A Gradient Sampling method based on Ideal direction for solving nonsmooth nonconvex optimization problems
convergence analysis and numerical experiments

M. Maleknia · M. Shamsi

Received: date / Accepted: date

Abstract In this paper, a modification to the Gradient Sampling (GS) method for minimizing nonsmooth nonconvex functions is presented. One drawback in GS method is the need of solving a Quadratic optimization Problem (QP) at each iteration, which is time-consuming especially for large scale objectives. To resolve this difficulty, we propose a new descent direction, namely Ideal direction, for which there is no need to consider any quadratic or linear optimization subproblem. It is shown that, this direction satisfies Armijo step size condition and can be used to make a substantial reduction in the objective function. Furthermore, we prove that using Ideal directions preserves the global convergence of the GS method. Moreover, under some moderate assumptions, we present an upper bound for the number of serious iterations. Using this upper bound, we develop a different strategy to study the convergence of the method. We also demonstrate the efficiency of the proposed method using small, medium and large scale problems in our numerical experiments.

Keywords nonsmooth and nonconvex optimization · subdifferential · steepest descent direction · gradient sampling · Armijo line search

Mathematics Subject Classification (2010) 49M05 · 65K05 · 90C26

1 Introduction

In this paper, we consider the following unconstrained minimization problem

\[ \min f(x) \quad \text{s.t.} \quad x \in \mathbb{R}^n \]  \hspace{1cm} (1)
where \( f : \mathbb{R}^n \to \mathbb{R} \) is locally Lipschitz and continuously differentiable on an open set \( D \) with full measure in \( \mathbb{R}^n \). These types of problems arise in many applications, such as optimal control, image processing, data analysis, economics, chemistry and biology ([1,2,3,4,5]). Therefore, it is worthwhile to develop efficient algorithms for solving such problems. During the last three decades, a lot of effort has gone into nonsmooth optimization. In particular, Clarke developed the concept of subgradient in [6] and soon after, it became the cornerstone of designing and developing many algorithms in nonsmooth optimization (see [1,7,8,9,10,11,12]).

When it comes to designing numerical methods for minimizing nonsmooth functions, there are serious challenges we need to deal with. The main difficulty is that, nonsmooth functions are generally not differentiable at stationary points. Finding a descent direction is not an easy task as well. In smooth case, any vector has an obtuse angle with the gradient is a descent direction. In particular, the vector \(-\nabla f(x)\) defines the steepest descent direction. In contrast, in the nonsmooth case a vector that is opposite to an arbitrary subgradient need not be a direction of descent and hence designing descent algorithms is much more complicated. These difficulties make most of classic methods in smooth optimization unsuitable for solving nonsmooth optimization problems. For instance, it is well known that when the ordinary steepest descent method is applied to a nonsmooth function, it generally fails to make a substantial reduction in objective function as it approaches a nonsmooth region.

There are various methods for locating minimizers of nonsmooth functions. The subgradient method, originally proposed by N. Shor [8], is one of the simplest methods for minimizing nonsmooth functions. Due to its simple structure, it is a popular method, although it suffers from some serious limitations such as slow convergence, lack of practical termination criterion and lack of descent. However, it is well known that some of its modifications are able to overcome these difficulties (see [13,14,15,16]). As another class of nonsmooth methods, we can refer to Bundle methods as one of the most efficient methods in nonsmooth optimization [9,10,7]. The key idea of bundle methods is to keep memory of computed subgradients at previous iterations to construct a piecewise linear model for objective function. These methods require a great deal of storage, therefore some of its modifications presented in [17,18] are more efficient for solving large scale nonsmooth problems. In class of subgradient and bundle methods, at each iteration, the user needs to supply at least one subgradient. However, in many cases computing only one subgradient is not an easy task. In such situations, Gradient Sampling (GS) methods are more efficient, because they obtain a search direction without explicit computation of subgradients.

The gradient sampling algorithm, originally developed by Burke, Lewis and Overton [19], is a descent method for solving problem (1). The method is robust and can be applied to a broad range of nonsmooth functions. A comprehensive discussion of the GS algorithm along with its last modifications can be found in the recent paper [20]. The convergence analysis and theoretical results of the GS algorithm were first developed in [19]. Soon after, these results were strengthen by the work of Kiwiel in [21]. In particular, under subtle modifications, Kiwiel derived a lower bound for step sizes and suggested a limited Armijo line search in which the number of backtracking steps can be managed through our choice of initial step size. Another version of the GS method in which the Clarke \( \varepsilon \)-subdifferential is approximated by sampling estimates of mollifier gradients presented by Kiwiel.
In the work of Curtis and Que [23], two novel strategies were introduced to improve the efficiency of the GS method. The first strategy is to combine the idea of sampling gradients and LBFGS update (see [24]) to approximate Hessian matrix. The second one uses the idea of sampling gradients to provide a model that overestimates the objective function. In both techniques, the dual of the quadratic subproblem is considered for warm-starting the QP solver. In the work of Curtis and Overton [25], the ideas of gradient sampling and Sequential Quadratic Programming (SQP) are combined for solving nonsmooth constrained optimization problems. In addition, the local convergence rate of the GS algorithm for the class of finite-max functions is studied in [26]. These continued developments clearly indicate that the GS technique has been a reach area of research.

However, the GS method suffers from two practical limitations. First, in order to obtain an approximation of \( \varepsilon \)-steepest descent direction, the GS method requires to compute gradient information at \( m \geq n + 1 \) randomly generated points during each iteration. It is noted that, for large-scale objectives, this may not be tractable at a moderate cost. To alleviate this difficulty, in [23], Curtis and Que proposed an adaptive strategy in which the convergence of the method is guaranteed only through \( O(1) \) gradient evaluations at each iteration. The second limitation is to solve the corresponding QP. As the number of variables increases, the size of this QP increases significantly which makes the method unsuitable for large-scale objectives. Furthermore, when we are far away from a nonsmooth region, since the information collected by sampling gradients are very close together, solving the quadratic subproblem is not reasonable. Nevertheless, since checking differentiability of \( f \) is not an easy task, the GS method and its modifications do not care about this fact. In this work, we address the second difficulty by introducing a new search direction.

In this paper, based on gradient sampling technique, we propose a new search direction, namely Ideal direction, for which there is no need to consider any kind of quadratic or linear subproblems. Furthermore, we show that this direction satisfies Armijo step size condition and provides a necessary optimality condition in the sense that it vanishes at optimality. The original GS method is modified using Ideal directions. By means of this new search direction, not only is there no need to solve the quadratic subproblem in smooth regions, but also we can reduce the number of quadratic subproblems once the method starts tracking a nonsmooth curve which leads to a stationary point. Moreover, we provide a comprehensive discussion for the convergence analysis of the method. We follow closely the work of Kiwiel in [21] to analyze the convergence of the proposed method. Of course, there are some differences due to using Ideal directions. In addition, thanks to the limited Armijo line search proposed in [21], we present an upper bound for the number of serious iterations generated in our method. Using this upper bound, a different strategy to study the global convergence behavior of the proposed method is developed assuming that the objective function is bounded below.

This paper is organized as follows. Section 2 provides some mathematical preliminaries used in this paper. A brief review over the GS method is presented in Sect. 3. In Sect. 4, we introduce Ideal direction and its main properties are examined. The proposed method and its convergence analysis are given in Sect. 5. Numerical results are reported in Sect. 6 and Sect. 7 concludes the paper.
2 Preliminaries

We use the following notations in this paper. As usual, \( \mathbb{R}^n \) is the \( n \)-dimensional Euclidean space and its inner product is denoted by \( \langle x, y \rangle := \sum_{i=1}^n x_i y_i \) that induces the associated Euclidean norm \( \|x\| := (x, x)^{1/2} \). \( B(x, \varepsilon) := \{ y \in \mathbb{R}^n : \|y - x\| \leq \varepsilon \} \) is the closed ball centered at \( x \) with radius \( \varepsilon \). Furthermore \( B_\varepsilon := B(0, \varepsilon) \).

A function \( f : \mathbb{R}^n \to \mathbb{R} \) is called locally Lipschitz \([27]\), if for every \( x \in \mathbb{R}^n \) there exist positive constants \( K_x \) and \( \varepsilon_x \) such that

\[
|f(z) - f(y)| \leq K_x \|z - y\|, \quad \text{for all } z, y \in B(x, \varepsilon_x).
\]

Let \( \Omega_f := \{ x \in \mathbb{R}^n : f \text{ is not differentiable at } x \} \), be the subset of \( \mathbb{R}^n \) where the function \( f \) is not differentiable. By Rademacher’s theorem \([28]\), every locally Lipschitz function is differentiable almost everywhere. Therefore, the Clarke subdifferential of a locally Lipschitz function \( f \) at a point \( x \in \mathbb{R}^n \) can be given by \([6]\)

\[
\partial f(x) := \text{co}\{ \xi \in \mathbb{R}^n : \exists \{x_k\} \subset \mathbb{R}^n \setminus \Omega_f \text{ s.t. } x_k \to x \text{ and } \nabla f(x_k) \to \xi \},
\]

in which \( \text{co} \) stands for the convex hull. It is shown in \([27]\) that the set valued map \( \partial f : \mathbb{R}^n \rightrightarrows \mathbb{R}^n \) is outer semicontinuous and the set \( \partial f(x) \) is a convex compact subset of \( \mathbb{R}^n \). The Clarke \( \varepsilon \)-subdifferential, which is the generalization of ordinary subdifferential, is defined by \([27]\)

\[
\partial_\varepsilon f(x) := \text{cl co} \partial f(x + B_\varepsilon).
\]

Clearly, the set valued map \( \partial_\varepsilon f : \mathbb{R}^n \rightrightarrows \mathbb{R} \) has a closed graph and hence it is outer semicontinuous \([29]\).

Suppose that \( f : \mathbb{R}^n \to \mathbb{R} \) is locally Lipschitz and continuously differentiable on an open set \( D \) with full measure in \( \mathbb{R}^n \). The following representation of the Clarke subdifferential is the key idea of approximating subdifferential set by sampling gradients \([6]\)

\[
\partial f(x) = \bigcap_{\varepsilon > 0} G_\varepsilon(x),
\]

such that

\[
G_\varepsilon(x) := \text{cl co}\{ \nabla f((x + B_\varepsilon) \cap D) \},
\]

where \( \text{cl} \) denotes the closure of a set. Let \( x \in \mathbb{R}^n \) be a differentiable point, \( \varepsilon > 0 \), \( m \in \mathbb{N} \) and \( u_1, \ldots, u_m \) be sampled uniformly and independently from \( B_1 \). If \( s_0 := x \) and \( s_i := x + \varepsilon u_i \) is a differentiable point for \( i = 1, \ldots, m \), then the gradient bundle \( G_\varepsilon^m(x) \) is defined as \([19, 21]\)

\[
G_\varepsilon^m(x) := \text{co} \{ \nabla f(s_0), \nabla f(s_1), \ldots, \nabla f(s_m) \}, \quad (2)
\]

in which \( \varepsilon > 0 \) and \( m \in \mathbb{N} \) are called sampling radius and sample size respectively. For a locally Lipschitz function \( f \), it is easy to see that

\[
G_\varepsilon^m(x) \subset G_\varepsilon(x) \subset \partial_\varepsilon f(x),
\]
and for $0 < \varepsilon_1 < \varepsilon_2$, we have $\partial_{\varepsilon_1} f(x) \subset G_{\varepsilon_2}(x)$. Furthermore, it is proved in [20] that for every $x \in \mathbb{R}^n$ and $\varepsilon > 0$

$$\partial f(x) \subset G_{\varepsilon}(x).$$

The following definition provides a useful point of view to study the concept of gradient sampling.

