



HAL
open science

Certified Multi-Fidelity Zeroth-Order Optimization

Étienne de Montbrun, Sébastien Gerchinovitz

► **To cite this version:**

Étienne de Montbrun, Sébastien Gerchinovitz. Certified Multi-Fidelity Zeroth-Order Optimization. 2023. hal-04174484

HAL Id: hal-04174484

<https://hal.science/hal-04174484>

Preprint submitted on 1 Aug 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Certified Multi-Fidelity Zeroth-order Optimization

Étienne de Montbrun* and Sébastien Gerchinovitz†

Abstract. We consider the problem of multi-fidelity zeroth-order optimization, where one can evaluate a function f at various approximation levels (of varying costs), and the goal is to optimize f with the cheapest evaluations possible. In this paper, we study *certified* algorithms, which are additionally required to output a data-driven upper bound on the optimization error. We first formalize the problem in terms of a min-max game between an algorithm and an evaluation environment. We then propose a certified variant of the MFDOO algorithm and derive a bound on its cost complexity for any Lipschitz function f . We also prove an f -dependent lower bound showing that this algorithm has a near-optimal cost complexity. We close the paper by addressing the special case of noisy (stochastic) evaluations as a direct example.

Key words. Multi-fidelity optimization, global optimization, bandit algorithms

MSC codes. 62L15, 65K10, 65Y20, 68Q32, 68T05, 68W40

1. Introduction. We consider the problem of multi-fidelity zeroth-order optimization, which unfolds roughly as follows (details are given in Section 1.1). Let $f : \mathcal{X} \subset \mathbb{R}^d \rightarrow \mathbb{R}$ be a function. Assume that at any $x \in \mathcal{X}$, we can query the value $f(x)$ with any desired accuracy $\alpha > 0$, at a cost of $c(\alpha)$. Accurate evaluations (small α) come at a high cost. The goal of multi-fidelity optimization is to maximize f with the cheapest evaluations possible.

A typical example is the optimization of a function f computed with finite element modeling. A case with two fidelity functions (two values of α) appears in [55], for sheet-metal forming design with the goal of having no defects in the products (automobile inner panel in that paper). Given three variables x_1, x_2, x_3 modeling strong restraining forces on the metal, the goal is to set these forces to a good value to avoid both rupture and wrinkling. Two different finite element solvers were used to approximate f at any point x : incremental finite element solvers, or a one-step finite element model, which is computationally cheap but provides worse estimates than the former model. Finding a good design of the forces at a reasonable computational cost is an example of multi (two) fidelity optimization problem. Many other examples can be found, e.g., in thermodynamics [12, 33], design of new aircrafts [17], or nuclear criticality safety [42].

Certified optimization. In practice, algorithms that achieve small optimization errors with small evaluation costs are desirable but may not inform the user when a small optimization error has been obtained. In the example above, an engineer might require to *certify* the output of the algorithm, that is, to get a guaranteed optimization error bound that they can compute by only using the observed data and some (light) prior knowledge on f , as is done, e.g., in [20, 1] in the single-fidelity setting. Such requirement can be important in industrial fields involving safety-critical systems (e.g., cars, aircrafts, health, nuclear engineering).

*ANITI & Toulouse School of Economics, France (edemontb@ens-paris-saclay.fr)

†IRT Saint Exupéry & Institut de Mathématiques de Toulouse, France (sebastien.gerchinovitz@irt-saintexupery.com)

In this paper, we study the problem of finding a *certifiably* approximate-maximizer of a Lipschitz function f in the multi-fidelity setting. We quantify the smallest evaluation cost to reach this goal, by deriving nearly-matching upper and lower bounds for any such f .

1.1. Setting. We now formally define the setting. Let $\mathcal{X} \subset \mathbb{R}^d$ be a non-empty¹ compact set endowed with a norm $\|\cdot\|$, and $f : \mathcal{X} \rightarrow \mathbb{R}$ be an L -Lipschitz function, with a maximizer $x^* \in \mathcal{X}$.² Let also $c : (0, +\infty) \rightarrow [0, +\infty)$ be a non-increasing cost function.

The problem, which we describe in the online protocol below, can be seen as an interaction between two players: the algorithm A whose goal is to maximize f , and an environment E which returns perturbed values of f . They interact together in the following way: at every round $t \geq 1$, A picks a query point $x_t \in \mathcal{X}$ and an evaluation accuracy $\alpha_t > 0$; it then observes $y_t = E_t(x_t, \alpha_t) \in [f(x_t) - \alpha_t, f(x_t) + \alpha_t]$ at a cost of $c(\alpha_t)$ ($E_t(\cdot, \alpha)$ is sometimes called α -fidelity function); finally A recommends some candidate $x_t^* \in \mathcal{X}$ for a maximizer of f , and outputs an error certificate $\xi_t \geq 0$ with the constraint that $\xi_t \geq \max_{x \in \mathcal{X}} f(x) - f(x_t^*)$ (see a more formal definition below). This way, when using the algorithm, we can not only find an ε -maximizer x_t^* , but we can *know* when it is ε -optimal by looking at ξ_t , and thus confidently stop searching.

Note that the case of the constant cost $c(\alpha) = 1$ for all $\alpha > 0$ can be reduced to the single-fidelity setting, where f is observed perfectly (see Appendix C for details).

Online Protocol: Certified multi-fidelity zeroth-order optimization

Init: The environment secretly observes f and picks $E = (E_t)_{t \geq 1}$ with $E_t(x, \alpha) \in [f(x) - \alpha, f(x) + \alpha]$ for all $t \geq 1$, $x \in \mathcal{X}$, and $\alpha > 0$ (we also call E *the environment*)

- 1: **for** $t = 1, 2, \dots$, **do**
 - 2: A chooses a query point $x_t \in \mathcal{X}$ and an evaluation accuracy $\alpha_t > 0$
 - 3: A incurs a cost $c(\alpha_t)$
 - 4: E returns $y_t = E_t(x_t, \alpha_t) \in [f(x_t) - \alpha_t, f(x_t) + \alpha_t]$ (inaccurate evaluation of $f(x_t)$)
 - 5: A returns a recommendation x_t^* for the maximum of f , with an error certificate $\xi_t \geq 0$
-

Next we introduce key definitions before describing the optimization goal, our contributions, related works, and useful notation.

Definitions: environments and certified algorithms. For any L -Lipschitz function $f : \mathcal{X} \rightarrow \mathbb{R}$, we define the set $\mathcal{E}(f)$ of all environments associated with f , which are sequences of functions $E = (E_t)_{t \geq 1}$ with $E_t(x, \alpha) \in [f(x) - \alpha, f(x) + \alpha]$ for all $t \geq 1$, $x \in \mathcal{X}$ and $\alpha > 0$. We assume for simplicity that the sequence $E \in \mathcal{E}(f)$ is fixed from the beginning of the online protocol.³

We can now formally define *certified algorithms*. As can be seen from the online protocol above, x_t and α_t are deterministic functions of the past observations y_1, \dots, y_{t-1} , while x_t^* and ξ_t are deterministic functions of y_1, \dots, y_t . We additionally require that the certificates ξ_t satisfy $\xi_t \geq \max_{x \in \mathcal{X}} f(x) - f(x_t^*)$ for all rounds $t \in \mathbb{N}^*$, all L -Lipschitz functions $f : \mathcal{X} \rightarrow \mathbb{R}$

¹All throughout the paper, \mathcal{X} is implicitly assumed to be non empty.

²In fact, all results of Section 2 still hold if f is simply L -Lipschitz around x^* , that is, if $f(x) \geq f(x^*) - L\|x - x^*\|$ for all $x \in \mathcal{X}$.

³Since we only consider deterministic algorithms A and work towards guarantees that hold uniformly over all environments, this is in fact equivalent to playing against adversarial environments.

and all environments $E \in \mathcal{E}(f)$. We call *certified algorithm* any such sequence of functions $A = (x_t(\cdot), \alpha_t(\cdot), x_t^*(\cdot), \xi_t(\cdot))_{t \geq 1}$, and let \mathcal{A} denote the set of all certified algorithms. With a slight abuse of notation, we also sometimes write $x_t(E)$, $\alpha_t(E)$, $x_t^*(E)$ and $\xi_t(E)$ to make the dependency on E more explicit.

