Inexact Direct-Search Methods for Bilevel Optimization Problems

Youssef Diouane* Vyacheslav Kungurtsev† Francesco Rinaldi‡
Damiano Zeffiro§

September 13, 2023

Abstract

In this work, we introduce new direct-search schemes for the solution of bilevel optimization (BO) problems. Our methods rely on a fixed accuracy blackbox oracle for the lower-level problem, and deal both with smooth and potentially nonsmooth true objectives. We thus analyze for the first time in the literature direct-search schemes in these settings, giving convergence guarantees to approximate stationary points, as well as complexity bounds in the smooth case. We also propose the first adaptation of mesh adaptive direct-search schemes for BO. Some preliminary numerical results on a standard set of bilevel optimization problems show the effectiveness of our new approaches.

1 Introduction

Bilevel optimization (see, e.g., [6, 9, 12, 13, 24] and references therein for a complete overview on the topic) has been subject of increasing interest, thanks to its application to hyperparameter tuning for machine learning algorithms and meta-learning (see, e.g., [17] and references therein). In this work, we are interested in the following bilevel optimization problem

$$\min_{(x,y)\in\mathbb{R}^{n_x}\times n_y} \quad f(x,y), \qquad \text{s.t.} \qquad y \in \arg\min_{z\in Z} g(x,z). \tag{1}$$

wherein we assume that the upper-level function $f(x,y) : \mathbb{R}^{n_x\times n_y} \to \mathbb{R}$ is continuous, and $g(x,z) : \mathbb{R}^{n_x\times n_y} \to \mathbb{R}$ is such that the lower-level problem $\min_{z\in Z} g(x,z)$ has a unique solution $y(x)$ for every $x \in \mathbb{R}^{n_x}$, and $Z \subset \mathbb{R}^{n_y}$. Uniqueness of the lower-level problem solution, also known as the Low-Level Singleton (LLS) assumption, is a quite common assumption in many real world applications, such as hyperparameter optimization, meta-learning, pruning, semi-supervised learning on multilayer graphs (see, e.g., [17, 20, 42, 45]). While for simplicity we focus on the setting described above, it is important to point out that our analysis still holds, for a specific class of BO problems, even when dropping the LLS assumption (see Remark 2.1).

The algorithms we study here are derivative free optimization (DFO) methods, which do not use derivatives of the upper-level objective function, but rather only the objective value itself.

*Department of Mathematics and Industrial Engineering, Polytechnique Montréal, Montréal, QC, Canada.(youssef.diouane@polymtl.ca)

†Department of Computer Science, Czech Technical University, Czech Republic. (kunguvya@fel.cvut.cz)

‡Dipartimento di Matematica "Tullio Levi-Civita", Università di Padova, Italy. (rinaldi@math.unipd.it)

§Dipartimento di Matematica "Tullio Levi-Civita", Università di Padova, Italy. (zeffiro@math.unipd.it)

Importantly, in this setting we also assume the availability of some blackbox oracle generating an approximation $\tilde{y}(x)$ of $y(x)$ for any given $x \in \mathbb{R}^n_x$. Among DFO methods, we are interested in particular in direct-search methods (see, e.g., [2, 26]), which sample the objective in suitably chosen tentative points without building a model for it. These algorithmic schemes allow us to prove convergence guarantees under very mild assumptions on our bilevel optimization problem.

1.1 Previous Work

Several gradient-based methods have been proposed in the literature to tackle bilevel optimization problems. Those methods usually require the computation of the true objective gradient, called "hypergradient", and rely on the LLS and suitable smoothness assumptions (see, e.g., [17, 18, 23, 27, 29] and references therein). In another line of research, some asymptotic results based on relaxations of the LLS assumption were also analyzed (see, e.g., [30, 31, 32] and references therein). Calculating the hypergradients can be however a notoriously challenging and time consuming task. It indeed requires the handling of $\nabla_x y(x)$, which in turns involves the calculation of the Hessian matrix related to the g function via the implicit differentiation theorem. In some contexts, the hypergradients might not be available at all due to the blackbox nature of the functions describing the problem. These are the reasons why the development of new and efficient zeroth-order/derivative-free approaches is crucial in the BO context.

As for derivative free approaches, classic direct-search (see, e.g., [2, 10, 26]) and trust-region methods (see, e.g.,[10, 26]) have been applied to BO in [11, 15, 37, 44]. In [37], a direct-search method for BO assuming the availability of the true objective is described. More specifically, their analysis does not allow for approximation errors in the solution of the lower-level problem, and relies on suitable assumptions making the true objective directionally differentiable. In [44], the analysis from [37] is extended considering lower-level inexact solutions with a stepsize-based adaptive error. In [11], an algorithm applying trust-region methods both in the inner level and on the true objective is described, with an adaptive estimation error for the true objective depending on the trust-region radius; in that work, a strategy to recycle function evaluations for the lower-level problem is described as well. In [15], the analysis of another trust-region method with adaptive error for bilevel optimization is carried out. The authors report worst-case complexity estimates both in terms of upper-level iterations and computational work from the lower-level problem, when considering a strongly convex lower-level problem solved by a suitable gradient descent approach. In the more recent works [8, 35], zeroth-order methods based on smoothing strategies [39] are analyzed. These studies, drawing inspiration from the complexity results provided in [21] for zeroth-order methods that handle nonsmooth and non-convex objectives, offer complexity estimates tailored for the BO setting. They rely on the assumptions that the lower-level problem can be solved with fixed precision, and that gradient descent on the lower level converges either polinomially or exponentially, respectively.

Finally, min-max DFO problems (which can be seen as a particular instance of BO) are also recently tackled in the literature [1, 36]. Relevant to our work are also direct-search methods under the presence of noise. While previous works analyze direct-search methods with adaptive deterministic [34] and stochastic noise [1, 4, 41], we are not aware of previous analyses of direct-search methods with bounded but non adaptive noise.

1.2 Contributions

Our contributions can be summarized as follows.

- We define and analyze the first inexact direct-search schemes for BO problems with general potentially nonsmooth true objectives. Those methods indeed never require exact lower-level problem solutions, but instead assume access to approximate solutions with fixed accuracy, a reasonable assumption in practice. We therefore operate in a different setting than the one considered in previous works on direct-search for BO, where true objectives are directionally differentiable [37, 44] and lower-level solutions are exact [37] or require an adaptive precision [44].
- We analyze mesh based direct-search schemes for BO, extending in particular the classic mesh adaptive direct-search (MADS) scheme from [3]. This is, to the best of our knowledge, the first analysis of this scheme that considers both inexact objective evaluation and the simple decrease condition for new iterates used originally in [3].
- We give the first convergence results for direct-search schemes with bounded and non-adaptive noise on the objective.
- We give the first convergence guarantees to (δ, ϵ)-Goldstein stationary points for direct-search schemes applied to general nonsmooth objectives. With respect to classic analyses considering Clarke stationary points (see, e.g., [5]), these are the first results for direct-search scheme involving some quantitative measure of approximate nonsmooth stationarity.

2 Background and Preliminaries

We now introduce the main assumptions considered in the paper, along with a set of helpful preliminary results that will support the subsequent convergence theory. As anticipated in the introduction, we will always assume the existence of a unique minimizer $y(x)$ for the lower-level problem, i.e., that the LLS assumption holds.

Assumption 2.1 *For any $x \in \mathbb{R}^{n_x}$, we have that* $\operatorname{argmin}_{z \in Z} g(x, z) = \{y(x)\}$.

Under Assumption 2.1, the bilevel optimization problem (1) can then be rewritten as

$$\min_{x \in \mathbb{R}^{n_x}} \quad F(x) := f\left(x, y(x)\right). \tag{2}$$

However, in practical applications, it is usually necessary to employ an iterative method to compute $y(x)$. Therefore, one cannot expect to obtain an exact value of $y(x)$, but rather some approximation. We will hance make use of the following assumption.

Assumption 2.2 *For all $x \in \mathbb{R}^{n_x}$ we can compute an approximation $\tilde{y}(x)$ of $y(x)$ such that:*

$$\|\tilde{y}(x) - y(x)\| \leq \varepsilon. \tag{3}$$

While the remaining assumptions introduced in this section are not always needed, in the rest of this manuscript we always assume that Assumptions 2.1 and 2.2 hold.

Remark 2.1 *Our analysis extends to the case where* $\operatorname{argmin}_{z \in Z} g(x, z)$ *is not a singleton, but an approximate solution $\tilde{y}(x)$ of the simple bilevel problem*

$$\min_{y \in \mathbb{R}^{n_y}} \quad f(x, y), \qquad s.t. \qquad y \in \arg\min_{z \in Z} g(x, z). \tag{4}$$

is available for every $x \in \mathbb{R}^{n_x}$. In fact our convergence proofs rely on (3) *rather than the singleton assumption. We refer the reader to the recent work [8] for a detailed discussion on the complexity and regularity properties of the simple bilevel problem* (4).