**Definition 2.1** For a locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}$ and optimality tolerance $\nu > 0$, a point $x \in \mathbb{R}^n$ is called a $(\nu, G^m_{\varepsilon}(x))$-stationary point if

$$\min\{\|g\| : g \in G^m_{\varepsilon}(x)\} \leq \nu.$$

We recall that, if $x \in \mathbb{R}^n$ is a local minimum of a locally Lipschitz function $f$, then it is necessary that [27]

$$0 \in \partial f(x).$$

The point $x \in \mathbb{R}^n$ satisfying the above condition is called Clarke stationary point. Furthermore, a point $x \in \mathbb{R}^n$ is called Clarke $\varepsilon$-stationary point if

$$0 \in \partial_{\varepsilon} f(x).$$

3 Background on GS method

The Gradient Sampling (GS) method [19] is a descent method for solving problem (1). At each iteration of the method, the steepest descent direction is approximated and utilized to make a substantial reduction in $f$. In this method, the concepts of steepest descent direction and choosing step size are crucial. In this respect, these concepts are briefly reviewed.

3.1 The steepest descent direction

We recall that, a direction $d \in \mathbb{R}^n$ is called a direction of descent for $f : \mathbb{R}^n \to \mathbb{R}$ at $x \in \mathbb{R}^n$, if there is $z_0 > 0$ such that

$$f(x + td) - f(x) < 0, \quad \text{for all } t \in (0, z_0].$$

(3)

In the descent methods, choosing a step size $t$ satisfying the above inequality does not guarantee the convergence of the method. In this regard, some sufficient decrease conditions are introduced [7]. Among them, an Armijo step size condition, which is commonly used in GS methods, is considered to state the following definition.

**Definition 3.1** Let $g \neq 0$, we say that $g$ induces an Armijo Descent Direction (ADD) $d := -\|g\|^{-1}g$ for $f$ at point $x$, if for each $c \in (0, 1)$ there is $z_0 > 0$ such that

$$f(x + td) - f(x) < -ct\|g\|, \quad \text{for all } t \in (0, z_0].$$

The above condition is called **sufficient decrease condition**.
If $f$ is smooth at a point $x \in \mathbb{R}^n$, then every descent direction at this point is an ADD too\[24\]. However, this is not the case when $f$ is not smooth at $x$. Therefore, in order to design a descent method for minimizing a nonsmooth function, providing an ADD is essential to ensure the convergence of the method.

To define the concept of steepest descent direction for a locally Lipschitz function more precisely, we need to generalize the classical directional derivative. For a smooth or convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, the classical directional derivative is defined as

\[
f'(x; d) := \lim_{t \downarrow 0} t^{-1}[f(x + td) - f(x)],
\]

and the steepest descent direction at the point $x \in \mathbb{R}^n$ is obtained by solving the problem

\[
\min_{\|d\| \leq 1} f'(x; d).
\]

However, the quantity (4) does not always exist for locally Lipschitz functions. In this regard, for locally Lipschitz functions, the Clarke generalized directional derivative is defined as \[27\]

\[
f^\circ(x; d) := \limsup_{y \rightarrow x, t \downarrow 0} t^{-1}[f(y + td) - f(y)].
\]

Moreover, it is shown that this directional derivative can be expressed as \[27\]

\[
f^\circ(x; d) = \max_{\xi \in \partial f(x)} \langle \xi, d \rangle,
\]

and naturally, the steepest descent direction of a locally Lipschitz function $f$ at the point $x \in \mathbb{R}^n$ can be obtained by solving the following min-max problem

\[
\min_{\|d\| \leq 1} \max_{\xi \in \partial f(x)} \langle \xi, d \rangle.
\]

(5)

Using von Neumann minimax theorem \[30\], the above problem is equivalent to the following problem

\[
\min_{\xi \in \partial f(x)} \|\xi\|,
\]

(6)

in the sense that, if $\xi^* \neq 0$ is the solution of the problem (6), then $d^* = -\|\xi^*\|^{-1}\xi^*$ solves the problem (5). As a result, to obtain the steepest descent direction it is sufficient to find the member of subdifferential set with minimum norm.

In an iterative descent method for minimizing a nonsmooth function, after some iterations, the method rapidly approaches a nonsmooth curve leading to a stationary point. Therefore, it is likely that, the method generates a sequence of smooth points that are close to the nonsmooth curve. In such cases, since the Clarke subdifferential is a singleton set, the steepest descent direction is not an efficient direction, while the Clarke $\varepsilon$-subdifferential is able to collect some information of the nonsmooth curve and hence the so-called $\varepsilon$-steepest descent direction is a more suitable search direction.

In a similar fashion, the $\varepsilon$-steepest descent direction at the point $x \in \mathbb{R}^n$ can be obtained from the solution of the following problem

\[
\min_{\xi \in \partial \varepsilon f(x)} \|\xi\|.
\]

(7)
However, since solving the problem (7) requires the knowledge of the whole subdifferential on \( B(x, \varepsilon) \), computing the \( \varepsilon \)-steepest descent direction is cumbersome. To overcome this drawback, since \( G^\varepsilon_m(x) \) is a proper inner approximation of \( \partial_\varepsilon f(x) \), the \( \varepsilon \)-steepest descent direction is approximated through replacing \( \partial_\varepsilon f(x) \) by \( G^\varepsilon_m(x) \) in the problem (7) and the approximate \( \varepsilon \)-steepest descent direction can be obtained from the following problem

\[
\min_{g \in G^\varepsilon_m(x)} \|g\|. \tag{8}
\]

Indeed, the problem (8) is a Quadratic Problem (QP). To state this fact, by Carathéodory’s theorem [31], this problem can be written in the following quadratic programming form

\[
\min_{\lambda} \frac{1}{2}\|G\lambda\|^2 \tag{9}
\]

s.t. \( \sum_{i=0}^{m} \lambda_i = 1, \ \lambda_i \geq 0, \ i = 0, \ldots, m, \)

where \( G := [\nabla f(s_0) \ \nabla f(s_1) \ldots \nabla f(s_m)] \in \mathbb{R}^{n \times (m+1)} \) and \( \lambda' := (\lambda_0, \lambda_1, \ldots, \lambda_m) \in \mathbb{R}^{m+1} \). We summarize the concept of approximate \( \varepsilon \)-steepest descent direction in the following definition.

**Definition 3.2** Suppose that \( 0 \not\in G^\varepsilon_m(x) \) and \( \lambda^* \) is the solution of the problem (9). Let \( g^*_\varepsilon := G\lambda^* \). Then the direction \( d^\varepsilon := -\|g^*_\varepsilon\|^{-1}g^*_\varepsilon \) is called (normalized) approximate \( \varepsilon \)-steepest descent direction.

It is noted that, the approximate \( \varepsilon \)-steepest descent direction is an ADD [19]. In other words, for the direction \( d^\varepsilon \) and each \( c \in (0, 1) \), there exists \( z_0 > 0 \) such that

\[
f(x + td^\varepsilon) - f(x) < -tc\|g^*_\varepsilon\|, \quad \text{for all } t \in (0, z_0]. \tag{10}
\]

### 3.2 Choosing stepsize and backtracking line search

Loosely speaking, the GS method is a descent method in which at each iteration the approximate \( \varepsilon \)-steepest descent direction is considered as the search direction. Therefore, the QP (9) is solved per iteration and then a backtracking line search is applied to find the step size \( t \) as stated in Algorithm 1.

**Algorithm 1:** Backtracking Armijo Line Search (BALS)

| inputs : \( x \in \mathbb{R}^n \), the directions \( d, g \in \mathbb{R}^n \), parameters \( \gamma, c \in (0, 1) \). |
| output : Step size \( t \). |
| required: \( d = -\|g\|^{-1}g \) is an ADD for \( f \) at \( x \). |

1. **Function** BALS\((x, g, d, \gamma, c)\):
   2. \( t := 1; \)
   3. while \( f(x + td) - f(x) \geq -ct\|g\| \) do
   4. \( t := \gamma t; \)
   5. end
   6. return \( t \);
7. **End Function**
In Algorithm 1, the following termination condition
\[ f(x + td) - f(x) < -ct\|g\|, \]  
(11)
is called sufficient decrease condition. It is stressed that, for an ADD this algorithm finds a step size \( t > 0 \) after finitely iterations.

3.3 GS method

The main aim of the GS method is to produce a Clarke stationary point through a sequence of \( (\nu_k, G^m_{\epsilon_k}(x_k)) \)-stationary points where \( \{\nu_k\} \) and \( \{\epsilon_k\} \) are decreasing sequences that tend to zero. To provide more details, at each iteration, the set of sampled points \( \{s_{k1}, \ldots, s_{kn}\} \cup \{s_{k0}\} \) is generated uniformly and independently from \( B(x_k, \epsilon_k) \) and then the gradient information of the objective function is computed at these points. Next, \( g_{\epsilon_k}^c \) is obtained by solving the QP \((9)\) and then the optimality condition
\[ \|g_{\epsilon_k}^c\| \leq \nu_k, \]  
(12)
is checked. If condition \((12)\) does not hold, \( x_k \) is not a \( (\nu_k, G^m_{\epsilon_k}(x_k)) \)-stationary point. In this case, the sampling radius and the optimality tolerance remain unchanged, the approximate \( \epsilon \)-steepest descent direction \( d_{\epsilon_k}^c = -\|g_{\epsilon_k}^c\|^{-1}g_{\epsilon_k}^c \) is considered as the search direction, the the step size \( t_k \) is computed by the Algorithm 1 using inputs \( (x_k, g_{\epsilon_k}^c, d_{\epsilon_k}^c, \gamma, c) \) and finally the current point \( x_k \) is updated by \( x_{k+1} := x_k + t_k d_{\epsilon_k}^c \). To ensure the convergence of the GS method, the point \( x_{k+1} \) must be a differentiable point. Consequently, if the objective function is not differentiable at \( x_{k+1} \), a differentiable point, say \( \hat{x} \), having the following properties
\[ f(\hat{x}) - f(x_k) < -ct_k\|g_{\epsilon_k}^c\|, \]  
(13)
\[ \|x_k + t_k d_{\epsilon_k}^c - \hat{x}\| \leq \min\{t_k, \epsilon_k\}, \]  
(14)
is selected as a perturbation of \( x_{k+1} \) and \( x_{k+1} \) is updated by \( \hat{x} \). A procedure to find such a perturbation can be found in [21]. We shall see later the motivation of the inequalities \((13)\) and \((14)\) in Sect. 5. On the other hand, if condition \((12)\) holds, \( x_k \) is a \( (\nu_k, G^m_{\epsilon_k}(x_k)) \)-stationary point. In this case, the sampling radius and the optimality tolerance are decreased by their corresponding reduction factors, the gradient bundle \( G^m_{\epsilon_k}(x_k) \) is resampled and the process is repeated.

The GS method is robust and can be used for solving a broad range of nonsmooth problems, but it still suffers from some limitations. Here, we highlight some of these drawbacks that are addressed in this paper.

1. For the large scale objectives the QP \((9)\) is a time-consuming subproblem that is solved at each iteration. This fact makes the GS method unsuitable for large scale nonsmooth problems [20].
2. When the objective function \( f \) is smooth on \( B(x_k, \epsilon_k) \) (this is the case in early iterations of the method) the information we collect by sampling gradients are close together and there is not any significant difference between \(-\|g_{\epsilon_k}^c\|^{-1}g_{\epsilon_k}^c\) and \(-\|\nabla f(x_k)\|^{-1}\nabla f(x_k)\) as a search direction. In such cases, solving the QP \((9)\) is not reasonable.
To resolve the above limitations, we introduce an alternative direction, namely Ideal direction, that can be computed fast in comparison to solving the QP (9). Moreover, this direction may help us to check the optimality condition (12) without solving the QP (9).

4 Ideal direction

To overcome the aforementioned difficulties, we introduce Ideal directions as an alternative of $d^\varepsilon $. The name “Ideal” originates from the concept of multicriteria optimization where Ideal points are used as reference points in compromise programming [32].