Optimization goal. Recall that $x^* \in \mathcal{X}$ denotes a maximizer of f . A classic goal in multi-fidelity optimization is to reach a small optimization error $f(x^*) - f(x_t^*)$ while minimizing the total cost $\sum_{s=1}^t c(\alpha_s)$ (see, e.g., [46, 47, 13]). In this paper, we address the stronger goal of finding a recommendation x_t^* with an error certificate ξ_t below ε (and thus an optimization error *known* to be bounded by ε), with the smallest cumulative cost $\sum_{s=1}^t c(\alpha_s)$ possible.

More formally, for any environment $E \in \mathcal{E}(f)$, we define the *cost complexity* $\sigma(A, E, \varepsilon)$ as the smallest total cost for which A can output a certificate below ε (when run against E). It can be expressed as follows (by convention, $\sigma(A, E, \varepsilon) = +\infty$ if no such C exists):

$$(1.1) \quad \sigma(A, E, \varepsilon) = \inf \left\{ C \in \mathbb{R} : \exists \tau \in \mathbb{N}^*, \sum_{t=1}^{\tau} c(\alpha_t(E)) \leq C \text{ and } \xi_{\tau}(E) \leq \varepsilon \right\}.$$

Equivalently, since costs are nonnegative, $\sigma(A, E, \varepsilon)$ is equal to the total cost incurred by A (when run against E) until its certificate ξ_t falls below ε for the first time.

In this paper, we are interested in certified algorithms with small cost complexity against any environment, that is, in algorithms $A \in \mathcal{A}$ that approximately reach the infimum

$$\inf_{A \in \mathcal{A}} \sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon)$$

for any unknown L -Lipschitz function f . Importantly, the above min-max quantity depends on f (through the set $\mathcal{E}(f)$), since some functions are easier than others to maximize with certified algorithms.

1.2. Main contributions and outline of the paper. In this paper, we study the cost complexity of certified algorithms to maximize L -Lipschitz functions in the multi-fidelity setting. We prove nearly-matching f -dependent upper and lower bounds, which extend the single-fidelity results of [1]. More precisely, we make the following set of contributions.

- On the modeling side, we formalize the problem of *certified* multi-fidelity zeroth-order optimization (see Section 1.1 above).
- In Section 2 we define a certified variant of the MFDOO algorithm [46]. We bound its cost complexity in terms of the key quantity $S_{\beta}(f, \varepsilon)$ defined in (1.2) below.
- In Section 3 we derive a nearly-matching f -dependent lower bound that holds for all certified algorithms (i.e., we derive a lower bound on the min-max quantity above).
- Finally, in Section 4 we address the special case of noisy evaluations of f . By a simple reduction to the (deterministic) multi-fidelity setting above, we derive a high-probability sample complexity bound which solves a conjecture by [1].

In the appendix we collect some technical proofs, useful simple geometric lemmas, together with a formal intuitive reduction from the single- to the multi-fidelity setting (Appendix C).

1.3. Related works. This paper has connections with several rich literatures, about single- or multi-fidelity settings, within deterministic or Bayesian frameworks, with or without certificates. Next we provide a (non-comprehensive) subset of references to these related works.

In the single-fidelity setting (where the unknown function f can be evaluated perfectly), the problem reduces to zeroth-order global optimization. Optimization algorithms *without certificates* have a very long history, in convex optimization (e.g., [39, 5, 6]), non-convex optimization (e.g., [20, 21, 24]), Bayesian optimization (e.g., [16]), stochastic optimization (e.g., [52, 3]), or bandit optimization (e.g., [38, 50]). Among the algorithmic techniques that are closest to this paper, we can mention the Piyavskii-Shubert algorithm [44, 49] for zeroth-order Lipschitz optimization, as well as discretized variants such as the branch-and-bound algorithm of [41] or the DIRECT algorithm by [25], to name a few. More recently several variants were also derived in the bandit community, since zeroth-order optimization can be cast as a bandit problem with continuously many arms. Examples of such bandit algorithms for perfect or noisy (stochastic) evaluations of f include DOO [37], HOO [7], (Sto)SOO [37, 56], POO [18], which are all based on a hierarchical partition of the input domain, as the c.MF-DOO algorithm of the present paper. Another discretization approach (yet computationally more challenging) is the Zooming algorithm for general metric spaces [30, 31]. We refer the reader to [38, 50] for further details and references on bandit algorithms.

Also close to our work is the Bayesian optimization literature, because of its rich contributions to the multi-fidelity setting. For single-fidelity optimization, we can mention the seminal work of Kushner [32], the use of the expected improvement function introduced in [36], and the EGO algorithm of [26], together with convergence rates in [9]. The kriging community also addressed global optimization with noisy observations [14, 43]. In [53, 54] the authors design and study a Gaussian Process-based bandit algorithm (GP-UCB) and derive regret bounds both under Bayesian and deterministic assumptions on the underlying function f . This algorithm was later adapted in [10] for the case of sequential mini-batch queries in parallel computing. A detailed review of the Bayesian optimization literature can be found in [16].

We now focus on two key features of our setting: multi-fidelity and error certificates.

Multi-fidelity optimization. Multiple works in the multi-fidelity setting come from the Bayesian optimization literature. For instance, Huang [23] designed a multi-fidelity variant of the EGO algorithm; a multi-fidelity counterpart of GP-UCB (MF-GP-UCB) was studied in [28, 27], while co-kriging (a multi-fidelity extension of kriging) was studied, e.g., in [15, 45]. A framework where the (finitely many) fidelity functions may be mutually dependent was introduced in [51], alongside a high probability guarantee. A more complete review on multi-fidelity Bayesian optimization can be found in [40] and in [16, Section 11.5].

Recently bandit algorithms were also extended to the multi-fidelity setting. Starting with the case of finitely-many arms [29], several algorithms were designed for continuously-many arms, which corresponds to multi-fidelity zeroth-order optimization *without* certificates. An extension of DOO (called MFDOO) was provided by [46], together with an optimization error bound (or simple regret bound) for a given overall cost budget. The authors also study a variant inspired from POO (MFPDOO) to handle the case of unknown smoothness (see a paragraph below). Similar algorithms based on HOO and POO (called MFHOO and MFPOO) were later introduced by [47] to cope with additional (stochastic) noise. In [13] the authors

develop the Kometo algorithm (based on StroquOOL) and prove nearly optimal upper bounds (with matching minimax lower bounds) on the optimization error given an overall cost budget. Their analysis also covers the cases of unknown smoothness and of possibly unbounded costs at accuracies α in the neighborhood of $\alpha = 0$. The case of delayed and noisy feedback was addressed in [57] with a generalization of HOO.

Error certificates. Though convergence results about the optimization error $f(x^*) - f(x_t^*)$ in terms of a total cost budget are now well established in the multi-fidelity setting, the question of *certifying* approximate maximizers with minimum total cost has not been addressed, to the best of our knowledge.

The notion of error certificate appeared in several other settings, such as, e.g., in convex optimization [5], where the *duality gap* between primal and dual feasible points plays the role of an error certificate. In zeroth-order Lipschitz optimization with perfect evaluations of f (single-fidelity setting), the Piyavskii-Shubert algorithm [44, 49] is naturally endowed with an error certificate, which is the difference between the maximum value of a guaranteed upper bounding function of f and the maximum value $f(x_s)$ observed so far. For one-dimensional inputs, a tight analysis of the number of evaluations before which this certificate falls below ε was given in [19] (see also [11]), with a simple integral expression. This result was generalized to multi-dimensional inputs by [4, 1], and shown in [1] to be achievable with the tractable⁴ c.DOO algorithm, with a nearly matching f -dependent lower bound. In this paper we extend the complexity analysis of [1] to the multi-fidelity setting, using a certified variant of MFDOO [46] for the upper bound.

Adaptivity to smoothness versus error certificates. Another important series of works is about adaptivity to the unknown smoothness of f , that is, the question of achieving nearly optimal optimization performances with an algorithm that has (almost) no prior knowledge on the smoothness of f . Among the many algorithms designed to that end, let us mention the seminal DIRECT algorithm of [25], the $Z(k)$ algorithm of [22], as well as bandit algorithms (with simple or cumulative regret guarantees) including (Sto)SOO [37, 56], the two-phase algorithm of [8], POO and GPO [18, 48], AdaLIPO [35], SequOOL and StroquOOL [2], and their multi-fidelity variants [46, 47, 13]. See also [34] for a detailed account on possible and impossible adaptivity results in the single-fidelity setting.

