 261–275 (1993)
5. Curtis, F.E., Overton, M.L.: A sequential quadratic programming algorithm for nonconvex, nonsmooth constrained optimization. *SIAM J. Optim.* **22**, 474–500 (2012)
6. Daniilidis, A., Georgiev, P.: Approximate convexity and submonotonicity. *J. Math. Anal. Appl.* **291**, 292–301 (2004)
7. Dao, M.N.: Bundle method for nonconvex nonsmooth constrained optimization. *J. Convex Anal.* **22**, 1061–1090 (2015)
8. Dao, M.N., Gwinner, J., Noll, D., Ovcharova, N.: Nonconvex bundle method with application to a delamination problem. *Comput. Optim. Appl.* **65**, 173–203 (2016)
9. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. *Math. Program.* **91**, 201–213 (2002)
10. Ferrier, C.: *Bornes Duales de Problemes d’Optimisation Polynomiaux*. Ph.D. thesis, Laboratoire Approximation et Optimisation, Université Paul, Toulouse (1997)
11. Gabarrou, M., Alazard, D., Noll, D.: Design of a flight control architecture using a non-convex bundle method. *Math. Control Signals Syst.* **25**, 257–290 (2013)
12. Hare, W., Sagastizábal, C.: Computing proximal points of nonconvex functions. *Math. Program.* **116**, 221–258 (2009)
13. Hare, W., Sagastizábal, C.: A redistributed proximal bundle method for nonconvex optimization. *SIAM J. Optim.* **20**, 2442–2473 (2010)
14. Hare, W., Sagastizábal, C., Solodov, M.: A proximal bundle method for nonsmooth nonconvex functions with inexact information. *Comput. Optim. Appl.* **63**, 1–28 (2016)
15. Hintermüller, M.: A proximal bundle method based on approximate subgradients. *Comput. Optim. Appl.* **20**, 245–266 (2001)
16. Hiriart-Urruty, J.B., Lemaréchal, C.: Convex analysis and minimization algorithms II. In: *Advanced Theory and Bundle Methods*. vol. 306 of *Grundlehren der mathematischen Wissenschaften* (1993)
17. Hoseini, N., Nobakhtian, S.: A new trust region method for nonsmooth nonconvex optimization. *Optimization* **67**, 1265–1286 (2018)
18. Joki, K., Bagirov, A.M., Karmitsa, N., Mäkelä, M.M.: A proximal bundle method for nonsmooth DC optimization utilizing nonconvex cutting planes. *J. Glob. Optim.* **68**, 501–535 (2017)
19. Karas, E., Ribeiro, A., Sagastizábal, C., Solodov, M.: A bundle filter method for nonsmooth convex constrained optimization. *Math. Program. Ser. B* **116**, 297–320 (2009)
20. Kiwiel, K.C.: A linearization algorithm for nonsmooth minimization. *Math. Oper. Res.* **10**, 185–194 (1985)
21. Kiwiel, K.C.: *Methods of Descent for Nondifferentiable Optimization*. Lecture Notes in Mathematics, vol. 1133. Springer, Berlin (1985)
22. Kiwiel, K.C.: Restricted step and Levenberg–Marquardt techniques in proximal bundle methods for nonconvex nondifferentiable optimization. *SIAM J. Optim.* **6**, 227–249 (1996)
23. Lemaréchal, C.: Bundle methods in nonsmooth optimization. In: Lemaréchal, C., Mifflin, R. (eds.) *Nonsmooth Optimization* (Laxenburg, 1977), IASA Proc. Ser., vol. 3, pp. 79–102. Pergamon Press, Oxford (1978)
24. Lemaréchal, C.: Lagrangian relaxation. In: *Computational Combinatorial Optimization*, Lecture Notes in Computer Science, vol. 2241, pp. 112–156. Springer, Berlin (2001)

25. Lv, J., Pang, L.P., Meng, F.Y.: A proximal bundle method for constrained nonsmooth nonconvex optimization with inexact information. *J. Glob. Optim.* **70**, 517–549 (2017)
26. Lv, J., Pang, L.P., Xu, N., Xiao, Z.-H.: An infeasible bundle method for nonconvex constrained optimization with application to semi-infinite programming problems. *Numer. Algorithm* **80**, 397–427 (2019)
27. Mäkelä, M.M., Neittaanmäki, P.: *Nonsmooth Optimization: Analysis and Algorithms with Applications to Optimal Control*. World Scientific, Singapore (1992)
28. Mifflin, R.: A modification and extension of Lemarechal's algorithm for nonsmooth minimization. In: Sorensen, D.C., Wets, R.B. (eds.) *Nondifferential and Variational Techniques in optimization* (Lexington, 1980), *Mathematical Programming Studies*, vol. 17, pp. 77–90. North-Holland, Amsterdam (1982)
29. Moré, J.J., Wild, S.: Benchmarking derivative-free optimization algorithms. *SIAM J. Optim.* **20**, 172–191 (2009)
30. Noll, D.: Cutting plane oracles to minimize non-smooth non-convex functions. *Set-Valued Var. Anal.* **18**, 531–568 (2010)
31. Rockafellar, R.T., Wets, R.J.B.: *Variational Analysis*. Springer, Berlin (1998)
32. Rustem, B., Nguyen, Q.: An algorithm for the inequality-constrained discrete min-max problem. *SIAM J. Optim.* **8**, 265–283 (1998)
33. Sagastizábal, C., Solodov, M.: An infeasible bundle method for nonsmooth convex constrained optimization without a penalty function or a filter. *SIAM J. Optim.* **16**, 146–169 (2005)
34. Shor, N.Z.: *Minimization Methods for Non-Differentiable Functions*. Springer, Berlin (1985)
35. Spingarn, J.E.: Submonotone subdifferentials of Lipschitz functions. *Trans. Am. Math. Soc.* **264**, 77–89 (1981)
36. Tang, C.M., Liu, S., Jian, J.B., Li, J.L.: A feasible SQP-GS algorithm for nonconvex, nonsmooth constrained optimization. *Numer. Algorithms* **65**, 1–22 (2014)
37. Yang, Y., Pang, L., Ma, X., Shen, J.: Constrained nonconvex nonsmooth optimization via proximal bundle method. *J. Optim. Theory Appl.* **163**, 900–925 (2014)

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