For a differentiable point $x \in \mathbb{R}^n, m \in \mathbb{N}, \varepsilon > 0$ and $i \in \{1, \ldots, n\}$ let

$$G^m_\varepsilon (x) := \text{co} \left\{ \frac{\partial f}{\partial x_i} (s_j) : j = 0, \ldots, m \right\},$$

and define the vector $g^I_\varepsilon = (g^I_\varepsilon, 1, \ldots, g^I_\varepsilon, n) \in \mathbb{R}^n$ by

$$g^I_\varepsilon, i := \arg\min \{ |g| : g \in G^m_\varepsilon (x) \}, \quad i = 1, \ldots, n. \quad (15)$$

In fact, the $i$-th component of $g^I_\varepsilon$ denotes the member of $G^m_\varepsilon (x)$ with minimum distance from zero. It is easy to see that $g^I_\varepsilon, i$ can be expressed by

$$g^I_\varepsilon, i = \frac{\text{sign}(m_i) + \text{sign}(M_i)}{2} \min \{|m_i|, |M_i|\}, \quad i = 1, \ldots, n,$$

where

$$m_i := \min \left\{ \frac{\partial f}{\partial x_i} (s_j) : j = 0, \ldots, m \right\},$$

and

$$M_i := \max \left\{ \frac{\partial f}{\partial x_i} (s_j) : j = 0, \ldots, m \right\}.$$

Therefore, in order to obtain the $i$-th component of $g^I_\varepsilon$, we only need to compute the partial derivative of function $f$ with respect to $x_i$ at sampled points $s_0, \ldots, s_m$ and find the ones with minimum and maximum value. Thus, there is no need to consider any complex problem, like QP (9), in order to obtain $g^I_\varepsilon$ and hence these directions are easy to compute. The preceding discussion is summarized in the following definition.

**Definition 4.1** Let $g^I_\varepsilon$ be as defined in (15). If $g^I_\varepsilon \neq 0$, the direction $d^I_\varepsilon = (d^I_\varepsilon, 1, \ldots, d^I_\varepsilon, n) := -\|g^I_\varepsilon\|^{-1} g^I_\varepsilon$ is called (normalized) Ideal direction of the set $G^m_\varepsilon (x)$.

In what follows we examine the most important properties of Ideal directions.

**Lemma 4.1** Let $x$ be a differentiable point, $m \in \mathbb{N}$ and $\varepsilon > 0$.

1. For the set $G^m_\varepsilon (x)$ we always have

$$\|g^I_\varepsilon\| \leq \|g^I_\varepsilon\|.$$

2. If $0 \in G^m_\varepsilon (x)$, then

$$g^I_\varepsilon = 0.$$
Proof

1. Since \( g^c_i \) is the solution of the QP (9), we have \( g^c_i \in G^m_{\varepsilon,i}(x) \) and hence \( g^c_{i,i} \in G^m_{\varepsilon,i}(x) \) for every \( i = 1, \ldots, n \). Now, since \( g^c_{i,i} \) is the solution of problem (15) and \( g^c_{i,i} \in G^m_{\varepsilon,i}(x) \), we conclude that

\[
|g^c_{i,i}| \leq |g^c_{i,i}|, \quad \text{for all } i = 1, \ldots, n,
\]
yielding \( \|g^c_i\| \leq \|g^c_i\| \).

2. Since \( 0 \in G^m_{\varepsilon,i}(x) \), we conclude that \( g^c_i = 0 \) and the result follows immediately from the first part of this lemma. \( \square \)

In the following lemma, we show that if \( g^c_i \neq 0 \), the Ideal direction \( d^i_c = -\|g^c_i\|^{-1}g^c_i \) is an ADD.

Lemma 4.2 Suppose that \( g^c_i \neq 0 \) for the set \( G^m_{\varepsilon}(x) \) and \( c \in (0, 1) \). Then there exists \( z_0 > 0 \) such that

\[
f(x + td^i_c) - f(x) < -tc\|g^c_i\|, \quad \text{for all } t \in (0, z_0].
\]

Proof Let \( A_i := G^m_{\varepsilon,i}(x) \). By noting that \( \frac{\partial f}{\partial x_i}(x) \in A_i \) for every \( i \in \{1, \ldots, n\} \), we have

\[
\nabla f(x)^t d^i_c = \sum_{i=1}^n \frac{\partial f}{\partial x_i}(x)d^i_{c,i} \leq \sum_{i=1}^n \sup_{g \in A_i} \{g d^i_{c,i}\} = -\|g^c_i\|^{-1} \sum_{i=1}^n \sup_{g \in A_i} \{g g^c_{i,i}\}.
\]

Since \( g^c_{i,i} \) is the solution of the problem (15), a necessary condition for minimum implies that for every \( i = 1, \ldots, n \)

\[
g g^c_{i,i} \geq (g^c_{i,i})^2, \quad \text{for all } g \in A_i,
\]

therefore

\[
\nabla f(x)^t d^i_c \leq -\|g^c_i\|^{-1} \sum_{i=1}^n (g^c_{i,i})^2 = -\|g^c_i\|,
\]

and the result follows from the fact that \( \nabla f(x)^t d^i_c = f'(x, d^i_c) \). \( \square \)

Therefore, when \( g^c_i \neq 0 \) an ADD is in hand without solving any complex sub-problem. We will use this key property to alleviate the first drawback we mentioned above. In the next lemma, let \( e_i \) be a vector with a 1 in the \( i \)-th coordinate and zeros elsewhere.

Lemma 4.3 For a differentiable point \( x \in \mathbb{R}^n \), \( \varepsilon > 0 \) and \( m \in \mathbb{N} \), let \( d^i_c \) be the Ideal direction of the set \( G^m_{\varepsilon}(x) \). Then \( d^i_c \neq 0 \) if and only if, for some \( i \in \{1, \ldots, n\} \) there exists a hyperplane \( H \) with the normal vector \( e_i \) which strongly separates \( G^m_{\varepsilon}(x) \) and origin.

Proof First, assume that for some \( i \in \{1, \ldots, n\} \) there is a hyperplane \( H \) with the normal vector \( e_i \) strongly separating \( G^m_{\varepsilon}(x) \) and origin. This means that \( 0 \notin G^m_{\varepsilon,i}(x) \) and hence

\[
g^c_{i,i} = \argmin \{|g| : g \notin G^m_{\varepsilon,i}(x)\} \neq 0,
\]

which yields \( g^c_i \neq 0 \) and consequently \( d^i_c \neq 0 \). Now, suppose that \( d^i_c \neq 0 \) for the set \( G^m_{\varepsilon}(x) \). Thus, \( g^c_i \neq 0 \) and there exits \( i \in \{1, \ldots, n\} \) such that \( g^c_{i,i} \neq 0 \) which
means that $0 \notin G^m_\varepsilon(x)$. Therefore, by convexity of $G^m_\varepsilon(x)$, we conclude that the set $G^m_\varepsilon(x)$ does not touch the hyperplane

$$H = \{(h_1,\ldots,h_{i-1},g^i_{\varepsilon,i}/2,h_{i+1},\ldots,h_n) : h_j \in \mathbb{R}, \text{ for all } j \neq i\}.$$ 

Clearly, $e_i$ is the normal vector of $H$ and this hyperplane strongly separates $G^m_\varepsilon(x)$ and origin.

The following lemma states that, if function $f$ is smooth at a nonstationary point $x \in \mathbb{R}^n$, there exists a sampling radius $\varepsilon > 0$ such that the Ideal direction of the set $G^m_\varepsilon(x)$ is nonzero.

**Lemma 4.4** Suppose that the function $f : \mathbb{R}^n \to \mathbb{R}$ is smooth at a point $x \in \mathbb{R}^n$ and $\nabla f(x) \neq 0$. Then, there exists a sampling radius $\varepsilon > 0$ such that $d^i_\varepsilon \neq 0$ for the set $G^m_\varepsilon(x)$.

**Proof** Since the map $\nabla f : D \subset \mathbb{R}^n \to \mathbb{R}^n$ is continuous at the point $x \in \mathbb{R}^n$, for every $\delta > 0$ there exists $\varepsilon > 0$ such that

$$\|\nabla f(x) - \nabla f(y)\| < \delta, \quad \text{for all } y \in B(x, \varepsilon).$$

Now since $\nabla f(x) \neq 0$, without loss of generality, there exists $i \in \{1,\ldots,n\}$ such that $\frac{\partial f(x)}{\partial x_i} > 0$. By taking $\delta := \frac{\partial f(x)}{\partial x_i}/2$, there exists $\varepsilon > 0$ such that

$$\left|\frac{\partial f(x)}{\partial x_i} - \frac{\partial f(y)}{\partial y_i}\right| \leq \|\nabla f(x) - \nabla f(y)\| < \frac{1}{2} \frac{\partial f(x)}{\partial x_i}, \quad \text{for all } y \in B(x, \varepsilon). \quad (16)$$

This means that $\frac{\partial f(y)}{\partial y_i} > 0$ for every $y \in B(x, \varepsilon)$ and consequently $0 \notin G^m_\varepsilon(x)$ that gives $g^i_{\varepsilon,i} \neq 0$. Therefore $d^i_\varepsilon = -\|g^i_\varepsilon\|^{-1}g^i_\varepsilon \neq 0$ for the set $G^m_\varepsilon(x)$. □

In the light of preceding lemma, one might think that, whenever we obtain a zero ideal direction, then we can reduce the sampling radius to obtain a nonzero one. However, reducing sampling radius may lead to losing information of the nearby nonsmooth curve and hence the resulting nonzero ideal direction would not be an effective search direction. In such cases, the approximate $\varepsilon$-steepest descent direction obtained by QP (9) may provide a more suitable search direction.

By the proof of the Lemma 4.4 one can see that, when we are far away from a nonsmooth curve, in the sense that $B(x, \varepsilon) \subset D$, since the information we collect by sampling gradients are very close together, we expect the inequalities (16) hold (without reducing the sampling radius) and for the set $G^m_\varepsilon(x)$ we obtain a nonzero Ideal direction. This important observation resolves the second drawback stated above.

It is noted that, the ideal direction is simply obtained without solving any time-consuming subproblem. In fact, computing gradient at sampled points is equivalent to having an ideal direction and whenever $g^i_\varepsilon \neq 0$ it decreases the value of function along $d^i_\varepsilon$ with a step size $t$ satisfying the Armijo condition. In the case that $g^i_\varepsilon = 0$ for the set $G^m_\varepsilon(x)$, according to the second part of Lemma 4.1, we consider $x$ as a candidate to be a Clarke $\varepsilon$-stationary point and this possibility is checked by solving the QP (9).
5 A Gradient Sampling method using Ideal direction

In this section, based on Ideal directions, we modify the GS algorithm presented in [21]. Next, following the work of Kiwiel in [21], we analyze the global convergence of the proposed method.

To modify the GS method, our method consider the Ideal direction as the first choice for the search direction. As mentioned, whenever \( g_I^k \neq 0 \) it presents an ADD without solving any complex subproblem. Thus, it is reasonable to compute the Ideal direction before solving the QP (9). Furthermore, during each iteration, the optimality condition (12) must be checked. By noting that \( \| g_I^k \| \leq \| g_s^k \| \), if for a point \( x_k \in \mathbb{R}^n \) we have \( \| g_I^k \| > \nu_k \), we conclude that \( \| g_s^k \| > \nu \) and hence \( x_k \) is not a \((\nu_k, G_m^k(x_k))\)-stationary point. In the case that \( \| g_I^k \| \leq \nu_k \), we need to check whether \( x_k \) is \((\nu_k, G_m^k(x_k))\)-stationary point or not and this is done by solving the QP (9). Therefore, in some cases we can check the optimality condition (12) without computing \( g_s^k \). In this way, through Ideal directions, we can reduce the number of quadratic subproblems significantly.

Based on the preceding discussion, a Gradient Sampling algorithm based on Ideal directions (GSI) is presented in Algorithm 2.

5.1 Convergence analysis

To study the convergence of the GSI algorithm, at first we recall three notations from [21, 19].

