Numerical Algorithms (2023) 94:765–787
https://doi.org/10.1007/s11075-023-01519-8

ORIGINAL PAPER

New proximal bundle algorithm based on the gradient sampling method for nonsmooth nonconvex optimization with exact and inexact information

N. Hoseini Monjezi[1] · S. Nobakhtian[2]

Received: 30 August 2022 / Accepted: 17 February 2023 / Published online: 25 April 2023

Abstract
In this paper, we focus on a descent algorithm for solving nonsmooth nonconvex optimization problems. The proposed method is based on the proximal bundle algorithm and the gradient sampling method and uses the advantages of both. In addition, this algorithm has the ability to handle inexact information, which creates additional challenges. The global convergence is proved with probability one. More precisely, every accumulation point of the sequence of serious iterates is either a stationary point if exact values of gradient are provided or an approximate stationary point if only inexact information of the function and gradient values is available. The performance of the proposed algorithm is demonstrated using some academic test problems. We further compare the new method with a general nonlinear solver and two other methods specifically designed for nonconvex nonsmooth optimization problems.

Keywords Proximal bundle method · Gradient sampling · Inexact information · Nonsmooth optimization · Nonconvex optimization

Mathematics Subject Classification (2010) 90C26 · 49J52 · 65K05

1 Introduction

There exists a wide collection of practical problems involving nonsmooth functions with inexact information and nonconvex characteristics. However, most nonsmooth solution methods are only designed to solve problems with exact data and are strongly

✉ S. Nobakhtian
nobakht@math.ui.ac.ir

[1] School of Mathematics, Institute for Research in Fundamental Sciences (IPM), P.O. Box: 19395-5746, Tehran, Iran

[2] Department of Applied Mathematics and Computer Science, Faculty of Mathematics and Statistics, University of Isfahan, Isfahan, Iran

based on the convexity of functions. In this paper, we introduce a new numerical method to solve an unconstrained optimization problem with a nonsmooth and nonconvex objective function that is able to handle both exact and inexact information.

Over the last decades, several numerical methods have been developed based on the Clarke subdifferential for solving nonsmooth optimization problems. The subgradient-type methods are the simplest methods for convex optimization [24, 34] which are further generalized for nonconvex problems [2, 3, 5, 29, 37]. The bundle-type methods were first introduced for convex problems [25] and have been developed over the years for nonconvex problems [11–13, 17, 19, 20, 27, 30, 32]. The discrete gradient algorithm is considered as a derivative free method to solve nonconvex problems [1]. Trust region algorithms are among the most popular methods for smooth optimization problems and have been developed for nonsmooth optimization [14, 33]. In [6, 23, 36] the gradient sampling (GS) algorithm is proposed for solving nonsmooth nonconvex optimization problems.

Inexact information has been considered in subgradient methods for the convex optimization in [22] and for the nonconvex case in [35]. Inexact information of function and subgradient values in convex bundle methods returns to [21], where vanishing noise is considered, that is evaluations need to be asymptotically tightened. Inexact information with nonvanishing perturbations in bundle methods has been studied in [8, 12, 13, 18, 30].

The bundle and GS algorithms belong to the most attractive methods for minimizing nonsmooth nonconvex problems. Bundle algorithms need to calculate a single subgradient at each iteration. The information already generated in previous iterations is kept in the bundle and using this information, a piecewise linear model for the objective function is generated at each iteration. Then by solving a quadratic program, a new candidate descent direction is obtained that either creates a descent in the objective function or finds new information that modifies the next model. On the other hand, GS algorithms do not need to calculate subgradients by the user. At each iteration, GS methods calculate the gradients at the current point and at some randomly generated nearby points. Then by solving a quadratic program, an ε-steepest descent direction is obtained. A standard Armijo line search along this direction produces a candidate for the next iterate and only needs to be perturbed to remain in the set of differentiable points of the objective function. The perturbation is random and small enough to preserve the Armijo sufficient descent property.

In this paper, we propose a minimization algorithm that combines the advantages of the GS algorithm [23] and the redistributed proximal bundle method [11, 12, 15, 16]. Following the same idea as GS algorithms, we calculate gradients and keep the information in a bundle using the technique of bundle methods. In the following, we use usual concepts in bundle methods, such as “serious iterate” and “null iterate”. We recall that a trial point is called a serious iterate if it decreases sufficiently the objective function, otherwise it is called a null iterate. In each iteration, gradients are required to be calculated at the current point and at some randomly generated nearby points. Using this information, a piecewise linear model is generated and a candidate descent direction is obtained by solving a quadratic program. Note that either the new direction reduces the objective function and a serious point is obtained, or we have a null point and the piecewise linear model must be improved. We need to perturb the null and serious iterates in the set of differentiable points of the objective function. Unlike bundle methods, after computation a new serious iterate the bundle is emptied and contrary to GS methods no line search procedure is employed.

We are interested in the situation where for a given point only inexact information of the function and subgradient (gradient) values is available. Nonvanishing perturbations are considered in both function and subgradient (gradient) values and should be bounded. We highlight that the proposed algorithm works completely the same trend for both exact and inexact information and it does not require any additional procedure to handle inexact information. Moreover, the global convergence is proved under mild conditions. More precisely, we show that if the number of serious iterates is finite and the exact gradient values (inexact function and gradient values) are provided, then with probability one the latest serious iterate is stationary (approximate stationary). On the other hand, if an infinite number of serious iterates is obtained, then with probability one every accumulation point of this sequence is stationary in the exact case and approximate stationary in the inexact case.

There are two other works where inexact information is studied in bundle methods for nonconvex single objective problems [12, 30]. The algorithm in [30] utilizes the downshift mechanism to deal with the nonconvexity of the objective function, while similar to our work the method in [12] uses the redistributed proximal bundle algorithm. Although these three research works use the bundle method with inexact information, their algorithms and convergence techniques are quite different. Moreover, to the best of our knowledge, inexact information in GS methods has never been studied explicitly before.

The remainder of the paper is organized as follows. In Section 2, we review some basic definitions and results from nonsmooth analysis. In Section 3, the details of the new algorithm are provided and its convergence is presented in Section 4. Results of computational experience are reported in Section 5 and, finally, concluding remarks are given in Section 6.

2 Preliminaries

Throughout the paper, we use the following notations and definitions. Suppose that $\mathbb{R}^n$ defines the n-dimensional Euclidean space. We denote by $\langle u, v\rangle = \sum_{i=1}^n u_i v_i$ the inner product of two vectors $u, v \in \mathbb{R}^n$ and by $\|\cdot\|$ the standard Euclidean norm. For $x \in \mathbb{R}^n$ and $\varepsilon > 0$, $B(x,\varepsilon)$ ($\bar{B}(x,\varepsilon)$) is an open (closed) ball of the radius ε centered at x.

The function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if $f(\lambda x + (1-\lambda)y) \le \lambda f(x) + (1-\lambda) f(y)$, for all $x, y \in \mathbb{R}^n$ and $\lambda \in [0,1]$. The subdifferential of a convex function f at x is given by $\partial_c f(x) := \{\xi \in \mathbb{R}^n \mid f(y) \ge f(x) + \langle \xi, y - x\rangle,\ \forall y \in \mathbb{R}^n\}$. For any $\varepsilon \ge 0$, the ε-subdifferential [28] of a convex function f at x is defined as

$$\partial_\varepsilon f(x) := \{\xi \in \mathbb{R}^n \mid f(y) \ge f(x) + \langle \xi, y - x\rangle - \varepsilon,\ \forall y \in \mathbb{R}^n\}. \tag{1}$$

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

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

The subset of $\mathbb{R}^n$ where the objective function f is differentiable is defined as $\mathcal{D} := \{x \in \mathbb{R}^n,\ f \text{ is differentiable at } x\}$, and the Clarke subdifferential of f at any point x is given by $\partial f(x) = \operatorname{conv}\{\lim_{j\to\infty} \nabla f(y^j),\ y^j \to x,\ y^j \in \mathcal{D}\}$, and it coincides with the convex subdifferential for every convex function. Each element $\xi \in \partial f(x)$ is called a subgradient of f at x. It is well-known that $\partial f(x)$ is a nonempty convex compact set in $\mathbb{R}^n$. Further, the Clarke subdifferential $\partial f(x)$ is upper semicontinuous at every $x \in \mathbb{R}^n$.

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

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a locally Lipschitz function. A point $x^* \in \mathbb{R}^n$ is called a stationary point of f if $0 \in \partial f(x^*)$. If x^* is a local minimizer of f, then x^* is a stationary point of f. Therefore, the stationary condition is a necessary condition for local minimizers.

3 The gradient sampling proximal bundle algorithm

In this section, a new algorithm based on the combination of the GS algorithm [23] and the redistributed proximal bundle methods [11, 12, 15, 16] is introduced to solve nonsmooth nonconvex unconstrained optimization problems. Consider the following optimization problem

$$\min_{x \in \mathbb{R}^n} \quad f(x), \tag{2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is locally Lipschitz, but possibly nonsmooth and nonconvex.

In various types of real-world applications, calculating the exact values of the objective function and/or its subgradient (gradient) can be very expensive or sometimes impossible. In these optimization problems, only approximations of function values and/or subgradient (gradient) values are accessible. It particularly happens when f is given by some optimization problems, e.g., $f(x) = \max_{t \in T} F(t, x)$.