Though adaptivity to unknown smoothness is a key robustness feature of optimization algorithms, we stress that it is in a way *incompatible* with the certificate requirement. For instance, as noted by [1] in the single-fidelity setting, when optimizing a Lipschitz function f with unknown Lipschitz constant $\text{Lip}(f)$, it is impossible to produce a finite certificate ξ_t after any number t of evaluations of f , since there could be an arbitrarily steep bump in a yet unobserved input region. More formally, if f has a maximizer x^* within the interior of \mathcal{X} , the lower bound of [1, Theorem 2] scales at least as $(L/\text{Lip}(f))^d$ when $L \rightarrow +\infty$, which implies that the minimum number of evaluations that certified algorithms need for the function f is arbitrarily large, if we require such algorithms to output valid certificates for all Lipschitz functions g with arbitrarily large Lipschitz constants $\text{Lip}(g)$. The same intuitive remark applies to our multi-fidelity setting, by the lower bound of Theorem 3.1 in Section 3.

⁴Tractability refers to a small (logarithmic) number of elementary operations per evaluation of f .

1.4. Notation. We collect below some notation that is used all throughout the paper, including the key quantity $S_\beta(f, \varepsilon)$ defined in (1.2) below.

Standard notation. $\mathbb{N} = \{0, 1, 2, \dots\}$ denotes the set of natural numbers, and $\mathbb{N}^* = \mathbb{N} \setminus \{0\}$ denotes the set of positive natural numbers. For any $x \in \mathbb{R}^d$ and $r > 0$, we write $B(x, r) = \{u \in \mathbb{R}^d : \|u - x\| \leq r\}$ for the closed ball centered at x with radius r .

Lipschitz functions, ε -optimal points, layers. Let \mathcal{F}_L denote the set of L -Lipschitz functions from \mathcal{X} to \mathbb{R} . Let also $\text{diam}(\mathcal{X}) = \sup_{x, y \in \mathcal{X}} \|x - y\|$ denote the diameter of \mathcal{X} . Since f is L -Lipschitz, the largest possible optimization error $f(x^*) - f(x_i^*)$ is bounded by $\varepsilon_0 := L \cdot \text{diam}(\mathcal{X})$. In the sequel, we will thus only consider values $\varepsilon \in (0, \varepsilon_0)$, and set $m_\varepsilon := \lceil \log_2(\varepsilon_0/\varepsilon) \rceil$. For any $1 \leq k \leq m_\varepsilon - 1$ we define the intermediate target error $\varepsilon_k := \varepsilon_0 2^{-k}$; we also set $\varepsilon_{m_\varepsilon} := \varepsilon$.

For any $0 \leq a < b$, we denote the set of a -optimal points by $\mathcal{X}_a := \{x \in \mathcal{X} : f(x^*) - f(x) \leq a\}$, and we define the *layer* $\mathcal{X}_{(a, b]} := \{x \in \mathcal{X} : a < f(x^*) - f(x) \leq b\}$, which is the set of b -optimal points that are not a -optimal.

Packing number. For any $r > 0$, the r -packing number $\mathcal{N}(\mathcal{X}', r)$ of a subset $\mathcal{X}' \subset \mathcal{X}$ is the largest number n of r -separated points $x'_1, \dots, x'_n \in \mathcal{X}'$, that is, such that $\|x'_i - x'_j\| > r$ for all $i \neq j \leq n$. (By convention, $\mathcal{N}(\mathcal{X}', r) = 0$ if \mathcal{X}' is empty. Note also that $\mathcal{N}(\mathcal{X}', r) < +\infty$ since \mathcal{X} is compact.)

The complexity quantity $S_\beta(f, \varepsilon)$. For any $\beta > 0$, we set

$$(1.2) \quad S_\beta(f, \varepsilon) := \mathcal{N}\left(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}\right) c(\beta\varepsilon) + \sum_{k=1}^{m_\varepsilon} \mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}, \frac{\varepsilon_k}{L}\right) c(\beta\varepsilon_k).$$

As we will see later, $S_\beta(f, \varepsilon)$ plays a key role in the optimal cost complexity of certified algorithms. We briefly explain why. Let $x \in \mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})}$. To realize that x belongs to the layer $\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})}$, a single call to f at x with (prophetic) evaluation accuracy $\alpha \approx \varepsilon_k$ would be enough, yielding a cost roughly of $c(\varepsilon_k)$. By L -Lipschitz continuity of f , this single evaluation also helps classify nearby points (at distance roughly ε_k/L) within the same layer. Therefore, the whole layer could (hopefully) be identified with roughly $\mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}, \frac{\varepsilon_k}{L}\right)$ evaluations of f at accuracy $\alpha \approx \varepsilon_k$. Repeating these arguments over all layers and using another (similar) argument over \mathcal{X}_ε suggests that the sum $S_\beta(f, \varepsilon)$ above characterizes the optimal cost complexity of certified algorithms. Next we prove upper and lower bounds supporting this intuition.

2. The Certified MFDOO Algorithm, and its Cost Complexity. In this section we define a certified version of the MFDOO algorithm [46], and then study its cost complexity (Theorem 2.4 below), which we will prove to be nearly optimal in Section 3.

Similarly to MFDOO [46] and its ancestors (e.g., the branch-and-bound algorithm of [41], HOO [7], DOO [37], POO [18] c.DOO [1], etc), our algorithm takes as input a hierarchical partitioning of \mathcal{X} , that is, a tree-based structure X whose each node represents a region of \mathcal{X} and has K children, which correspond to a K -partition of the parent region. More precisely, X is an infinite sequence of subsets $(X_{h,i})_{h \in \mathbb{N}, i \in \{0, \dots, K^h - 1\}}$ of \mathcal{X} called *cells* such that $\mathcal{X} \subset X_{0,0}$, and for any depth $h \in \mathbb{N}$ and location $0 \leq i \leq K^h - 1$, the cells $X_{h+1, Ki}, \dots, X_{h+1, K(i+1)-1}$ form a partition of $X_{h,i}$ (the nodes $(h+1, Ki), \dots, (h+1, K(i+1)-1)$ are the *children* of node (h, i)). Each cell has a *representative* $x_{h,i} \in X_{h,i}$. We assume that $x_{h,i} \in \mathcal{X}$ whenever $X_{h,i} \cap \mathcal{X} \neq \emptyset$. (A typical example is the barycenter of the cell, if inside \mathcal{X} .)

In all the paper, we make the following two assumptions. The first one is classical (e.g., [7, 37]). The second one appeared in [1] and was useful to derive bounds on DOO or c.DOO in terms of packing numbers of layers $\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}$. Both assumptions can always be satisfied when \mathcal{X} is compact. For example, if $\mathcal{X} = [0, 1]^d$ and $\|\cdot\|$ is the sup norm, we can take the regular dyadic partitioning $(X_{h,i})_{h \in \mathbb{N}, i \in \{0, \dots, 2^{dh}-1\}}$ consisting of 2^{dh} cubes of size 2^{-h} at depth $h \geq 0$, with centers $x_{h,i}$. Then, Assumptions 2.1 and 2.2 hold true with $R = 1$ and $\delta = \nu = 1/2$.

Assumption 2.1. There exist two positive constants $\delta \in (0, 1)$ and $R > 0$ such that, for all $h \in \mathbb{N}$, $i \in \{0, \dots, K^h - 1\}$, and all $u, v \in X_{h,i}$, we have $\|u - v\| \leq R\delta^h$.

Assumption 2.2. There exists $\nu > 0$ such that, with δ as in Assumption 2.1, for any $h, h' \in \mathbb{N}$, $i \in \{0, \dots, K^h - 1\}$ and $i' \in \{0, \dots, K^{h'} - 1\}$ with $(h, i) \neq (h', i')$, we have $\|x_{h,i} - x_{h',i'}\| \geq \nu\delta^{\max\{h, h'\}}$.