For \( \varepsilon > 0 \) and \( \bar{x} \in \mathbb{R}^n \), let \( \rho_\varepsilon(\bar{x}) \) be the distance between \( G_\varepsilon(\bar{x}) \) and origin. In mathematical terms

\[ \rho_\varepsilon(\bar{x}) := \text{dist}(0, G_\varepsilon(\bar{x})), \]

and for \( x \in \mathbb{R}^n \) and \( m \in \mathbb{N} \), let

\[ D_\varepsilon^m(x) := \prod_{1}^{m} (B(x, \varepsilon) \cap D) \subset \prod_{1}^{m} \mathbb{R}^n. \]

Moreover, for \( x, \bar{x} \in \mathbb{R}^n, m \in \mathbb{N} \) and \( \varepsilon, \delta > 0 \) define

\[ V_\varepsilon^m(x, \bar{x}, \delta) := \{ (y_1, \ldots, y_m) \in D_\varepsilon^m(x) : \text{dist}(0, \text{co}\{\nabla f(y_k)\}_{k=1}^{m}) \leq \rho_\varepsilon(\bar{x}) + \delta \}. \]

Next, we recall the next three lemmas from [21]. Before it, suppose that the objective function in problem (1) fulfills Assumption 5.1 throughout this section [20].

**Assumption 5.1** The function \( f : \mathbb{R}^n \to \mathbb{R} \) is locally Lipschitz and continuously differentiable on an open set \( D \) with full measure in \( \mathbb{R}^n \).

In the next three lemmas, the first one provides a variational inequality for some elements of \( G_\varepsilon(\bar{x}) \) and the second one examines the local behavior of the set \( V_\varepsilon^m(\bar{x}, \cdot, \delta) \) in the vicinity of \( \bar{x} \). Finally, Lemma 5.3 derives a lower bound for the step size \( t_k \) when the sampled points \((s_{k1}, \ldots, s_{km})\) lie in a special open subset of \( V_\varepsilon^m(\bar{x}, x_k, \delta) \).
Algorithm 2: Gradient Sampling method using Ideal direction (GSI)

inputs: $x_0 \in \mathbb{R}^n$ as initial guess, initial sampling radius $\varepsilon_0 > 0$, initial stationarity tolerance $\nu_0 > 0$, sample size $m \in \mathbb{N}$, parameters $\nu_{\text{opt}}, \varepsilon_{\text{opt}} \geq 0$ as tolerances in stopping condition, parameters $\theta, \mu \in (0, 1)$ as reduction factors for optimality tolerance and sampling radius, and the backtracking Armijo line search parameters $\gamma, c \in (0, 1)$.

output: An approximation of a Clarke stationary point of $f$.

required: $f$ is differentiable at the starting point $x_0 \in \mathbb{R}^n$.

1. $k := 0$;
2. while $\nu_k \geq \nu_{\text{opt}}$ or $\varepsilon_k \geq \varepsilon_{\text{opt}}$ do
3.   Sample $u_1, \ldots, u_m$ independently and uniformly from $B_1$ and set $s_{kj} := x_k + \varepsilon_k u_j$ for $j = 1, \ldots, m$;
4.   if $s_{kj}$ is not a differentiable point for some $j \in \{1, \ldots, m\}$ then
5.     Stop;
6.   end
7.   Compute $g^*_k$ for the set $G^m_{\varepsilon_k}(x_k)$ and set $g_k := g^*_k$;
8.   if $\|g_k\| \leq \nu_k$ then
9.     Solve QP (9) to compute $g^*_k$ and set $g_k := g^*_k$;
10. end
11. if $\|g_k\| \leq \nu_k$ then
12.   $\nu_{k+1} := \theta\nu_k$, $\varepsilon_{k+1} := \mu\varepsilon_k$;
13.   $x_{k+1} := x_k$;
14.   $k := k + 1$;
15.   Continue;
16. end
17. $d_k := -\|g_k\|^{-1}g_k$;
18. $t_k := \text{BLS}(x_k, g_k, d_k, \gamma, c)$;
19. $x_{k+1} := x_k + t_k d_k$;
20. if $f$ is not differentiable at $x_{k+1}$ then
21.   Find a point $\bar{x}$ at which $f$ is differentiable and satisfies $f(\bar{x}) - f(x_k) < -c\|g_k\|$;
22.   $\|x_k + t_k d_k - \bar{x}\| \leq \min\{t_k, \varepsilon_k\}$;
23.   Set $x_{k+1} := \bar{x}$;
24. end
25. $\nu_{k+1} := \nu_k$, $\varepsilon_{k+1} := \varepsilon_k$;
26. $k := k + 1$;
27. end
28. Introduce $x_k$ as an approximation of a Clarke stationary point.

Lemma 5.1 ([21]) For a point $\mathbf{x} \in \mathbb{R}^n$, suppose that $\mathbf{0} \notin G_c(\mathbf{x})$ and $c \in (0, 1)$, then there is $\delta > 0$ such that for every $\mathbf{u} \in G_c(\mathbf{x})$ satisfying $\|\mathbf{u}\| \leq \rho_c(\mathbf{x}) + \delta$, we have $\langle \mathbf{u}, \mathbf{v} \rangle > c\|\mathbf{u}\|^2$, for all $\mathbf{v} \in G_c(\mathbf{x})$.

Lemma 5.2 ([21]) For any $\delta > 0$, there exist $\tau > 0, \overline{m} \geq n + 1$ and a nonempty open set $\mathcal{V}$, such that $\mathcal{V} \subset V^\overline{m}_c(\mathbf{x}, \mathbf{x}, \delta)$, for all $\mathbf{x} \in B(\mathbf{x}, \tau)$.

Lemma 5.3 ([21]) Assume that $\mathbf{0} \notin G_c(\mathbf{x})$, choose $\delta > 0$ as in Lemma 5.1 and $\tau, \overline{m}, \mathcal{V}$ as in Lemma 5.2. Suppose that at iteration $k$ of Algorithm 2, $(s_k, \ldots, s_{\overline{m}m}) \in \mathcal{V}$ and the backtracking Armijo line search in Line 18 is reached with $x_k \in B(\mathbf{x}, \min\{\tau, \varepsilon_k/3\})$ and $d_k = d_{\tau k}$. Then $t_k \geq \min\{1, 7\varepsilon_k/3\}$. 

In order to extend Lemma 5.3 to Ideal directions, at first we need to introduce the following notations. For $\mathbf{x} \in \mathbb{R}^n$ and $i \in \{1, \ldots, n\}$, let

$$\rho_{\epsilon,i}(\mathbf{x}) := \text{dist}(0, G_{\epsilon,i}(\mathbf{x})), $$

where

$$G_{\epsilon,i}(\mathbf{x}) := c_1 \text{co} \left\{ \frac{\partial f}{\partial x_i}(B(\mathbf{x}, \epsilon) \cap D) \right\}. $$

Furthermore, for $x, \mathbf{x} \in \mathbb{R}^n, m \in \mathbb{N}$ and $\epsilon, \delta, \omega > 0$ we define

$$W^m_{\epsilon}(\mathbf{x}, x, \delta, \omega) := \left\{ \begin{array}{l} \text{dist}(0, \text{co} \left\{ \frac{\partial f}{\partial x_i}(y_k) \right\}_{k=1}^m) \leq \rho_{\epsilon,i}(\mathbf{x}) + \delta, i \in A(\mathbf{x}) \text{ and} \\
(y_1, \ldots, y_m) \in D^m_{\epsilon}(\mathbf{x}) : \text{dist}(0, \text{co} \left\{ \frac{\partial f}{\partial x_i}(y_k) \right\}_{k=1}^m) = 0, i \in A_{\text{int}}(\mathbf{x}) \text{ and} \\
\text{dist}(0, \text{co} \left\{ \frac{\partial f}{\partial x_i}(y_k) \right\}_{k=1}^m) \leq \omega, i \in A_{\text{bd}}(\mathbf{x}) \end{array} \right\}, $$

in which

$$A(\mathbf{x}) := \{ i \in \{1, \ldots, n\} : 0 \notin G_{\epsilon,i}(\mathbf{x}) \}, $$

$$A_{\text{int}}(\mathbf{x}) := \{ i \in \{1, \ldots, n\} : 0 \in \text{int} G_{\epsilon,i}(\mathbf{x}) \}, $$

$$A_{\text{bd}}(\mathbf{x}) := \{ i \in \{1, \ldots, n\} : 0 \in \text{bd} G_{\epsilon,i}(\mathbf{x}) \}. $$

Here $\text{bd}$ and $\text{int}$ denote the boundary and interior of a set respectively.

To study the global convergence behavior of our method, we have to understand the local behavior of the set $W^m_{\epsilon}(\mathbf{x}, \cdot, \delta, \omega)$ in the vicinity of $\mathbf{x}$. Then, we derive a lower bound for the step size $t_k$ when the search direction is defined by an Ideal direction and the sampled points $(s_{k1}, \ldots, s_{km})$ are located in a special open subset of $W^m_{\epsilon}(\mathbf{x}, x, \delta, \omega)$. To this end, we start with the following lemma which is an immediate consequence of Lemma 5.1.

**Lemma 5.4** For a point $\mathbf{x} \in \mathbb{R}^n, c \in (0, 1)$ and any $i \in A(\mathbf{x})$, there is $\delta_i > 0$ such that for every $u_i \in G_{\epsilon,i}(\mathbf{x})$ satisfying $|u_i| \leq \rho_{\epsilon,i}(\mathbf{x}) + \delta_i$, we have

$$u_i v_i > cu_i^2, \quad \text{for all } v_i \in G_{\epsilon,i}(\mathbf{x}). $$

**Proof** This follows immediately from Lemma 5.1. $\square$

The following lemma states that for the points sufficiently close to $\mathbf{x}$, the set valued map $W^m_{\epsilon}(\cdot, \cdot, \delta, \omega)$ contains a nonempty open set.

**Lemma 5.5** For any $\epsilon, \delta, \omega > 0$, there exist $\tau > 0, \mathbf{m} \in \mathbb{N}$ and a nonempty open set $\mathbf{W}$, such that

$$\mathbf{W} \subset W^m_{\epsilon}(\mathbf{x}, \mathbf{x}, \delta, \omega), \quad \text{for all } \mathbf{x} \in B(\mathbf{x}, \tau). $$

**Proof** First, assume that $i \in A(\mathbf{x})$. Since $\delta > 0$, there exists $u_i \in \text{co} \left\{ \frac{\partial f}{\partial x_i}(B(\mathbf{x}, \epsilon) \cap D) \right\}$ such that

$$|u_i| \leq \rho_{\epsilon,i}(\mathbf{x}) + \delta, \quad \text{for all } i \in A(\mathbf{x}), $$

where

$$G_{\epsilon,i}(\mathbf{x}) := c_1 \text{co} \left\{ \frac{\partial f}{\partial x_i}(B(\mathbf{x}, \epsilon) \cap D) \right\}. $$
then, by Carathéodory’s theorem, there exist $s_1^i, s_2^i \in B(\bar{x}, \varepsilon) \cap D$ and nonnegative scalars $\bar{\lambda}_1^i, \bar{\lambda}_2^i$ such that $\bar{\lambda}_1^i + \bar{\lambda}_2^i = 1$ and
\[
    u_i = \bar{\lambda}_1^i \frac{\partial f}{\partial x_i}(s_1^i) + \bar{\lambda}_2^i \frac{\partial f}{\partial x_i}(s_2^i).
\]
Now according to Assumption 5.1, for each $i \in \{1, \ldots, n\}$ the map $\frac{\partial f}{\partial x_i}()$ is continuous in $D$. Thus, for every $i \in \mathcal{A}(\bar{x})$ there is $\varepsilon_1 \in (0, \varepsilon)$ such that for all $y_1^i \in B(s_1^i, \varepsilon_1) \cap D$ and $y_2^i \in B(s_2^i, \varepsilon_1) \cap D$, we have
\[
    \left| \lambda_1^i \frac{\partial f}{\partial x_i}(y_1^i) + \lambda_2^i \frac{\partial f}{\partial x_i}(y_2^i) \right| \leq \rho_{\varepsilon_1,i}(\bar{x}) + \delta, \quad \text{for all } i \in \mathcal{A}(\bar{x}). \quad (17)
\]
Second, suppose that $i \in \mathcal{A}_{\text{int}}'(\bar{x})$. Then we have $0 \in \text{co} \left\{ \frac{\partial f}{\partial x_i}(B(\bar{x}, \varepsilon) \cap D) \right\}$ which implies the existence of $0 < u_1^+ \in \text{co} \left\{ \frac{\partial f}{\partial x_i}(B(\bar{x}, \varepsilon) \cap D) \right\}$ and $0 > u_1^- \in \text{co} \left\{ \frac{\partial f}{\partial x_i}(B(\bar{x}, \varepsilon) \cap D) \right\}$.