In the next proposition, we show how condition (3) can be satisfied, by applying gradient descent to $g(x, \cdot)$, under a suitable error bound condition on $\nabla_y g(x, y)$ generalizing strong convexity (see, e.g, [22] for a detailed comparison with other conditions). We also give an explicit bound on the number of iterations needed to satisfy (3).

Proposition 2.1 *Assume that there exists $c_g > 0$ such that for all $y \in Z$,*

$$c_g \|y - y(x)\| \leq \|\nabla_y g(x, y)\| \,. \tag{5}$$

Furthermore, let $\nabla_y g$ be L_g Lipschitz continuous in y, uniformly in x. Define $y_0(x)$ to be any arbitrary initialization mapping onto the domain of $g(x, \cdot)$. Then consider the sequence,

$$y_{k+1}(x) = y_k(x) - \frac{1}{L_g} \nabla_y g(x, y_k(x)) \,. \tag{6}$$

Define the solution estimate to be:

$$\tilde{y}(x) = \text{argmin}_{k \in [0:K(x)]} \|\nabla_y g(x, y_k(x))\| \tag{7}$$

It holds that $\tilde{y}(x)$ satisfies (3)*, for*

$$K(x) = \left\lceil \frac{2L_g(g(x, y_0(x)) - g(x, y(x)))}{\varepsilon^2 c_g^2} \right\rceil \,. \tag{8}$$

Proof. This follows from the well known iteration complexity of gradient descent for smooth non convex objectives. ■

We introduce now some technical assumptions on the objective function needed in our analysis.

Assumption 2.3 *The function f is lower bounded by f_{low}.*

Assumption 2.4 *The function f is Lipschitz continuous with respect to y with Lipschitz constant L_f (independent of x).*

We remark that these assumptions are an adaptation to our bilevel setting of standard assumptions made in the analysis of direct-search methods [10, 34]. Assumption 2.2 together with Assumption 2.4 imply that $\tilde{F}(x) := f(x, \tilde{y}(x))$ is an approximation of $F(x)$ with accuracy $L_f \varepsilon$. Indeed,

$$|\tilde{F}(x) - F(x)| = |f(x, \tilde{y}(x)) - f(x, y(x))| \leq L_f \|\tilde{y}(x) - y(x)\| \leq L_f \varepsilon \,. \tag{9}$$

Some regularity on the true objective $F(x)$ will always be necessary for our analyses. We consider both the differentiable and the potentially non differentiable setting.

Assumption 2.5 *$F(x)$ is Lipschitz continuous with constant L_F.*

Assumption 2.6 *The function F is continuously differentiable with Lipschitz continuous gradient, of Lipschitz constant L.*

Note that if f is Lipschitz with respect to x, and $y(x)$ Lipschitz continuous with respect to x, then Assumption 2.5 is satisfied. Furthermore, in the strongly convex lower-level setting there is an explicit expression for ∇F (see, e.g., [8, Equation (3)]), implying that its Lipschitz continuity follows from that of $y(x)$ together with suitable regularity assumptions on f and g.

2.1 Algorithm

In this section, we introduce a general direct-search algorithm for bilevel optimization that embeds both directional direct-search methods with *sufficient decrease* and mesh adaptive direct-search methods with *simple decrease*, as defined in [10]. The methods in the first class sample tentative points along a suitable set of descent directions and then select as the new iterate a point satisfying a sufficient decrease condition. The methods in the second class sample the points in a suitably defined mesh, and then select the new iterate according to a simple decrease condition. A tentative point t is hence accepted if the decrease condition

$$f(t, \tilde{y}(t)) < f(x_k, y_k) - \rho(\alpha_k) \tag{10}$$

is satisfied, for ρ nonnegative function. We have a sufficient decrease when $\rho(t) > 0$ with $\lim_{t\to 0^+} \rho(t)/t = 0$, and a simple decrease in case $\rho(t) = 0$. These two classes of decrease conditions lead to significant differences in convergence properties and consequently require different choices in the algorithm parameters. They will therefore be analyzed separately in Sections 3 and 4 respectively.

Algorithm 1: DS for bilevel optimization

1: **Initialization:** Choose $x_0 \in \mathbb{R}^{n_x}$, α_0 initial stepsize, $\rho : \mathbb{R}_{>0} \to \mathbb{R}_{\geq 0}$. Let $y_0 = \tilde{y}(x_0)$ be an approximate minimizer for the lower-level problem in x_0. **Optional:** Let $\Delta_0 = \alpha_0$ be the initial frame size parameter.
2: **for** $k = 0, 1, 2, \ldots$ **do**
3: Let $M_k \subset \mathbb{R}^{n_x}$ be a mesh depending on α_k and x_k. Let S_k be a finite subset of M_k.
4: **if** $f(t, \tilde{y}(t)) < f(x_k, y_k) - \rho(\alpha_k)$ for some $t \in S_k$ **then**
5: Set $x_{k+1} = t$, declare the iteration successful, and go to step 13.
6: **end if**
7: Choose a set of descent directions D_k, possibly depending on Δ_k and such that $\{x_k + \alpha_k d \mid d \in D_k\} \subset M_k$. For a given $d \in D_k$, compute the approximate minimizer $y_k^{\alpha_k d} = \tilde{y}(x_k + \alpha_k d_k)$ for the lower problem. Evaluate f at the poll points belonging to $\{(x_k + \alpha_k d, y_k^{\alpha_k d}) : d \in D_k\}$.
8: **if** there exists $d_k \in D_k$ such that $f(x_k + \alpha_k d_k, y_k^{\alpha_k d_k}) < f(x_k, y_k) - \rho(\alpha_k)$ **then**
9: Declare the iteration as successful. Set $x_{k+1} = x_k + \alpha_k d_k$ and $y_{k+1} = y_k^{\alpha_k d_k}$.
10: **else**
11: Declare the iteration as unsuccessful. Set $x_{k+1} = x_k$ and $y_{k+1} = y_k$.
12: **end if**
13: Update the frame size parameter Δ_k and the stepsize α_k.
14: **Optional:** If some approximate stationarity condition is satisfied, terminate the algorithm.
15: **end for**

The detailed scheme (see Algorithm 1) follows the lines of the general schemes proposed in [10] and [26], with the addition of calls to the lower-level oracle $\tilde{y}(x)$, and an explicit reference to the mesh used in mesh-based schemes. At Step 1, the algorithm searches for a new iterate

by testing the upper level objective in $(t, \tilde{y}(t))$ for t in S_k subset of the mesh M_k. In case Step 1 is not successful, the method generates, at Step 2, a new iterate by selecting a set of descent directions D_k and testing the upper level objective in $(t, \tilde{y}(t))$ for t chosen along the descent directions using a stepsize α_k. Step 3 and Step 4 perform updates on the algorithm iterate and parameters based on the outcome of Step 1 and 2. For the set of directions D_k, we require in some cases a positive cosine measure, that is

$$\mathrm{cm}(D_k) \stackrel{d}{=} \min_{v \neq 0_{\mathbb{R}^{n_x}}} \max_{d \in D_k} \frac{d^\top v}{\|d\|\|v\|} \geq \kappa\,, \tag{11}$$

for some $\kappa > 0$.

3 Sufficient decrease condition

In this section, we analyze directional direct-search methods using a sufficient decrease condition with $\rho(t) = \frac{c}{2}t^2$. We first focus on potentially nonsmooth objectives, and then on smooth ones. In both cases we consider the scheme presented in Algorithm 2, which can be viewed as an adaption to BO of classic generating set of search directions (GSS) schemes (see, e.g., [25, Algorithm 3.2]). In order to handle the error introduced by the approximate solution in the lower level, we lower bound the stepsize with a constant $\alpha_{\min}$. We further notice that, thanks to the sufficient decrease condition, maintaining a mesh is not necessary, and therefore we simply set $M_k = \mathbb{R}^{n_x}$.

