Gebken, Bennet

A note on the convergence of deterministic gradient sampling in nonsmooth optimization

Computational Optimization and Applications

Provided in Cooperation with:
Springer Nature

Suggested Citation: Gebken, Bennet (2024) : A note on the convergence of deterministic gradient sampling in nonsmooth optimization, Computational Optimization and Applications, ISSN 1573-2894, Springer US, New York, NY, Vol. 88, Iss. 1, pp. 151-165, https://doi.org/10.1007/s10589-024-00552-0

Computational Optimization and Applications (2024) 88:151–165
https://doi.org/10.1007/s10589-024-00552-0

A note on the convergence of deterministic gradient sampling in nonsmooth optimization

Bennet Gebken[1]

Received: 5 July 2023 / Accepted: 4 January 2024 / Published online: 6 February 2024

Abstract
Approximation of subdifferentials is one of the main tasks when computing descent directions for nonsmooth optimization problems. In this article, we propose a bisection method for weakly lower semismooth functions which is able to compute new subgradients that improve a given approximation in case a direction with insufficient descent was computed. Combined with a recently proposed deterministic gradient sampling approach, this yields a deterministic and provably convergent way to approximate subdifferentials for computing descent directions.

Keywords Nonsmooth optimization · Nonsmooth analysis · Nonconvex optimization · Gradient sampling

1 Introduction

Nonsmooth optimization is concerned with the optimization of a function $f : \mathbb{R}^n \to \mathbb{R}$ which is not necessarily continuously differentiable. For such functions, one cannot rely on the gradient for describing the local behavior around a given point. As a replacement, generalized concepts from nonsmooth analysis can be employed. If the objective is still locally Lipschitz continuous, as is the case in many practical applications, then the standard approach is to use the *Clarke subdifferential* ∂f [1]. However, since $\partial f(x)$ reduces to the gradient if f is continuously differentiable at x, and since f is typically continuously differentiable almost everywhere, the Clarke subdifferential cannot be used to capture nonsmoothness numerically. To circumvent this issue, the *(Goldstein)* ε*-subdifferential* $\partial_\varepsilon f$ [2] may be used instead, which is the convex hull of all Clarke subdifferentials in an ε-ball around a given point. For the ε-subdifferential, it is sufficient for x to have a distance of at most ε to a nonsmooth point of f to capture

✉ Bennet Gebken
bgebken@math.upb.de

[1] Insitute of Mathematics, Paderborn University, Warburger Str. 100, Paderborn 33098, Germany

the nonsmoothness. As such, it can be interpreted as a stabilized version of the Clarke subdifferential.

It is well-known that the element $\bar{v} \in -\partial_\varepsilon f(x_0)$ with the smallest norm is a descent direction for f at x_0 [1, 2]. Unfortunately, the full ε-subdifferential that is required to compute this direction is rarely available in practice and has to be approximated instead. To this end, in the *gradient sampling method* [3–5], the idea is to approximate $\partial_\varepsilon f(x_0)$ by the convex hull of the gradients at randomly generated points in the ε-ball around x_0 where f is differentiable. While this is easy to implement and convergence can be shown with probability 1, randomly computing gradients means that one generally computes more gradients than would be necessary. Furthermore, a good approximation may require a large and a priori unknown number of sample points (which is highlighted in Appendix 1). As an alternative, in [6–8], a deterministic sampling approach was used. There, the idea is to compute the approximation of $\partial_\varepsilon f(x_0)$ iteratively, by starting with $W = \{\xi\}$ for a subgradient $\xi \in \partial f(x_0)$ at x_0, and then adding new elements of $\partial_\varepsilon f(x_0)$ to W until $\operatorname{conv}(W)$ is a satisfactory approximation. The mechanism for finding new elements of $\partial_\varepsilon f(x_0)$ is based on the observation that if v is a direction that yields less descent than expected (based on the current approximation $\operatorname{conv}(W)$), then there has to be a point $x_0 + tv$ with $t > 0$ in the ε-ball at which a new subgradient $\xi' \in \partial f(x_0 + tv) \subseteq \partial_\varepsilon f(x_0)$ with $\xi' \notin \operatorname{conv}(W)$ can be sampled. To find such a t, a subroutine based on bisection of the interval $(0, \varepsilon/\|v\|)$ is used. While in [6–8], it was analyzed why this subroutine likely works (i.e., terminates) in practice and while termination was also observed in all numerical examples, a full proof (under reasonable assumptions for f) was not given.

The goal of this note is to close the above mentioned gap in the convergence theory of the deterministic gradient sampling approach. The bisection algorithm in [6–8] is based on reformulating the problem of finding a new element of $\partial_\varepsilon f(x_0)$ as finding a point $t > 0$ in which a certain nonsmooth function $h : \mathbb{R} \to \mathbb{R}$ is increasing. The convergence issues arise in cases where the bisection converges to a critical point $\bar{t}$ of h (i.e., to a point $\bar{t}$ with $0 \in \partial h(\bar{t})$). To fix these issues, we replace h by a slightly modified function $h_{\tilde{c}}$. We then show convergence of the resulting method for the case where f is weakly lower semismooth [9, 10] (meaning that $-f$ is weakly upper semismooth). Since semismooth functions [11] are weakly lower semismooth, this case includes continuously differentiable functions, convex functions and piecewise differentiable functions [12]. As our result is essentially just concerned with the deterministic computation of new ε-subgradients (not necessarily in the context of [6–8]), it has also use in other methods based on gradient sampling, like [13, 14]. Our method has strong similarities to Procedure 4.1 in [15], which is used for the computation of new subgradients in a bundle framework. It is based on the same idea, but differs in the condition used for bisection and the stopping criterion.

The remainder of this article is structured as follows: In Sect. 2 we introduce the basics of gradient sampling and the bisection algorithm from [7] (which is identical to the one in [8] and almost identical to the one in [6]). In [7], the bisection was only a small part of a larger algorithm, but we will introduce it here in a stand-alone way for convenience. In Sect. 3 we construct the improved bisection algorithm and show its convergence when f is weakly lower semismooth. In Sect. 4 we visualize the

behavior of the improved bisection method in a simple example. Finally, in Sect. 5, we summarize our results and discuss possible directions for future research.