Using Carathéodory’s theorem, for each $i \in \mathcal{A}_{\text{int}}'(\bar{x})$, there exist $s_1^i, s_2^i, s_3^i, s_4^i \in B(\bar{x}, \varepsilon) \cap D$ and nonnegative scalars $\bar{\lambda}_1^i, \bar{\lambda}_2^i, \bar{\lambda}_3^i, \bar{\lambda}_4^i$ such that
\[
    0 < u_1^+ = \bar{\lambda}_1^i \frac{\partial f}{\partial x_i}(s_1^i) + \bar{\lambda}_2^i \frac{\partial f}{\partial x_i}(s_2^i), \quad \bar{\lambda}_1^i + \bar{\lambda}_2^i = 1,
\]
and
\[
    0 > u_1^- = \bar{\lambda}_3^i \frac{\partial f}{\partial x_i}(s_3^i) + \bar{\lambda}_4^i \frac{\partial f}{\partial x_i}(s_4^i), \quad \bar{\lambda}_3^i + \bar{\lambda}_4^i = 1.
\]
Now since the map $\frac{\partial f}{\partial x_i}()$ is continuous in $D$, there is $\varepsilon_2 \in (0, \varepsilon)$ such that for every $i \in \mathcal{A}_{\text{int}}'(\bar{x})$ one can write
\[
    0 < \lambda_1^i \frac{\partial f}{\partial x_i}(y_1^i) + \lambda_2^i \frac{\partial f}{\partial x_i}(y_2^i), \quad \text{for all } y_1^i \in B(s_1^i, \varepsilon_2), y_2^i \in B(s_2^i, \varepsilon_2), \quad (18)
\]
and
\[
    0 > \lambda_3^i \frac{\partial f}{\partial x_i}(y_3^i) + \lambda_4^i \frac{\partial f}{\partial x_i}(y_4^i), \quad \text{for all } y_3^i \in B(s_3^i, \varepsilon_2), y_4^i \in B(s_4^i, \varepsilon_2). \quad (19)
\]
Finally, let $i \in \mathcal{A}_{\text{cad}}'(\bar{x})$. Since $\omega > 0$, there is $u_i \in \text{co} \left\{ \frac{\partial f}{\partial x_i}(B(\bar{x}, \varepsilon) \cap D) \right\}$ such that
\[
    |u_i| \leq \rho_{\varepsilon_1,i}(\bar{x}) + \omega = \omega, \quad \text{for all } i \in \mathcal{A}_{\text{cad}}'(\bar{x}).
\]
By Carathéodory’s theorem, for each $i \in \mathcal{A}_{\text{cad}}'(\bar{x})$, there exist $s_1^i, s_2^i \in B(\bar{x}, \varepsilon) \cap D$ and nonnegative scalars $\bar{\lambda}_1^i, \bar{\lambda}_2^i$ such that $\bar{\lambda}_1^i + \bar{\lambda}_2^i = 1$ and
\[
    u_i = \bar{\lambda}_1^i \frac{\partial f}{\partial x_i}(s_1^i) + \bar{\lambda}_2^i \frac{\partial f}{\partial x_i}(s_2^i).
\]
Using continuity of the map $\frac{\partial f}{\partial x_i}()$, there is $\varepsilon_3 \in (0, \varepsilon)$ such that for all $y_1^i \in B(s_1^i, \varepsilon_3)$ and $y_2^i \in B(s_2^i, \varepsilon_3)$ we have
\[
    \left| \lambda_1^i \frac{\partial f}{\partial x_i}(y_1^i) + \lambda_2^i \frac{\partial f}{\partial x_i}(y_2^i) \right| \leq \omega. \quad (20)
\]
Now there exists $0 < \bar{\varepsilon} \leq \min\{\bar{\varepsilon}_1, \bar{\varepsilon}_2, \bar{\varepsilon}_3\}$ such that $\overline{W} := \bigcap_{i,j} \text{int} B(s_i^j, \bar{\varepsilon})$ lies in $D_{\bar{x}}(\bar{x})$ in which

$$2n \leq \overline{m} := 2(|A(\bar{x})| + |A_{bd}(\bar{x})|) + 4|A_{\text{int}}'(\bar{x})| \leq 4n.$$ 

Furthermore, in view of (17), (18), (19) and (20) and the fact that $B(\bar{x}, \varepsilon - \tau) \subset B(x, \varepsilon)$ for all $x \in B(\bar{x}, \tau)$ with $\tau := \varepsilon$, we can conclude that $\overline{W} \subset W_\varepsilon^m(\bar{x}, x, \delta, \omega)$, for all $x \in B(\bar{x}, \tau)$.

The following corollary is an immediate consequence of the preceding lemma.

**Corollary 5.1** Let $\varepsilon, \delta, \omega > 0$, $\bar{x}, x \in \mathbb{R}^n$ and $\overline{W} \subset W_\varepsilon^m(\bar{x}, x, \delta, \omega)$ be as in Lemma 5.5. For the set $G_\varepsilon^m(x)$ suppose that $(s_1, \ldots, s_m) \in \overline{W}$ and $g_i^\varepsilon = (g_{i,1}^\varepsilon, \ldots, g_{i,n}^\varepsilon)$ is defined as (15). Then

$$|g_i^\varepsilon| \leq \rho_{\varepsilon,i}(\bar{x}) + \delta, \quad \text{for all } i \in A(\bar{x}),$$

$$g_i^\varepsilon = 0, \quad \text{for all } i \in A'_{\text{int}}(\bar{x}),$$

$$|g_i^\varepsilon| \leq \omega, \quad \text{for all } i \in A_{bd}(\bar{x}).$$

The next lemma indicates that, if the set of sampled points lies in the open set $\overline{W}$ and the search direction is defined by an Ideal direction, a lower bound for the step size $t$ can be derived.

**Lemma 5.6** Suppose that $A(\bar{x}) \neq \emptyset$. For each $i \in A(\bar{x})$ choose $\delta_i$ as in Lemma 5.4 and $\overline{m}, \overline{W}$ as in Lemma 5.5. Assume that at iteration $k$ of Algorithm 2, $(s_{k1}, \ldots, s_{km}) \in \overline{W}$ and the backtracking Armijo line search in Line 18 is reached with $x_k \in B(\bar{x}, \min\{\tau, \varepsilon_k/3\})$ and $d_k = d_k^\varepsilon$. Then $t_k \geq \min\{1, \gamma \varepsilon_k/3\}$.

**Proof.** For the sake of simplicity in notations, let $m := \overline{m}$ and $\varepsilon := \varepsilon_k$. First, we show that for the sampled points $\{s_{k1}, \ldots, s_{km}\} \cup \{s_{k0}\}$, we have $g_{i}^{\varepsilon} \in G_{\varepsilon,i}(\bar{x})$ for all $i \in \{1, \ldots, n\}$. Let $G_{\varepsilon,i} := \text{co}\langle \partial f_i, \bar{x} \rangle_{i=1}^m$. Since $(s_{k1}, \ldots, s_{km}) \in \overline{W} \subset D_{\bar{x}}(\bar{x})$ with $\varepsilon = \tau$, we have $G_{\varepsilon,i} \subset G_{\varepsilon,i}(\bar{x})$. Moreover, $x_k \in B(\bar{x}, \varepsilon/3)$ is a differentiable point and hence $\frac{df}{dx_i}(x_k) = \frac{df}{dx_i}(s_{k0}) \in G_{\varepsilon,i}(\bar{x})$. Thus, $g_{i}^{\varepsilon} \in G_{\varepsilon,i}(\bar{x})$ for all $i \in \{1, \ldots, n\}$. Now, set $\delta := \min\{\delta_i : i \in A(\bar{x})\}$, then the fact that $(s_{k1}, \ldots, s_{km}) \in \overline{W} \subset W_\varepsilon^m(\bar{x}, \bar{x}, \delta, \omega)$ along with Corollary 5.1 imply that

$$|g_i^\varepsilon| < \rho_{\varepsilon,i}(\bar{x}) + \delta \quad \text{for all } i \in A(\bar{x}).$$

Now by Lemma 5.4, for each $v_i \in G_{\varepsilon,i}(\bar{x})$ and $c \in (0, 1)$ one can write

$$g_i^\varepsilon v_i > c (g_i^\varepsilon)^2 \quad \text{for all } i \in A(\bar{x}).$$

(21)

Furthermore, Corollary 5.1 yields that

$$g_i^\varepsilon = 0, \quad \text{for all } i \in A'_{\text{int}}(\bar{x}),$$

(22)

and for arbitrary $\omega > 0$, we have

$$|g_i^\varepsilon| \leq \omega, \quad \text{for all } i \in A_{bd}(\bar{x}).$$

(23)
Using (21), (22), (23) and choosing \( \omega > 0 \) sufficiently small in Lemma 5.5, for all \( v \in G_\varepsilon(x) \) we have

\[
\langle g^i_k, v \rangle = \sum_{i \in A(k)} g^1_{i, i} v_i + \sum_{i \in A_0^i(k)} g^1_{i, i} v_i + \sum_{i \in d_{\varepsilon}^i(k)} g^1_{i, i} v_i \\
= \sum_{i \in A(k)} g^1_{i, i} v_i + \sum_{i \in A_0^i(k)} g^1_{i, i} v_i > c \sum_{i \in A(k)} (g^1_{i, i})^2 \\
= c\|g^i_k\|^2.
\]

We stress that, the last inequality follows from (21) and the fact that \( \omega > 0 \) is sufficiently small. So far, we have proved that

\[
\langle g^i_k, v \rangle > c\|g^i_k\|^2, \quad \text{for all } v \in G_\varepsilon(x).
\]  \hspace{1cm} (24)

By contradiction, suppose that \( t_k < \min\{1, \gamma\varepsilon/3\} \). In view of Algorithm 1, we conclude

\[-c \gamma^{-1} t_k \|g^i_k\| \leq f(x_k + \gamma^{-1} t_k d^i_k) - f(x_k),\]

and using Lebourg’s mean value theorem, there exist \( x_k \in [x_k + \gamma^{-1} t_k d^i_k, x_k] \) and \( \xi_k \in \partial f(x_k) \) such that

\[f(x_k + \gamma^{-1} t_k d^i_k) - f(x_k) = \gamma^{-1} t_k \langle \xi_k, d^i_k \rangle.\]

Therefore, using \( d^i_k = -\|g^i_k\|^{-1} g^i_k \), we obtain \( \langle \xi_k, g^i_k \rangle \leq c\|g^i_k\|^2 \) and (24) implies that \( \xi_k \notin G_\varepsilon(x) \). On the other hand, \( \gamma^{-1} t_k \|d^i_k\| < \varepsilon/3 \) and \( \|x_k - x\| \leq \varepsilon/3 \) yield \( x_k \in B(x, 2\varepsilon/3) \) and hence \( \xi_k \notin G_\varepsilon(x) \), which is a contradiction. \( \square \)

In the next lemma, we recall a highly useful inequality from [21].

**Lemma 5.7 ([21])** Algorithm 2 ensures that

\[f(x_{k+1}) \leq f(x_k) - \frac{1}{2} \|x_{k+1} - x_k\| \|g_k\|, \quad \text{for all } k, \]  \hspace{1cm} (25)

in which \( c \) is the Armijo parameter.