3.1 Nonsmooth objectives

First, we present convergence guarantees and proofs thereof for a variant of Algorithm 2 designed for the case of Lipschitz continuous true objectives, i.e., under Assumption 2.5. With respect to the general scheme presented as Algorithm 2, here $D_k = \{g_k\}$ with g_k generated in the unit sphere. We remark that this is a standard choice for direct-search algorithms applied to nonsmooth objectives (see, e.g., [16, Algorithm DFN$_{simple}$]). The stepsize lower bound here must be strictly positive (i.e. $\alpha_{\min} > 0$).This together with the sufficient decrease conditions ensures that the sequence generated by the algorithm is eventually constant, as proved in Lemma 3.1. We then use a novel argument to prove that the limit point of the sequence is a (δ, ϵ)-Goldstein stationary point. Although such a notion of stationarity has recently gained attention in the analysis of zeroth-order smoothing-based approaches [21, 28, 40], including extensions to BO [8, 35], to the best of our knowledge, it has never been used for the analysis of direct-search methods.It is further important to notice that convergence of directional direct-search methods to $(\delta, \epsilon)-$Goldstein stationary points in the nonsmooth case is a novel result also for classic optimization problems. We now recall some useful definitions. If $B_\delta(x)$ is the ball of radius δ centered in x, then the δ-Goldstein subdifferential (see, e.g., [28]) is defined as

$$\partial_\delta F(x) = \mathrm{conv}\left\{ \bigcup_{y \in B_\delta(x)} \partial F(y) \right\}, \tag{12}$$

and x is an (δ, ϵ)-Goldstein stationary point for the function F if, for some $g \in \partial_\delta F(x)$, we have $\|g\| \leq \epsilon$.

Algorithm 2: Inexact directional DS for bilevel optimization

1: **Initialization:** Choose starting point $x_0 \in \mathbb{R}^{n_x}$, stepsize lower bound $\alpha_{\min} \geq 0$, initial stepsize $\alpha_0 \geq \alpha_{\min}$, coefficient for stepsize contraction $0 < \theta < 1$, coefficient for stepsize expansion $\gamma \geq 1$, sufficient decrease condition coefficient c. Let $y_0 = \tilde{y}(x_0)$ be an approximate minimizer for the lower-level problem at x_0.
2: **for** $k = 0, 1, 2, \ldots$ **do**
3: Let $S_k \subset \mathbb{R}^{n_x}$ with $|S_k| < +\infty$.
4: **if** $f(t, \tilde{y}(t)) < f(x_k, y_k) - \frac{c}{2}\alpha_k^2$ for some $t \in S_k$ **then**
5: Set $x_{k+1} = t$, declare the iteration successful, and go to step 13.
6: **end if**
7: Choose a set of descent directions D_k. For a given $d \in D_k$, compute the approximate minimizer $y_k^{\alpha_k d} = \tilde{y}(x_k + \alpha_k d_k)$ for the lower problem. Evaluate f at the poll points belonging to $\{(x_k + \alpha_k d, y_k^{\alpha_k d}) : d \in D_k\}$.
8: **if** for some $d_k \in D_k$, $f(x_k + \alpha_k d_k, y_k^{\alpha_k d_k}) < f(x_k, y_k) - \frac{c}{2}\alpha_k^2$ **then**
9: Declare the iteration as successful. Set $x_{k+1} = x_k + \alpha_k d_k$ for d_k satisfying the condition and $y_{k+1} = y_k^{\alpha_k d_k}$.
10: **else**
11: Declare the iteration as unsuccessful. Set $x_{k+1} = x_k$ and $y_{k+1} = y_k$.
12: **end if**
13: **If** the iteration was successful **then** maintain or increase the corresponding stepsize parameter – set $\alpha_{k+1} \in [\alpha_k, \gamma\alpha_k]$. **Else** decrease the stepsize parameter, by choosing $\alpha_{k+1} = \max\{\alpha_{\min}, \theta\alpha_k\}$.
14: **[Optional]** If some approximate stationarity condition is satisfied, terminate the algorithm.
15: **end for**

We can now proceed with our convergence analysis. As anticipated, we start by proving that the sequence of iterates generated by our method is eventually constant.

Lemma 3.1 *Let Assumptions 2.3 and 2.4 hold. Then there exists $\bar{k} \in \mathbb{N}_0$ such that the sequence $\{x_k\}$ generated by Algorithm 2 is constant for $k \geq \bar{k}$.*

Proof. Notice that $\{\tilde{F}(x_k)\}$ is non-increasing, with $\tilde{F}(x_k) = \tilde{F}(x_{k+1})$ after an unsuccessful step, and

$$\tilde{F}(x_{k+1}) < \tilde{F}(x_k) - \frac{c}{2}\alpha_k^2 \leq \tilde{F}(x_k) - \frac{c}{2}\alpha_{\min}^2 \tag{13}$$

after a successful step. Thus there can be at most

$$\frac{2\left(\tilde{F}(x_0) - \inf_{x\in\mathbb{R}^n} \tilde{F}(x)\right)}{c\alpha_{\min}^2} \leq \frac{2\left(\tilde{F}(x_0) - f_{\text{low}} + L_f\varepsilon\right)}{c\alpha_{\min}^2} \tag{14}$$

successful steps, where we used $\tilde{F}(x) \geq F(x) - L_f\varepsilon \geq f_{\text{low}} - L_f\varepsilon$ in the inequality. Since this quantity is finite, this implies that $\{x_k\}$ is eventually constant. ■

We now prove convergence of our algorithm to (δ, ϵ)-Goldstein stationary points. In order to get our convergence result, we need to assume that the sequence $\{g_k\}$ is dense in the unit sphere. We remark that such a dense sequence can be generated using a suitable quasirandom sequence (see, e.g., [19, 33]).

Theorem 3.1 *Let Assumptions 2.3, 2.4 and 2.5 hold. Assume that $\{g_k\}$ is dense in the unite sphere. Then the sequence $\{x_k\}$ generated by Algorithm 2 is eventually constant, with the unique limit point (δ, ϵ)-Goldstein stationary, for*

$$\epsilon = \frac{4L_f\varepsilon}{\alpha_{\min}} + c\alpha_{\min} \quad \textit{and} \quad \delta = \alpha_{\min}\,. \tag{15}$$

Proof. First, $\{x_k\}$ is eventually constant as seen in Lemma 3.1. Let $\bar{x}$ be the unique limit point. By the stepsize updating rule, we have that every iteration must be unsuccessful with $\alpha_k = \alpha_{\min}$ for k large enough. Then, there exists $\bar{k} \in \mathbb{N}$ large enough such that for every $k \geq \bar{k}$

$$\tilde{F}(\bar{x}) < \tilde{F}(\bar{x} + \alpha_k g_k) + \frac{c}{2}\alpha_{\min}^2 = \tilde{F}(\bar{x} + \alpha_{\min} g_k) + \frac{c}{2}\alpha_{\min}^2 \tag{16}$$

implying

$$F(\bar{x}) < F(\bar{x} + \alpha_{\min} g_k) + \frac{c}{2}\alpha_{\min}^2 + 2L_f\varepsilon\,. \tag{17}$$

By the density of $\{g_k\}$ it follows

$$F(\bar{x}) < F(\bar{x} + d) + \frac{c}{2}\alpha_{\min}^2 + 2L_f\varepsilon \tag{18}$$

for every d such that $\|d\| = \alpha_{\min}$.
We now define the function $\bar{F}_{\bar{x}}(d) := F(\bar{x} + d) + (\frac{c}{2} + \frac{2L_f\varepsilon}{\alpha_{\min}^2})\|d\|^2$. Since

$$\bar{F}_{\bar{x}}(0) < \bar{F}_{\bar{x}}(d) \tag{19}$$

for every d such that $\|d\| = \alpha_{\min}$ by (18), there must be a $\tilde{d} \in \operatorname{argmin}_{\|d\| \leq \alpha_{\min}} \bar{F}_{\bar{x}}(d)$ with $\|\tilde{d}\| < \alpha_{\min}$. We can conclude

$$0 \in \partial \bar{F}_{\bar{x}}(\tilde{d}) = \partial F(x + \tilde{d}) - \left(c + \frac{4L_f\varepsilon}{\alpha_{\min}^2}\right)\tilde{d} \tag{20}$$

Equivalently, $g = (c + \frac{4L_f\varepsilon}{\alpha_{\min}^2})\tilde{d} \in \partial F(x + \tilde{d})$ and since $\partial F(x + \tilde{d}) \subset \partial_{\alpha_{\min}} F(\bar{x})$ we have $g \in \partial_{\alpha_{\min}} F(\bar{x})$. To conclude, observe $\|g\| < c\alpha_{\min} + \frac{4L_f\varepsilon}{\alpha_{\min}}$. ■

As a corollary of Theorem 3.1, for $\alpha_{\min} \propto \sqrt{\varepsilon}$ we are able to get a $(\mathcal{O}(\sqrt{\varepsilon}), \mathcal{O}(\sqrt{\varepsilon}))-$Goldstein stationary point. Interestingly, the order of magnitude $\mathcal{O}(\sqrt{\varepsilon})$ of the approximation error coincides with that of typical gradient approximation methods [7], as well as with that of direct-search in the smooth setting, as we shall see in the next section.