The definition of inexact information for the function value is simple. For a given point x and a noise tolerance $\delta \geq 0$, the approximate value for $f(x)$, which is denoted by $\widehat{f}(x)$, is defined as $|\widehat{f}(x) - f(x)| \leq \delta$. On the other hand, the definition of approximate subgradients is more complicated. When the objective function f is convex, the approximation of subgradients in the convex bundle methods refers to the ε-subdifferential $\partial_\varepsilon f(\cdot)$. Its beneficial property is that, $0 \in \partial_\varepsilon f(\bar{x})$ implies $f(\bar{x}) \leq \min_{x \in \mathbb{R}^n} f(x) + \varepsilon$. In this text, we deal with locally Lipschitz functions, which without convexity we can not expect a tool with the similar global convergence property. Therefore, we are working with the following natural approximate subdifferential for locally Lipschitz functions $\partial f(\cdot) + \bar{B}(0, \varepsilon)$. In what follows, we present our motivation for this choice. A point $x^* \in \mathbb{R}^n$ is called an approximate stationary point of f if $0 \in \partial f(x^*) + \bar{B}(0, \varepsilon)$. If x^* is a local minimizer of $f(\cdot) + \varepsilon\| \cdot - x^*\|$, then x^* is an approximate stationary point of f, which means that x^* is a stationary point of a small perturbation of f, i.e., $f(\cdot) + \varepsilon\| \cdot - x^*\|$. Therefore, at a point x, an element $g \in \mathbb{R}^n$ approximates within tolerance $\theta \geq 0$ some subgradients of f at x if $g \in \partial f(x) + \bar{B}(0, \theta)$. Hence, when the function f is differentiable at x, we use the approximate gradient $\widehat{\nabla} f(x) = \nabla f(x) + v$, where $v \in \bar{B}(0, \theta)$.

Assume that the algorithm is at the k-th outer iteration and at the ℓ-th inner iteration. Moreover, suppose that $x^k \in \mathbb{R}^n$ is the latest serious iterate. Motivated by the GS method, we assume that f is differentiable at x^k. Consider $B(x^k, \varepsilon_\ell)$ as the sample ball with the radius $\varepsilon_\ell > 0$. Let $x_0^k = x^k$ and choose $m \in \mathbb{N}$ points $\{x_j^k\}_{j=1}^m$ independently and uniformly from $B(x^k, \varepsilon_\ell) \cap \mathcal{D}$. Since the locally Lipschitz function f is almost everywhere differentiable, this step is successful with probability one. Since the algorithm is at ℓ-th inner iteration, we have ℓ null iterates in hand. Therefore, the index set is defined by

$$\begin{aligned} \mathcal{L}_\ell^k :&= \mathcal{L}_0^k \cup \{m+1, m+2, \ldots, m+\ell\} \\ &= \{0, 1, 2, \ldots, m\} \cup \{m+1, m+2, \ldots, m+\ell\} = \{0, 1, 2, \ldots, m+\ell\}. \end{aligned}$$

As usual in the bundle methods, already generated information is used to obtain a piecewise linear model for the objective function. Here by using the sampling points and combined with the bundle technique, the piecewise linear model is defined. If the objective function is convex, the piecewise linear model is stated as a lower approximation for it [28]. In our case f is locally Lipschitz and therefore it is possibly nonconvex. Hence motivated by the presented method in [11, 12, 15, 16], we use the augmented function as $f_{\eta_\ell^k}(d, x^k) := f(x^k + d) + \frac{\eta_\ell^k}{2}\|d\|^2$, where $\eta_\ell^k \in \mathbb{R}$ is a positive parameter, that adjusted dynamically. Since we handle with inexact information the augmented objective function with the approximate function value is considered, i.e., $\widehat{f}_{\eta_\ell^k}(d, x^k) := \widehat{f}(x^k + d) + \frac{\eta_\ell^k}{2}\|d\|^2$.

The inexact function and gradient values at x_j^k are defined as $\widehat{f}(x_j^k) = f(x_j^k) - \delta_j^k$ and $\widehat{\nabla} f(x_j^k) = \nabla f(x_j^k) + v_j^k$, where $v_j^k \in B(0, \theta_j^k)$ for all $j \in \mathcal{L}_\ell^k$, ℓ and k. Note that δ_j^k can be positive or negative, so the true function value can be either overestimated or underestimated, however θ_j^k is nonnegative. In addition, both noise terms are assumed to be bounded, thus there exist $\bar{\delta} > 0$ and $\bar{\theta} > 0$ such that $|\delta_j^k| \leq \bar{\delta}$ and $0 \leq \theta_j^k \leq \bar{\theta}$ for all $j \in \mathcal{L}_\ell^k$ and $\ell, k \in \mathbb{N}$. It is worth mentioning that the noise terms and their bounds are generally unknown and we do not assume any link between $\bar{\delta}$ and $\bar{\theta}$.

The piecewise linear model for the augmented function $\widehat{f}_{\eta_\ell^k}$ is formed at the ℓ-th iteration as follows:

$$M_\ell(d, x^k) := \widehat{f}(x^k) + \max_{j \in \mathcal{L}_\ell^k}\{-c_j^k + \langle \xi_j^k, d\rangle\}, \tag{3}$$

where for all $j \in \mathcal{L}_\ell^k$ we have $e_j^k = \widehat{f}(x^k) - \widehat{f}(x_j^k) - \langle \widehat{\nabla} f(x_j^k), x^k - x_j^k \rangle$, $c_j^k = e_j^k + \eta_\ell^k b_j^k$, $\xi_j^k = \widehat{\nabla} f(x_j^k) + \eta_\ell^k (x_j^k - x^k)$, $b_j^k = \frac{\|x_j^k - x^k\|^2}{2}$ and the bundle is defined as $\mathcal{B}_\ell^k := \bigcup_{j \in \mathcal{L}_\ell^k} \left\{ \left(\widehat{\nabla} f(x_j^k), e_j^k, b_j^k, x_j^k - x^k \right) \right\}$. Our aim is to keep c_j^k nonnegative, for all $j \in \mathcal{L}_\ell^k$. For this purpose, we take

$$\eta_\ell^k \geq \max\{ \max_{j \in \mathcal{L}_\ell^k \setminus \{0\}} \frac{-2e_j^k}{\|x_j^k - x^k\|^2}, \omega\} + \omega, \tag{4}$$

where $\omega > 0$ is a positive constant. By using (4), for all $j \in \mathcal{L}_\ell^k \setminus \{0\}$, we have $c_j^k = e_j^k + \frac{\eta_\ell^k}{2} \|x_j^k - x^k\|^2 \geq \frac{\omega}{2} \|x_j^k - x^k\|^2 \geq 0$. On the other hand, for $x_0^k = x^k$ we obtain $c_0^k = 0$. Therefore, $c_j^k \geq 0$ for all $j \in \mathcal{L}_\ell^k$. Since the latest serious iterate is one of the bundle points, it follows that $M_\ell(0, x^k) = \widehat{f}(x^k) + \max_{j \in \mathcal{L}_\ell^k} \{-c_j^k\} = \widehat{f}(x^k)$. In addition, by (3) and $M_\ell(0, x^k) = \widehat{f}(x^k)$ we deduce $M_\ell(d, x^k) \geq M_\ell(0, x^k) + \langle \xi_j^k, d \rangle - c_j^k$ for all $j \in \mathcal{L}_\ell^k$ and $d \in \mathbb{R}^n$. Using the definition of the ε-subdifferential in (1), we obtain

$$\xi_j^k \in \partial_{c_j^k} M_\ell(0, x^k), \quad \forall j \in \mathcal{L}_\ell^k. \tag{5}$$

To generate the candidate descent direction d_ℓ^k, our bundle method chooses a proximal parameter $\mu_\ell^k > 0$ and solves the following quadratic problem

$$\min_{d \in \mathbb{R}^n} \quad M_\ell(d, x^k) + \frac{\mu_\ell^k}{2} \|d\|^2. \tag{6}$$

Clearly d_ℓ^k is unique, since the objective function is strictly convex. Set

$$v_\ell^k := M_\ell(d_\ell^k, x^k) - \widehat{f}(x^k). \tag{7}$$

If $d_\ell^k = 0$, then $v_\ell^k = 0$ and the algorithm stops. Therefore, we assume $d_\ell^k \neq 0$. By uniqueness of d_ℓ^k as the solution of Problem (6), we get $M_\ell(d_\ell^k, x^k) + \frac{\mu_\ell^k}{2} \|d_\ell^k\|^2 < M_\ell(0, x^k) + \frac{\mu_\ell^k}{2} \|0\|^2 = M_\ell(0, x^k) = \widehat{f}(x^k)$. On the other hand, since $\frac{\mu_\ell^k}{2} \|d_\ell^k\|^2 \geq 0$, we have $M_\ell(d_\ell^k, x^k) < \widehat{f}(x^k)$ and so $v_\ell^k < 0$.

Problem (6) can be rewritten in the following smooth form

$$\begin{aligned} \min_{d \in \mathbb{R}^n, v \in \mathbb{R}} \quad & v + \frac{\mu_\ell^k}{2} \|d\|^2 \\ & \langle \xi_j^k, d \rangle - c_j^k \leq v, \quad \forall j \in \mathcal{L}_\ell^k. \end{aligned} \tag{8}$$

The quadratic dual problem of (8) is formulated as follows:

$$\begin{aligned} \min_{\lambda_j \geq 0, \forall j \in \mathcal{L}_\ell^k} \quad & \frac{1}{2\mu_\ell^k} \| \sum_{j \in \mathcal{L}_\ell^k} \lambda_j \xi_j^k \|^2 + \sum_{j \in \mathcal{L}_\ell^k} \lambda_j c_j^k \\ & \sum_{j \in \mathcal{L}_\ell^k} \lambda_j = 1. \end{aligned} \tag{9}$$

By using the relationship between the primal and dual solutions, if λ_j for all $j \in \mathcal{L}_\ell^k$ solve Problem (9), then we have $v_\ell^k = -\Big(\frac{1}{\mu_\ell^k} \| \sum_{j \in \mathcal{L}_\ell^k} \lambda_j \xi_j^k \|^2 + \sum_{j \in \mathcal{L}_\ell^k} \lambda_j c_j^k\Big)$ and $d_\ell^k = -\frac{1}{\mu_\ell^k} \sum_{j \in \mathcal{L}_\ell^k} \lambda_j \xi_j^k$.