Now, we are in a position to analyze the global convergence behavior of the GSI algorithm. Our analysis is similar to the work of Kiwiel in [21]. However, there are some differences due to using Ideal directions. So, for the sake of completeness, we present a comprehensive proof for the following theorem.

**Theorem 5.1** Let \( \{x_k\} \) be a sequence generated by Algorithm 2 with \( \varepsilon_0, \nu_0 > 0, \nu_{opt} = \varepsilon_{opt} = 0 \) and \( \mu, \theta < 1 \). Then, with probability 1, the algorithm does not terminate in Line 5 and either \( f(x_k) \to -\infty \) or \( \nu_k \downarrow 0, \varepsilon_k \downarrow 0 \) and every cluster point of the sequence \( \{x_k\} \) is a Clarke stationary point.

**Proof** From measure theory, we know that the termination in Line 5 has zero probability. Also, in the case \( f(x_k) \to -\infty \), we have nothing to prove. So, we consider the following case

\[\inf_{k \to \infty} f(x_k) > \infty.\]

Then, sufficient decrease condition (11) yields

\[\sum_{k=0}^{\infty} t_k \|g_k\| < \infty\]  \hspace{1cm} (26)
and from (25), we have

\[
\sum_{k=0}^{\infty} \|x_{k+1} - x_k\| g_k < \infty.
\]

(27)

Now, we prove that, with probability 1, the sequences \(\{r_k\}\) and \(\{\epsilon_k\}\) tend to zero. For this purpose, assume that there exist \(k_1 \in \mathbb{N}\), \(\bar{\epsilon} > 0\) and \(\bar{\nu} > 0\) such that \(\epsilon_k = \bar{\epsilon}\) and \(r_k = \bar{\nu}\) for all \(k \geq k_1\). Therefore, we conclude that \(\|g_k\| > \bar{\nu}\) in (26) and (27) which implies \(r_k \rightarrow 0\) and \(\|x_{k+1} - x_k\| \rightarrow 0\) as \(k \rightarrow \infty\). Thus, we may assume that there is \(\bar{x} \in \mathbb{R}^n\) such that \(x_k \rightarrow \bar{x}\). Suppose in the GSI algorithm

\[
m \geq \overline{m} = 2(|A(\bar{x})| + |A_{sd}(\bar{x})|) + 4|A_{int}(\bar{x})| \geq n + 1.
\]

For the sake of simplicity in notations, let \(\epsilon := \bar{\epsilon}\). We need to consider four cases.

**Case 1.** Let \(0 \notin G_\epsilon(\bar{x})\) and \(K\) be an infinite subset of \(\mathbb{N}\) such that \(d_k = -\|g_{e_k}\|^{-1} g_{e_k}\) for all \(k \geq k_1\) and \(k \in K\). Then, for \(\tau, \delta, \nu\) chosen as in Lemma 5.3, since \(t_k \rightarrow 0\) and \(x_k \rightarrow \bar{x}\), we can choose \(k_2 \geq k_1\) such that \(x_k \in B(\bar{x}, \min\{\tau, \epsilon/3\})\) and \(t_k < \min\{1, \gamma \epsilon/3\}\) for all \(k \geq k_2\) which means that \((s_{k1}, \ldots, s_{km}) \notin \overline{\nu}\) for all \(k \geq k_2\) and \(k \in K\). Since \((s_{k1}, \ldots, s_{km})\) is sampled uniformly and independently from \(D_{00}(x_k)\) containing the nonempty open set \(\overline{\nu}\), this event has probability zero.

**Case 2.** Let \(0 \notin G_{\epsilon, i}(\bar{x})\) for some \(i \in \{1, \ldots, n\}\) and \(K\) be an infinite subset of \(\mathbb{N}\) such that \(d_k = -\|g_{e_k}\|^{-1} g_{e_k}\) for all \(k \geq k_1\) and \(k \in K\). Then, for \(\tau, \delta, \omega\) and \(\overline{\nu}\) chosen as in Lemma 5.6, one can choose \(k_2 \geq k_1\) such that \(x_k \in B(\bar{x}, \min\{\tau, \epsilon/3\})\) and \(t_k < \min\{1, \gamma \epsilon/3\}\) for all \(k \geq k_2\) which implies that \((s_{k1}, \ldots, s_{km}) \notin \overline{\nu}\) for all \(k \geq k_2\) and \(k \in K\). This event occurs with probability zero as well.

**Case 3.** Let \(0 \notin G_\epsilon(\bar{x})\) and \(K\) be an infinite subset of \(\mathbb{N}\) such that \(d_k = -\|g_{e_k}\|^{-1} g_{e_k}\) for all \(k \geq k_1\) and \(k \in K\). Clearly

\[
\bar{\nu} \leq \|g_{e_k}\| \leq \text{dist}\left(0, \text{co}(\nabla f(s_{k1}), \ldots, \nabla f(s_{km}))\right).
\]

Then, for \(\delta := \frac{\bar{\nu}}{2}\) and \(\overline{\nu}\) as in Lemma 5.2, one can choose \(k_2 \geq k_1\) such that \(x_k \in B(\bar{x}, \tau)\) for all \(k \geq k_2\) and \(k \in K\) and since \(\rho_\epsilon(\bar{x}) = 0\), we can write

\[
\text{dist}\left(0, \text{co}(\nabla f(s_{k1}), \ldots, \nabla f(s_{km}))\right) > \rho_\epsilon(\bar{x}) + \frac{\bar{\nu}}{2} = \frac{\bar{\nu}}{2},
\]

which means that \((s_{k1}, \ldots, s_{km}) \notin \overline{\nu}\) for all \(k \geq k_2\) and \(k \in K\). This event takes place with probability zero.

**Case 4.** Let \(0 \notin G_{\epsilon, i}(\bar{x})\) for every \(i \in \{1, \ldots, n\}\) and \(K\) be an infinite subset of \(\mathbb{N}\) such that \(d_k = -\|g_{e_k}\|^{-1} g_{e_k}\) for all \(k \geq k_1\) and \(k \in K\). If \(A_{sd}(\bar{x}) = 0\), then for \(\tau, \overline{\nu}\) as in Lemma 5.5, one can choose \(k_2 \geq k_1\) such that \(x_k \in B(\bar{x}, \tau)\) for all \(k \geq k_2\) and the fact that

\[
0 < \bar{\nu} \leq \|g_{e_k}\|, \quad \text{for all } k \geq k_2 \text{ and } k \in K,
\]

gives \((s_{k1}, \ldots, s_{km}) \notin \overline{\nu}\) for all \(k \geq k_2\) and \(k \in K\). Clearly, this event occurs with probability zero. Now, suppose that \(A_{sd}(\bar{x}) \neq 0\). Then, for \(\omega := |A_{sd}(\bar{x})|^{-1/2} \bar{\nu}/2\) and \(\tau, \overline{\nu}\) as in Lemma 5.5, one can choose \(k_2 \geq k_1\) such that \(x_k \in B(\bar{x}, \tau)\) for all \(k \geq k_2\) and the fact that

\[
0 < \omega < \bar{\nu} \leq \|g_{e_k}\|, \quad \text{for all } k \geq k_2 \text{ and } k \in K,
\]
yields \((s_{k1}, \ldots, s_{kn}) \notin \mathbb{R}^w\) for all \(k \geq k_2\) and \(k \in K\). This event has also probability zero.

Now, we are in a position to consider the case that \(\nu_k \downarrow 0, \varepsilon_k \downarrow 0\) and \(x\) is a cluster point of \(\{x_k\}\). Since \(\nu_k \downarrow 0\) we conclude that 0 is a cluster point of the sequence \(\{\|g_{s_k}^*\|\}\). Therefore, in the case \(x_k \to x\) we have

\[
\liminf_{k \to \infty} \max \\{\|x_k - x\|, \|g_{s_k}^*\|, \varepsilon_k\} = 0.
\]

Note that, \(g_{s_k}^* \in \partial_{x_k} f(x_k)\) and hence outer semicontinuity of the map \(\partial f(\cdot)\) yields \(0 \in \partial f(\bar{x})\). On the other hand, if \(x_k \to \bar{x}\) we need to prove that

\[
\liminf_{k \to \infty} \max \\{\|x_k - \bar{x}\|, \|g_{s_k}^*\|\} = 0. \tag{28}
\]

By contradiction, suppose that there exist \(\bar{\nu} > 0, k_1 \in \mathbb{N}\) and an infinite set \(K := \{k : k \geq k_1, \|x_k - \bar{x}\| \leq \bar{\nu}, \|g_{s_k}^*\| > \bar{\nu}\}\). Then, by (27) we have

\[
\sum_{k \in K} \|x_{k+1} - x_k\| < \infty. \tag{29}
\]

By noting that \(x_k \to \bar{x}\), there is \(\varepsilon > 0\) such that for all \(k \in K\) with \(\|x_k - \bar{x}\| \leq \bar{\nu}/2\) there exists \(k_2 > k\) satisfying \(\|x_{k_2} - x_k\| > \varepsilon\) and \(\|x_i - \bar{x}\| \leq \bar{\nu}\) for each \(k \leq i \leq k_2\). Therefore, using the triangle inequality, we have \(\varepsilon < \|x_{k_2} - x_k\| \leq \sum_{i=k}^{k_2-1} \|x_{i+1} - x_i\|\). However, for \(k \in K\) sufficiently large, (29) yields that the right-hand side of this inequality is less than \(\varepsilon\), which is a contradiction. Therefore (28) holds and consequently \(0 \in \partial f(\bar{x})\).

5.2 More results

In this subsection, we modify the Algorithm 2, such that under some moderate conditions on the objective function \(f\), one can obtain an upper bound for the number of serious iterations required for obtaining a \((\nu, G_{s_{k_2}}(x_k))\)-stationary point. Next, the convergence of the proposed method is studied. Indeed, instead of using the backtracking Armijo line search presented in Algorithm 1, following [21], we will apply the following limited backtracking Armijo line search.

**Algorithm 3:** Limited Backtracking Armijo Line Search (LBALS) [21]

**inputs:** \(x \in \mathbb{R}^n\), the directions \(d, g \in \mathbb{R}^n\), \(\varepsilon > 0\), parameters \(\gamma, c \in (0, 1)\).

**output:** \(d = -\|g\|^{-1} g\) is an ADD for \(f\) at \(x\).

1. **Function LBALS**: \(x, g, d, \varepsilon, \gamma, c\):
   2. \(t := 1;\)
   3. while \(t > \min\{1, \gamma \varepsilon/3\}\) do
     4. if \(f(x + td) - f(x) < -ct\|g\|\) then
       5. return \(t;\)
     6. end
   7. \(t := \gamma t;\)
   8. end
   9. return \(t := 0;\)
10. End Function
By this substitution, in some iterations of the Algorithm 2, the returned step size \( t \) by LBALS may be zero. In these iterations, which are called “null iterations”, no improvement is achieved. When a null iteration takes place, then in the next iteration(s), the gradient bundle \( G^m_{x_k}(x_k) \) is resampled, such that eventually a serious iteration \( (t_k \neq 0) \) occurs when the resulting gradient bundle improves the search direction sufficiently.

At the first glance, the null iterations are undesirable, since we do some computations without any reduction in the objective function. However, according to [21], this change has some advantages in reducing the number of function evaluations. Moreover, as we will show, it does not affect the convergence of the GSI method and interesting results can be explored. Maybe the most important of them is the subject of the following theorem in which the LBALS is considered as an alternative of BALS.

**Theorem 5.2** In the GSI algorithm, let \( \nu_0 = \nu_{opt} =: \nu > 0, \varepsilon_0 = \varepsilon_{opt} =: \varepsilon > 0 \) and \( m := 4n \). If the function \( f \) is bounded below, then the algorithm terminates after finitely iterations and produces a \( (\nu, G^m_{x_k}(x_k)) \)-stationary point, with probability 1.