We now define c.MF-DOO (Certified Multi-Fidelity Deterministic Optimistic Optimization). The pseudo-code is given in Algorithm 2.1 below. The algorithm maintains a set \mathcal{L}_t of active nodes (or *leaves*) whose associated cells cover \mathcal{X} . At the end of each iteration (Line 14), c.MF-DOO picks the most promising leaf (h^*, i^*) by maximizing the surrogate $y_{h,i} + LR\delta^h + \alpha_{h,i}$, which is an upper bound on $f(x)$ for any $x \in X_{h,i}$.⁵ Then, during the next iteration, c.MF-DOO develops the tree by querying one after the other all the (feasible) children of (h^*, i^*) . At Line 9, it picks $x_t = x_{h^*+1,j}$ as the next query point and $\alpha_t = LR\delta^{h^*+1}$ for the accuracy. (A much smaller value of α_t could be counter-productive: it could come at a much higher cost, while not improving the optimization process by much, since the surrogate is an over-approximation of f with a mistake possibly of the order of $LR\delta^{h^*+1}$ on the cell $X_{h^*+1,j}$ even if f were observed exactly.) After receiving the approximate evaluation y_t of $f(x_t)$, c.MF-DOO returns the recommendation $x_t^* = x_{\bar{t}}$ (see Line 11), which is the point with the currently best guaranteed value of f . Finally, the certificate ξ_t at Lines 12 or 15 is the difference between a guaranteed upper bound $y_{h^*,i^*} + LR\delta^{h^*} + \alpha_{h^*,i^*}$ on $\max(f)$ and a guaranteed lower bound $y_{\bar{t}} - \alpha_{\bar{t}}$ on $f(x_t^*)$.

Note that in Algorithm 2.1 and in the rest of the paper, we identify any round $t \geq 1$ with the node (h, i) that is queried at time t .⁶ Depending on our needs, we index quantities either by rounds or nodes (writing, e.g., y_t or $y_{h,i}$).

Before bounding the cost complexity of c.MF-DOO, we start by proving that the ξ_t 's defined at Lines 3, 12 and 15 are valid certificates.

Lemma 2.3. *Suppose Assumption 2.1 holds, and that $f : \mathcal{X} \rightarrow \mathbb{R}$ is an L -Lipschitz function, with a maximizer $x^* \in \mathcal{X}$. Then, for any environment $E \in \mathcal{E}(f)$ and any $t \in \mathbb{N}^*$, the quantity ξ_t defined at Lines 3, 12 and 15 of Algorithm 2.1 is a valid certificate, that is: $f(x^*) - f(x_t^*) \leq \xi_t$.*

Proof. Since f is L -Lipschitz and $R \geq \text{diam}(\mathcal{X})$, note that $\xi_1 = LR \geq f(x^*) - f(x_1^*)$. We now prove the lemma for any subsequent round. For any $t' \geq 2$, consider the moment when the algorithm reaches Line 12 with $t = t'$. Next we show that the certificate $\xi_t = y_{h^*,i^*} + LR\delta^{h^*} + \alpha_{h^*,i^*} - \max_{s \leq t} (y_s - \alpha_s)$ defined at that time satisfies $\xi_t \geq f(x^*) - f(x_t^*)$ (the potential update at Line 15 will be addressed at the end of the proof). The associated node

⁵Indeed, $y_{h,i}$ is an $\alpha_{h,i}$ -approximation of $f(x_{h,i})$, f is L -Lipschitz, and the maximum distance from a point in $X_{h,i}$ to $x_{h,i}$ is at most $R\delta^h$, so that $\max_{x \in X_{h,i}} f(x) \leq f(x_{h,i}) + LR\delta^h \leq y_{h,i} + LR\delta^h + \alpha_{h,i}$.

⁶By definition of Algorithm 2.1, there is indeed an injection $t \in \mathbb{N}^* \mapsto (h, i)$ with $h \in \mathbb{N}$, $i \in \{0, \dots, K^h - 1\}$.

Algorithm 2.1 c.MF-DOO (Certified Multi-Fidelity Deterministic Optimistic Optimization)

Inputs: \mathcal{X} , K , $(X_{h,i})_{h \in \mathbb{N}, i \in \{0, \dots, K^h - 1\}}$, $(x_{h,i})_{h \in \mathbb{N}, i \in \{1, \dots, K^h - 1\}}$, δ , R , L

Initialization Let $t \leftarrow 1$ and $\mathcal{L}_1 \leftarrow \{(0, 0)\}$

- 1: Pick the first query point $x_1 \leftarrow x_{0,0}$, and the first accuracy $\alpha_1 \leftarrow LR$
- 2: Observe the value $y_1 = E_1(x_1, \alpha_1) \in [f(x_1) - \alpha_1, f(x_1) + \alpha_1]$
- 3: Output the recommendation $x_1^* \leftarrow x_1$ and certificate $\xi_1 \leftarrow LR$
- 4: Pick the first node $(h^*, i^*) \leftarrow (0, 0)$
- 5: **for** *iteration* = 1, 2, ... **do**
- 6: **for each** child $(h^* + 1, j)$ of (h^*, i^*) **do**
- 7: **if** $X_{h^*+1,j} \cap \mathcal{X} \neq \emptyset$ **then**
- 8: Let $t \leftarrow t + 1$ and $\mathcal{L}_t \leftarrow \mathcal{L}_{t-1} \cup \{(h^* + 1, j)\}$
- 9: Pick the query point $x_t \leftarrow x_{h^*+1,j}$ and accuracy $\alpha_t \leftarrow LR\delta^{h^*+1}$
- 10: Observe the value $y_t = E_t(x_t, \alpha_t) \in [f(x_t) - \alpha_t, f(x_t) + \alpha_t]$ given by E
- 11: Output the recommendation $x_t^* = x_{\tilde{t}}$, with $\tilde{t} \in \arg \max_{1 \leq s \leq t} \{y_s - \alpha_s\}$
- 12: Output the certificate $\xi_t = y_{h^*, i^*} + LR\delta^{h^*} + \alpha_{h^*, i^*} - (y_{\tilde{t}} - \alpha_{\tilde{t}})$
- 13: Remove (h^*, i^*) from \mathcal{L}_t
- 14: Let $(h^*, i^*) \in \arg \max_{(h,i) \in \mathcal{L}_t} \{y_{h,i} + LR\delta^h + \alpha_{h,i}\}$
- 15: Update the last certificate $\xi_t = y_{h^*, i^*} + LR\delta^{h^*} + \alpha_{h^*, i^*} - (y_{\tilde{t}} - \alpha_{\tilde{t}})$

(h^*, i^*) was either defined at Line 4 if the outer **for** loop is still at iteration 1, or at Line 14 otherwise. In both cases, we have $(h^*, i^*) \in \arg \max_{(h,i) \in \mathcal{L}_m} \{y_{h,i} + LR\delta^h + \alpha_{h,i}\}$ for a number $m \leq t - 1$ of evaluations of f .

Note that, by induction on the **iteration** variable, the cells $X_{h,i}$ associated with the leaves $(h, i) \in \mathcal{L}_m$ form a partition of a superset of \mathcal{X} . Let $(\bar{h}, \bar{i}) \in \mathcal{L}_m$ be the node of the cell $X_{\bar{h}, \bar{i}}$ containing x^* . By the maximizing property of (h^*, i^*) , we have

$$(2.1) \quad \begin{aligned} y_{h^*, i^*} + LR\delta^{h^*} + \alpha_{h^*, i^*} &\geq y_{\bar{h}, \bar{i}} + LR\delta^{\bar{h}} + \alpha_{\bar{h}, \bar{i}} \\ &\geq f(x_{\bar{h}, \bar{i}}) + LR\delta^{\bar{h}} \geq f(x^*), \end{aligned}$$

where the last line follows from $|y_{\bar{h}, \bar{i}} - f(x_{\bar{h}, \bar{i}})| \leq \alpha_{\bar{h}, \bar{i}}$, Assumption 2.1, and the fact that f is L -Lipschitz. To conclude, note that $y_s - \alpha_s \leq f(x_s)$ for all $s \leq t$, and therefore $y_{\tilde{t}} - \alpha_{\tilde{t}} \leq f(x_{\tilde{t}}) = f(x_t^*)$. Combining this with (2.1) entails that $\xi_t \geq f(x^*) - f(x_t^*)$. Noting that the same arguments apply (with $m = t$) if ξ_t is redefined at Line 15 concludes the proof. \blacksquare

We just proved that c.MF-DOO is a certified algorithm. We now show that its cost complexity can be controlled in terms of the quantity $S_\beta(f, \varepsilon)$ defined in (1.2). We recall that $\varepsilon_0 := L \operatorname{diam}(\mathcal{X})$, $m_\varepsilon := \lceil \log_2(\varepsilon_0/\varepsilon) \rceil$, $\varepsilon_{m_\varepsilon} := \varepsilon$, and $\varepsilon_k := \varepsilon_0 2^{-k}$ for all $1 \leq k \leq m_\varepsilon - 1$.