2 Computing descent directions for nonsmooth functions

In this section, we summarize the basic ideas of gradient sampling from [3–8]. To this end, let $f : \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz continuous. The *Clarke subdifferential* [1] of f at $x \in \mathbb{R}^n$ is given by

$$\partial f(x) := \operatorname{conv}\left(\left\{\xi \in \mathbb{R}^n : \exists (x^j)_j \in \mathbb{R}^n \setminus \Omega \text{ with } \lim_{j\to\infty} x^j = x \text{ and } \lim_{j\to\infty} \nabla f(x^j) = \xi\right\}\right), \tag{1}$$

where $\Omega \subseteq \mathbb{R}^n$ is the set of points at which f is not differentiable. (By Rademacher's Theorem [1], Ω has measure zero.) The elements of $\partial f(x)$ are called *(Clarke) subgradients*. In theory, the Clarke subdifferential can be used similarly to the standard gradient, as it can be used in generalized versions of results like the mean value theorem, the chain rule and optimality conditions [1]. In practice however, there are severe problems: $\partial f(x)$ only captures the nonsmoothness of f if x is a point where f is not continuously differentiable, which is typically a null set. Furthermore, when $x \in \Omega$, there is no general way of computing the full subdifferential, i.e., all subgradients. Instead, a reasonable assumption is that we can only evaluate a single, arbitrary subgradient from $\partial f(x)$ at any $x \in \mathbb{R}^n$. Further explanations and examples for these issues can be found in [16]. To circumvent these problems, a more suitable object to use as a generalized derivative for constructing descent methods is the *(Goldstein) ε-subdifferential* [2]

$$\partial_\varepsilon f(x) := \operatorname{conv}\left(\bigcup_{y\in B_\varepsilon(x)} \partial f(y)\right),$$

where $\varepsilon \geq 0$, $B_\varepsilon(x) := \{y \in \mathbb{R}^n : \|y - x\| \leq \varepsilon\}$ and $\|\cdot\|$ is the Euclidean norm. The elements of $\partial_\varepsilon f(x)$ are called *ε-subgradients*. The ε-subdifferential can be interpreted as a "stabilized" Clarke subdifferential. In particular, it can be used to compute descent directions, as we demonstrate in the following.

Let $x_0 \in \mathbb{R}^n$, $v \in \mathbb{R}^n \setminus \{0\}$ and $\varepsilon > 0$. Then a simple application of the mean value theorem ([1], Theorem 2.3.7) shows that

$$f(x_0 + tv) \leq f(x_0) + t \max_{\xi\in\partial_\varepsilon f(x_0)} \langle \xi, v\rangle \quad \forall t \in \left(0, \frac{\varepsilon}{\|v\|}\right]. \tag{2}$$

Thus, directions v with $\langle \xi, v\rangle < 0$ for all $\xi \in \partial_\varepsilon f(x_0)$ are descent directions for f at x_0. Based on convex analysis [17], the direction that minimizes the maximum of the

inner products on the right-hand side of (2), called the *ε-steepest descent direction*, can be computed as

$$\bar{v} := -\arg\min_{\xi \in \partial_\varepsilon f(x_0)} \|\xi\|^2. \tag{3}$$

It holds either $\bar{v} = 0$, in which case $0 \in \partial_\varepsilon f(x_0)$ and x_0 is called *ε-critical*, or

$$\langle \xi, \bar{v} \rangle \leq -\|\bar{v}\|^2 < 0 \quad \forall \xi \in \partial_\varepsilon f(x_0) \tag{4}$$

and, due to (2),

$$f\left(x_0 + \frac{\varepsilon}{\|\bar{v}\|}\bar{v}\right) \leq f(x_0) - \varepsilon\|\bar{v}\|. \tag{5}$$

Unfortunately, the full ε-subdifferential required to solve Problem (3) is rarely available in practice. Thus, the direction $\bar{v}$ has to be approximated.

2.1 Random gradient sampling

In the standard gradient sampling framework [3–5], the ε-subdifferential is approximated by randomly (independently and uniformly) sampling $m \geq n + 1$ elements $x^1, \ldots, x^m \in B_\varepsilon(x_0) \backslash \Omega$ and setting

$$W := \{\nabla f(x_0), \nabla f(x^1), \ldots, \nabla f(x^m)\} \subseteq \partial_\varepsilon f(x_0).$$

(The differentiability of f at the current iterate x_0 is enforced via a differentiability check.) As an approximation of $\bar{v}$ from (3), the direction

$$v^{\mathrm{GS}} := -\arg\min_{\xi \in \mathrm{conv}(W)} \|\xi\|^2 \tag{6}$$

is computed, and an Armijo-like backtracking line search is used to assure decrease in f. It can be shown that when dynamically reducing the sampling radius ε, then the resulting descent method (cf. [4], GS Algorithm) produces a sequence converging to a critical point of f with probability 1. Unfortunately, sampling randomly means that a large number of sample points m may be required to assure that v^{GS} is a good approximation of $\bar{v}$, i.e., to assure that meaningful decrease is achieved in every descent step. This drawback is highlighted in Appendix 1.

2.2 Deterministic gradient sampling

Instead of randomly sampling points from $B_\varepsilon(x_0)$ for the approximation of $\partial_\varepsilon f(x_0)$, there is also a deterministic approach [7]. (In [7] *multiobjective* problems are considered, but we will only consider the special case of a single objective here.) Assume that x_0 is not ε-critical. The idea is to start with a subset $W \subseteq \partial_\varepsilon f(x_0)$ (e.g., $W = \{\xi\}$ for $\xi \in \partial f(x_0)$) and to then iteratively add new subgradients to W until a direction

$$\tilde{v} := -\arg\min_{\xi \in \mathrm{conv}(W)} \|\xi\|^2 \tag{7}$$

is found that yields sufficient descent. The meaning of "sufficient descent" can be derived from (5): For some fixed $c \in (0, 1)$, we want to have at least

$$f\left(x_0 + \frac{\varepsilon}{\|\tilde{v}\|}\tilde{v}\right) \leq f(x_0) - c\varepsilon\|\tilde{v}\|. \tag{8}$$