If $x^k + d_\ell^k$ satisfies the descent test

$$\widehat{f}(x^k + d_\ell^k) \leq \widehat{f}(x^k) + m_L v_\ell^k, \tag{10}$$

where $0 < m_L < 1$, then we have a new serious iterate and set $x^{k+1} = x^k + d_\ell^k$. This implies that the function value obtained at this serious iterate is significantly better than the function value at the previous serious point. We just only need to perturb the point x^{k+1} in $\mathcal{D}$. Otherwise, when the condition (10) is not satisfied, we set $x_{m+\ell+1}^k = x^k + d_\ell^k$ and perform a null step and the model will be modified by adding new information to the bundle. As we have done in the previous case, if $x_{m+\ell+1}^k \notin \mathcal{D}$, then we replace it with any point in $\mathcal{D}$ which satisfies in some conditions (we exactly state the necessary conditions in the algorithm). Then, we will augment the piecewise linear model $M_\ell(\cdot, x^k)$ and generate $M_{\ell+1}(\cdot, x^k)$. For this purpose, the cutting plane with respect to $x_{m+\ell+1}^k$ should be added in the model. This is done by updating the index set and the bundle, i.e., $\mathcal{L}_{\ell+1}^k = \mathcal{L}_\ell^k \cup \{m + \ell + 1\}$ and $\mathcal{B}_{\ell+1}^k = \mathcal{B}_\ell^k \bigcup \{(\widehat{\nabla} f(x_{m+\ell+1}^k), e_{m+\ell+1}^k, b_{m+\ell+1}^k, d_{m+\ell+1}^k)\}$ and define $M_{\ell+1}(d, x^k) := \widehat{f}(x^k) + \max_{j \in \mathcal{L}_{\ell+1}^k} \{-c_j^k + \langle \xi_j^k, d \rangle\}$.

When after a fixed serious iteration, the algorithm executes a number of null iterations without calculating a new serious iterate, there are two possibilities. The first possibility is that the piecewise linear model is not a good approximation of the objective function. Most bundle methods try to correct this situation by adding new null iterates' information to the bundle in order to improve the model function. The second possibility is that the algorithm is close to a stationary point. GS algorithms try to overcome this situation by reducing the sample radius. In this paper, we try to use the ideas of both methods. For this purpose, first we try to improve the piecewise linear model by adding the information of new null iterates. We consider a counter (denoted by n_{null}) to account the number of null iterations and choose an upper bound $\max_{\text{null}}$ for it. If the null steps have been already performed the maximum number of times, i.e., $n_{\text{null}} = \max_{\text{null}}$, the algorithm may close to a stationary point. To investigate this issue, we reduce the radius of the sample ball and continue the algorithm. It is worth mentioning that although sample size $m \geq n + 1$ is required in most GS methods [6, 7, 23], we have no condition for m here. In numerical experiences, $\max_{\text{null}}$ is chosen equal to $2n$, because according to the numerical results in [6, 7] it yields faster convergence and good numerical results.

Now we introduce the gradient sampling proximal bundle algorithm (GSPB) for solving optimization problem (2).

Algorithm 1 The gradient sampling proximal bundle algorithm.

```
1: Initialization: Choose the line search parameter $m_L \in (0, 1)$, the reduction factor $\mu_\varepsilon \in (0, 1)$, the stopping tolerance tol $\geq 0$, the initial sample radius $\bar{\varepsilon} > 0$ and the upper bound $\max_{\mathrm{null}} \in \mathbb{N}$. Choose a starting point $x^1 \in \mathcal{D}$ and calculate $\widehat{f}(x^1)$ and $\widehat{\nabla} f(x^1)$. Set $k := 1$, $n_{\mathrm{null}} := 0$, $\varepsilon_0 = \bar{\varepsilon}$, $\mathcal{L}_0^1 := \{0\}$ and $\mathcal{B}_0^1 := \{(\widehat{\nabla} f(x^1), e_0^1, 0, 0)\}$.
2: Approximate $\varepsilon$–subdifferential by gradient sampling: Set $x_0^k := x^k$. Let $\{x_j^k\}_{j=1}^m$ be the sample points independently and uniformly from $B(x^k, \varepsilon_0) \cap \mathcal{D}$. Calculate $\widehat{f}(x_j^k)$ and $\widehat{\nabla} f(x_j^k)$ for $j = 1, 2, \ldots, m$ and set $\mathcal{B}_0^k = \cup_{j=0}^m \{(\widehat{\nabla} f(x_j^k), e_j^k, b_j^k, x_j^k - x^k\}$, $\mathcal{L}_0^k := \{0, 1, \ldots, m\}$, $\ell = 0$ and $n_{\mathrm{null}} = 0$.
3: New point generation and stopping test: Select $\eta_\ell^k > 0$ as in (4) and using (3) formulate $M_\ell(d, x^k)$. Select a proximal parameter $\mu_\ell^k > 0$ and by solving the subproblem (6) obtain $d_\ell^k$. Compute $v_\ell^k$, if $-v_\ell^k \leq$ tol, stop.
4: Test serious iterate: If (10) does not satisfy go to Step 5, otherwise we have a serious iterate. If $x^k + d_\ell^k \in \mathcal{D}$ then set $x^{k+1} = x^k + d_\ell^k$. Otherwise (i.e., $x^k + d_\ell^k \notin \mathcal{D}$), let $x^{k+1}$ be any point in $\mathcal{D}$ satisfying $\widehat{f}(x^{k+1}) \leq \widehat{f}(x^k) + m_L v_\ell^k$ and $\|x^k + d_\ell^k - x^{k+1}\| \leq \varepsilon_\ell$. Call $x^{k+1}$ as a new serious iterate. Set $k = k + 1$ and $\varepsilon_0 = \bar{\varepsilon}$. Go to Step 2.
5: Test null iterate: If $x^k + d_\ell^k \in \mathcal{D}$, set $x_{m+\ell+1}^k = x^k + d_\ell^k$. Otherwise let $x_{m+\ell+1}^k$ be any point in $\mathcal{D}$ satisfying
       $$\widehat{f}(x_{m+\ell+1}^k) > \widehat{f}(x^k) + m_L v_\ell^k, \tag{11a}$$
       $$\|x^k + d_\ell^k - x_{m+\ell+1}^k\| \leq \varepsilon_\ell. \tag{11b}$$
   Improve the piecewise linear model by adding the corresponding information with $x_{m+\ell+1}^k$ to the bundle, that is, $\mathcal{B}_{\ell+1}^k = \mathcal{B}_\ell^k \cup \{\widehat{\nabla} f(x_{m+\ell+1}^k), e_{m+\ell+1}^k, d_{m+\ell+1}^k, b_{m+\ell+1}^k)\}$, $\mathcal{L}_{\ell+1}^k = \mathcal{L}_\ell^k \cup \{m + \ell + 1\}$ and $n_{\mathrm{null}} = n_{\mathrm{null}} + 1$. If $n_{\mathrm{null}} = \max_{\mathrm{null}}$, set $n_{\mathrm{null}} = 1$ and go to Step 6. Otherwise set $\ell = \ell + 1$ and go to Step 3.
6: Update sample radius: Set $\varepsilon_{\ell+1} = \mu_\varepsilon \varepsilon_\ell$, $\ell = \ell + 1$ and go to Step 3.
```

Some explanations to Algorithm 1 are essential. We note that, along with the standard gradient sampling [6, 23], the algorithm keeps every iterates x^k and x_ℓ^k in the set $\mathcal{D}$. In Step 4, if $x^k + d_\ell^k \notin \mathcal{D}$, x^{k+1} can be chosen as follows. For $i = 1, 2, \ldots$ sample x^{k+1} can be found from an uniform distribution on $B(x^k + d_\ell^k, \varepsilon_\ell / i)$ until $x^{k+1} \in \mathcal{D}$ and (10) holds. By continuity of f this process terminates with probability one. In Step 5, if $x^k + d_\ell^k \notin \mathcal{D}$, $x_{m+\ell+1}^k$ can be determined like the previous procedure such that $x_{m+\ell+1}^k \in \mathcal{D}$, (11a) and (11b) hold.

Remark 1 In the sequel of this paper, we suppose that $\{\eta_\ell^k\}_\ell$ is bounded. Using (4), we deduce that $\eta_\ell^k \geq 2\omega$ for all ℓ, therefore we assume that there exists $\bar{\eta}$ such that $2\omega \leq \eta_\ell^k \leq \bar{\eta}$ for all ℓ. Since the value of $\bar{\eta}$ is not needed in the performance and the analysis of the algorithm, this assumption is not restrictive on the implementation of the algorithm. The boundedness of $\{\eta_\ell^k\}_\ell$ has been considered in [12] for the unconstrained problems with lower $-C^1$ functions and in [15, 16] for constrained problems with regular functions. However, we consider this assumption for unconstrained optimization problems with locally Lipschitz functions.

4 Global convergence

First, for analytical purposes and motivated by [15, 16, 31], we define the upper envelope model associated with the cutting planes. Then, as a lemma we state some of its useful properties. The upper envelope model is a helpful tool to prove the global convergence.

Consider a given point $x \in \mathbb{R}^n$. Suppose that $\bar{B}(x, \bar{\varepsilon})$ is a fixed closed ball such that it contains all possible trial steps y^+. Set $\mathcal{B}(x) := \{y^+ |\ y^+ \in \bar{B}(x, \bar{\varepsilon}),\ y^+ \text{ is a trial point}\}$. The upper envelope model $M^{\uparrow}(\cdot, x) : \mathbb{R}^n \to \mathbb{R}$ for the objective function f is defined as

$$M^{\uparrow}(d, x) := \widehat{f}(x) + \sup_{2\omega \le \eta \le \bar{\eta},\ y^+ \in \mathcal{B}(x),\ \xi \in \partial f(y^+) + B(0, \bar{\theta})} \{m_{y^+, \xi, \eta}(d, x)\},$$

and $\bar{\eta}$ is determined in Remark 1. The plane $m_{y^+, \xi, \eta}(y, x)$ is the cutting plane at the serious iterate x and the trial step y^+ as following

$$m_{y^+, \xi, \eta}(d, x) := -\frac{\omega}{2} \|y^+ - x\|^2 + \langle \xi + \eta(y^+ - x), d\rangle.$$

The boundedness of $\bar{B}(x, \bar{\varepsilon})$ and the definition of η imply that $M^{\uparrow}(\cdot, x)$ is defined everywhere. Some useful properties of the upper envelope model $M^{\uparrow}(\cdot, x)$ are stated in Lemma 1. The proof of this lemma follows immediately from the proof of [16, Lemma 5] for unconstrained problems (only item (iv) needs to be modified).