Corollary 3.1 *Let Assumptions 2.3, 2.4 and 2.5 hold. Assume that $\{g_k\}$ is dense in the unite sphere. Then the sequence $\{x_k\}$ generated by Algorithm 2 with $\alpha_{\min} = 2\sqrt{\frac{L_f\varepsilon}{c}}$ is eventually constant, with the unique limit point (δ, ϵ)-Goldstein stationary, for*

$$\epsilon = 4\sqrt{L_f\varepsilon c} \quad \textit{and} \quad \delta = 2\sqrt{\frac{L_f\varepsilon}{c}}\,. \tag{21}$$

3.2 Smooth objectives

We now focus on the case where the objective F is smooth, in particular under Assumption 2.6. We consider here a variant of Algorithm 2 with D_k positive spanning set. When the stepsize lower bound is strictly positive we set as termination criterion $\alpha_k = \alpha_{k+1} = \alpha_{\min}$. Our scheme can hence be seen as a variant of classic direct-search methods for smooth objectives [10, 25]. It is important to highlight that this is the first analysis of direct-search methods for smooth objectives under bounded noise. The only analysis of direct-search methods we are aware of in the smooth case is the one given in [14] under stochastic noise, where, however, the author only focuses on classic optimization problems.

We first extend to our bounded error setting a standard result that allows to get an upper bound on the gradient norm for unsuccessful iterations (see, e.g., [25, Theorem 3.3]).

Lemma 3.2 *Let Assumptions 2.4 and 2.6 hold, together with* (11)*. Let $\{x_k\}$ be a sequence generated by Algorithm 2. If the iteration k is unsuccessful, then*

$$\|\nabla F(x_k)\| \leq \frac{1}{\kappa}\left(\frac{(L+c)\alpha_k}{2} + \frac{2L_f\varepsilon}{\alpha_k}\right). \tag{22}$$

Proof. Let $d \in D_k$ be such that

$$-\nabla F(x_k)^\top d \geq \kappa \|\nabla F(x_k)\|\|d\|\,. \tag{23}$$

We have

$$\begin{aligned} &\kappa\alpha_k\|\nabla F(x_k)\|\|d\| - \alpha_k^2\frac{L}{2}\|d\|^2 \leq -\alpha_k\nabla F(x_k)^\top d - \alpha_k^2\frac{L}{2}\|d\|^2 \\ &\leq F(x_k) - F(x_k + \alpha_k d) \leq \tilde{F}(x_k) - \tilde{F}(x_k + \alpha_k d) + 2L_f\varepsilon \leq \frac{c}{2}\alpha_k^2 + 2L_f\varepsilon\,, \end{aligned} \tag{24}$$

where we used (23) in the first inequality, the standard descent lemma in the second inequality, (9) in the third inequality, and that the step is unsuccessful in the last inequality. Therefore, since by assumption $\|d\| = 1$

$$\kappa\alpha_k\|\nabla F(x_k)\| = \kappa\alpha_k\|\nabla F(x_k)\|\|d\| \leq \frac{c}{2}\alpha_k^2 + 2L_f\varepsilon\alpha_k^2\frac{L}{2}\|d\|^2 = \frac{c}{2}\alpha_k^2 + 2L_f\varepsilon + \alpha_k^2\frac{L}{2}\,, \tag{25}$$

implying the thesis. ■

We now prove convergence and complexity bounds when $\alpha_{\min} > 0$, extending those given in [43] for the exact oracle case, and $\alpha_{\min} = 0$. We notice that in this second case we lose finite convergence and our guarantees are thus somewhat weaker, i.e., we are only able to prove that the stepsize converges to 0 and that at some point the gradient norm is $\mathcal{O}(\sqrt{\varepsilon})$.

Theorem 3.2 *Let Assumptions 2.3, 2.4 and 2.6 hold, together with* (11) *for every $k \in \mathbb{N}_0$. Let $\{x_k\}$ be a sequence generated by Algorithm 2.*

1. *If $\alpha_{\min} > 0$, then the algorithm terminates after $\bar{k}$ iterations, with*

$$\bar{k} < 1 + \frac{2}{\alpha_{\min}^2 c}(\tilde{F}(x_0) - f_{low} + 2L_f\varepsilon)\left(1 - \frac{\ln\gamma}{\ln\theta}\right) + \frac{\ln\alpha_{\min} - \ln\alpha_0}{\ln\theta}\,, \tag{26}$$

and its last iterate $x_{\bar{k}}$ is such that

$$\|\nabla F(x_{\bar{k}})\| \leq \frac{1}{\kappa}\left(\frac{(L+c)\alpha_{\min}}{2} + \frac{2L_f\varepsilon}{\alpha_{\min}}\right)\,. \tag{27}$$

2. *If, furthermore, it holds that $\alpha_{\min} = 2\sqrt{\frac{L_f\varepsilon}{L+c}}$, then*

$$\|\nabla F(x_{\bar{k}})\| \leq \frac{2}{\kappa}\sqrt{(c+L)L_f\varepsilon}\,. \tag{28}$$

3. *If $\alpha_{\min} = 0$, then $\alpha_k \to 0$, and if additionally $\alpha_0 \geq \bar{\alpha}_{\min} = 2\sqrt{\frac{L_f\varepsilon}{L+c}}$, for some $\bar{k} \in \mathbb{N}_0$ we have*

$$\|\nabla F(x_{\bar{k}})\| \leq \frac{1}{\theta\kappa}\left(\frac{(L+c)\bar{\alpha}_{\min}}{2} + \frac{2L_f\varepsilon}{\bar{\alpha}_{\min}}\right)\,, \tag{29}$$

and

$$F(x_k) \leq F(x_{\bar{k}}) + 2L_f\varepsilon \quad \text{for all } k \geq \bar{k}\,. \tag{30}$$

Proof. 1. Let k_s and k_{ns} be the number of successful and unsuccessful steps, so that $k_s + k_{ns} = k$. Reasoning as in Lemma 3.1, we obtain by (14)

$$k_s < \frac{2}{\alpha_{\min}^2 c}(F(x_0) - f_{\text{low}} + 2L_f\varepsilon)\,. \tag{31}$$

Furthermore, since

$$\alpha_{\min} \leq \alpha_k \leq \alpha_0\gamma^{k_s}\theta^{k_{ns}-1}\,, \tag{32}$$

we get

$$\begin{aligned} k_{ns} &\leq 1 - \frac{1}{\ln(\theta)}(\ln(\alpha_0) - \ln(\alpha_{\min}) + k_s \ln(\gamma)) \\ &\leq 1 - \frac{1}{\ln(\theta)}(\ln(\alpha_0) - \ln(\alpha_{\min}) + \frac{2}{\alpha_{\min}^2 c}(\tilde{F}(x_0) - f_{\text{low}} + 2L_f\varepsilon)\ln(\gamma))\,, \end{aligned} \tag{33}$$

where we applied (31) in the second inequality. Combining the bounds on the successful and unsuccessful steps (31) and (33), we have

$$k = k_s + k_{ns} < 1 + \frac{2}{\alpha_{\min}^2 c}(\tilde{F}(x_0) - f_{low} + 2L_f\varepsilon)\left(1 - \frac{\ln\gamma}{\ln\theta}\right) + \frac{\ln\alpha_{\min} - \ln\alpha_0}{\ln\theta}\,, \tag{34}$$

as desired.
2. Follows from a direct application of the first result.
3. Reasoning as in the first result, the number of successful steps with stepsize above a certain threshold is bounded, hence $\alpha_k \to 0$. Furthermore, for any $\bar{k} \in \mathbb{N}_0$, if $k \geq \bar{k}$

$$F(x_k) \leq \tilde{F}(x_k) + L_f\varepsilon \leq \tilde{F}(x_{\bar{k}}) + L_f\varepsilon \leq F(x_{\bar{k}}) + 2L_f\varepsilon\,, \tag{35}$$

which proves (30). Let $\bar{\alpha}_{\min} = 2\sqrt{\frac{L_f\varepsilon}{L+c}}$. Since $\alpha_0 \geq \bar{\alpha}_{\min}$, and $\alpha_k \to 0$ with contraction factor θ, we must have $\alpha_{\bar{k}} \in [\theta\bar{\alpha}_{\min}, \bar{\alpha}_{\min}]$ for some $\bar{k} \in \mathbb{N}_0$. Then (29) follows from (22) for $\alpha_k = \alpha_{\bar{k}}$. ■

We now extend to our setting the $\mathcal{O}(n^2/\epsilon^2)$ complexity result given in [43, Corollary 2]. For a fixed precision ϵ, an approximation error $\varepsilon = \mathcal{O}(\epsilon^2)$ is required, as for classic gradient approximation schemes [7].