To find a new subgradient (that is not already contained in $\mathrm{conv}(W)$) in case $\tilde{v}$ does not yield sufficient descent, note that the mean value theorem implies that there are $t' \in (0, \varepsilon/\|\tilde{v}\|)$ and $\xi' \in \partial f(x_0 + t'\tilde{v})$ with

$$\begin{aligned} & \frac{\varepsilon}{\|\tilde{v}\|}\langle \xi', \tilde{v}\rangle = f\left(x_0 + \frac{\varepsilon}{\|\tilde{v}\|}\tilde{v}\right) - f(x_0) > -c\varepsilon\|\tilde{v}\| \\ \Leftrightarrow \quad & \langle \xi', \tilde{v}\rangle > -c\|\tilde{v}\|^2. \end{aligned} \tag{9}$$

Analogously to (3) and (4), it holds

$$\langle \xi, \tilde{v}\rangle \leq -\|\tilde{v}\|^2 < 0 \quad \forall \xi \in \mathrm{conv}(W).$$

Thus, (9) implies that $\xi' \notin \mathrm{conv}(W)$, so adding ξ' to W improves the approximation of $\partial_\varepsilon f(x_0)$. In [6, 7], it was proven that iteratively computing $\tilde{v}$ via (7) while adding new subgradients to W as above yields a finite algorithm which deterministically computes descent directions satisfying (8) (as long as $0 \notin \partial_\varepsilon f(x_0)$). (For a formal definition of this algorithm, see Algorithm 2 in [7] for $k = 1$.)

Unfortunately, the above application of the mean value theorem does not yield an explicit formula for the computation of ξ' as in (9), so additional effort is required in practice. To this end, a strategy based on bisection can be used: Let

$$h : \mathbb{R} \to \mathbb{R}, \quad t \mapsto f(x_0 + t\tilde{v}) - f(x_0) + ct\|\tilde{v}\|^2. \tag{10}$$

By the chain rule ([1], Theorem 2.3.9) it holds

$$\partial h(t) \subseteq \langle \partial f(x_0 + t\tilde{v}), \tilde{v}\rangle + c\|\tilde{v}\|^2, \tag{11}$$

so $\partial h(t') \cap \mathbb{R}^{>0} \neq \emptyset$ for $t' \in (0, \varepsilon/\|\tilde{v}\|)$ would imply that there is some $\xi' \in \partial f(x_0 + t'\tilde{v})$ as in (9). Thus, roughly speaking, the idea is to search for some interval in $(0, \varepsilon/\|\tilde{v}\|)$ on which h is monotonically increasing. In [7] this was done via Algorithm 1.

It performs bisections such that $h(a_j) < h(b_j)$ for all $j \in \mathbb{N}$, while checking whether a ξ' was found satisfying (9). In [6–8], it was argued why the algorithm is likely to terminate in practice, and termination was also observed in all numerical examples, but a proper analysis was not carried out. There are basically two issues that may cause the algorithm not to terminate:

Algorithm 1 Computation of new ε-subgradients

Require: Point $x_0 \in \mathbb{R}^n$, radius $\varepsilon > 0$, descent parameter $c \in (0, 1)$, direction $\tilde{v} \in \mathbb{R}^n \setminus \{0\}$ violating (8).
1: Initialize $j = 1$, $a_1 = 0$, $b_1 = \frac{\varepsilon}{\|\tilde{v}\|}$ and $t_1 = \frac{1}{2}(a_1 + b_1)$.
2: Compute $\xi' \in \partial f(x_0 + t_j \tilde{v})$.
3: If $\langle \xi', \tilde{v} \rangle > -c\|\tilde{v}\|^2$ then stop.
4: If $h(b_j) > h(t_j)$ then set $a_{j+1} = t_j$ and $b_{j+1} = b_j$. Otherwise set $a_{j+1} = a_j$ and $b_{j+1} = t_j$.
5: Set $t_{j+1} = \frac{1}{2}(a_{j+1} + b_{j+1})$, $j = j + 1$ and go to step 2.

1. One may never encounter a t_j with $\partial h(t_j) \cap \mathbb{R}^{\geq 0} \neq \emptyset$ during the bisection, even if $\partial h(\bar{t}) \cap \mathbb{R}^{>0} \neq \emptyset$ holds for the limit $\bar{t} \in [0, \varepsilon/\|\tilde{v}\|]$ of $(t_j)_j$.
2. The subgradient ξ' evaluated in step 2 may not correspond to a subgradient $\langle \xi', \tilde{v} \rangle + c\|\tilde{v}\|^2 \in \partial h(t_j)$, since we do not have equality in (11). So one could have $\langle \xi', \tilde{v} \rangle \leq -c\|\tilde{v}\|^2$ even when $\partial h(t_j) \subseteq \mathbb{R}^{>0}$.

In [8], Example 4.3.4, an example was constructed for which Algorithm 1 does not terminate with a function h that is not semismooth (cf. [11]). In the following, we will construct a more nuanced example that shows that the algorithm may also fail for semismooth functions.

Example 1 For $i \in \mathbb{N} \cup \{0\}$ let

$$x_i^1 := 1 - 7 \cdot 2^{-i-3}, \ \varphi_i^1 := 1 - 9 \cdot 2^{-2i-3},$$
$$x_i^2 := 1 - 5 \cdot 2^{-i-3}, \ \varphi_i^2 := 1 - 3 \cdot 2^{-2i-4}.$$

Then $x_i^1 < x_i^2 < x_{i+1}^1$ for all $i \in \mathbb{N} \cup \{0\}$. We construct a function $\varphi : \mathbb{R} \to \mathbb{R}$ as follows: For $x < 0$ let $\varphi(x) := -\frac{1}{2}x$, for $x \geq 1$ let $\varphi(x) := 1$ and on $[0, 1)$ let φ be the piecewise linear function with $\varphi(0) = 0$ and

$$\varphi(x_i^1) = \varphi_i^1 \quad \text{and} \quad \varphi(x_i^2) = \varphi_i^2 \quad \forall i \in \mathbb{N} \cup \{0\}.$$

The graph of φ is shown in Fig. 1a. Figure 1b shows the gradient of φ at points where φ is differentiable, and a vertical line from smallest to the largest subgradient at points where φ is not differentiable. We see that in the limit $x \to 1$, all subgradients tend to 0. Based on this observation, it can be shown that φ is semismooth.