Lemma 1 *Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is a locally Lipschitz function, then:*

(i) $M^{\uparrow}(\cdot, x)$ *is a convex function.*
(ii) $M_\ell(\cdot, x) \le M^{\uparrow}(\cdot, x)$, $\forall \ell$.
(iii) $M^{\uparrow}(0, x) = \widehat{f}(x)$.
(iv) $\partial_c M^{\uparrow}(0, x) \subseteq \partial f(x) + \bar{B}(0, \bar{\theta})$.

Now we examine the convergence properties of Algorithm 1. We consider various cases that may be happened during its execution.

Theorem 1 *Assume that f is locally Lipschitz, $tol = 0$ and $\mu_\ell^k > 0$, for all k and ℓ. If Algorithm 1 stops with a finite number of iterations, then with probability one the latest serious iterate x^k is an approximate stationary point of Problem (2).*

Proof According to the assumptions, Algorithm 1 stops and this happens at Step 3 with $-v_\ell^k \le 0$. Since we always have $-v_\ell^k \ge 0$, we conclude that $-v_\ell^k = 0$.

By definition $-v_\ell^k = \frac{1}{\mu_\ell^k}\|\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k\|^2 + \sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k$ and using $\mu_\ell^k > 0$, we deduce

$$\|\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k\|^2 = 0, \quad \text{and} \quad \sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k = 0. \tag{12}$$

By (5) we have $\xi_j^k \in \partial_{c_j^k} M_\ell(0,x^k)$, and consequently $M_\ell(d,x^\ell) \geq M_\ell(0,x^k) + \langle\xi_j^k, d\rangle - c_j^k$ for all $j \in \mathcal{L}_\ell^k$ and $d \in \mathbb{R}^n$. Taking into account that $\lambda_j \geq 0$ and $\sum_{j\in\mathcal{L}_\ell^k}\lambda_j = 1$, we get $M_\ell(d,x^k) \geq M_\ell(0,x^k) + \langle\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k, d\rangle - \sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k$. By $M_\ell(0,x^k) = \widehat{f}(x^k)$ and Lemma 1 (ii)-(iii), we conclude that $M^\uparrow(d,x^k) \geq M^\uparrow(0,x^k) + \langle\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k, d\rangle - \sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k$. Using (12) and Lemma 1 (i), (iv), we get $0 \in \partial_c M^\uparrow(0,x^k)$ and hence $0 \in \partial f(x^k) + \bar{B}(0,\bar{\theta})$. Consequently, the latest serious iterate x^k is an approximate stationary point.

From now on, we assume that Algorithm 1 does not stop and generates an infinite sequence of iterates. We consider various cases that may occur during the execution of Algorithm 1 with infinite cycles.

Theorem 2 *Assume that f is locally Lipschitz, $\{\eta_\ell^k\}_\ell$ is bounded above, there exists $\bar{\mu} > 0$ such that $\mu_\ell^k \leq \bar{\mu}$ for all k and ℓ and the level set $A(x^1) := \{x \in \mathbb{R}^n,\ f(x) \leq f(x^1)\}$ is bounded. If Algorithm 1 performs infinite serious iterates, then with probability one every accumulation point of the sequence of serious iterates is an approximate stationary point of Problem* (2).

Proof By assumption there exists a sequence $\{x^k\}_k$ and with probability one $\{x^k\}_k \subseteq \mathcal{D}$. The method is descent type thus we have $\{x^k\}_k \subseteq A(x^1)$. Since f is locally Lipschitz and $A(x^1)$ is bounded, the sequences $\{\widehat{f}(x^k)\}_k$ is bounded below. On the other hand, for each k we have $\widehat{f}(x^{k+1}) \leq \widehat{f}(x^k) + m_L v_\ell^k$ and hence $v_\ell^k \to 0$, as $k \to \infty$. By the definition of v_ℓ^k, we have

$$-v_\ell^k = \Big(\frac{1}{\mu_\ell^k}\|\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k\|^2 + \sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k\Big).$$

Since $c_j^k \geq 0$ and $\lambda_j \geq 0$, we deduce that $\sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k$ and $\frac{1}{\mu_\ell^k}\|\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k\|^2$ are nonnegative. Therefore, both are convergence to zero, i.e.,

$$\sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k \to 0 \quad \text{and} \quad \frac{1}{\mu_\ell^k}\|\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k\|^2 \to 0.$$

Since $\mu_\ell^k \leq \bar{\mu}$, we have $\frac{1}{\mu_\ell^k} \geq \frac{1}{\bar{\mu}}$ and hence

$$\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k \to 0, \quad \text{and} \quad \sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k \to 0, \quad k \to \infty. \tag{13}$$

We have $\{x^k\}_k \subseteq A(x^1)$ and due to boundedness of the set $A(x^1)$, there exist a convergent subsequence $\{x^k\}_{k\in\mathcal{A}} \subseteq \{x^k\}_k$ and $x^* \in \mathbb{R}^n$ satisfying $x^k \to_{k\in\mathcal{A}} x^*$. By (5) we have $\xi_j^k \in \partial_{c_j^k} M_\ell(0, x^k)$ and consequently $M_\ell(d, x^k) \geq M_\ell(0, x^k) + \langle \xi_j^k, d\rangle - c_j^k$, for all $j \in \mathcal{L}_\ell^k$ and $d \in \mathbb{R}^n$. By Lemma 1 (ii) and (iii), $\lambda_j \geq 0$ and $\sum_{j\in\mathcal{L}_\ell^k} \lambda_j = 1$ we have $M^\uparrow(d, x^k) \geq M^\uparrow(0, x^k) + \langle \sum_{j\in\mathcal{L}_\ell^k} \lambda_j \xi_j^k, d\rangle - \sum_{j\in\mathcal{L}_\ell^k} \lambda_j c_j^k$. Passing to the limit in this relation when $k \in \mathcal{A}$ and $k \to \infty$ and using (13), we obtain $M^\uparrow(d, x^*) \geq M^\uparrow(0, x^*) + \langle 0, d\rangle$. By Lemma 1 (i) and (iv), we deduce $0 \in \partial f(x^*) + \bar{B}(0, \bar{\theta})$ and thus x^* is an approximate stationary point.

Theorem 3 *Assume that f is a locally Lipschitz function, $\{\eta_\ell^k\}_\ell$ is bounded above and $\mu_\ell^k \leq \mu_{\ell+1}^k \leq \bar{\mu}$ for all ℓ. If Algorithm 1 performs infinite iterations with a finite number of serious iterations, then with probability one the latest serious iterate x^k is an approximate stationary point of Problem (2).*

Proof Suppose Algorithm 1 produces a finite number of serious iterations followed by an infinite number of null iterations. Therefore, throughout this proof, k and the latest serious point x^k are constant. We demonstrate that it is an approximate stationary point. Suppose that d_ℓ^k is the optimal solution of Problem (8) and $x_{m+\ell+1}^k = x^k + d_\ell^k$. First, we show that the sequence $\{M_\ell(d_\ell^k, x^k) + \frac{\mu_k}{2}\|d_\ell^k\|^2\}_\ell$ is bounded above and nondecreasing.

Since Problem (6) is strictly convex, it follows that its solution (i.e., d_ℓ^k) is unique thus $M_\ell(d_\ell^k, x^k) + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 < M_\ell(d, x^k) + \frac{\mu_\ell^k}{2}\|d\|^2$, for all $d \in \mathbb{R}^n$ and $d \neq d_\ell^k$. Now set $d = 0$, then $M_\ell(d_\ell^k, x^k) + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 < M_\ell(0, x^k) = \widehat{f}(x^k)$. Therefore, the sequence $\{M_\ell(d_\ell^k, x^k) + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2\}_\ell$ is bounded from above by $\widehat{f}(x^k)$. Next, let us prove that this sequence is nondecreasing. We have

$$\begin{aligned}
&M_{\ell+1}(d_{\ell+1}^k, x^k) + \frac{\mu_{\ell+1}^k}{2}\|d_{\ell+1}^k\|^2 \\
&\qquad\geq M_{\ell+1}(d_{\ell+1}^k, x^k) + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k\|^2 \\
&\qquad\geq \widehat{f}(x^k) - c_j^k + \langle \xi_j^k, d_{\ell+1}^k\rangle + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k\|^2 \\
&\qquad= \widehat{f}(x^k) - c_j^k + \langle \xi_j^k, d_\ell^k\rangle + \langle \xi_j^k, d_{\ell+1}^k - d_\ell^k\rangle + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k + d_\ell^k\|^2 \\
&\qquad= \widehat{f}(x^k) - c_j^k + \langle \xi_j^k, d_\ell^k\rangle + \langle \xi_j^k, d_{\ell+1}^k - d_\ell^k\rangle + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2 \\
&\qquad\quad + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 + \mu_\ell^k\langle d_{\ell+1}^k - d_\ell^k, d_\ell^k\rangle,
\end{aligned} \tag{14}$$

where the first inequality follows from $\mu_\ell^k \leq \mu_{\ell+1}^k$, the second inequality holds by the definition of $M_\ell(\cdot, x^k)$ and the other relations are obvious. The above relation is satisfied for all $j \in \mathcal{L}_{\ell+1}^k$ and further $\mathcal{L}_{\ell+1}^k = \mathcal{L}_\ell^k \bigcup \{m + \ell + 1\}$. For all $j \in$

$\mathcal{L}_\ell^k$, multiplying the relation (14) with corresponding λ_j, summing up and due to $\sum_{j\in\mathcal{L}_\ell^k}\lambda_j = 1$, we arrive at

$$M_{\ell+1}(d_{\ell+1}^k, x^k) + \frac{\mu_{\ell+1}^k}{2}\|d_{\ell+1}^k\|^2 \geq \widehat{f}(x^k) - \sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k + \langle \sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k, d_\ell^k\rangle$$

$$+ \langle \sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k, d_{\ell+1}^k - d_\ell^k\rangle + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2 + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 + \mu_\ell^k\langle d_{\ell+1}^k - d_\ell^k, d_\ell^k\rangle.$$