**Proof** We know that the GSI algorithm does not terminate in Line 5, with probability 1. So, we consider the case that the Algorithm 2 does not stop in Line 5. By contradiction, suppose that an infinite sequence \( \{x_k\} \) is generated by this algorithm. Then, with probability 1, there exists an infinite set \( K \) such that \( t_k \neq 0 \) for all \( k \in K \). Let \( K = K_1 \cup K_2 \) such that \( \|g^s_{x_k}\| \leq \nu \) at any iteration \( k \in K_1 \) and \( \|g^s_{x_k}\| > \nu \) at any iteration \( k \in K_2 \). For every \( k \in K_2 \), the direction \( d_k = -\|g^s_{x_k}\|^{-1}g^s_{x_k} \) is considered as the search direction and we have

\[
\begin{align*}
f(x_k + t_k d_k) - f(x_k) \leq -ct_k \|g^s_{x_k}\|, & \quad \text{for all } k \in K_2, \\
f(x_{k+1}) - f(x_k) \leq -ct_k \nu, & \quad \text{for all } k \in K_2. \tag{30}
\end{align*}
\]

On the other hand, for every \( k \in K_1 \), since the algorithm does not terminate, we have \( \|g^s_{x_k}\| > \nu \) for all \( k \in K_1 \) and similarly, we can write

\[
\begin{align*}
f(x_{k+1}) - f(x_k) \leq -ct_k \nu, & \quad \text{for all } k \in K_1. \tag{31}
\end{align*}
\]

Combining (30) with (31) we obtain

\[
f(x_{k+1}) - f(x_k) \leq -ct_k \nu, \quad \text{for all } k \in K.
\]

Furthermore, for every \( k \in \mathbb{N}_0 \setminus K \), we have \( t_k = 0 \) and hence

\[
f(x_{k+1}) - f(x_k) = -ct_k = 0, \quad \text{for all } k \in \mathbb{N}_0 \setminus K,
\]

and consequently

\[
f(x_{k+1}) - f(x_k) \leq -ct_k \nu, \quad \text{for all } k.
\]

Using the above inequality inductively, one can write

\[
f(x_{k+1}) \leq f(x_0) - c
\]

\[
\sum_{j=0}^{k} t_j, \quad \text{for all } k. \tag{32}
\]
Since $t_j > \min\{1, \gamma/3\}$ for every $j \in K$, we have
\[ \sum_{j \in K} t_j = \infty, \]
and hence the left hand side of the inequality (32) tends to $-\infty$ as $k \to \infty$, while function $f$ is supposed to be bounded from below. \[ \square \]

**Corollary 5.2** Under assumptions of Theorem 5.2, with probability 1, the GSI Algorithm terminates after finitely serious iterations $k_{\text{max}} \geq 0$ such that
\[ k_{\text{max}} \leq \left\lfloor \frac{f(x_0) - f_l}{c \nu \min\{1, \gamma \varepsilon / 3\}} \right\rfloor + 1, \]
where $f_l$ is a lower bound for $f$. 

**Proof** By contradiction, suppose that an infinite sequence $\{x_k\}$ is generated by this algorithm. Similar to the proof of Theorem 5.2, we have
\[ f(x_{k+1}) - f(x_k) \leq -ct_k \nu, \quad \text{for all } k, \]
and therefore
\[ f(x_{k+1}) \leq f(x_0) - K c \nu \min\{1, \gamma \varepsilon / 3\}, \quad \text{for all } k, \]
in which
\[ K := \left| \{j \in K : j \leq k\} \right|. \]
Since $K \to \infty$ as $k \to \infty$, we conclude that $f(x_k)$ tends to $-\infty$ as $k \to \infty$ which is a contradiction. Moreover, since $f_l$ is a lower bound for the function $f$, one can write
\[ f_l \leq f(x_{k+1}) \leq f(x_0) - K c \nu \min\{1, \gamma \varepsilon / 3\}. \]

Hence, an upper bound for the number of serious iterations is obtained as follows
\[ k_{\text{max}} \leq \left\lfloor \frac{f(x_0) - f_l}{c \nu \min\{1, \gamma \varepsilon / 3\}} \right\rfloor + 1. \]

\[ \square \]

In the next theorem, we examine the convergence analysis of Algorithm 2 under the proposed modifications. In this theorem, the lower $\alpha$-level set of a function $f : \mathbb{R}^n \to \mathbb{R}$ is given by [29]
\[ \text{lev}_{\leq \alpha} f := \{x \in \mathbb{R}^n : f(x) \leq \alpha\}. \]

**Theorem 5.3** Suppose that the objective function $f$ is bounded from below and $\text{lev}_{\leq f(x_0)} f$ is bounded. In Algorithm 2, let $\varepsilon_0, \nu_0 > 0, \nu_{\text{opt}} = \varepsilon_{\text{opt}} = 0$ and $0 < \mu, \theta < 1$. Then, with probability 1, any cluster point of the sequence $\{x_k\}$ generated by this algorithm is a Clarke stationary point.
Proof According to Theorem 5.2, with probability 1, the GSI Algorithm produces a sequence of \( (\nu_i, G^m_{\epsilon_i}(x_k)) \)-stationary points, say \( \{x_i\} \). Hence

\[
\min\{\|g\| : g \in G^m_{\epsilon_i}(x_i)\} \leq \nu_i.
\]

Clearly, \( x_i \in \text{lev}_{\leq f(x_0)} f \) for every \( i \). Thus, the boundedness of \( \text{lev}_{\leq f(x_0)} f \) implies that the sequence \( \{x_i\} \) has at least one cluster point. So, assume that \( \{x_{ij}\} \subset \{x_i\} \) be such that \( x_{ij} \rightarrow \bar{x} \) as \( j \rightarrow \infty \). Therefore

\[
\min\{\|g\| : g \in G^m_{\epsilon_{ij}}(x_{ij})\} \leq \nu_{ij}.
\]  \( (33) \)

Now, let \( \omega > 0 \) be arbitrary. Since \( \nu_i \rightarrow 0 \), there exists \( j_1 > 0 \) such that \( \nu_{ij} < \omega \) for all \( j > j_1 \). In view of \((33)\) we have

\[
\min\{\|g\| : g \in G^m_{\epsilon_{ij}}(x_{ij})\} < \omega,
\]

for all \( j > j_1 \).

Now, we have

\[
\|g^*_{\epsilon_{ij}}\| = \min\{\|g\| : g \in G^m_{\epsilon_{ij}}(x_{ij})\},
\]

for all \( j > j_1 \), and since \( \|g^*_{\epsilon_{ij}}\| < \omega \) for every \( j > j_1 \), we may assume that \( g^*_{\epsilon_{ij}} \rightarrow \bar{g} \) as \( j \rightarrow \infty \). On the other hand, for every \( j > j_1 \) we have \( g^*_{\epsilon_{ij}} \in G^m_{\epsilon_{ij}}(x_{ij}) \) and therefore \( g^*_{\epsilon_{ij}} \in \partial \epsilon_{ij} f(x_{ij}) \). Now, using outer semicontinuity of the set valued map \( \partial f(\cdot) \), we conclude

\[
g \in \partial f(\bar{x}),
\]

which yields

\[
\min\{\|\xi\| : \xi \in \partial f(\bar{x})\} < \omega.
\]

Since \( \omega > 0 \) is arbitrary we conclude that

\[
0 \in \partial f(\bar{x}).
\]

\( \square \)

According to the results of this subsection, when the function \( f \) is bounded below, using LBALS instead of BALS guarantees an upper bound for the number of serious iterations required for obtaining a \((\nu, G^m_{\epsilon}(x_k))\)-stationary point.

6 Numerical results

In this section, we demonstrate the efficiency of the proposed method through the results of numerical experiments. Following [27], we divide test problems into three categories based on the scale of the problems in which problems with \( n \leq 50 \) are considered as small scale problems, medium scale problems have a dimension \( 50 < n \leq 200 \) and large scale problems are identified by \( n > 200 \).

Based on this classification, this section is divided into three experiments. In Experiment 1, we apply the GSI method to a set of nonsmooth small scale problems. Medium and large scale problems are considered in Experiments 2 and 3, where performance profiles are provided to compare the proposed method with the GS method. At each experiment, a set of nonsmooth test problems including convex and nonconvex objectives is considered.
The experiments are implemented in MATLAB software on PC Intel Core i7 2700k CPU 3.5 GHz and 8 GB of RAM. In Experiment 3, we also used MATLAB API for C++ language in order to manage the CPU time elapsed by each run. To provide fair results, we run each problem 5 times using starting points generated randomly from a ball centered at $x_0$ (suggested in the literature) with radius $\|x_0\|/n$. Note that, $x_0 \neq 0$ for all problems, so the starting points for each run were different. In the test problems for which $x_0$ is not available in the literature, we used $x_0 = e := (1, 1, \ldots, 1)$ when $x^* \neq e$ and $x_0 = 2e$ when $x^* = e$.

For the GS method, we used the default setting proposed in [19] and for the GSI method we have the following choices for the parameters.

- **Sample size.** We have experimented different sizes of sample points. Although Lemma 5.5 suggests a sample size $m \geq \bar{m}$, we don’t see any favorable change in search direction by choosing $m > 2n$. Following [19], for the sample size $m = 2n$, a suitable approximation of $\partial_\epsilon f(x)$ is obtained. Since both GS and GSI methods need a great deal of random access memory, it is not recommended to choose a sample size more than $2n$, specially for large scale objectives.

- **Sampling radius.** We choose $\epsilon_0 = 10^{-3}$ for problems with $n \leq 10$ and $\epsilon_0 = 10^{-2}$ for the problems having $n > 10$. Furthermore, we set $\mu = 0.5$. Note that, there is a trade-off between sampling radius and the efficiency of the Ideal direction. Although a larger sampling radius leads to collecting more information of nonsmooth curves, it may cause $\nu_\epsilon^k$ to become zero. The GSI has its best performance when there is a balance between these two factors.

- **Backtracking Armijo line search.** In Algorithm 1, we set $\gamma = 0.5$, $c = 10^{-6}$ and we start with $t = 1$. Following [19], we also limit the number of backtracking steps to 50. Once this limit is reached, the sampling radius is reduced by reduction factor $\mu$ and we set $x_{k+1} := x_k$, and then we go to Line 3. So, a null iteration occurs and the gradient bundle is resampled with a smaller sampling radius.

- **Optimality tolerance.** We select $\nu_0$ with respect to the scale of the problems. For small, medium and large scale problems, we set $\nu_0 = 10^{-3}$, $\nu_0 = 10^{-2}$ and $\nu_0 = 10^{-1}$ respectively. Furthermore, we set $\theta = 0.5$.

- **Stopping criterion.** Since the (local) minimizers of the objective functions are available in our test problems, we terminate the GS and GSI algorithms, if they find a solution which fulfills the following condition

$$\frac{|f(x_k) - f(x^*)|}{|f(x^*)| + 1} < \epsilon.$$  

Here, $\epsilon > 0$ is a given tolerance and $f(x^*)$ is the known (local) minimum of the objective function. We set $\epsilon = 5 \times 10^{-4}$ for small scale problems and $\epsilon = 10^{-3}$ for medium and large scale problems. We also limit the number of iterations to 2000.