Now let

$$f : \mathbb{R} \to \mathbb{R}, \quad x \mapsto \varphi(x) - \frac{1}{2}x$$

as shown in Fig. 1c. Let $x_0 = 0$, $\varepsilon = 1$ and $c = 1/2$. Assume that we have evaluated the subgradient $\xi := -1 \in \partial f(0) = [-3/2, -1]$. Then for $W = \{\xi\}$, the direction $\tilde{v}$ from (7) is simply $\tilde{v} = -\xi = 1$. When checking whether $\tilde{v}$ yields sufficient decrease, we see that

$$f\left(x_0 + \frac{\varepsilon}{\|\tilde{v}\|}\tilde{v}\right) - f(x_0) = f(1) - f(0) = \frac{1}{2} > -\frac{1}{2} = -c\varepsilon\|\tilde{v}\|,$$

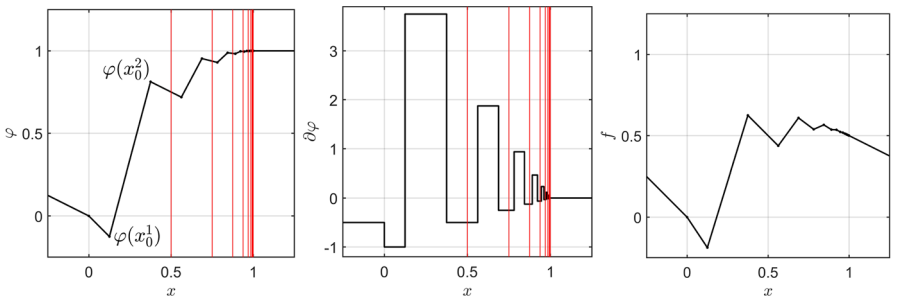

Fig. 1 The graphs of **a** $\varphi = h$, **b** $\partial\varphi = \partial h$ and **c** f in Example 1. The red lines show the values of $t_j = 1 - 2^{-j}$ for $j \in \mathbb{N}$ (color figure online)

i.e., $\tilde{v}$ does not yield sufficient (or even any) decrease.
For Algorithm 1 we have

$$h(t) = f(x_0 + t\tilde{v}) - f(x_0) + ct\|\tilde{v}\|^2 = f(t) + \frac{1}{2}t = \varphi(t).$$

Since $1 = h(1) = h(b_1) > h(t)$ for all $t \in [0, 1)$, we have

$$a_j = 1 - 2^{-j+1},\ b_j = 1,\ t_j = 1 - 2^{-j} \quad \forall j \in \mathbb{N},$$

as indicated in Fig. 1a and b. By construction (cf. Figure 1b), f is continuously differentiable at all $x_0 + t_j\tilde{v} = t_j$ and

$$\partial f(x_0 + t_j\tilde{v}) = \{\nabla f(t_j)\} = \left\{-2^{-j} - \frac{1}{2}\right\} \quad \forall j \in \mathbb{N},$$

so

$$\langle \xi', \tilde{v}\rangle = -2^{-j} - \frac{1}{2} < -\frac{1}{2} = -c\|\tilde{v}\|^2 \quad \forall j \in \mathbb{N}.$$

Thus, Algorithm 1 does not terminate.

Note that in the previous example, only the nonsmoothness of f at $x = 1$ is relevant for the failure of the algorithm. Thus, by "smoothing" the objective f at all kinks except 1, one could even construct a semismooth function which is continuously differentiable everywhere outside of a single point for which the algorithm fails.

3 Improved bisection method

In this section, we propose a slightly modified version of Algorithm 1 and prove termination for the case where the objective f is weakly lower semismooth. For any

$x \in \mathbb{R}^n$, we assume that we are only able to evaluate a single, arbitrary subgradient of the locally Lipschitz continuous f at x.

3.1 Derivation of the improved method

Roughly speaking, the idea is to use a smaller parameter $\tilde{c} < c$ in h (cf. (10)) to try to find a new subgradient that satisfies a stricter version of inequality (9). This will solve the first of the issues described in Sect. 2.2, as even points in which ∂h is negative but close to zero then suffice to find subgradients that satisfy the weaker requirement with respect to c.

More precisely, note that $\tilde{v}$ is an unacceptable descent direction if and only if

$$\begin{aligned} & f\left(x_0 + \frac{\varepsilon}{\|\tilde{v}\|}\tilde{v}\right) > f(x_0) - c\varepsilon\|\tilde{v}\| \\ \Leftrightarrow \quad & c_{\min} := -\frac{f\left(x_0 + \frac{\varepsilon}{\|\tilde{v}\|}\tilde{v}\right) - f(x_0)}{\varepsilon\|\tilde{v}\|} < c. \end{aligned}$$

Thus, if $\tilde{v}$ is an unacceptable direction, then it is also unacceptable if we replace c in (8) by any $\tilde{c} \in (c_{\min}, c) \neq \emptyset$. In other words, we could apply Algorithm 1 for any $\tilde{c} \in (c_{\min}, c)$ and the method would still produce sequences with $h_{\tilde{c}}(a_j) < h_{\tilde{c}}(b_j)$ for all $j \in \mathbb{N}$, where, analogously to (10),

$$h_{\tilde{c}} : \mathbb{R} \to \mathbb{R}, \quad t \mapsto f(x_0 + t\tilde{v}) - f(x_0) + \tilde{c}t\|\tilde{v}\|^2.$$

In step 3, the method would check whether $\langle \xi', \tilde{v}\rangle > -\tilde{c}\|\tilde{v}\|^2$. But since $\tilde{c} < c$, this inequality is stricter than (9). Thus, instead, we apply Algorithm 1 with only h being replaced by $h_{\tilde{c}}$ and the rest unchanged. In terms of the subdifferential of $h_{\tilde{c}}$, this means that the method may stop as soon as

$$\exists g \in \partial h_{\tilde{c}}(t_j) : g = \langle \xi', \tilde{v}\rangle + \tilde{c}\|\tilde{v}\|^2 > (\tilde{c} - c)\|\tilde{v}\|^2, \tag{12}$$

where $(\tilde{c} - c)\|\tilde{v}\|^2 < 0$. For clarity, this modified algorithm is denoted in Algorithm 2.