Since $d_\ell^k = -\frac{1}{\mu_\ell^k}\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k$, we have

$$\begin{aligned}
&M_{\ell+1}(d_{\ell+1}^k, x^k) + \frac{\mu_{\ell+1}^k}{2}\|d_{\ell+1}^k\|^2 \\
&\geq \widehat{f}(x^k) - \sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 - \frac{1}{\mu_\ell^k}\langle \sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k, \sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k\rangle + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2 \\
&= \widehat{f}(x^k) - \big(\sum_{j\in\mathcal{L}_\ell^k}\lambda_j c_j^k + \frac{1}{\mu_\ell^k}\|\sum_{j\in\mathcal{L}_\ell^k}\lambda_j\xi_j^k\|^2\big) + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2 \\
&= \widehat{f}(x^k) + v_\ell^k + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2 \\
&= M_\ell(d_\ell^k, x^k) + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2 \\
&\geq M_\ell(d_\ell^k, x^k) + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2.
\end{aligned}$$

Therefore, the sequence $\{M_\ell(d_\ell^k, x^k) + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2\}_\ell$ is nondecreasing and due to its boundedness, we deduce it is convergent. Assume that there exists $M^* \in \mathbb{R}$ such that $M_\ell(d_\ell^k, x^k) + \frac{\mu_k}{2}\|d_\ell^k\|^2 \to M^*$ as $\ell \to \infty$. From the above relation we have

$$M_{\ell+1}(d_{\ell+1}^k, x^k) + \frac{\mu_{\ell+1}^k}{2}\|d_{\ell+1}^k\|^2 \geq M_\ell(d_\ell^k, x^k) + \frac{\mu_\ell^k}{2}\|d_\ell^k\|^2 + \frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2.$$

Passing to the limit in this inequality when $\ell \to \infty$, we get

$$M^* \geq M^* + \lim_{\ell\to\infty}\frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2,$$

hence $\lim_{\ell\to\infty}\frac{\mu_\ell^k}{2}\|d_{\ell+1}^k - d_\ell^k\|^2 \leq 0$. By assumption we have $\mu_\ell^k \leq \mu_{\ell+1}^k \leq \bar{\mu}$, therefore $\mu_\ell^k \geq \mu_{\ell^*}^k$. That is $\{\mu_\ell^k\}_\ell$ is bounded below by $\mu_{\ell^*}^k$, where $\ell^* := \max\{\ell | \ell \text{ is a serious iteration}\}$. This implies that $d_{\ell+1}^k - d_\ell^k \to 0$, as $\ell \to \infty$. Using the definition of $M_{\ell+1}(d_{\ell+1}^k, x^k)$ for all $j \in \mathcal{L}_{\ell+1}^k$ we have $M_{\ell+1}(d_{\ell+1}^k, x^k) \geq \widehat{f}(x^k) - c_j^k + \langle \xi_j^k, d_{\ell+1}^k \rangle$. Set $j = m + \ell + 1$ in this relation, we obtain

$$\begin{aligned}
&M_{\ell+1}(d_{\ell+1}^k, x^k) \\
&\geq \widehat{f}(x^k) - c_{m+\ell+1}^k + \langle \xi_{m+\ell+1}^k, d_{\ell+1}^k \rangle \\
&= \widehat{f}(x_{m+\ell+1}^k) + \langle \widehat{\nabla} f(x_{m+\ell+1}^k), x^k - x_{m+\ell+1}^k \rangle - \frac{\eta_{\ell+1}^k}{2} \|x_{m+\ell+1}^k - x^k\|^2 \\
&\quad + \langle \xi_{m+\ell+1}^k, d_{\ell+1}^k \rangle \\
&\geq \widehat{f}(x_{m+\ell+1}^k) + \langle \widehat{\nabla} f(x_{m+\ell+1}^k), x^k - x_{m+\ell+1}^k \rangle - \eta_{\ell+1}^k \|x^k - x_{m+\ell+1}^k\|^2 \\
&\quad + \langle \xi_{m+\ell+1}^k, d_{\ell+1}^k \rangle \\
&= \widehat{f}(x_{m+\ell+1}^k) - \langle \widehat{\nabla} f(x_{m+\ell+1}^k) + \eta_{\ell+1}^k (x_{m+\ell+1}^k - x^k), x_{m+\ell+1}^k - x^k \rangle \\
&\quad + \langle \xi_{m+\ell+1}^k, d_{\ell+1}^k \rangle \\
&= \widehat{f}(x_{m+\ell+1}^k) - \langle \xi_{m+\ell+1}^k, x_{m+\ell+1}^k - x^k \rangle + \langle \xi_{m+\ell+1}^k, d_{\ell+1}^k \rangle \\
&= \widehat{f}(x_{m+\ell+1}^k) + \langle \xi_{m+\ell+1}^k, d_{\ell+1}^k - (x_{m+\ell+1}^k - x^k) \rangle \\
&= \widehat{f}(x_{m+\ell+1}^k) + \langle \xi_{m+\ell+1}^k, d_{\ell+1}^k - d_\ell^k \rangle - \langle \xi_{m+\ell+1}^k, x_{m+\ell+1}^k - (x^k + d_\ell^k) \rangle \\
&> \widehat{f}(x^k) + m_L v_\ell^k + \langle \xi_{m+\ell+1}^k, d_{\ell+1}^k - d_\ell^k \rangle - \langle \xi_{m+\ell+1}^k, x_{m+\ell+1}^k - (x^k + d_\ell^k) \rangle.
\end{aligned}$$

Due to the definition of $v_{\ell+1}^k$ on (7), we have $v_{\ell+1}^k = M_{\ell+1}(d_{\ell+1}^k, x^k) - \widehat{f}(x^k)$ and get

$$\begin{aligned}
0 \leq -v_{\ell+1}^k &= \widehat{f}(x^k) - M_{\ell+1}(d_{\ell+1}^k, x^k) \\
&< -m_L v_\ell^k + \langle \xi_{m+\ell+1}^k, d_\ell^k - d_{\ell+1}^k \rangle + \langle \xi_{m+\ell+1}^k, x_{m+\ell+1}^k - (x^k + d_\ell^k) \rangle \\
&\leq -m_L v_\ell^k + \langle \xi_{m+\ell+1}^k, d_\ell^k - d_{\ell+1}^k \rangle + \|\xi_{m+\ell+1}^k\| \|x_{m+\ell+1}^k - (x^k + d_\ell^k)\| \\
&\leq -m_L v_\ell^k + \langle \xi_{m+\ell+1}^k, d_\ell^k - d_{\ell+1}^k \rangle + \|\xi_{m+\ell+1}^k\| \varepsilon_\ell.
\end{aligned}$$

Since the number of the null iterates is infinite, by the algorithm process, we deduce $\varepsilon_\ell \to 0$. On the other hand, we have $d_{\ell+1}^k - d_\ell^k \to 0$, as $\ell \to \infty$, $m_L \in (0, 1)$ and $\{\xi_\ell^k\}_\ell$ is bounded, hence $-v_\ell^k \to 0$. By the definition of v_ℓ^k and $c_j^k \geq 0$ for all $j \in \mathcal{L}_\ell^k$ and this fact that $\mu_\ell^k \leq \bar{\mu}$, we get

$$\sum_{j \in \mathcal{L}_\ell^k} \lambda_j \xi_j^k \to 0, \quad \text{and} \quad \sum_{j \in \mathcal{L}_\ell^k} \lambda_j c_j^k \to 0, \quad \ell \to \infty. \tag{15}$$

By using the relation (5), we have $\xi_j^k \in \partial_{c_j^k} M_\ell(0, x^k)$, and by the definition of ε-subdifferential we deduce $M_\ell(d, x^k) \geq M_\ell(0, x^k) + \langle \xi_j^k, d \rangle - c_j^k$, for all $j \in \mathcal{L}_\ell^k$ and $d \in \mathbb{R}^n$. By multiplying $\lambda_j \geq 0$ in this relation, summing up and due to $\sum_{j \in \mathcal{L}_\ell^k} \lambda_j = 1$ we obtain $M_\ell(d, x^k) \geq \widehat{f}(x^k) + \langle \sum_{j \in \mathcal{L}_\ell^k} \lambda_j \xi_j^k, d \rangle - \sum_{j \in \mathcal{L}_\ell^k} \lambda_j c_j^k$. From Lemma 1 (ii)-(iii), we get $M^{\uparrow}(y, x^k) \geq M^{\uparrow}(0, x^k) + \langle \sum_{j \in \mathcal{L}_\ell^k} \lambda_j \xi_j^k, d\rangle - \sum_{j \in \mathcal{L}_\ell^k} \lambda_j c_j^k$. Taking the limit as $\ell \to \infty$ and using (15), we deduce $M^{\uparrow}(d, x^k) \geq M^{\uparrow}(0, x^k) + \langle 0, d\rangle$. Therefore, due to Lemma 1 (i) and (iv) we obtain $0 \in \partial_c M^{\uparrow}(0, x^k)$ and so $0 \in \partial f(x^k) + \bar{B}(0, \bar{\theta})$.

In the sequel, we state the results in the exact case.

Remark 2 Consider the obtained necessary conditions, we observe that:

(i) If $\bar{\delta} = \bar{\theta} = 0$, that is we have the exact function and gradient values then we have that if Algorithm 1 generates an infinite serious sequence, then every accumulation point of the generated sequence is a stationary point. On the other hand, if Algorithm 1 produces a finite number of serious iterates, then the latest serious iterate is a stationary point.

(ii) If $\bar{\theta} = 0$, that is no error in the gradient values but errors in the function values are allowed, then the results in (i) are still correct.

(iii) Adding inexact information does not cause any additional difficulty to the performance of the algorithm.

5 Numerical experiments

In this section, the performance and efficiency of the Gradient Sampling Proximal Bundle (GSPB) algorithm are presented. We first investigate the performance of GSPB when the information is exact (i.e., $\bar{\delta} = \bar{\theta} = 0$), by comparing it with other concurrent nonsmooth optimization algorithms. Furthermore, we numerically test GSPB for various kinds of inexactness and study the effect of noise on the accuracy of the solution.