Corollary 3.2 *Let Assumptions 2.3, 2.4 and 2.6 hold, together with* (11) *for every* $k \in \mathbb{N}_0$. *Let* $\{x_k\}$ *be a sequence generated by Algorithm 2. Assume also* $\varepsilon \leq \epsilon^2\kappa^2$, *that at every iterations there are at most* $d_1 n$ *function evaluations and that* $\kappa \geq d_2/\sqrt{n}$, *for* $d_1, d_2 > 0$. *Then if* $\alpha_{\min} = 2\sqrt{\frac{L_f\varepsilon}{L+c}}$, *the algorithm terminates after* $\mathcal{O}(n^2/\epsilon^2)$ *function evaluations with* $\|\nabla f(x_{\bar{k}})\| \leq d_3\epsilon$, *for* $d_3 > 0$ *depending only on* c, L *and* L_f.

Proof. Follows from point 1 and 2 of Theorem 3.2, plugging in the parameters specified in the assumptions. ■

4 Simple decrease condition

In this section, we analyze two methods based on simple decrease condition (i.e., with $\rho(t) = 0$, in (10)), one for potentially nonsmooth objectives and one for smooth objectives. Both methods follow the scheme presented in Algorithm 3, which is an adaptation to the BO setting of the mesh adaptive direct-search algorithm (MADS, see [2] and references therein). Again we lower bound the stepsize by a constant $\alpha_{\min}$. The stepsize updating rule we use to handle unsuccessful iterations depends on the mesh size parameter Δ_k and the contraction coefficient θ, and smoothness of the true objective (i.e., update varies between the smooth and the nonsmooth case).

It is a standard assumption in the analysis of MADS that all the iterates lie in a compact set (see, e.g., [3, Section 3]). In our framework, this can be ensured if the following boundedness assumption is satisfied.

Assumption 4.1 *The set*

$$\mathcal{L}_\varepsilon = \{x \in \mathbb{R}^{n_x} \mid F(x) \leq F(x_0) + 2L_f\varepsilon\} \tag{36}$$

is bounded.

The mesh, as defined in the literature (see,e.g., [5, 10] and references therein for further details), is a discrete set of points from which the algorithm selects candidate trial points. Its coarseness is parameterised by the mesh size parameter δ. The goal of each algorithm iteration is to get a mesh point whose objective function value improves with respect to the incumbent value. Given a positive spanning set D and a center x the related mesh is formally defined as follows:

$$M = \{x + \delta Dy \mid y \in \mathbb{N}^p\}, \tag{37}$$

where, with a slight abuse of notation, we use D also for the matrix $D \in \mathbb{R}^{n\times p}$ with columns corresponding to the elements of the set D. We notice that the mesh is just a conceptual tool, and is never actually constructed.

Algorithm 3: Inexact mesh based DS for bilevel optimization

Initialization: Choose starting point $x_0 \in \mathbb{R}^{n_x}$, stepsize lower bound $\alpha_{\min} \geq 0$, initial mesh size parameter $\alpha_0 = \alpha_{\min}\theta^{-\mu_0}$, with $\mu_0 \in \mathbb{N}_0$, starting frame parameter $\Delta_0 = \alpha_0$, stepsize contraction/expansion parameter $\theta \in (0,1) \cap \mathbb{Q}$, $G \in \mathbb{R}^{n\times n}$ invertible and $Z \in \mathbb{Z}^{n\times p}$ with columns forming a positive spanning set. Let $D = GZ$. Let $y_0 = \tilde{y}(x_0)$ be an approximate minimizer for the lower-level problem in x_0.
for $k = 0, 1, 2, \ldots$ **do**
  [**Optional**] Let M_k be the mesh with size parameter α_k, positive spanning set D and center x_k. Select a finite subset S_k of M_k.
  if $f(t, \tilde{y}(t)) < f(x_k, y_k)$ for some $t \in S_k$ **then**
    Set $x_{k+1} = t$, declare the iteration successful, and **go to step 7**.
  end if
  Choose a positive spanning set D_k such that $\{x_k + \alpha_k d \mid d \in D_k\} \subset M_k$. Compute the approximate minimizer $y_k^{\alpha_k d} = \tilde{y}(x_k + \alpha_k d_k)$ for the lower problem. Evaluate f at the poll points belonging to $\{(x_k + \alpha_k d, y_k^{\alpha_k d}) : d \in D_k\}$.
  if there exists $d_k \in D_k$ such that $f(x_k + \alpha_k d_k, y_k^{\alpha_k d_k}) < f(x_k, y_k)$ **then**
    Declare the iteration as successful. Set $x_{k+1} = x_k + \alpha_k d_k$ and $y_{k+1} = y_k^{\alpha_k d_k}$.
  else
    Declare the iteration as unsuccessful. Set $x_{k+1} = x_k$ and $y_{k+1} = y_k$.
  end if
  If the iteration was successful **then** set $\Delta_{k+1} = \theta^{-1}\Delta_k$ and $\alpha_k = \min(\Delta_k, \Delta_k^2)$. **Else** set $\Delta_{k+1} = \max\{\alpha_{\min}, \theta\Delta_k\}$ and $\alpha_{k+1} = \alpha_u(\alpha_k, \Delta_k, \theta)$.
  [**Optional**] If some approximate stationarity condition is satisfied, terminate the algorithm.
end for

4.1 Nonsmooth objectives

With respect to the general scheme presented in Algorithm 3, here the stepsize updating rule for unsuccessful iterations is given by $\alpha_u(\alpha_k, \Delta_k, \theta) = \min(\Delta_k, \Delta_k^2, \theta\alpha_k)$, ensuring that $\alpha_k \to 0$ and the mesh gets infinitely dense if the algorithm gets stuck in a certain point. The set of search directions D_k must be such that

$$\frac{\Delta_k}{\alpha_k} b_1(\alpha_k) \leq \|d\| \leq \frac{\Delta_k}{\alpha_k} b_2(\alpha_k) \tag{38}$$

for all $d \in D_k$, with $b_i : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ such that $\lim_{t\to 0} b_i(t) = 1$ for $i \in \{1, 2\}$. Thus with respect to the classic MADS scheme here the frame size Δ_k defines also a lower bound and not only an upper bound on the distance between the current iterate and tentative points selected in the poll step. This adjustment is necessary due to the error on the true objective evaluation. As shown in the next lemma, Condition (38) ensures that as the stepsize converges to 0 the tentative steps get closer and closer the boundary of a ball of radius $\alpha_{\min}$.

Lemma 4.1 *Assume that $\alpha_{\min} > 0$ and that (38) holds. Then if $\lim_{k\in K} \alpha_k = 0$, the set of limit points of $\{\alpha_k D_k\}_{k\in K}$ is contained in $S^{n_x-1}(\alpha_{\min})$.*

Proof. If $\lim_{k\in K} \alpha_k = 0$ then it holds that, for $k \in K$ large enough, $\Delta_k = \alpha_{\min}$. Consider $\{d_k\} = D_k$. It holds that, for all d_k,

$$\limsup_{k\in K} \|\alpha_k d_k\| \leq \limsup_{k\in K} \Delta_k b_2(\alpha_k) = \alpha_{\min}\,, \tag{39}$$

where we applied (38) in the inequality. Analogously, we can prove $\liminf_{k\in K} \|\alpha_k d_k\| \geq \Delta_k$, whence $\lim_{k\in K} \|\alpha_k d_k\| = \alpha_{\min}$, which implies the thesis. ■

We now extend to this scheme the (δ, ϵ)-Goldstein stationarity result proved under the sufficient decrease condition in Section 3.1. Also in this case we are not aware of any analogous result for the standard MADS scheme, which is instead known to convergence to Clarke stationary points [3].
We start with a lemma that extends a well known property of MADS (see, e.g., [3, Proposition 3.1]) to our bilevel setting.

Lemma 4.2 *Let Assumptions 2.4, 2.5 and 4.1 hold. Then the sequence $\{\alpha_k\}$ generated by Algorithm **??** is such that $\liminf \alpha_k = 0$.*

Proof. Since $\{\tilde{F}(x_k)\}$ is non-increasing (and strictly decreasing for successful iterations), $\{x_k\}$ is contained in the set $\mathcal{L}_\varepsilon$, which is compact by Assumptions 2.5 and 4.1. Thus $\liminf \alpha_k = 0$ follows from the finiteness of feasible points generated in $\mathcal{L}_\varepsilon$ when keeping the parameter α_k lower bounded, which can be proved with the same arguments used for MADS in [3, Proposition 3.1].
■

We can now state our main result.