Algorithm 2 Improved computation of new ε-subgradients

Require: Point $x_0 \in \mathbb{R}^n$, radius $\varepsilon > 0$, descent parameters $c \in (0, 1)$, $\tilde{c} \in (c_{\min}, c)$, direction $\tilde{v} \in \mathbb{R}^n \setminus \{0\}$ violating (8).

1: Initialize $j = 1$, $a_1 = 0$, $b_1 = \frac{\varepsilon}{\|\tilde{v}\|}$ and $t_1 = \frac{1}{2}(a_1 + b_1)$.
2: Compute $\xi' \in \partial f(x_0 + t_j\tilde{v})$.
3: If $\langle \xi', \tilde{v}\rangle > -c\|\tilde{v}\|^2$ then stop.
4: If $h_{\tilde{c}}(b_j) > h_{\tilde{c}}(t_j)$ then set $a_{j+1} = t_j$ and $b_{j+1} = b_j$. Otherwise set $a_{j+1} = a_j$ and $b_{j+1} = t_j$.
5: Set $t_{j+1} = \frac{1}{2}(a_{j+1} + b_{j+1})$, $j = j + 1$ and go to step 2.

3.2 Proof of termination

We begin the analysis of Algorithm 2 with some simple, technical results.

Lemma 1 *If Algorithm 2 does not terminate, then*

(i) $(a_j)_j$, $(b_j)_j$ *and* $(t_j)_j$ *have the same limit* $\bar{t} \in [0, \varepsilon/\|\tilde{v}\|]$,
(ii) $\bar{t} \in [a_j, b_j]$ *for all* $j \in \mathbb{N}$,
(iii) $(h_{\tilde{c}}(b_j))_j$ *is monotonically increasing and* $h_{\tilde{c}}(a_j) < h_{\tilde{c}}(b_j)$ *for all* $j \in \mathbb{N}$,
(iv) $t_j < \bar{t}$ *for infinitely many* $j \in \mathbb{N}$.

Proof (i) By construction $(a_j)_j$ is monotonically increasing and $(b_j)_j$ is monotonically decreasing in $[0, \varepsilon/\|\tilde{v}\|]$, so both sequences converge. Since Algorithm 2 does not terminate, it holds $\lim_{j\to\infty} b_j - a_j = 0$, so they must converge to the same limit $\bar{t} \in [0, \varepsilon/\|\tilde{v}\|]$. Since $t_j \in (a_j, b_j)$ for all $j \in \mathbb{N}$, $(t_j)_j$ must have the same limit.

(ii) Assume that $\bar{t} \notin [a_j, b_j]$ for some $j \in \mathbb{N}$. Then either $\bar{t} < a_j$ or $\bar{t} > b_j$. Due to the monotonicity of $(a_j)_j$ and $(b_j)_j$, this is a contradiction to $\bar{t}$ being the limit of both sequences.

(iii) By construction, the value of $(b_j)_j$ only changes when $h_{\tilde{c}}(b_j) \leq h_{\tilde{c}}(t_j)$ in step 4. In this case b_{j+1} is set to t_j, so we have $h_{\tilde{c}}(b_{j+1}) = h_{\tilde{c}}(t_j) \geq h_{\tilde{c}}(b_j)$ and $h_{\tilde{c}}(a_{j+1}) = h_{\tilde{c}}(a_j)$. If, on the other hand, $h_{\tilde{c}}(b_j) > h_{\tilde{c}}(t_j)$, then $a_{j+1} = t_j$, so $h_{\tilde{c}}(a_{j+1}) = h_{\tilde{c}}(t_j) < h_{\tilde{c}}(b_j) = h_{\tilde{c}}(b_{j+1})$. The proof follows by induction.

(iv) Assume that this does not hold. Then there is some $N \in \mathbb{N}$ such that $t_j \geq \bar{t}$ for all $j > N$. By construction of $(t_j)_j$, $t_j = \bar{t}$ may only hold once, so we can assume w.l.o.g. that $t_j > \bar{t}$ for all $j > N$. Since $\bar{t} \in [a_j, b_j]$ for all $j \in \mathbb{N}$ and t_j is the midpoint of $[a_j, b_j]$, this implies that $a_j = a_N$ for all $j > N$. In particular, $\bar{t} = \lim_{j\to\infty} a_j = a_N$. Since $h_{\tilde{c}}(\bar{t}) = h_{\tilde{c}}(a_N) < h_{\tilde{c}}(b_N)$ and $(h_{\tilde{c}}(b_j))_j$ is monotonically increasing with $\lim_{j\to\infty} h_{\tilde{c}}(b_j) = h_{\tilde{c}}(\bar{t})$ due to continuity, this is a contradiction. □

To be able to fix the second of the two issues mentioned in Sect. 2.2, we need a stronger assumption for f. An easy way to solve the issue would be to force equality in the chain rule (11) by assuming that f is *regular* (cf. [1], Definition 2.3.4). While the class of regular functions includes convex functions, even simple nonconvex functions like $x \mapsto -|x|$ are not regular. As such, this assumption would heavily restrict the applicability of the method. Fortunately, we do not actually need equality in (11) as we are only interested in the behavior of $\langle \partial f(x_0 + t\tilde{v}), \tilde{v}\rangle$ for $t \to \bar{t}$, and not necessarily in $\langle \partial f(x_0 + \bar{t}\tilde{v}), \tilde{v}\rangle$ itself. Thus, we will see that it suffices to assume that f is *weakly lower semismooth* [9, 10], which means that it is locally Lipschitz and for $x \in \mathbb{R}^n$, $v \in \mathbb{R}^n$ and sequences $(s_i)_i \in \mathbb{R}^{>0}$, $(\xi_i)_i \in \mathbb{R}^n$ with $s_i \searrow 0$ and $\xi_i \in \partial f(x + s_i v)$ for all $i \in \mathbb{N}$, it holds

$$\limsup_{i\to\infty} \langle \xi_i, v\rangle \leq \liminf_{s\searrow 0} \frac{f(x + sv) - f(x)}{s}. \tag{13}$$

Roughly speaking, weak lower semismoothness means that there is a semicontinuous relationship between directional derivatives and sequences of subgradients (via the inner product). In our case, we are interested in inequality (13) for $x = x_0 + \bar{t}\tilde{v}$ and

$v = -\tilde{v}$. This will give us a lower estimate for $\langle \xi', \tilde{v} \rangle$ in step 3 of Algorithm 2, which we can use to show termination.