5.1 Solvers and their implementations

We employ three optimization algorithms in our comparison; including Hybrid Algorithm for Nonsmooth Optimization (HANSO 2.2) [6, 26], the unconstrained version of the infeasible proximal bundle method (IPBM) [16] and a solver for local nonlinear optimization problems, which is an implementation of Shor's algorithm (SolvOpt) [34]. All codes are written in MATLAB R2016a and run on a PC Intel Core I5 with CPU 2.5 GHz and 4GB of RAM. For HANSO 2.2, IPBM and SolvOpt, the parameters are set to the default values suggested by the respective codes and the stopping parameter is chosen tol $:= 10^{-8}$. We set the parameters of Algorithm 1 as tol $:= 10^{-8}$, $\bar{\varepsilon} := 0.1$, $\mu_\varepsilon := 0.1$, $m_L := 10^{-2}$, $\omega := 1.2$, $\max_{\text{null}} := 2n$ and $m := n$. The proximal parameter is considered one in all iterations, i.e., $\mu_\ell^k := 1$ for all k and ℓ. The quadratic programming solver is quadprog.m, which is available in the MATLAB optimization toolbox. In our results, to select η_ℓ^k we use the relation (4) with equality, i.e., $\eta_\ell^k = \max\{\max_{j \in \mathcal{L}_\ell^k \setminus \{0\}} \frac{-2e_j^k}{\|x_j^k - x^k\|^2}, \omega\} + \omega$.

5.2 Comparison with different solvers in the exact case

To evaluate the performance of GSPB and compare it with the aforementioned algorithms, some nonsmooth test problems of [4] are used; for more details see Table 1. We use the notation n for the number of variables and f_{opt} for the optimal value. The number of function and subgradient (gradient) evaluations are considered as measures of efficiency.

First, we applied algorithms for solving problems P1–P20 with the given starting points in [4]. Results are presented in Table 2. The numerical experiments demonstrate that GSPB has an acceptable behavior for these problems as comparing with the previously mentioned algorithms; for more details see Table 2. We use the following notations: f_{final} is the value of the objective function at the final point and n_f, n_ξ and $n_{\nabla f}$ are the number of function evaluations, subgradient evaluations, and gradient evaluations, respectively.

In the next step, for each problem we use 20 randomly generated starting points (by applying rand.m and randn.m functions in MATLAB) and the starting points are the same for all algorithms. To compare the performance of the algorithms, we apply an indicator: n_b — the number of successful runs considering the best known solution. We say that an algorithm finds a solution to a problem with tolerance $\varepsilon > 0$

Table 1 Description of test problems

No.	Name Problem	Problem type	n	f_{opt}
P1	CB2	Nonsmooth convex	2	1.9522245
P2	CB3	Nonsmooth convex	2	2
P3	DEM	Nonsmooth convex	2	-3
P4	QL	Nonsmooth convex	2	7.2
P5	LQ	Nonsmooth convex	2	$-\sqrt{2}$
P6	Miffilin 1	Nonsmooth convex	2	-1
P7	Wolfe	Nonsmooth convex	2	-8
P8	Rosen-Suzuki	Nonsmooth convex	4	-44
P9	Davidon	Nonsmooth convex	4	115.70644
P10	Shor	Nonsmooth convex	5	22.600162
P11	Crescent	Nonsmooth nonconvex	2	0
P12	Miffilin 2	Nonsmooth nonconvex	2	-1
P13	WF	Nonsmooth nonconvex	2	0
P14	Spiral	Nonsmooth nonconvex	2	0
P15	EVD 52	Nonsmooth nonconvex	3	3.5997193
P16	PBC 3	Nonsmooth nonconvex	3	$0.42021427 \times 10^{-2}$
P17	Brad	Nonsmooth nonconvex	3	$0.50816327 \times 10^{-1}$
P18	Kowalik-Osborne	Nonsmooth nonconvex	4	$0.80843684 \times 10^{-2}$
P19	OET 5	Nonsmooth nonconvex	4	$0.26359735 \times 10^{-2}$
P20	OET 6	Nonsmooth nonconvex	4	$0.20160753 \times 10^{-2}$

Table 2 Results of P1–P20 with given starting points

	GSPB			HANSO 2.2		IPBM		SolvOpt		
No.	n_f	$n_{\nabla f}$	f_{final}	$n_f, n_{\nabla f}$	f_{final}	n_f, n_ξ	f_{final}	n_f	n_ξ	f_{final}
P1	63	83	1.9522	128	1.9522	48	1.9522	99	32	1.9522
P1	67	89	1.9522	108	1.9522	47	1.9522	92	31	1.9522
P2	7	11	2	192	2.0000	20	2	81	30	2.0000
P3	104	136	-3.0000	92	-2.9998	36	-3.0000	250	92	-3.0000
P4	45	65	7.2000	131	7.2000	24	7.2000	85	27	7.2000
P5	28	50	-1.4142	156	-1.4142	17	-1.4142	59	14	-1.4142
P6	126	148	-1.0000	32	4.0000	128	-1.0000	55	34	-0.8286
P7	238	274	-8.0000	96	-8.0000	69	-8.0000	120	34	-8
P8	738	839	-44.0000	140	-44.0000	96	-44.0000	147	55	-44.0000
P9	369	481	115.7064	356	115.7064	69	115.7064	211	75	115.7064
P10	616	756	22.6002	410	22.6002	92	22.6002	118	46	22.6002
P11	38	66	0.0000	175	0.0000	15	0.0002	261	50	0.0000
P12	60	88	-1.0000	247	-1.0000	28	-1.0000	85	27	-1.0000
P13	88	120	0.0000	91	0.0000	73	0.0000	172	33	0.0000
P14	475	526	0.1152	96	0.0774	615	0.0769	291	128	0.0743
P15	92	125	3.5997	131	3.5997	53	3.5997	144	46	3.5997
P16	319	505	0.0042	254	0.0042	90	0.0042	263	73	0.0042
P17	304	487	0.0508	468	0.0508	87	0.0508	118	37	0.0508
P18	47	69	0.0081	860	0.0081	615	0.0081	248	79	0.0081
P19	2039	2979	0.0027	2622	0.0029	61	0.0074	2445	662	0.0027
P20	1999	2640	0.0020	2701	0.0020	678	0.0020	938	258	0.0020

if $|f_{\text{final}} - f_{\text{opt}}|/(1 + |f_{\text{opt}}|) \leq \varepsilon$. In our experiment $\varepsilon = 5 \times 10^{-4}$. Results of this part are presented in Table 3. We use the following notations: n_f^{ave}, n_ξ^{ave} and $n_{\nabla f}^{\text{ave}}$ are the average number of function evaluations, subgradient evaluations and gradient evaluations, respectively. These results show that GSPB, SolvOpt, IPBM and HANSO 2.2 can solve 388, 373, 368 and 344 problems, respectively.

In order to provide a better picture of the algorithms, we analyze the results using performance profiles [9]. We compare the performance of the solvers both in terms of the number of function evaluations and in terms of the number of subgradient (gradient) evaluations.

In the performance profiles, the value of $\rho_s(\tau)$ at $\tau = 0$ determines the ratio of test problems for which the solver s is the best, i.e., the solver s uses the least number of function evaluations or the least number of subgradient (gradient) evaluations. Note that the value of $\rho_s(\tau)$ on the rightmost abscissa shows the ratio of test problems that the solver s can solve, that is, the robustness of the solver s. In addition, the higher is a particular curve, the better is the corresponding solver.

It is clear from the performance profile figures that the GSPB method is more efficient and accurate with given and randomly starting points than the HANSO 2.2. with the number of function evaluations (see parts (a) of Figs. 1 and 3) and the number

Table 3 Average results of P1–P20 with 20 randomly starting points

	GSPB			HANSO 2.2		IPBM		SolvOpt		
No.	n_b	n_f^{ave}	$n_{\nabla f}^{\text{ave}}$	n_b	$n_f^{\text{ave}}, n_{\nabla f}^{\text{ave}}$	n_b	$n_f^{\text{ave}}, n_\xi^{\text{ave}}$	n_b	n_f^{ave}	n_ξ^{ave}
P1	20	64.60	87.10	20	109.50	20	50.40	20	90.25	29.55
P2	20	90.05	114.60	20	193.95	20	33.80	20	94.80	31.55
P3	20	83.20	114.40	13	115.05	20	35.55	20	187.90	54.20
P4	20	32.25	48.25	20	120.90	20	40.05	20	105.30	32.65
P5	20	25.20	42.20	20	170.65	20	29.60	20	91.00	29.55
P6	19	125.00	150.35	14	131.15	18	167.20	8	183.05	82.60
P7	20	306.65	349.60	20	182.10	20	70.65	20	95.65	29.25
P8	20	817.55	924.45	20	257.05	20	185.00	20	136.55	46.50
P9	20	428.80	542.00	20	459.30	20	82.05	20	256.60	84.00
P10	20	695.05	865.35	20	422.15	20	112.75	20	155.45	54.95
P11	15	85.10	109.30	20	169.25	10	56.95	20	214.95	41.90
P12	20	46.20	70.50	20	274.25	20	27.90	20	90.00	27.00
P13	16	81.30	105.65	17	69.70	10	39.25	20	120.30	26.25
P14	18	64.50	117.15	3	35.15	18	116.70	16	434.95	123.68
P15	20	110.30	152.90	20	244.85	20	55.70	20	124.25	40.4.00
P16	20	205.05	337.75	19	333.30	19	135.75	20	182.90	55.85
P17	20	194.30	254.25	13	585.95	19	155.65	18	188.60	50.70
P18	20	770.90	947.20	15	881.50	18	103.70	16	198.70	63.15
P19	20	329.65	3975.45	20	728.50	20	433.60	20	2086.30	567.60
P20	20	4649.90	6049.10	10	1793.30	16	617.40	15	903.00	285.40
Sum	388			344		368		373		

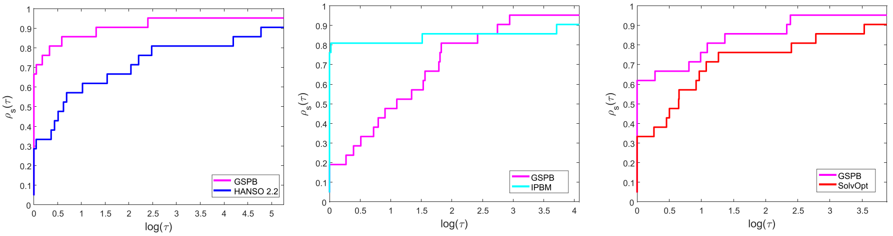
of gradient evaluations (see parts (a) of Figs. 2 and 4). Since it is superior to HANSO 2.2 in all figures.