Theorem 2.4. *Assume that $c : \mathbb{R}^+ \rightarrow \mathbb{R}$ is non-increasing and that $\mathcal{X} \subset \mathbb{R}^d$ is compact. Suppose that Assumptions 2.1 and 2.2 hold, and that for some known $L > 0$, c.MF-DOO is run with evaluation accuracies $\alpha_{h,i} = LR\delta^h$ (cf. Algorithm 2.1). Then there exists a constant $a > 0$ (e.g., $a = K$ if $\nu \geq 3R$ or $a = K(1 + 6R/\nu)^d$ otherwise) such that, for any L -Lipschitz function $f : \mathcal{X} \rightarrow \mathbb{R}$, any environment $E \in \mathcal{E}(f)$, and any $\varepsilon \in (0, \varepsilon_0)$,*

$$\sigma(\text{c.MF-DOO}, E, \varepsilon) \leq aS_{\frac{\varepsilon}{3}}(f, \varepsilon) + c(LR).$$

We make several comments before proving the theorem.

Related upper bounds. A similar upper bound was proved in [1, Theorem 1] when f is evaluated perfectly, which corresponds to the special case $c(\alpha) = 1$ for all α . Theorem 2.4 above generalizes this result (up to constants) to the multi-fidelity setting.

Other related results are bounds for MFDOO [46] and Kometo [13], which are multi-fidelity algorithms *without certificates*. In that setting, the performances are measured differently. The cost complexity can be defined as the total cost incurred by the algorithm before outputting an ε -optimal recommendation (the difference with the certified setting is that the learner has no *observable* proof that an ε -maximizer has been found.) With such performance measure, MFDOO satisfies a complexity bound similar to $S_\beta(f, \varepsilon)$ but *without the first term* $\mathcal{N}(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}) c(\beta\varepsilon)$ in (1.2).⁷ This difference can be negligible for some functions (e.g., if $c_1 \|x - x^*\|^\nu \leq f(x^*) - f(x) \leq c_2 \|x - x^*\|^\nu$ for all $x \in \mathcal{X}$ and some $c_1, c_2 > 0$ and $0 < \nu \leq 1$, where x^* is a maximizer of f), but it can be dramatic for other functions. For instance, for constant functions, the term $\mathcal{N}(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}) c(\beta\varepsilon)$ is of the order of $(L/\varepsilon)^d c(\beta\varepsilon)$. The reason behind this large additional term in the certified setting is intuitive: a constant function f is perfectly optimized after one evaluation only, but *certifying* the result at accuracy ε somehow requires to evaluate the function on a ε/L -cover of \mathcal{X} with accuracies $\alpha_t \approx \varepsilon$, so as to make sure no bumps of size ε were forgotten.

Note also that, contrary to [13], we work with a *known* bias function but an *unknown* cost function. This is because we aim at *certifying* an ε -maximizer of f , rather than optimally allocating a total evaluation budget Λ .

On the choice of ε_k . As can be seen from the proof below, the upper bound is actually true for any decreasing sequence $\varepsilon_0 = L \operatorname{diam}(\mathcal{X}) > \varepsilon_1 > \dots > \varepsilon_{m-1} > \varepsilon_m = \varepsilon$ and any $m \geq 1$. The specific sequence $\varepsilon_k := \varepsilon_0 2^{-k}$ however realizes a good trade-off between small ratios $\varepsilon_{k-1}/\varepsilon_k \leq 2$ and a small number of terms $m_\varepsilon = \lceil \log_2(\varepsilon_0/\varepsilon) \rceil$. The nearly-matching lower bound of Section 3 will indeed imply that this sequence is nearly optimal.

Possible improvements or consequences. Note that the constant a was not optimized and could likely be improved. Furthermore, similarly to [1], under a mild geometric condition on \mathcal{X} recalled in Section 4, the sum $S_{\delta/3}(f, \varepsilon)$ can be bounded (up to multiplicative constants) in between two integrals of the form $\int_{\mathcal{X}} c(b \cdot (\Delta_x + \varepsilon)) / (\Delta_x + \varepsilon)^d dx$, where $\Delta_x = f(x^*) - f(x)$ and $b \in \{\delta/3, \delta/12\}$ (provided $\varepsilon < \varepsilon_0/2$). This integral form is omitted due to lack of space.

Proof of Theorem 2.4. The proof generalizes that of [1, Theorem 1] to the multi-fidelity setting, with similar arguments yet a few technical subtleties. In order to bound the total cost incurred by c.MF-DOO against environment E , we control the index $I_\varepsilon \geq 1$ of the first iteration (cf Line 5) at the end of which the certificate falls below ε . More precisely, let (h_ℓ^*, i_ℓ^*)

⁷This bound can be proved along the same lines as those of Theorem 2.4. See also [46, Theorem 1] and [13, Theorem 3] for similar bounds under slightly weaker assumptions (relating f directly to the hierarchical partitioning) but that are expressed in terms of a near-optimality dimension of f , and thus do not reflect the fact that constant functions are easy to optimize. Roughly speaking, the bound of [46, Theorem 1] for MFDOO is in spirit close to $\sum_{k=1}^{m_\varepsilon} \mathcal{N}(\mathcal{X}_{\varepsilon_{k-1}}, \varepsilon_k/L) c(\beta\varepsilon_k)$, instead of the tighter bound $\sum_{k=1}^{m_\varepsilon} \mathcal{N}(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}, \varepsilon_k/L) c(\beta\varepsilon_k)$.

be the node chosen at the end of each iteration $\ell \geq 1$ (Line 14). Then, we define I_ε by⁸

$$I_\varepsilon = \inf \left\{ \ell \in \mathbb{N}^* : y_{h_\ell^*, i_\ell^*} + LR\delta^{h_\ell^*} + \alpha_{h_\ell^*, i_\ell^*} \leq \max_{s \leq T_\ell} \{y_s - \alpha_s\} + \varepsilon \right\},$$

where for any $\ell \geq 1$, the quantity T_ℓ denotes the total number of evaluations of f until the leaf (h_ℓ^*, i_ℓ^*) is selected at Line 14. Next we focus on $\tau := T_{I_\varepsilon}$. Note that $\xi_\tau \leq \varepsilon$ by definition of I_ε and ξ_τ (in Line 15). Recalling that $\sigma(\text{c.MF-DOO}, E, \varepsilon)$ is the total cost that c.MF-DOO incurs until outputting a certificate below ε for the first time, this entails

$$(2.2) \quad \sigma(\text{c.MF-DOO}, E, \varepsilon) \leq \sum_{t=1}^{\tau} c(\alpha_t).$$

We now split the right-hand side into several terms involving the layers $\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}$. We set $(h_0^*, i_0^*) = (0, 0)$. Note that the points x_t queried at times $t \in \{2, \dots, \tau\}$ are all associated with nodes (h, i) that are children of some (h_ℓ^*, i_ℓ^*) , $\ell = 0, \dots, I_\varepsilon - 1$, and that these (h, i) are queried only once. Therefore,

$$(2.3) \quad \sum_{t=1}^{\tau} c(\alpha_t) \leq c(\alpha_1) + \sum_{\ell=0}^{I_\varepsilon-1} \sum_{j=K i_\ell^*}^{K(i_\ell^*+1)-1} c(\alpha_{h_\ell^*+1, j}) = c(LR) + K \sum_{x_{h^*, i^*} \in \mathcal{E}_\varepsilon} c(LR\delta^{h^*+1}),$$

where we set $\mathcal{E}_\varepsilon := \{x_{h_0^*, i_0^*}, \dots, x_{h_{I_\varepsilon-1}^*, i_{I_\varepsilon-1}^*}\}$ (the $x_{h, i}$ are pairwise-distinct by Assumption 2.2).