To this end, we will first derive an upper bound for the right-hand side of (13) in the following lemma.

Lemma 2 *Assume that Algorithm 2 does not terminate. Let $\bar{t}$ as in Lemma 1. Then*

$$\liminf_{s \searrow 0} \frac{f(x_0 + \bar{t}\tilde{v} - s\tilde{v}) - f(x_0 + \bar{t}\tilde{v})}{s} \leq \tilde{c}\|\tilde{v}\|^2. \tag{14}$$

Proof By Lemma 1 we have $t_j < \bar{t}$ for infinitely many $j \in \mathbb{N}$. Let $(j_i)_i$ be the sequence of such j. Note that monotonicity of $(h_{\tilde{c}}(b_j))_j$ and $t_{j_i} < \bar{t}$ imply that

$$h_{\tilde{c}}(\bar{t}) = \lim_{j \to \infty} h_{\tilde{c}}(b_j) \geq h_{\tilde{c}}(b_{j_i}) > h_{\tilde{c}}(t_{j_i}) \quad \forall i \in \mathbb{N}.$$

In particular, writing $f(x_0 + t_{j_i}\tilde{v}) = f(x_0 + \bar{t}\tilde{v} - (\bar{t} - t_{j_i})\tilde{v})$, it holds

$$\begin{aligned} & 0 > h_{\tilde{c}}(t_{j_i}) - h_{\tilde{c}}(\bar{t}) = f(x_0 + \bar{t}\tilde{v} - (\bar{t} - t_{j_i})\tilde{v}) - f(x_0 + \bar{t}\tilde{v}) + \tilde{c}(t_{j_i} - \bar{t})\|\tilde{v}\|^2 \\ \Leftrightarrow \quad & \frac{f(x_0 + \bar{t}\tilde{v} - (\bar{t} - t_{j_i})\tilde{v}) - f(x_0 + \bar{t}\tilde{v})}{\bar{t} - t_{j_i}} < \tilde{c}\|\tilde{v}\|^2 \end{aligned}$$

for all $i \in \mathbb{N}$. Since $\bar{t} - t_{j_i} \searrow 0$ and the limit inferior is taken in (14), this completes the proof. □

The previous lemma enables us to prove our main result.

Theorem 1 *Assume that f is weakly lower semismooth. Then Algorithm 2 terminates.*

Proof Assume that Algorithm 2 does not terminate. Choose $(j_i)_i$ as in the proof of Lemma 2 and let $\xi_i' \in \partial f(x_0 + t_{j_i}\tilde{v})$ be the subgradient evaluated in step 3 in iteration j_i. Let $s_i := \bar{t} - t_{j_i}$. Then $\xi_i' \in \partial f(x_0 + t_{j_i}\tilde{v}) = \partial f(x_0 + \bar{t}\tilde{v} - s_i\tilde{v})$ and $s_i \searrow 0$, so by Lemma 2 and weak lower semismoothness, it holds

$$\begin{aligned} \liminf_{i \to \infty} \langle \xi_i', \tilde{v} \rangle &= -\limsup_{i \to \infty} \langle \xi_i', -\tilde{v} \rangle \geq -\liminf_{s \searrow 0} \frac{f(x_0 + \bar{t}\tilde{v} - s\tilde{v}) - f(x_0 + \bar{t}\tilde{v})}{s} \\ &\geq -\tilde{c}\|\tilde{v}\|^2 > -c\|\tilde{v}\|^2, \end{aligned}$$

since $\tilde{c} \in (c_{\min}, c)$. In particular, we must have

$$\langle \xi_i', \tilde{v} \rangle > -c\|\tilde{v}\|^2$$

after finitely many iterations, causing the algorithm to stop in step 3. □

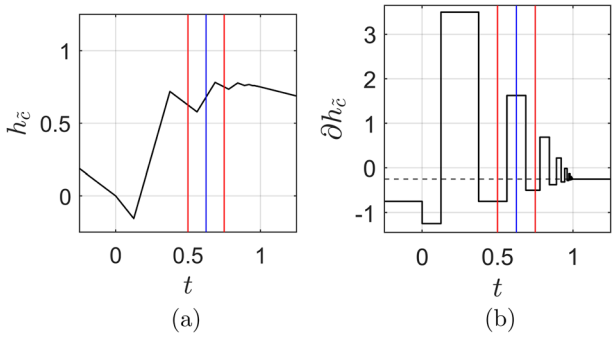

Fig. 2 The graphs of **a** $h_{\tilde{c}}$ and **b** $\partial h_{\tilde{c}}$ for $c = 1/2$, $\tilde{c} = 1/4$ in Example 2. The vertical lines show the sequence $(t_j)_j$ from Algorithm 2, with the final value colored in blue. The dashed horizontal line in **b** marks the value $(\tilde{c} - c)\|\tilde{v}\|^2 = -1/4$ above which $\partial h_{\tilde{c}}(t_j)$ must lie for the method to stop (cf. (12)) (color figure online)

4 Example

In this section, we visualize the difference between Algorithms 1 and 2 by revisiting Example 1.

Example 2 (a) In the situation of Example 1, it holds

$$c_{\min} = -\frac{f(0+1) - f(0)}{1} = -\frac{1}{2},$$

so any $\tilde{c} \in (-1/2, c) = (-1/2, 1/2)$ can be chosen for Algorithm 2. Figure 2 shows the graphs of $h_{\tilde{c}}$ and $\partial h_{\tilde{c}}$ when choosing $\tilde{c} = 1/4$. We see that the algorithm terminates after two iterations with $t_3 = 5/8$ and the new ε-subgradient $\xi' = 11/8 \notin \operatorname{conv}(W) = \{-1\}$.