By comparing the performance profiles of GSPB with IPBM, we deduce that IPBM is better than GSPB for the most of the test problems with given and randomly starting points. On the other hand, in all cases IPBM can solve approximately 90% of problems while GSPB can solve 95% of problems; see parts (b) of Figs. 1, 2, 3 and 4. This means that GSPB is more robust than IPBM.

Due to parts (c) of Figs. 1 and 3, we get GSPB is more accurate and efficient than SolvOpt for both given and randomly starting points with the number of function evaluations. But if we consider the number of subgradient (gradient) evaluations as a measure of efficiency, we deduce that SolvOpt is better solver than GSPB; see parts (c) of Figs. 2 and 4.

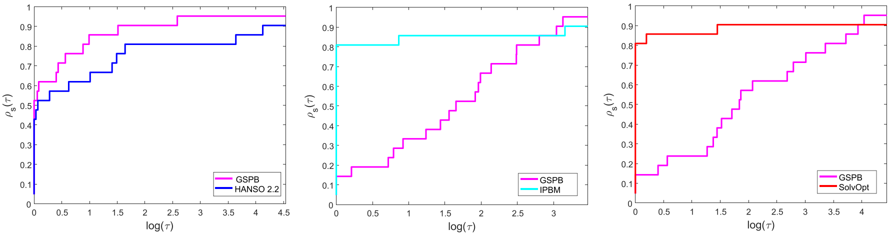
5.3 Impact of noise on solution accuracy

In order to study the impact of the inexact information on the GSPB method, we use the Ferrier polynomials as a collection of nonconvex test problems, which further used in



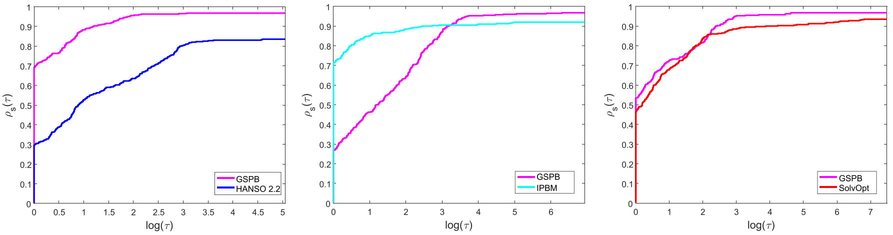
(a) (b) (c)

Fig. 1 Performance profile with the number of function evaluations, given starting points



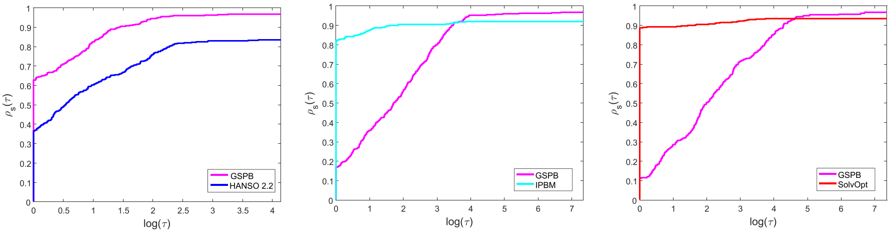
(a) (b) (c)

Fig. 2 Performance profile with the number of subgradient (gradient) evaluations, given starting points



(a) (b) (c)

Fig. 3 Performance profile with the number of function evaluations, random starting points



(a) (b) (c)

Fig. 4 Performance profile with the number of subgradient (gradient) evaluations, random starting points

[11, 12]. For each $i = 1, 2, 3, \dots, n$, the function $h_i : \mathbb{R}^n \to \mathbb{R}$ is defined as $h_i(x) = (ix_i^2 - 2x_i) + \sum_{j=1}^n x_j$. There are five classes of test functions defined by h_i (see [10]) as $f_1(x) := \sum_{i=1}^n |h_i(x)|$, $f_2(x) := \sum_{i=1}^n (h_i(x))^2$, $f_3(x) := \max_{i=1,2,\dots,n} |h_i(x)|$, $f_4(x) := \sum_{i=1}^n |h_i(x)| + \frac{1}{2}\|x\|^2$ and $f_5(x) := \sum_{i=1}^n |h_i(x)| + \frac{1}{2}\|x\|$. It has been proved in [11] that these test functions are nonconvex and globally lower $- C^1$ in $\mathbb{R}^2$ and they are nonsmooth except f_2. These properties carry to higher dimensions as well. We consider the following test problems

$$\min_{x \in \mathbb{R}^n} \quad f_k(x),$$

for $k = 1, 2, 3, 4, 5$ and $n = 2, 3, 4, \dots, 20$ which are called Problem 1–Problem 5. We set $x^1 = [1, \frac{1}{2}, \frac{1}{3}, \dots, \frac{1}{n}]$ as a starting point.

To introduce perturbations to the available information, at each evaluation we add randomly generated elements to the exact function values and gradient values, with norm less than or equal to $\bar{\delta}$ and $\bar{\varepsilon}$, respectively. We test five different forms of the noise:

- N_0 : No noise, $\bar{\varepsilon} = \varepsilon_j = 0$ and $\bar{\delta} = \delta_j = 0$ for all $j \in \mathcal{L}_\ell^k$.
- $N_c^{f,\xi}$: Constant noise, $\bar{\varepsilon} = \varepsilon_j = 0.01$ and $\bar{\delta} = \delta_j = 0.01$ for all $j \in \mathcal{L}_\ell^k$.
- $N_v^{f,\xi}$: Changing noise, $\bar{\varepsilon} = 0.01$, $\varepsilon_j = \min\{0.01, \frac{\|x_j\|}{100}\}$, $\bar{\delta} = 0.01$ and $\delta_j = \min\{0.01, \frac{\|x_j\|}{100}\}$ for all $j \in \mathcal{L}_\ell^k$.
- N_c^{ξ} : Constant gradient noise, $\bar{\varepsilon} = \varepsilon_j = 0$ and $\bar{\delta} = \delta_j = 0.01$ for all $j \in \mathcal{L}_\ell^k$.
- N_v^{ξ} : Changing gradient noise, $\bar{\varepsilon} = \varepsilon_j = 0$ and $\delta_j = \min\{0.01, \frac{\|x_j\|}{100}\}$ for all $j \in \mathcal{L}_\ell^k$.

The first noise form, N_0, is used as a benchmark for comparison. The noise form $N_c^{f,\xi}$ is representative of inexact function and gradient values with a constant noise. The third, $N_v^{f,\xi}$, is representative of inexact values with a changing noise. The fourth and fifth, N_c^{ξ} and N_v^{ξ}, represent the exact function values with inexact gradient values. In order to preserve the random nature, for noise forms $N_c^{f,\xi}$, $N_v^{f,\xi}$, N_c^{ξ} and N_v^{ξ}, we repeat each test 20 times.

For all functions the global minimum is zero. We use the following formula Accuracy $= |\log_{10}(f_j^k)|$ to check the accuracy of GSPB with different noises. In Figs. 5 and 6, we plot the accuracy of the achieved results, when running the GSPB algorithm until satisfaction of its stopping test. For ease the interpretation of the graphs, we also report the results with no noise. We used colors blue, red, magenta, green and black for Problem 1–Problem 5 respectively and we employ these colors for all dimensions $n = 2, 3, \dots, 20$. Figure 5 reports the results for constant noises (for N_0, N_c^{ξ} and $N_c^{f,\xi}$) and Fig. 6 states the results for changing noises (for N_0, N_v^{ξ} and $N_v^{f,\xi}$). We see that when function values are exact the accuracy is better than the situations where both the function and gradients values are inexact. In the most cases the accuracy with a changing noise is better than the accuracy with a constant noise.

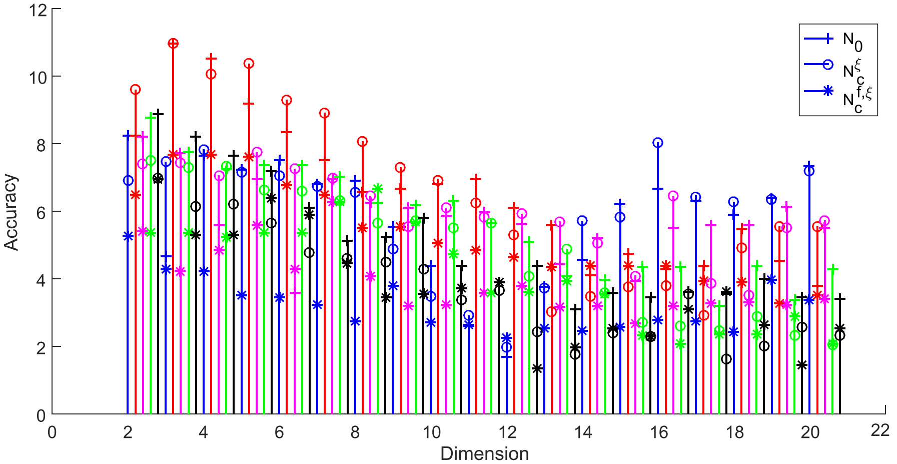


Fig. 5 Accuracy at termination for noise forms N_0, N_c^{ξ} and $N_c^{f,\xi}$ (blue, red, magenta, green and black are used for Problem 1–Problem 5, respectively)

6 Conclusions

We have proposed and analyzed a new algorithm for unconstrained nonsmooth nonconvex optimization problems. This method combines the advantages of the well-known GS and proximal bundle methods. It is a descent method, easy to implement and supports both exact and inexact information. At each iteration, the objective function is approximated by a piecewise linearworking model $M_{\ell}(y, x^k)$ based on the bundle