We now split the sum over \mathcal{E}_ε above into $m_\varepsilon + 1 = \lceil \log_2(\varepsilon_0/\varepsilon) \rceil + 1$ terms. Recall from Section 1.4 that $\varepsilon_0 = L \text{diam}(\mathcal{X})$, $\varepsilon_k = \varepsilon_0 2^{-k}$ for $1 \leq k \leq m_\varepsilon - 1$, and $\varepsilon_{m_\varepsilon} = \varepsilon$. Since the sets \mathcal{X}_ε (all ε -optimal points) and $\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}$, $k = 1, \dots, m_\varepsilon$ (all points in between ε_k and ε_{k-1} optimal) form a partition of \mathcal{X} ,

$$(2.4) \quad \mathcal{E}_\varepsilon = (\mathcal{E}_\varepsilon \cap \mathcal{X}_\varepsilon) \cup \bigcup_{k=1}^{m_\varepsilon} (\mathcal{E}_\varepsilon \cap \mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}) .$$

Let $N_{\varepsilon, k}$ be the cardinality of $\mathcal{E}_\varepsilon \cap \mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}$ for all $1 \leq k \leq m_\varepsilon$ and $N_{\varepsilon, m_\varepsilon+1}$ be the cardinality of $\mathcal{E}_\varepsilon \cap \mathcal{X}_\varepsilon$. Moreover, let $h_{\varepsilon, k}$ be the maximum depth h^* reached by points x_{h^*, i^*} in $\mathcal{E}_\varepsilon \cap \mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}$ for $1 \leq k \leq m_\varepsilon$, and $h_{\varepsilon, m_\varepsilon+1}$ be the maximum depth reached by points in $\mathcal{E}_\varepsilon \cap \mathcal{X}_\varepsilon$. By (2.3), (2.4), and the fact that $\alpha \mapsto c(\alpha)$ is non-increasing, we have:

$$(2.5) \quad \sum_{t=1}^{\tau} c(\alpha_t) \leq c(LR) + K \sum_{k=1}^{m_\varepsilon+1} N_{\varepsilon, k} c(LR\delta^{h_{\varepsilon, k}+1}).$$

We now bound $N_{\varepsilon, k}$ from above and $LR\delta^{h_{\varepsilon, k}+1}$ from below (see (2.7), (2.8), (2.9), and (2.10)).

We start by proving (2.6) below. Let $x^* \in \mathcal{X}$ be a maximizer of f . Following the same arguments as before (2.1) (using f being L -Lipschitz and $E \in \mathcal{E}(f)$), we can see that, for any node (h^*, i^*) selected at Line 14,

$$y_{h^*, i^*} + LR\delta^{h^*} + \alpha_{h^*, i^*} \geq f(x^*).$$

⁸The rest of the proof implies that the set is never empty, so that $I_\varepsilon < +\infty$.

This implies that $f(x_{h^*,i^*}) + LR\delta^{h^*} + 2\alpha_{h^*,i^*} \geq f(x^*)$, and thus $3LR\delta^{h^*} \geq f(x^*) - f(x_{h^*,i^*})$ (by $\alpha_{h^*,i^*} = LR\delta^{h^*}$). Therefore, for any $\ell \in \{0, \dots, I_\varepsilon - 1\}$ (the case $\ell = 0$ is straightforward),

$$(2.6) \quad x_{h_\ell^*, i_\ell^*} \in \mathcal{X}_{3LR\delta^{h_\ell^*}}.$$

Now, let $k \in \{1, \dots, m_\varepsilon\}$ and $x_{h_\ell^*, i_\ell^*} \in \mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})} \cap \mathcal{E}_\varepsilon$. By (2.6) and the fact that $x_{h_\ell^*, i_\ell^*} \in \mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})}$ is not ε_k -optimal, we have $3LR\delta^{h_\ell^*} > \varepsilon_k$. This and the definition of $h_{\varepsilon, k}$ entail

$$(2.7) \quad 3LR\delta^{h_{\varepsilon, k}} > \varepsilon_k.$$

Also, let $x_{h, j}$ and $x_{h', j'}$ be two distinct elements of $\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})} \cap \mathcal{E}_\varepsilon$. By Assumption 2.2 and (2.7), we have $\|x_{h, j} - x_{h', j'}\| \geq \nu\delta^{\max\{h, h'\}} > \frac{\nu\varepsilon_k}{3LR}$. Therefore, and by definition of a packing number, we get that for all $k \in \{1, \dots, m_\varepsilon\}$, the cardinality $N_{\varepsilon, k}$ of $\mathcal{E}_\varepsilon \cap \mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})}$ satisfies

$$(2.8) \quad N_{\varepsilon, k} \leq \mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})}, \frac{\nu\varepsilon_k}{3LR}\right) \leq \underbrace{\left(1_{\frac{\nu}{3R} \geq 1} + 1_{\frac{\nu}{3R} < 1} \left(1 + \frac{6R}{\nu}\right)^d\right)}_{=: b} \mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})}, \frac{\varepsilon_k}{L}\right)$$

from Lemma B.2 in Appendix B.

Now, let $x_{h_\ell^*, i_\ell^*} \in \mathcal{X}_\varepsilon \cap \mathcal{E}_\varepsilon$, with $\ell \in \{0, \dots, I_\varepsilon - 1\}$. If $\ell \geq 1$, we have, by definition of I_ε ,

$$y_{h_\ell^*, i_\ell^*} + LR\delta^{h_\ell^*} + \alpha_{h_\ell^*, i_\ell^*} > \max_{s \leq T_\ell} \{y_s - \alpha_s\} + \varepsilon \geq y_{h_\ell^*, i_\ell^*} - \alpha_{h_\ell^*, i_\ell^*} + \varepsilon.$$

Again, replacing $\alpha_{h_\ell^*, i_\ell^*}$ with $LR\delta^{h_\ell^*}$, we get $3LR\delta^{h_\ell^*} > \varepsilon$, which is also true if $\ell = 0$. Therefore,

$$(2.9) \quad 3LR\delta^{h_{\varepsilon, m_\varepsilon+1}} > \varepsilon.$$

Combining this inequality with Assumption 2.2, we get that $\|x_{h, j} - x_{h', j'}\| > \frac{\nu\varepsilon}{3LR}$ for any two distinct elements $x_{h, j}, x_{h', j'}$ of $\mathcal{X}_\varepsilon \cap \mathcal{E}_\varepsilon$. Therefore, and by definition of a packing number,

$$(2.10) \quad N_{\varepsilon, m_\varepsilon+1} \leq \mathcal{N}\left(\mathcal{X}_\varepsilon, \frac{\nu\varepsilon}{3LR}\right) \leq b \cdot \mathcal{N}\left(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}\right),$$

by Lemma B.2 again. Putting (2.2), (2.5), (2.8), (2.10) together and setting $a := Kb$, we get

$$\begin{aligned} \sigma(\text{c.MF-DOO}, E, \varepsilon) &\leq c(LR) + a\mathcal{N}\left(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}\right) c\left(LR\delta^{h_{\varepsilon, m_\varepsilon+1}+1}\right) + a \sum_{k=1}^{m_\varepsilon} \mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})}, \frac{\varepsilon_k}{L}\right) c\left(LR\delta^{h_{\varepsilon, k}+1}\right) \\ &\leq c(LR) + a\mathcal{N}\left(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}\right) c\left(\frac{\delta\varepsilon}{3}\right) + a \sum_{k=1}^{m_\varepsilon} \mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1})}, \frac{\varepsilon_k}{L}\right) c\left(\frac{\delta\varepsilon_k}{3}\right), \end{aligned}$$

where we used (2.7), (2.9), and the fact that c is non-increasing. This concludes the proof. \blacksquare

3. Lower Bound. In this section, for any fixed L -Lipschitz function $f : \mathcal{X} \rightarrow \mathbb{R}$, we derive a lower bound on the worst-environment cost complexity $\sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon)$ (see (1.1)) of any certified algorithm A . Our main result below, which depends on f through the key quantity $S_\beta(f, \varepsilon)$ defined in (1.2), generalizes [1, Theorem 2] from perfect evaluations of f to the multi-fidelity setting. We recall that $\varepsilon_0 = L \operatorname{diam}(\mathcal{X})$ and $m_\varepsilon = \lceil \log_2(\varepsilon_0/\varepsilon) \rceil$.