(b) Note that part (a) essentially solved the problem by simply choosing a different value for c in h (leading to a more well-behaved function $h_{\tilde{c}}$). To better visualize differences of Algorithms 1 and 2, assume that we chose $c = 3/4$, such that we may choose $\tilde{c} = 1/2 \in (c_{\min}, c)$. Then Algorithm 2 has to deal with the same problematic function as Algorithm 1 in Example 1. The resulting graphs of $h_{\tilde{c}}$ and $\partial h_{\tilde{c}}$ are shown in Fig. 3. Since, in Algorithm 2, it is sufficient to have $g > (\tilde{c} - c)\|\tilde{v}\|^2$ for a subgradient $g = \langle \xi', \tilde{v} \rangle + \tilde{c}\|\tilde{v}\|^2 \in \partial h_{\tilde{c}}(t_j)$ (cf. (12)), and since all subgradients at $h_{\tilde{c}}(t)$ tend to 0 as $t \to 1$, the method already stops in $t_3 = 7/8$ with the new ε-subgradient $\xi' = -5/8 \notin \operatorname{conv}(W) = \{-1\}$.

5 Conclusion and future work

In this article, we showed how the gap in the convergence theory of the deterministic gradient sampling methods from [6–8] can be closed for weakly lower semismooth functions by a more careful handling of the sufficient decrease condition in the bisection.

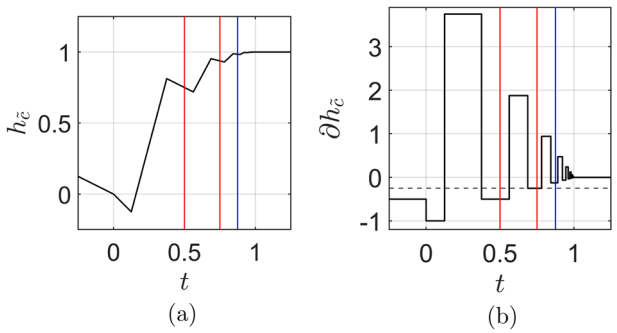

Fig. 3 Same as Fig. 2 but for $\tilde{c} = 1/2$ and $c = 3/4$

For future work, it might be worth to analyze the behavior of Algorithm 2 in a more general setting. In [18], the convergence theory for the original gradient sampling method from [4] was generalized to directionally Lipschitzian functions, and already in [4], gradient sampling was successfully applied to even more general, non-Lipschitzian functions. Furthermore, the general strategy of approximating the ε-subdifferential in a deterministic fashion as in [6–8, 15] may lead to interesting new methods when combined with other methods that rely on random sampling, like [13, 14]. In the long run, we believe that this strategy may lead one step closer to a unified framework for both gradient sampling and bundle methods. For example, when comparing the standard gradient sampling method in [4] and the bundle method of [15], one could "hide" all the null steps in the bundle method in a subroutine and end up with a method that is similar to the gradient sampling method, just with a different way to approximate the ε-subdifferential.

Funding Open Access funding enabled and organized by Projekt DEAL.

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

Declarations

Conflicts of interest The author has no competing interests to declare that are relevant to the content of this article.

Appendix A A drawback of random sampling

Example 3 For $n > 1$ let $\mathrm{pr} : \mathbb{R}^n \to \mathbb{R}^{n-1}, x \mapsto (x_1, \ldots, x_{n-1})^\top$, be the projection onto the first $n-1$ components. Let

$$f : \mathbb{R}^n \to \mathbb{R}, \quad x \mapsto |x_n - \|\mathrm{pr}(x)\|| + \frac{1}{2}x_n.$$

Then f is continuously differentiable in $D^1 \cup D^2$, where

$$D^1 := \{x \in \mathbb{R}^n : \mathrm{pr}(x) \neq 0, x_n < \|\mathrm{pr}(x)\|\},$$
$$D^2 := \{x \in \mathbb{R}^n : \mathrm{pr}(x) \neq 0, x_n > \|\mathrm{pr}(x)\|\}.$$

For $n = 2$, the graph of f and the sets D^1, D^2 are shown in Fig. 4.

For the gradient we have

$$\nabla f(x) = \begin{pmatrix} \mathrm{pr}(x)/\|\mathrm{pr}(x)\| \\ -1/2 \end{pmatrix} \ \forall x \in D^1, \ \nabla f(x) = \begin{pmatrix} -\mathrm{pr}(x)/\|\mathrm{pr}(x)\| \\ 3/2 \end{pmatrix} \ \forall x \in D^2.$$

Thus, it is easy to see that $x = 0$ is a critical point of f. In particular, if $x_0 \in \mathbb{R}^n$ and $\varepsilon > 0$ such that $0 \in B_\varepsilon(x_0)$, then the solution of (3) is $\bar{v} = 0$.

Clearly, to have $v^{\mathrm{GS}} = 0$ in (6), it is necessary to sample at least one gradient from $D^2_\varepsilon := D^2 \cap B_\varepsilon(x_0)$. For $x_0 = 0$ and $\varepsilon = 1$, the probability that $y \in D^2_\varepsilon$ for a uniformly sampled $y \in B_\varepsilon(x_0)$ can be computed by comparing the hypervolume $V_n(D^2_\varepsilon)$ of D^2_ε (which can be computed via a partition of D^2_ε into a hypercone and a hyperspherical cap) to $V_n(B_\varepsilon(x_0))$. In [4], $m = 2n$ gradients are sampled in every iteration for the approximation of $\partial_\varepsilon f(x_0)$. The resulting probabilities of having at least one of the $2n$ sample points in D^2_ε are shown in Table 1.

We see that when increasing the dimension n, it quickly becomes highly unlikely that random gradient sampling correctly identifies $x_0 = 0$ as ε-critical. Note that this is not related to $x_0 = 0$ being a nonsmooth point of f, and we get a similar (or even worse) result when choosing some $x_0 \in D^1$ close to zero. (It would just become more difficult to compute the exact probabilities as in the table for $x_0 \neq 0$.) In this case, the method from [4] would perform descent steps with short step lengths (and little descent), which would require many function evaluations due to the backtracking nature of the line search, without recognizing that the iterates are already ε-critical.