Theorem 4.1 *Let Assumptions 2.4, 2.5 and 4.1 hold. Let K be a subset of unsuccessful iteration indices related to Algorithm **??**. Let us further assume that:*

- $\lim_{k\in K} x_k = \bar{x}$;

- $\lim_{k\in K} \alpha_k = 0$;
- $\{\hat{D}_k\}_{k\in K}$ *is dense in the unit sphere, with* $\hat{D}_k = \{\frac{d}{\|d\|} \mid d \in D_k\}$;
- *Condition* (38) *holds.*

Then, the limit point $\bar{x}$ *of* $\{x_k\}_{k\in K}$ *is* (δ, ϵ)*-Goldstein stationary, for*

$$\epsilon = \frac{4L_f\varepsilon}{\alpha_{\min}} \quad \textit{and} \quad \delta = \alpha_{\min}. \tag{40}$$

Proof. Let $\bar{d} \in \mathbb{R}^n$ with $\|\bar{d}\| = 1$, and let $L \subset K$ be such that $\lim_{k\in L} \frac{d_k}{\|d_k\|} \to \bar{d}$, with $d_k \in D_k$. Then $\alpha_k d_k \to \alpha_{\min}\bar{d}$ by Lemma 4.1. Now, for every $k \in L$

$$F(x_k) - F(x_k + \alpha_k d_k) \leq \tilde{F}(x_k) - \tilde{F}(x_k + \alpha_k d_k) + 2L_f\varepsilon \leq 2L_f\varepsilon\,, \tag{41}$$

where the first inequality follows from (9), and we used that the step k is unsuccessful in the second inequality. Passing to the limit, we obtain

$$F(\bar{x}) \leq F(\bar{x} + \alpha_{\min}\bar{d}) + 2L_f\varepsilon\,. \tag{42}$$

Now let $\bar{F}_{\bar{x}}(d) = F(\bar{x} + d) + \frac{2L_f\varepsilon}{\alpha_{\min}^2}\|d\|^2$. By applying (41) we get

$$\bar{F}_{\bar{x}}(0) \leq \bar{F}_{\bar{x}}(\alpha_{\min}\bar{d})\,,$$

and given that $\bar{d}$ is arbitrary, this holds for any d such that $\|d\| = \alpha_{\min}$. The thesis then follows as in the proof of Theorem 3.1. ■

As in Section 3.1, here we also have a corollary showing that for $\alpha_{\min} \propto \sqrt{\varepsilon}$ we are able to get a $(\mathcal{O}(\sqrt{\varepsilon}), \mathcal{O}(\sqrt{\varepsilon}))$-Goldstein stationary point.

Corollary 4.1 *Under the assumptions of Theorem 4.1, the limit point* $\bar{x}$ *of the sequence* $\{x_k\}$ *generated by Algorithm* **??** *with* $\alpha_{\min} = 2\sqrt{L_f\varepsilon}$ *is* (δ, ϵ)*-Goldstein stationary, for*

$$\epsilon = \delta = 2\sqrt{L_f\varepsilon}\,. \tag{43}$$

4.2 Smooth objectives

Now we consider the case where the true objective is smooth, i.e., Assumption 2.6 holds. With respect to the general scheme reported in Algorithm 3, we have $\alpha_u(\alpha_k, \Delta_k, \theta) = \min(\Delta_k, \Delta_k^2)$, and the algorithm terminates if $\alpha_k = \alpha_{k+1} = \alpha_{\min}$. As for D_k, it must always satisfy $\mathrm{cm}(D_k) \geq \kappa$ for some positive κ independent from k, as well as

$$\frac{\Delta_k}{\alpha_k} b_1 \leq \|d\| \leq \frac{\Delta_k}{\alpha_k} b_2 \tag{44}$$

for every $d \in D_k$.
We remark that convergence of mesh based schemes for smooth objectives is well understood (see, e.g., [5, Chapter 7]), so that once again our main contribution here is the adaptation to the bilevel setting. We begin our analysis by extending Lemma 3.2 under the simple decrease condition and condition (44) on the descent directions.

Lemma 4.3 *Let Assumptions 2.4 and 2.6 hold, together with* (11). *Let* $\{x_k\}$ *be a sequence generated by Algorithm* **??**. *If the step* k *is unsuccessful, then*

$$\|\nabla F(x_k)\| \leq \frac{1}{\kappa}\left(\frac{b_2\Delta_k L}{2} + \frac{2L_f\varepsilon}{b_1\Delta_k}\right). \tag{45}$$

Proof. Since the step is unsuccessful, by considering $d \in D_k$ such that

$$-\nabla F(x_k)^\top d \geq \kappa\|\nabla F(x_k)\|\|d\| \tag{46}$$

we have, reasoning as in (24) with $c = 0$

$$\kappa\alpha_k\|\nabla F(x_k)\|\|d\| - \alpha_k^2\frac{L}{2}\|d\|^2 \leq 2L_f\varepsilon\,. \tag{47}$$

Finally, we get

$$\|\nabla F(x_k)\| \leq \frac{1}{\kappa}\left(\frac{\alpha_k L\|d_k\|}{2} + \frac{2L_f\varepsilon}{\alpha_k\|d_k\|}\right) \leq \frac{1}{\kappa}\left(\frac{b_2\Delta_k L}{2} + \frac{2L_f\varepsilon}{b_1\Delta_k}\right). \tag{48}$$

■

We now extend Theorem 3.2 to our mesh based scheme. The main difference is the absence of complexity estimates, which to our knowledge are not available for MADS schemes.

Theorem 4.2 *Let Assumptions 2.4, 2.5 and 4.1 hold. Let* $\{x_k\}$ *be a sequence generated by Algorithm* **??**.

1. *If* $\alpha_{\min} > 0$, *then the algorithm terminates in a finite number of iterations, with the last iterate* $x_{\bar{k}}$ *satisfying,*
$$\|\nabla F(x_{\bar{k}})\| \leq \frac{1}{\kappa}\left(\frac{b_2\alpha_{\min}L}{2} + \frac{2L_f\varepsilon}{\alpha_{\min}b_1}\right). \tag{49}$$

2. *If, furthermore, it holds that* $\alpha_{\min} = 2\sqrt{\frac{L_f\varepsilon}{b_1 b_2 L}}$, *then*
$$\|\nabla F(x_{\bar{k}})\| \leq \frac{1}{\kappa}\sqrt{Lb_2L_f\varepsilon/b_1}\,. \tag{50}$$

3. *If* $\alpha_{\min} = 0$, *then* $\liminf \alpha_k = 0$, *and if additionally* $\alpha_0 \geq \bar{\alpha}_{\min} = 2\sqrt{\frac{L_f\varepsilon}{b_1 b_2 L}}$, *for some* $\bar{k} \in \mathbb{N}_0$ *we have*
$$\|\nabla F(x_{\bar{k}})\| \leq \frac{1}{\theta\kappa}\left(\frac{L\alpha_{\min}b_2}{2} + \frac{2L_f\varepsilon}{b_1\alpha_{\min}}\right), \tag{51}$$
and
$$F(x_k) \leq F(x_{\bar{k}}) + 2L_f\varepsilon \quad \text{for all } k \geq \bar{k}\,. \tag{52}$$

Proof. 1. Since the frame parameter Δ_k is lower bounded, the mesh parameter α_k is lower bounded as well, and, by the subsequent finiteness of $\bigcup_{k\in\mathbb{N}_0} M_k$, the algorithm terminates in a finite number of iterations. By the termination criterion, at the last iteration $\bar{k}$ we have $\Delta_{\bar{k}} = \alpha_{\min}$. Since the last iteration is unsuccessful, we hence get

$$\|\nabla F(x_{\bar{k}})\| \leq \frac{1}{\kappa}\left(\frac{b_2\Delta_k L}{2} + \frac{2L_f\varepsilon}{b_1\Delta_k}\right) = \frac{1}{\kappa}\left(\frac{b_2\alpha_{\min}L}{2} + \frac{2L_f\varepsilon}{b_1\alpha_{\min}}\right), \tag{53}$$

where we applied Lemma 4.3 in the second inequality.
2. Follows from the previous point replacing $\alpha_{\min}$ with the given value in (49).
3. The property $\liminf \alpha_k = 0$ follows from standard arguments used in the analysis of MADS schemes, already mentioned in the proof of Lemma 4.1. The result then follows from point 1 and 2 (similarly to point 3 in Theorem 3.2). ■

5 Numerical illustration

In this section, we evaluate the performance of the proposed algorithms on a large collection of nonlinear bilevel optimization problems.