6.1 Experiment 1. Small scale problems

In this subsection, we apply the proposed method to a set of small scale problems. A description of the considered set of test problems is presented in Table 1. Numerical results for this set of test problems are given in Table 2. In this table, by the term “Iters” we refer to the number of iterations. Furthermore, the
Table 1: List of test problems for Experiment 1.

| Problem | Name       | Function Type | Type | n | Ref. |
|---------|------------|---------------|------|---|------|
| 1       | QL         | Convex        | 2    |   | [1]  |
| 2       | Wong1      | Convex        | 7    |   | [33] |
| 3       | Wolfe      | Convex        | 2    |   | [33] |
| 4       | SPIRAL     | Convex        | 2    |   | [33] |
| 5       | Rosenbrock | Nonconvex     | 2    |   | [34] |
| 6       | Crescent   | Nonconvex     | 2    |   | [7]  |
| 7       | Mifflin2   | Nonconvex     | 2    |   | [1]  |
| 8       | EVD52      | Nonconvex     | 2    |   | [33] |
| 9       | HS78       | Nonconvex     | 5    |   | [33] |
| 10      | Condition  | Nonconvex     | 45   |   | [34] |

Table 2: Numerical results for small scale problems.

| GSI | Problem | Iters | NII | PII | f_{eval} | g_{eval} |
|-----|---------|-------|-----|-----|----------|----------|
| 1   | QL      | 14    | 12  | 86 %| 76       | 70       |
| 2   | Wong1   | 82    | 75  | 91 %| 730      | 1233     |
| 3   | Wolfe   | 29    | 29  | 100%| 172      | 145      |
| 4   | SPIRAL  | 1735  | 1712| 99 %| 16390    | 8676     |
| 5   | Rosenbrock | 114  | 76  | 67 %| 918      | 568      |
| 6   | Crescent | 22    | 18  | 82 %| 130      | 109      |
| 7   | Mifflin2| 7     | 7   | 100%| 40       | 36       |
| 8   | EVD52   | 28    | 25  | 89 %| 207      | 199      |
| 9   | HS78    | 85    | 51  | 60 %| 1043     | 937      |
| 10  | Condition Number | 20 | 20  | 100%| 43       | 1801     |

number of iterations, in which the search direction is obtained by ideal directions and not by the QP (9), is denoted by “NII” (Number of Ideal Iterations) and the corresponding percentage is abbreviated by “PII” (Percentage of Ideal Iterations). In fact, in such iterations the GSI method only needs to compute the ideal direction. Also, the number of function and gradient evaluations are denoted by “f_{eval}” and “g_{eval}” respectively. Since the CPU time elapsed by the GSI method is almost zero for most of these problems, we don’t report this factor in this experiment.

It can be seen from Table 2 that, the proposed method has been able to solve all the test problems without observing any failure, which means that the GSI method is reliable for this type of problems. In addition, the method has a great potential to find a descent direction without solving the QP (9). In particular, we see that, for Problems 3, 7 and 10 the method reached its termination tolerance without solving any QP.

6.2 Experiment 2. Medium scale problems

In this experiment, we apply the GS and GSI methods to a set of medium scale problems given in Table 3. All of these problems can be formulated with any number of variables.

We compare our method with the GS method by means of performance profiles [37]. Fig. 1 and Fig. 2 demonstrate results for problems with $n = 100$ and $n = 200$. It can be seen from Fig. 1 that, for 60% of the problems the proposed method has
been more successful than the GS method in the sense of using CPU time. Note that, the efficiency of the GSI method depends on the number of Ideal iterations. Therefore, the GSI method is more successful than the GS method as long as Ideal directions define the search direction in majority of the iterations. Moreover, the efficiency of the Ideal iterations in reducing the CPU time elapsed by the QP solver can obviously be seen in Fig. 2.

Fig. 1: Performance profiles based on CPU time for medium scale problems with $n = 100$ (left) and $n = 200$ (right).

Fig. 2: Performance profiles based on QP solver time for medium scale problems with $n = 100$ (left) and $n = 200$ (right).
6.3 Experiment 3. Large scale problems

In this experiment, we consider a set of nonsmooth test problems for which we set \( n = 500 \) and \( n = 1000 \). A description of the test problems is presented in Table 4. To the best of our knowledge, it is the first time that the GS method is applied to a set of problems having more than 200 variables. Note that, due to heavy computational cost, an unreasonable amount of CPU time may be required for each run. Accordingly, we used MATLAB API for C++ language to manage the CPU time for the considered large scale problems.

Fig. 3 and Fig. 4 show results for problems with 500 and 1000 variables. As Fig. 3 demonstrates, at 70% of the problems having 500 variables, the GSI method used less CPU time than the GS method. In addition, Fig. 3 also demonstrates the superiority of the GSI method over the GS method for problems with 1000 variables. It can be seen from Fig. 4 that, the proposed method has a remarkable potential to reduce the CPU time consumed by the QP solver. Thus, in the problems for which the QP (9) is a demanding part of the GS method, the GSI method is a more desirable choice.

| Problem | Name                          | Function Type | Ref. |
|---------|-------------------------------|---------------|------|
| 1       | Titled Norm Function          | Convex        | [38] |
| 2       | A Convex Partly Smooth Function | Convex        | [38] |
| 3       | MAXQ                          | Convex        | [17] |
| 4       | Chained LQ                    | Convex        | [17] |
| 5       | Nesterov’s Chebyshev-Rosenbrock Function1 | Nonconvex | [38] |
| 6       | Nesterov’s Chebyshev-Rosenbrock Function2 | Nonconvex | [38] |
| 7       | Chained Mifflin 2              | Nonconvex     | [17] |
| 8       | Chained Crescent I            | Nonconvex     | [17] |
| 9       | Chained Crescent II           | Nonconvex     | [17] |
| 10      | A Nonconvex Partly Smooth Function. | Nonconvex | [38] |

Fig. 3: Performance profiles based on CPU time for large scale problems with \( n = 500 \) (left) and \( n = 1000 \) (right).
7 Conclusion

In this article, by introducing the Ideal direction as an alternative of the approximate $\epsilon$-steepest descent direction, we reduced the need of solving quadratic subproblem in the GS method. We have shown that, Ideal directions are easy to compute and a nonzero Ideal direction can make a substantial reduction in the objective function. Furthermore, we showed that by using Ideal directions, not only is there no need to consider the quadratic subproblem in the smooth regions, but also it can be an efficient search direction when the method starts to track a nonsmooth curve towards a stationary point. We studied the convergence of the proposed method and we proved that using Ideal directions preserves the global convergence property of the GS method. Furthermore, by using limited backtracking Armijo line search and under some moderate assumptions, we proposed an upper bound for the number of serious iterations in the GSI algorithm. We presented numerical results using small, medium and large scale nonsmooth test problems. Our numerical results clearly demonstrate that, the GSI method inherits robustness from the GS method and it is a significant enhancement in reducing the number of quadratic subproblems.

Acknowledgements The authors would like to thank Dr. L. E. A. Simões and Dr. Frank. E. Curtis for their valuable comments that improved the quality of the paper.

References

1. Mäkelä, M.M., Neittaanmäki, P.: Nonsmooth optimization: Analysis and algorithms with applications to optimal control. Singapore: World Scientific Publishing Co. (1992)
2. Nikolova, M.: Minimizers of cost-functions involving nonsmooth data-fidelity terms, application to the processing of outliers. SIAM Journal on Numerical Analysis 40(3), 965–994 (2002)
3. Astorino, A., Gaudioso, M.: Polyhedral separability through successive LP. J. Optim. Theory Appl 112(2), 265–293 (2002)
4. Lavor, C., Liberti, L., Maculan, N.: Molecular distance geometry problem (2nd ed.). Springer (2009)
5. Witten, I.H., Frank, E.: Data mining: Practical machine learning tools and techniques (2nd ed.). Elsevier Inc (2005)
6. Clarke, F.H.: Optimization and nonsmooth analysis. SIAM (1990)
7. Kiwiel, K.C.: Methods of descent for nondifferentiable optimization. Springer-Verlag (1985)
8. Shor, N.Z.: Minimization methods for non-differentiable functions. Springer (1985)
9. Hiriart-Urruty, J.B., Chal, C.L.: Convex Analysis and Minimization Algorithms I. A Series of Comprehensive Studies in Mathematics. Springer-Verlag (1993)
10. Hiriart-Urruty, J.B., Chal, C.L.: Convex Analysis and Minimization Algorithms II. A Series of Comprehensive Studies in Mathematics. Springer-Verlag (1993)
11. Uryasev, S.P.: New variable metric algorithms for nondifferentiable optimization problems. J. Optim. Theory Appl 71, 359–388 (1991)
12. Vlˇcék, J., Lukˇšan, L.: Globally convergent variable metric method for nonconvex nondifferentiable unconstrained minimization. J. Optim. Theory Appl 111(2), 407–430 (2001)
13. Bagirov, A.M., Jin, L., Karmitsa, N., Al Nuaimat, A., Sultanova, N.: A subgradient method for nonconvex nonsmooth optimization. J. Optim. Theory Appl 157, 416–435 (2013)
14. Mahdavi-Amiri, N., Yousefpour, R.: An effective nonsmooth optimization algorithm for locally Lipschitz functions. J. Optim. Theory Appl 155(1), 180–195 (2012)
15. Nesterov, Y.: Primal-dual subgradient methods for convex problems. Math. Program 120(1), 221–259 (2009)
16. Bagirov, A.M., Ganjehlou, A.N.: A quasisecant method for minimizing nonsmooth functions. Optim. Methods Softw 25(1), 3–18 (2010)
17. Haarala, M., Miettinen, K., Mäkelä, M.M.: New limited memory bundle method for large-scale nonsmooth optimization. Optim. Methods Softw 19(6), 673–692 (2004)
18. Lukˇšan, L., Vlˇcˇek, J.: A bundle-Newton method for nonsmooth unconstrained minimization. Math. Program 83(1), 373–391 (1998)
19. Burke, J.V., Lewis, A.S., Overton, M.L.: A robust gradient sampling algorithm for nonconvex, nonconvex optimization. SIAM J. Optim 15(3), 751–779 (2005)
20. Burke, J.V., Curtis, F.E., Lewis, A.S., Overton, M.L., Simões, L.E.A.: Gradient sampling methods for nonconvex optimization. arXiv : 1804.11003 (2018)
21. Kiwiel, K.C.: Convergence of the gradient sampling algorithm for nonconvex nonconvex optimization. SIAM J. Optim 18(2), 379–388 (2007)
22. Kiwiel, K.C.: A nonderivative version of the gradient sampling algorithm for nonconvex nonconvex optimization. SIAM J. Optim 20(4), 1983–1994 (2010)
23. Curtis, F.E., Que, X.: An adaptive gradient sampling algorithm for nonconvex nonconvex optimization. Optim. Methods Softw 28(6), 1302–1324 (2013)
24. Nocedal, J., Wright, S.J.: Numerical Optimization, second edition. Springer (2006)
25. Curtis, F.E., Overton, M.L.: A sequential quadratic programming algorithm for nonconvex, nonsmooth constrained optimization. SIAM J. Optim 22(2), 474–500 (2012)
26. Helou, E.S., Santos, A.S., Simões, L.E.A.: On the local convergence analysis of the gradient sampling method for finite max-functions. J. Optim. Theory
27. Bagirov, A., Karimtsa, N., Mäkelä, M.M.: Introduction to nonsmooth optimization. Springer (2014)
28. Evans, L.C., Gariepy, R.F.: Measure theory and fine properties of functions, Revised Edition. CRC Press (1992)
29. Rockafellar, R.T., Wets, R.J.B.: Variational analysis. Springer (2004)
30. Rockafellar, R.T.: Convex Analysis. Princeton University Press (1970)
31. Bazaraa, M.S., Sherali, H.D., Shetty, C.M.: Nonlinear Programming. John Wiley & Sons, Inc. (2006)
32. Ehrgott, M.: Multicriteria Optimization. Springer (2005)
33. Lukšan, L., Vlček, J.: Test problems for nonsmooth unconstrained and linearly constrained optimization. Tech. rep., Institute of Computer Science, Academy of Sciences of the Czech Republic (2000)
34. Skaaja, M.: Limited memory BFGS for nonsmooth optimization. Master’s thesis, New York University (2010)
35. Lukšan, L., Tcma, M., Siska, M., Vlček, J., Ramesova, N.: Ufo 2002. interactive system for universal functional optimization. Tech. rep., Institute of Computer Science, Academy of Sciences of the Czech Republic (2002)
36. Grothey, A.: Decomposition methods for nonlinear nonconvex optimization problems. Ph.D. thesis, University of Edinburgh (2001)
37. Dolan, E., Moré, J.: Benchmarking optimization software with performance profiles. Math. Program 91(2), 201–213 (2002)
38. Lewis, A.S., Overton, M.L.: Nonsmooth optimization via BFGS URL https://cs.nyu.edu/overton/papers/nonsmoothalg.html