If deterministic sampling is used instead (cf. Sect. 2.2), then a simple computation shows that if the first two subgradients ξ^1 and ξ^2 were sampled from D^1, then the next $\tilde{v}$ from (7) (for $W = \{\xi^1, \xi^2\}$) must be $\tilde{v} = (0, \ldots, 0, 1/2)^\top$, so the next subgradient is sampled at $x_0 + 1/2\,\tilde{v}$ (cf. Algorithm 2). If x_0 is close to zero, then $x_0 + 1/2\,\tilde{v} \in D^2$ such that a subgradient from D^2 is sampled. After at most one additional iteration, the procedure stops and correctly obtains $\tilde{v} = \bar{v} = 0$. Thus, for arbitrary n, at most 4 subgradients have to be sampled when sampling deterministically.

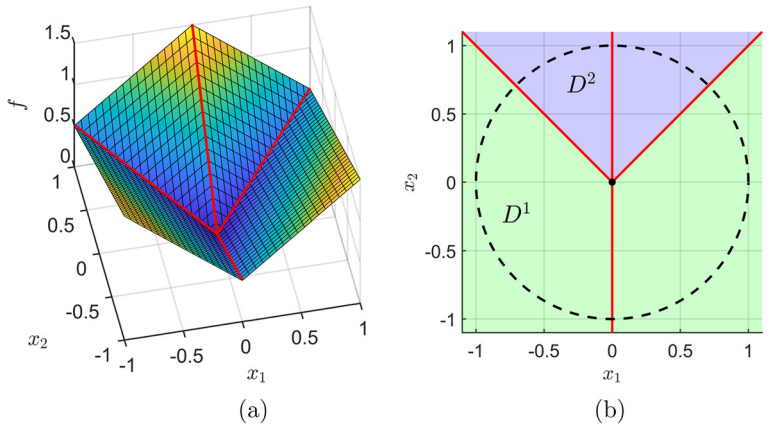

Fig. 4 **a** The graph of f for $n = 2$ in Example 3. The red lines indicate the points in which f is not differentiable. **b** The boundary of the unit sphere $B_1(0)$ (dashed) and the sets D^1 (green) and D^2 (blue) (color figure online)

Table 1 Probability that at least one of the $2n$ sample points lies in D_ε^2 in Example 3

n	2	3	5	10	20	50	100
Prob	0.6836	0.6133	0.4502	0.1394	0.0067	$3.3 \cdot 10^{-7}$	$2.2 \cdot 10^{-14}$

References

1. Clarke, F.H.: Optimization and nonsmooth analysis. Soc. Ind. Appl. Math. (1990). https://doi.org/10.1137/1.9781611971309
2. Goldstein, A.A.: Optimization of lipschitz continuous functions. Math. Progr. **13**(1), 14–22 (1977). https://doi.org/10.1007/bf01584320
3. Burke, J.V., Lewis, A.S., Overton, M.L.: Approximating subdifferentials by random sampling of gradients. Math. Operat. Res. **27**(3), 567–584 (2002). https://doi.org/10.1287/moor.27.3.567.317
4. Burke, J.V., Lewis, A.S., Overton, M.L.: A robust gradient sampling algorithm for nonsmooth, nonconvex optimization. SIAM J. Optimiz. **15**(3), 751–779 (2005). https://doi.org/10.1137/030601296
5. Burke, J.V., Curtis, F.E., Lewis, A.S., Overton, M.L., Simões, L.E.A.: Gradient sampling methods for nonsmooth optimization. In: Numerical Nonsmooth Optimization: State of the Art Algorithms, pp. 201–225. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-34910-3_6
6. Mahdavi-Amiri, N., Yousefpour, R.: An effective nonsmooth optimization algorithm for locally lipschitz functions. J. Optimizat. Theory Appl. **155**(1), 180–195 (2012). https://doi.org/10.1007/s10957-012-0024-7
7. Gebken, B., Peitz, S.: An efficient descent method for locally lipschitz multiobjective optimization problems. J. Optimizat. Theory Appl. **80**, 3–29 (2021). https://doi.org/10.1007/s10957-020-01803-w
8. Gebken, B.: Computation and analysis of Pareto critical sets in smooth and nonsmooth multiobjective optimization. PhD Thesis, Paderborn University (2022) https://doi.org/10.17619/UNIPB/1-1327
9. Mifflin, R.: An algorithm for constrained optimization with semismooth functions. Math. Operat. Res. **2**(2), 191–207 (1977). https://doi.org/10.1287/moor.2.2.191
10. Lewis, A.S., Overton, M.L.: Nonsmooth optimization via quasi-Newton methods. Math. Progr. **141**, 135–163 (2013). https://doi.org/10.1007/s10107-012-0514-2
11. Mifflin, R.: Semismooth and semiconvex functions in constrained optimization. SIAM J. Control Optimiz. **15**(6), 959–972 (1977). https://doi.org/10.1137/0315061
12. Sun, D., Sun, J.: Löwner's operator and spectral functions in euclidean jordan algebras. Math. Operat. Res. **33**(2), 421–445 (2008). https://doi.org/10.1287/moor.1070.0300

13. Curtis, F.E., Que, X.: An adaptive gradient sampling algorithm for non-smooth optimization. Optimiz. Methods Softw. **28**(6), 1302–1324 (2013). https://doi.org/10.1080/10556788.2012.714781
14. Curtis, F.E., Que, X.: A quasi-Newton algorithm for nonconvex, nonsmooth optimization with global convergence guarantees. Math. Progr. Comput. **7**(4), 399–428 (2015). https://doi.org/10.1007/s12532-015-0086-2
15. Kiwiel, K.C.: Improved convergence result for the discrete gradient and secant methods for nonsmooth optimization. J. Optimiz. Theory Appl **144**(1), 69–75 (2009). https://doi.org/10.1007/s10957-009-9584-6
16. Lemaréchal, C.: Chapter VII. Nondifferentiable optimization. In: Handbooks in Operations Research and Management Science, pp. 529–572. Elsevier, Amsterdam (1989). https://doi.org/10.1016/s0927-0507(89)01008-x
17. Cheney, W., Goldstein, A.A.: Proximity maps for convex sets. Proc. Am. Math. Soc. **10**(3), 448–448 (1959). https://doi.org/10.1090/s0002-9939-1959-0105008-8
18. Burke, J.V., Lin, Q.: Convergence of the gradient sampling algorithm on directionally lipschitz functions. Set-Valued Var. Anal. **29**(4), 949–966 (2021). https://doi.org/10.1007/s11228-021-00610-3