Three direct-search solvers derived from Algorithm 2 and Algorithm 3 were implemented in Matlab: **Mesh-DS** (related to Algorithm 3) with the mesh defined as in [5, Algorithm 8.2], **Coordinate-DS** (related to Algorithm 2) with $D_k = [\mathcal{B}_\oplus, -\mathcal{B}_\oplus]$ (where $\mathcal{B}_\oplus$ is the canonical basis of $\mathbb{R}^n$), and **Random-DS** (related to Algorithm 2) with $D_k = [\frac{v}{\|v\|}, -\frac{v}{\|v\|}]$, where $v \in \mathbb{R}^n$ is a uniformly generated vector.

In our tests, the parameters used for Algorithm 2 and Algorithm 3 were set as follows: $\alpha_{\min} = 10^{-6}$, $\theta = \frac{1}{2}$, $\alpha_0 = 1$, $c = 10^{-3}$, and $\gamma = 2$. For all the tested approaches, the optional search step (Step 1) was not included. Instead, in the poll step, when we observed a decrease along a specific direction, we further explored it by using a simple extrapolation strategy (i.e., we multiplied the step-size α_k by γ and re-evaluated the function).

In our implementation, the lower-level problem is solved using the **fmincon** Matlab procedure. To quantify the impact of inexact lower-level solutions on the performances, we used 2 different accuracies when solving the lower-level problem (i.e., LL_tol $\in \{10^{-3}, 10^{-6}\}$). The rest of the **fmincon** default parameters were kept unchanged. A feasibility tolerance of 10^{-6} for constraints violation was used in the solution of the lower-level problem.

The three solvers, **Mesh-DS**, **Coordinate-DS**, and **Random-DS**, were evaluated using 33 small-scale bilevel optimization problems from the BOLIB Matlab library [46]. This library consists of a collection of academic and real-world problems. The dimensions of the tested instances, with respect to the upper-level problem, do not exceed 10 variables. Since an initial point is not provided, we generated five problem instances by randomly selecting five different initial points, thus getting a total of 175 problem instances.

The computational analysis is carried out by using well-known tools from the literature, that is data and performance profiles (see,e.g., [38] for further details). We briefly recall here their definitions. Given a set S of algorithms and a set P of problems, for $s \in S$ and $p \in P$, let $t_{p,s}$ be the number of function evaluations required by algorithm s on problem p to satisfy the condition

$$\tilde{F}(x_k) \leq \tilde{F}_{\text{low}} + \alpha(\tilde{F}(x_0) - \tilde{F}_{\text{low}}), \tag{54}$$

where $\gamma_p \in (0,1)$ and $\tilde{F}_{\text{low}}$ is the best objective function value achieved by any solver on problem p. Then, the performance and data profiles of solver s are defined by

$$\begin{aligned} \rho_s(\gamma) &= \frac{1}{|P|}\left|\left\{p \in P : \frac{t_{p,s}}{\min\{t_{p,s'} : s' \in S\}} \leq \gamma\right\}\right|, \\ d_s(\kappa) &= \frac{1}{|P|}\left|\{p \in P : t_{p,s} \leq \kappa(n_p+1)\}\right|, \end{aligned}$$

where n_p is the dimension of problem p. We used a budget of 500 upper level function evaluations in our experiments.

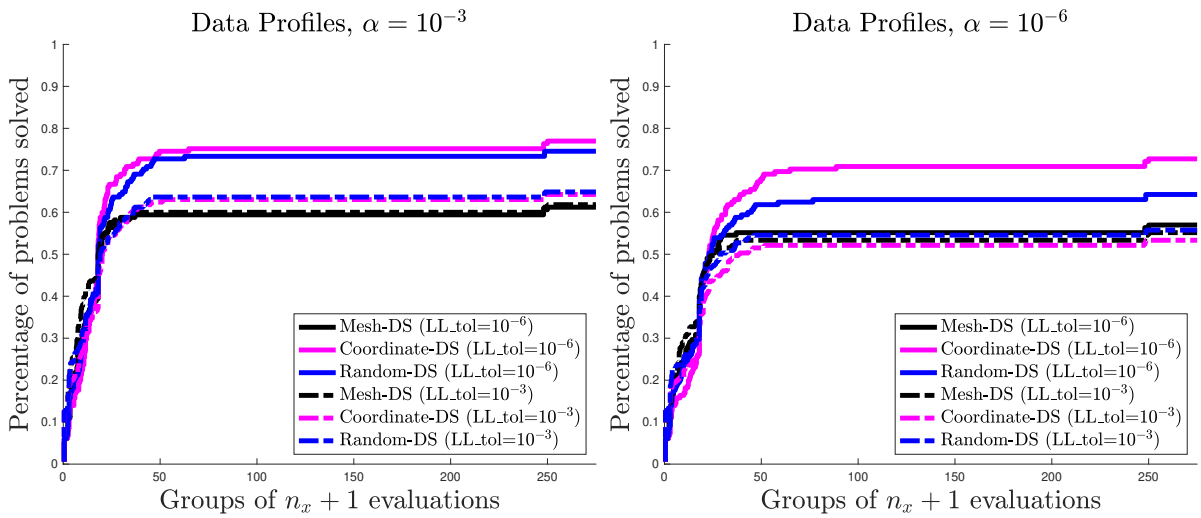

Figure 1: Data profiles using two type of tolerances to get an approximate minimizer for the lower-level problem.

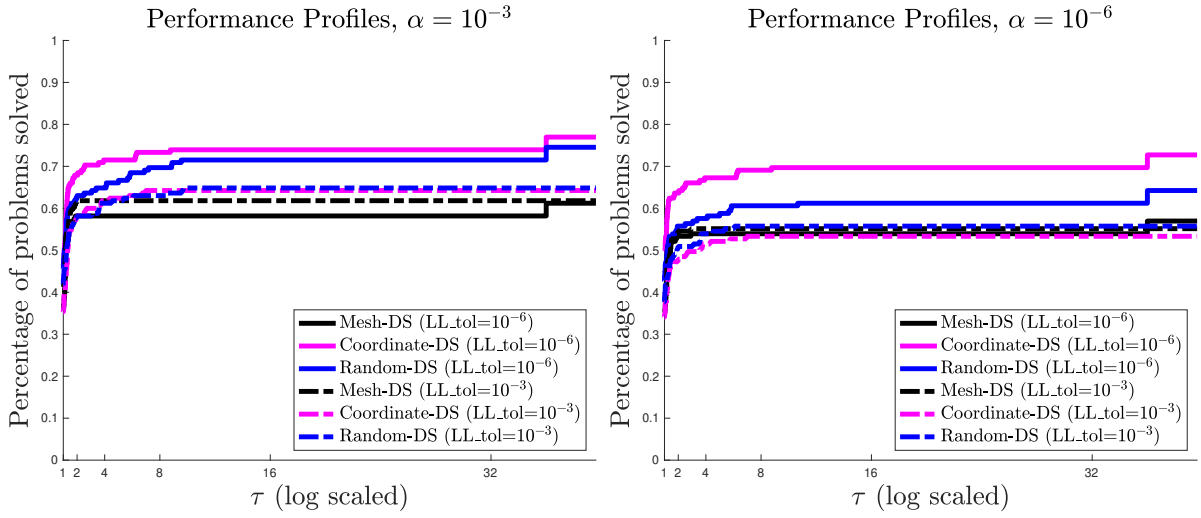

Figure 2: Performance profiles using two type of tolerances to get an approximate minimizer for the lower-level problem.

Figures 1-2 depict the resulting performance and data profiles, respectively, considering two levels of accuracy α: 10^{-3} and 10^{-6}. From Figure 2, it can be observed that the **Coordinate-DS** approach performs the best in terms of both efficiency (i.e., $\tau = 1$) and robustness (i.e., larger τ), particularly when the lower problem is solved accurately (i.e., LL_tol=10^{-6}). The data profiles (see Figure 2) indicate that all the direct-search approaches perform similarly for small budgets. However, as the budget increases, the accuracy of the lower problem becomes impactful on the solver's performance. Overall, on the tested problems, the directional direct-search approaches

seem to outperform the mesh-based direct-search approach.

6 Conclusion

In this work, we proposed an inexact direct-search based algorithmic framework for bilevel optimization, under the assumption that the lower-level problem can be solved within a fixed accuracy. We then proved convergence of two different classes of methods fitting our scheme, that is directional direct-search methods with sufficient decrease and mesh based schemes with simple decrease. Our results include complexity estimates for a directional direct-search scheme tailored for BO with smooth true objective, which extends previously known complexity estimates for the single level case. We also considered the nonsmooth case and gave convergence guarantees to (δ, ϵ)-Goldstein stationary points for both classes, thus nicely extending the known Clarke stationary point convergence properties of analogous schemes in the single level case. A lower bound on the stepsize allows these method to convergence to a point with the desired stationarity properties in a finite number of iterations. Preliminary numerical results suggest that directional direct-search methods might lead to better performance than mesh based strategies in this context.
Future developments include the extensions of our algorithms to constrained and stochastic objectives, as well as numerical comparisons with recent zeroth order smoothing based approaches for BO.