Theorem 3.1. *Assume that $c : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is a non-increasing function and that $\mathcal{X} \subset \mathbb{R}^d$ is a compact and connected set. Then, for some constant $a_d > 0$ (e.g., $a_d = 1/65^d$), the cost complexity of any certified algorithm A satisfies, for any L -Lipschitz function $f : \mathcal{X} \rightarrow \mathbb{R}$ and any target optimization error $\varepsilon \in (0, \varepsilon_0/2)$,*

$$\sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon) \geq \frac{a_d(1 - \operatorname{Lip}(f)/L)^d}{1 + m_\varepsilon} S_{16}(f, \varepsilon).$$

We make three comments before proving the theorem.

On the optimality of the bound. First note that a_d depends exponentially on the dimension d . While removing such exponential dependence completely is challenging without stronger assumptions on f (if not impossible), the constant 65 was not optimized and could be improved. Besides, the quantity $(1 - \operatorname{Lip}(f)/L)^d$ vanishes as L approaches $\operatorname{Lip}(f)$. Importantly, the case $L = \operatorname{Lip}(f)$ is not really relevant in practice, because it scarcely happens that one knows exactly the best Lipschitz constant $\operatorname{Lip}(f)$ without knowing the function itself. In the more realistic case when one only knows a strict upper bound L on $\operatorname{Lip}(f)$, and under the mild assumption $\sup_{\alpha > 0} c(\alpha)/c(2\alpha) < +\infty$ (which holds, e.g., if $c(\alpha)$ is polynomial in $1/\alpha$), Theorems 2.4 and 3.1 imply that c.MF-DOO is nearly optimal (among all certified algorithms) in terms of cost complexity, up to logarithmic and dimension-dependent multiplicative factors.

Earlier lower bounds. Similarly to Section 2, Theorem 3.1 can be compared to (at least) two types of existing lower bounds. First, our lower bound generalizes that of [1, Theorem 2] (where f can be evaluated perfectly at the same cost as coarse evaluations) to the multi-fidelity setting, where costs play a crucial role. Note that a study of the boundary case $L = \operatorname{Lip}(f)$ was provided by [1, Section 4], with different phenomena appearing in dimensions $d = 1$ or $d \geq 2$. Though out of the scope of this paper and with limited practical consequences, it would be interesting to investigate whether similar phenomena occur in our multi-fidelity setting.

A second type of lower bound (of a minimax form) was proved in [13, Theorem 1] for *non-certified* algorithms, under several assumptions on the cost function (more precisely, on a so-called *cost-to-bias* function) and a near-optimality dimension of f . Unlike the minimax approach, our lower bound is f -dependent. This is possible since we work with *certified* algorithms, whose data-driven certificates must be robust to yet unobserved values of f .

On more collaborative environments. The lower bound of Theorem 3.1 holds for the worst case among all environments. However, the cost complexity can be improved for some specific environments. Indeed one could think of the following collaborative environment: when asked two times for an approximation of $f(x)$ with two accuracies α and α' at the same $x \in \mathcal{X}$, it first returns $f(x) - \alpha$ and then $f(x) + \alpha'$. Then even with $\alpha = \alpha' = \varepsilon_0$, the algorithm has an exact knowledge of $f(x)$ after only two queries at the same x . Against such an environment, we would thus be in the same setting as in [1] (perfect evaluations of f) with only twice as much queries, and could therefore achieve a cost complexity of the order of

$c(\varepsilon_0) \cdot (\mathcal{N}(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}) + \sum_{k=1}^{m_\varepsilon} \mathcal{N}(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}, \frac{\varepsilon_k}{L}))$. Since $c(\varepsilon_0)$ can be much smaller than $c(\varepsilon)$ in practice, this would greatly improve over the upper bound of Thm 2.4, which (by Thm 3.1) is nearly optimal when considering worst-case environments $E \in \mathcal{E}(f)$. In practice we could expect the environment to lie between the collaborative and worst-case extremes. The question of deriving environment-dependent lower and upper bounds is left for future work.

The proof of Theorem 3.1 is inspired from that of [1, Theorem 2] who addressed the case of perfect evaluations of f . Our generalization to the multi-fidelity setting however requires additional technicalities. Before the proof, we introduce several useful quantities and lemmas. Recall that \mathcal{F}_L denotes the set of all L -Lipschitz functions from \mathcal{X} to \mathbb{R} . We first define the quantity $\text{err}_\tau(A, E)$ for any $\tau \geq 1$, as the best certificate ξ_τ that algorithm A could output given the sequence $(x_t, \alpha_t, y_t)_{t \leq \tau}$ and given x_τ^* (note that we consider all L -Lipschitz functions $g : \mathcal{X} \rightarrow \mathbb{R}$ that are compatible with the observations $(y_t)_{t \leq \tau}$):

$$\text{err}_\tau(A, E) = \sup \left\{ \max(g) - g(x_\tau^*) : g \in \mathcal{F}_L \text{ and } \forall t \leq \tau, g(x_t) \in [y_t - \alpha_t, y_t + \alpha_t] \right\}.$$

As can be seen from the next lemma, for any certified algorithm A , its certificate ξ_τ at any time $\tau \geq 1$ is bounded from below by $\text{err}_\tau(A, E)$. The proof is postponed to Appendix A.

Lemma 3.2. *Let $f : \mathcal{X} \rightarrow \mathbb{R}$ be an L -Lipschitz function, $E \in \mathcal{E}(f)$ be an environment and A be a certified algorithm with certificates $(\xi_t(E))_{t \geq 1}$ when run against E .⁹ Then for all $\tau \in \mathbb{N}^*$, $\xi_\tau(E) \geq \text{err}_\tau(A, E)$.*

Denoting the set of all certified algorithms by \mathcal{A} , we can now define¹⁰

$$C_{\text{inf}}(f, \varepsilon) = \inf \left\{ C \in \mathbb{R} : \exists A \in \mathcal{A}, \forall E \in \mathcal{E}(f), \exists \tau \in \mathbb{N}^*, \sum_{t=1}^{\tau} c(\alpha_t(E)) \leq C \text{ and } \text{err}_\tau(A, E) \leq \varepsilon \right\}$$

which represents the minimum cost that certified algorithms must incur to maximize f with an error certifiably below ε against any environment. This intuitive fact is formalized in the following lemma, which is proved in Appendix A.

Lemma 3.3. *Let A be a certified algorithm, $f : \mathcal{X} \rightarrow \mathbb{R}$ be an L -Lipschitz function, and $\varepsilon \in (0, \varepsilon_0/2)$. Then, $\sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon) \geq C_{\text{inf}}(f, \varepsilon)$.*

Another intuitive result is that an algorithm cannot output a certificate $\xi_t \leq \varepsilon$ unless it has already requested some value of f with an evaluation accuracy $\alpha_t \leq \varepsilon$. This implies that the total cost needed to certify an error at level ε must be at least of $c(\varepsilon)$. This is stated formally below and proved in Appendix A. Interestingly, this result would not hold if we worked with specific, possibly collaborative, environments (see a remark above).

Lemma 3.4. *Assume $\mathcal{X} \subset \mathbb{R}^d$ is compact and connected, and $c : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ non-increasing. Then $C_{\text{inf}}(f, \varepsilon) \geq c(\varepsilon)$ for any L -Lipschitz function $f : \mathcal{X} \rightarrow \mathbb{R}$ and any $\varepsilon \in (0, \varepsilon_0/2)$.*

We can now prove Theorem 3.1.

⁹As mentioned in the introduction, ξ_t is a function of all values y_1, \dots, y_t observed so far. We stress the (implicit) dependency on E since it is key in the proof.

¹⁰To see that the set of C 's is never empty, take $A = \text{c.MF-DOO}$ and apply Theorem 2.4 and Lemma 3.2.