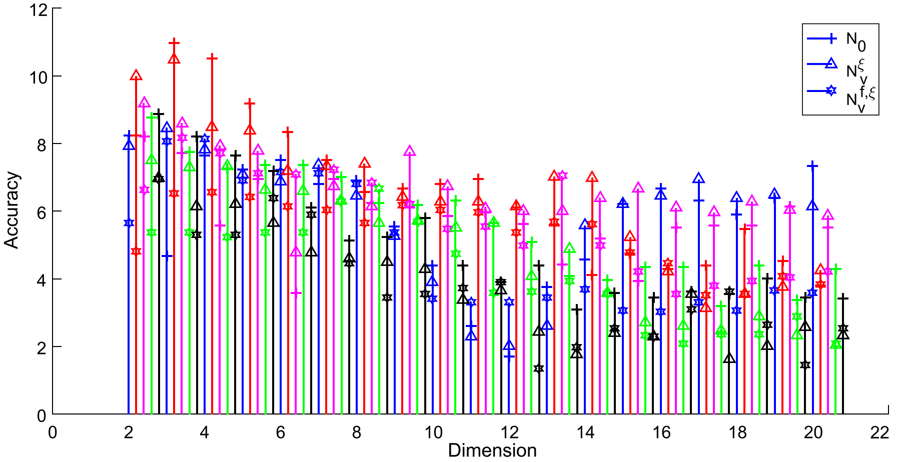


Fig. 6 Accuracy at termination for noise forms N_0, N_v^{ξ} and $N_v^{f,\xi}$ (blue, red, magenta, green and black are used for Problem 1–Problem 5, respectively)

methods. Then the proximal term is added to the model $M_\ell(y, x^k)$ to guarantee the existence and uniqueness of the minimum point of the subproblem (6) and also to keep the approximation local enough. The algorithm is globally convergent to a stationary point with exact information and to an approximately stationary point with inexact information. On the other hand, we need the proximal parameter μ_ℓ^k to be positive and the sequence $\{\mu_\ell^k\}$ be bounded above and as a sequence of ℓ, it must be nondecreasing. Therefore, it can be considered as a fixed sequence, i.e., $\mu_\ell^k = c > 0$ for all ℓ and k. The value of c can be considered arbitrarily small without any affect on the convergence of algorithm (since all required assumptions are satisfied).

The new method was tested using different nonsmooth unconstrained test problems and compared with several other nonsmooth solvers. Furthermore, in order to better analyze the numerical results we use the performance profile. The obtained results demonstrate that the proposed method is efficient and robust for solving nonsmooth nonconvex optimization problems, although in some cases it may require a large number of gradient evaluations; where it is usual in gradient sampling based methods. As mentioned in our numerical experiments, the proximal parameter is considered one in all iterations, i.e., $\mu_\ell^k = 1$ for all k and ℓ. Although the value of this parameter does not affect the convergence analysis, it is effective in the performance of the algorithm and its execution speed (numerical results). Investigating the effect of the proximal parameter value on the performance and speed of the algorithm is an interesting topic that is beyond the scope of this research. This can be considered as a topic for future research. Furthermore, we are interested in extending the proposed algorithm to solve nonsmooth nonconvex constrained optimization problems with exact and inexact information in future.

Acknowledgements The authors would like to thank anonymous referees for their comments that helped to improve the quality of the paper.

Author Contributions N. Hoseini Monjezi and S. Nobakhtian contributed equally to drafting this manuscript (material preparation and data collection, design, analysis and implementation of the algorithm, writing and editing the first draft of the manuscript). Both authors read and approved the final manuscript.

Funding The research was supported by a grant from IPM.

Data Availability Data availability not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Ethics approval and consent to participate Not applicable

Consent for publication Not applicable

Human and animal ethics Not applicable

Competing interests The authors declare no competing interests.

References

1. Bagirov, A.M.: Continuous subdifferential approximations and their applications. J. Math. Sci., 115, 2567–2609 (2003)
2. Bagirov, A.M., Hoseini Monjezi, N., Taheri, S.: An augmented subgradient method for minimizing nonsmooth DC functions. Comput. Optim. Appl. 80, 411–438 (2021)
3. Bagirov, A.M., Jin, L., Karmitsa, N., Nuaimat, A.Al., Sultanova N.: Subgradient method for nonconvex nonsmooth optimization. J. Optim. Theory Appl., 157, 416–435 (2013)
4. Bagirov, A.M., Karmitsa, N., Mäkelä, M.M.: Introduction to Nonsmooth Optimization: theory, practice and software. Springer (2014)
5. Bagirov, A.M., Taheri, S., Joki, K., Karmitsa, N., Mäkelä, M.M.: Aggregate subgradient method for nonsmooth DC optimization. Optim. Lett., 15, 83–96 (2021)
6. Burke, J., Lewis, A., Overton, M.: A robust gradient sampling algorithm for nonsmooth, nonconvex optimization. SIAM J. Optim., 15, 571–779 (2005)
7. Curtis, F.E., Overton, M.L.: A sequential quadratic programming algorithm for nonconvex, nonsmooth constrained optimization. SIAM J. Optim., 22(2), 474–500 (2012)
8. de Oliveira, W., Sagastizábal, C., Lemaréchal, C.: Convex proximal bundle methods in depth: a unified analysis for inexact oracles. Math. Program., 148(1–2), 241–277 (2014)
9. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Math. Program., 91(2), 201–213 (2002)
10. Ferrier, C.: Bornes Duales de Problémes d'Optimisation Polynomiaux, Ph.D. thesis, Laboratoire Approximation et Optimisation, Université Paul Sabatier, France (1997)
11. Hare, W., Sagastizábal, C.: A redistributed proximal bundle method for nonconvex optimization. SIAM J. Optim., 20, 2442–2473 (2010)
12. Hare, W., Sagastizábal, C., Solodov, M.: A proximal bundle method for nonsmooth nonconvex functions with inexact information. Comput. Optim. Appl., 63, 1–28 (2016)
13. Hintermüller, M.: A proximal bundle method based on approximate subgradients. Comput. Optim. Appl., 20, 245–266 (2001)
14. Hoseini, N., Nobakhtian, S.: A new trust region method for nonsmooth nonconvex optimization. Optimization, 67, 1265–1286 (2018)
15. Hoseini Monjezi, N., Nobakhtian, S.: A filter proximal bundle method for nonsmooth nonconvex constrained optimization. J. Glob. Opti., 79, 1–37 (2021)
16. Hoseini Monjezi, N., Nobakhtian, S.: A new infeasible proximal bundle algorithm for nonsmooth nonconvex constrained optimization. Comput. Optim. Appl., 74(2), 443–480 (2019)
17. Hoseini Monjezi, N., Nobakhtian, S.: A proximal bundle-based algorithm for nonsmooth constrained multiobjective optimization problems with inexact data. Numer. Algor., 89, 637–674 (2022)
18. Hoseini Monjezi, N., Nobakhtian, S.: An inexact multiple proximal bundle algorithm for nonsmooth nonconvex multiobjective optimization problems. Ann. Oper. Res., 311, 1123–1154 (2022)
19. Hoseini Monjezi, N., Nobakhtian, S.: Convergence of the proximal bundle algorithm for nonsmooth nonconvex optimization problems. Optim. Lett., 16, 1495–1511 (2022)
20. Hoseini Monjezi, N., Nobakhtian, S. Pouryayevali, M.R.: Proximal bundle algorithm for nonsmooth optimization on riemannian manifolds, IMA J. Numer. Anal., **43**(1), 293–325 (2023)
21. Kiwiel, K.C.: Approximations in proximal bundle methods and decomposition of convex programs. J. Optim. Theory Appl., 84, 529–548 (1995)
22. Kiwiel, K.C.: Convergence of approximate and incremental subgradient methods for convex optimization. SIAM. J. Optim., 14(3), 807–840 (2004)
23. Kiwiel, K.C.: Convergence of the gradient sampling algorithm for nonsmooth nonconvex optimization. SIAM J. Optim., 18, 379–388 (2007)
24. Kiwiel, K.C.: Efficiency of Proximal Bundle Methods. J. Optim. Theory Appl., 104, 589–603 (2000)
25. Lemaréchal, C.: Bundle methods in nonsmooth optimization, in: Nonsmooth Optimization (Laxenburg, 1977), Lemaréchal C., Mifflin, R. (eds.), IIASA Proc. Ser. 3, Pergamon Press, Oxford, 79–102 (1978)
26. Lewis, A.S., Overton, M.L.: Nonsmooth Optimization via Quasi-Newton Methods. Math. Program., 141(1–2), 135–163 (2013)
27. Lv, J., Pang, LP., Xu, N., Xiao, Z.H.: An infeasible bundle method for nonconvex constrained optimization with application to semi-infinite programming problems. Numer. Algor., 80, 397–427 (2019)
28. Mäkelä, M.M., Neittaanmäki, P.: Nonsmooth Optimization: Analysis and Algorithms with Applications to Optimal Control. World Scientific, Singapore (1992)

29. Nesterov, Y.: Primal-dual subgradient methods for convex problems. Math. Program., 120(1), 221–259 (2009)
30. Noll, D.: Bundle method for non-convex minimization with inexact subgradients and function values. In: Computational and Analytical Mathematics, vol. **50**, pp. 555–592. Springer Proceedings in Mathematics (2013)
31. Noll, D.: Cutting plane oracles to minimize non-smooth non-convex functions. Set-Valued Var. Anal., 18, 531–568 (2010)
32. Pang, LP., Wu, Q.: A feasible proximal bundle algorithm with convexification for nonsmooth, nonconvex semi-infinite programming. Numer. Algor., 90, 387–422 (2022)
33. Qi, L., Sun, J.: A trust region algorithm for minimization of locally Lipschitzian functions. Math. Program., 66, 25–43 (1994)
34. Shor, N.Z.: Minimization methods for non-differentiable functions. Springer (1985)
35. Solodov, M.V., Zavriev, S.K.: Error stability properties of generalized gradient-type algorithms. J. Optim. Theory Appl., 98, 663–680 (1998)
36. Tang, C., Liu, S., Jian, J., Li. J.: A feasible SQP-GS algorithm for nonconvex, nonsmooth constrained optimization. Numer. Algor., **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)