Data availability. The data analysed during the current study are available in the BOLIB library and the code will be made available by the authors upon reasonable request.

References

[1] Sotirios-Konstantinos Anagnostidis, Aurelien Lucchi, and Youssef Diouane. Direct-search for a class of stochastic min-max problems. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130, pages 3772–3780. PMLR, 2021.

[2] Charles Audet. *A survey on direct search methods for blackbox optimization and their applications*. Springer, 2014.

[3] Charles Audet and John E Dennis Jr. Mesh adaptive direct search algorithms for constrained optimization. *SIAM Journal on optimization*, 17(1):188–217, 2006.

[4] Charles Audet, Kwassi Joseph Dzahini, Michael Kokkolaras, and Sébastien Le Digabel. Stomads: Stochastic blackbox optimization using probabilistic estimates. *arXiv preprint arXiv:1911.01012*, 2019.

[5] Charles Audet and Warren Hare. Derivative-free and blackbox optimization. 2017.

[6] Yasmine Beck and Martin Schmidt. A gentle and incomplete introduction to bilevel optimization. 2021.

[7] Albert S Berahas, Liyuan Cao, Krzysztof Choromanski, and Katya Scheinberg. A theoretical and empirical comparison of gradient approximations in derivative-free optimization. *Foundations of Computational Mathematics*, 22(2):507–560, 2022.

[8] Lesi Chen, Jing Xu, and Jingzhao Zhang. On bilevel optimization without lower-level strong convexity. *arXiv preprint arXiv:2301.00712*, 2023.

[9] Benoît Colson, Patrice Marcotte, and Gilles Savard. An overview of bilevel optimization. *Annals of operations research*, 153:235–256, 2007.

[10] Andrew R Conn, Katya Scheinberg, and Luis N Vicente. *Introduction to derivative-free optimization*. SIAM, 2009.

[11] Andrew R Conn and Luís Nunes Vicente. Bilevel derivative-free optimization and its application to robust optimization. *Optimization Methods and Software*, 27(3):561–577, 2012.

[12] Stephan Dempe. *Foundations of bilevel programming*. Springer Science & Business Media, 2002.

[13] Stephan Dempe. Bilevel optimization: theory, algorithms, applications and a bibliography. *Bilevel Optimization: Advances and Next Challenges*, pages 581–672, 2020.

[14] Kwassi Joseph Dzahini. Expected complexity analysis of stochastic direct-search. *Computational Optimization and Applications*, 81:179–200, 2022.

[15] Matthias J Ehrhardt and Lindon Roberts. Inexact derivative-free optimization for bilevel learning. *Journal of Mathematical Imaging and Vision*, 63(5):580–600, 2021.

[16] Giovanni Fasano, Giampaolo Liuzzi, Stefano Lucidi, and Francesco Rinaldi. A linesearch-based derivative-free approach for nonsmooth constrained optimization. *SIAM journal on optimization*, 24(3):959–992, 2014.

[17] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*, pages 1568–1577. PMLR, 2018.

[18] Riccardo Grazzi, Luca Franceschi, Massimiliano Pontil, and Saverio Salzo. On the iteration complexity of hypergradient computation. In *International Conference on Machine Learning*, pages 3748–3758. PMLR, 2020.

[19] J. H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numer. Math.*, 2:84–90, 1960.

[20] Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Convergence analysis and enhanced design. In *International conference on machine learning*, pages 4882–4892. PMLR, 2021.

[21] Michael I Jordan, Guy Kornowski, Tianyi Lin, Ohad Shamir, and Manolis Zampetakis. Deterministic nonsmooth nonconvex optimization. *arXiv preprint arXiv:2302.08300*, 2023.

[22] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part I 16*, pages 795–811. Springer, 2016.

[23] Prashant Khanduri, Siliang Zeng, Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A near-optimal algorithm for stochastic bilevel optimization via double-momentum. *Advances in neural information processing systems*, 34:30271–30283, 2021.

[24] Thomas Kleinert, Martine Labbé, Ivana Ljubić, and Martin Schmidt. A survey on mixed-integer programming techniques in bilevel optimization. *EURO Journal on Computational Optimization*, 9:100007, 2021.

[25] Tamara G Kolda, Robert Michael Lewis, and Virginia Torczon. Optimization by direct search: New perspectives on some classical and modern methods. *SIAM review*, 45(3):385–482, 2003.

[26] Jeffrey Larson, Matt Menickelly, and Stefan M Wild. Derivative-free optimization methods. *Acta Numerica*, 28:287–404, 2019.

[27] Yingbin Liang et al. Lower bounds and accelerated algorithms for bilevel optimization. *Journal of Machine Learning Research*, 24(22):1–56, 2023.

[28] Tianyi Lin, Zeyu Zheng, and Michael Jordan. Gradient-free methods for deterministic and stochastic nonsmooth nonconvex optimization. *Advances in Neural Information Processing Systems*, 35:26160–26175, 2022.

[29] Bo Liu, Mao Ye, Stephen Wright, Peter Stone, and Qiang Liu. Bome! bilevel optimization made easy: A simple first-order approach. *Advances in Neural Information Processing Systems*, 35:17248–17262, 2022.

[30] Risheng Liu, Xuan Liu, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. A value-function-based interior-point method for non-convex bi-level optimization. In *International Conference on Machine Learning*, pages 6882–6892. PMLR, 2021.

[31] Risheng Liu, Yaohua Liu, Shangzhi Zeng, and Jin Zhang. Towards gradient-based bilevel optimization with non-convex followers and beyond. *Advances in Neural Information Processing Systems*, 34:8662–8675, 2021.

[32] Risheng Liu, Pan Mu, Xiaoming Yuan, Shangzhi Zeng, and Jin Zhang. A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In *International Conference on Machine Learning*, pages 6305–6315. PMLR, 2020.

[33] G. Liuzzi, S. Lucidi, F. Rinaldi, and L. N. Vicente. Trust-region methods for the derivative-free optimization of nonsmooth black-box functions. 29:3012–3035, 2019.

[34] Stefano Lucidi and Marco Sciandrone. A derivative-free algorithm for bound constrained optimization. *Computational Optimization and applications*, 21:119–142, 2002.

[35] Chinmay Maheshwari, S Shankar Sasty, Lillian Ratliff, and Eric Mazumdar. Convergent first-order methods for bi-level optimization and stackelberg games. *arXiv preprint arXiv:2302.01421*, 2023.

[36] M. Menickelly and S. M. Wild. Derivative-free robust optimization by outer approximations. *Mathematical Programming*, 179:157–193, 2020.

[37] Ayalew Getachew Mersha and Stephan Dempe. Direct search algorithm for bilevel programming problems. *Computational Optimization and Applications*, 49(1):1–15, 2011.

[38] J. J. Moré and S. M. Wild. Benchmarking derivative-free optimization algorithms. 20:172–191, 2009.

[39] Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17:527–566, 2017.

[40] Marco Rando, Cesare Molinari, Lorenzo Rosasco, and Silvia Villa. An optimal structured zeroth-order algorithm for non-smooth optimization. *arXiv preprint arXiv:2305.16024*, 2023.

[41] Francesco Rinaldi, Luis Nunes Vicente, and Damiano Zeffiro. A weak tail-bound probabilistic condition for function estimation in stochastic derivative-free optimization. *arXiv preprint arXiv:2202.11074*, 2022.

[42] Sara Venturini, Andrea Cristofari, Francesco Rinaldi, and Francesco Tudisco. Learning the right layers: a data-driven layer-aggregation strategy for semi-supervised learning on multilayer graphs. *arXiv preprint arXiv:2306.00152*, 2023.

[43] Luís Nunes Vicente. Worst case complexity of direct search. *EURO Journal on Computational Optimization*, 1(1-2):143–153, 2013.

[44] Dali Zhang and Gui-Hua Lin. Bilevel direct search method for leader–follower problems and application in health insurance. *Computers & operations research*, 41:359–373, 2014.

[45] Yihua Zhang, Yuguang Yao, Parikshit Ram, Pu Zhao, Tianlong Chen, Mingyi Hong, Yanzhi Wang, and Sijia Liu. Advancing model pruning via bi-level optimization. *Advances in Neural Information Processing Systems*, 35:18309–18326, 2022.

[46] S. Zhou, A. B. Zemkoho, and A. Tin. Bolib: Bilevel optimization library of test problems. *arXiv preprint arXiv:1812.00230v3*, 2020.