Proof of Theorem 3.1. We assume without loss of generality that $\text{Lip}(f) < L$ and set $\Omega_f = \left(\frac{1-\text{Lip}(f)/L}{65}\right)^d$. We want to show that

$$\sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon) \geq \frac{\Omega_f}{1+m_\varepsilon} S_{16}(f, \varepsilon).$$

Since $\sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon) \geq C_{\text{inf}}(f, \varepsilon)$ (by Lemma 3.3), it is sufficient to show that $C_{\text{inf}}(f, \varepsilon) \geq \frac{\Omega_f}{1+m_\varepsilon} S_{16}(f, \varepsilon)$. We set $K = \frac{16L}{L-\text{Lip}(f)}$ and note that $\Omega_f \leq \frac{1}{(1+4K)^d}$. We can distinguish between two cases:

First case: Assume first that $\frac{S_{16}(f, \varepsilon)}{1+m_\varepsilon} \leq (1+4K)^d c(\varepsilon)$. Then

$$\frac{\Omega_f}{1+m_\varepsilon} S_{16}(f, \varepsilon) \leq c(\varepsilon) \leq C_{\text{inf}}(f, \varepsilon),$$

where the last inequality follows from Lemma 3.4. In this case, the theorem is proved.

Second case: We now assume that $\frac{S_{16}(f, \varepsilon)}{1+m_\varepsilon} > (1+4K)^d c(\varepsilon)$. The idea is to upper bound the average of the $(1+m_\varepsilon)$ terms that define $S_{16}(f, \varepsilon)$ by the largest one.

Let $\tilde{\varepsilon}$ be the scale with maximum contribution in (1.2) with $\beta = 16$, that is:

$$\tilde{\varepsilon} = \begin{cases} \varepsilon & \text{if } \mathcal{N}\left(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}\right) c(16\varepsilon) \geq \max_{1 \leq k \leq m_\varepsilon} \mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}, \frac{\varepsilon_k}{L}\right) c(16\varepsilon_k) \\ \varepsilon_{k^*-1} & \text{otherwise, where } k^* \in \arg \max_{1 \leq k \leq m_\varepsilon} \mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}, \frac{\varepsilon_k}{L}\right) c(16\varepsilon_k) \end{cases}$$

Since $\mathcal{N}\left(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L}\right) \leq \mathcal{N}\left(\mathcal{X}_\varepsilon, \frac{\varepsilon}{2L}\right)$ and $\mathcal{N}\left(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}, \frac{\varepsilon_k}{L}\right) \leq \mathcal{N}\left(\mathcal{X}_{\varepsilon_{k-1}}, \frac{\varepsilon_{k-1}}{2L}\right)$ for all $1 \leq k \leq m_\varepsilon$, and since c is non-increasing, we then have $S_{16}(f, \varepsilon) \leq (1+m_\varepsilon) \mathcal{N}\left(\mathcal{X}_{\tilde{\varepsilon}}, \frac{\tilde{\varepsilon}}{2L}\right) c(8\tilde{\varepsilon})$.

Then, using Lemma B.2, the previous result, and the assumption of the second case, we have:

$$\begin{aligned} \mathcal{N}\left(\mathcal{X}_{\tilde{\varepsilon}}, \frac{K\tilde{\varepsilon}}{L}\right) c(8\tilde{\varepsilon}) &\geq \left(\frac{1}{1+4K}\right)^d \mathcal{N}\left(\mathcal{X}_{\tilde{\varepsilon}}, \frac{\tilde{\varepsilon}}{2L}\right) c(8\tilde{\varepsilon}) \\ (3.1) \qquad \qquad \qquad &\geq \left(\frac{1}{1+4K}\right)^d \frac{S_{16}(f, \varepsilon)}{(1+m_\varepsilon)} > c(\varepsilon). \end{aligned}$$

To prove our result, we assume for a moment that $C_{\text{inf}}(f, \varepsilon) < \frac{\Omega_f}{1+m_\varepsilon} S_{16}(f, \varepsilon)$ and will show that it raises a contradiction. Combining with (3.1) and $\Omega_f \leq \frac{1}{(1+4K)^d}$, this indeed yields

$$C_{\text{inf}}(f, \varepsilon) < \Omega_f (1+4K)^d \mathcal{N}\left(\mathcal{X}_{\tilde{\varepsilon}}, \frac{K\tilde{\varepsilon}}{L}\right) c(8\tilde{\varepsilon}) \leq \mathcal{N}\left(\mathcal{X}_{\tilde{\varepsilon}}, \frac{K\tilde{\varepsilon}}{L}\right) c(8\tilde{\varepsilon}).$$

Then, by definition of $C_{\text{inf}}(f, \varepsilon)$, there exists an algorithm $A \in \mathcal{A}$ such that for all environments $E \in \mathcal{E}(f)$, there exists $\tau \in \mathbb{N}^*$ such that

$$(3.2) \qquad \sum_{t=1}^{\tau} c(\alpha_t(E)) < \mathcal{N}\left(\mathcal{X}_{\tilde{\varepsilon}}, \frac{K\tilde{\varepsilon}}{L}\right) c(8\tilde{\varepsilon}) \text{ and } \text{err}_\tau(A, E) \leq \varepsilon.$$

We now consider the “noiseless” environment $E = (E_t)_{t \geq 1} \in \mathcal{E}(f)$ defined by $E_t(x, \alpha) = f(x)$ for all $t \geq 1$, $x \in \mathcal{X}$, and $\alpha > 0$. Let $\tau \in \mathbb{N}^*$ be such that (3.2) holds. Let $M = \mathcal{N}\left(\mathcal{X}_{\tilde{\varepsilon}}, \frac{K\tilde{\varepsilon}}{L}\right)$,

and let $\{\tilde{x}_1, \dots, \tilde{x}_M\}$ be a $(K\tilde{\varepsilon}/L)$ -packing of $\mathcal{X}_{\tilde{\varepsilon}}$. Note that the closed balls $B(\tilde{x}_m, \frac{K\tilde{\varepsilon}}{2L})$ with centers $\tilde{x}_1, \dots, \tilde{x}_M$ and radius $K\tilde{\varepsilon}/2L$ are pairwise disjoint. Note also that $M \geq 2$ from $\mathcal{N}(\mathcal{X}_{\tilde{\varepsilon}}, \frac{K\tilde{\varepsilon}}{L}) > \frac{c(\varepsilon)}{c(8\tilde{\varepsilon})} \geq 1$ by (3.1), $8\tilde{\varepsilon} \geq \varepsilon$, and c being non-increasing ($c(8\tilde{\varepsilon}) > 0$ by (3.1)).

For any $1 \leq m \leq M$, let c_m be the maximum cost spent at any round on the m -th ball, that is $c_m = \max\{c(\alpha_t) : t \in \mathcal{T}_m\}$ if the set $\mathcal{T}_m := \{t = 1, \dots, \tau : \|x_t(E) - \tilde{x}_m\| \leq K\tilde{\varepsilon}/2L\}$ is non-empty, and $c_m = 0$ otherwise. We know from (3.2) that the total cost up to round τ is smaller than $Mc(8\tilde{\varepsilon})$. By the pigeonhole principle, there is at least one $m \leq M$ for which $c_m < c(8\tilde{\varepsilon})$. Assume without loss of generality that this is true for $m = 1$. Then for any $t \in \mathcal{T}_1$ (if such t exists), the cost $c(\alpha_t(E))$ is smaller than $c(8\tilde{\varepsilon})$. Therefore, either $\alpha_t(E) > 8\tilde{\varepsilon}$ whenever the ball $B(\tilde{x}_1, \frac{K\tilde{\varepsilon}}{2L})$ is visited (since c is non-increasing) or this ball is never visited.

We just showed that, on the ball $B(\tilde{x}_1, \frac{K\tilde{\varepsilon}}{2L})$, algorithm A never queried f with an evaluation accuracy $\alpha_t \leq 8\tilde{\varepsilon}$. Next we show the following consequence: that the inequality $\text{err}_\tau(A, E) \leq \varepsilon$ in (3.2) cannot be true, by exhibiting an L -Lipschitz function $g \in \mathcal{F}_L$ compatible with the observations $y_t = E_t(x_t(E), \alpha_t(E)) = f(x_t(E))$ and such that $\max(g) - g(x_\tau^*(E)) > \varepsilon$. This will raise a contradiction in (3.2) and conclude the proof. To that end, we consider the two functions $g = f \pm h_{\tilde{\varepsilon}}$, with $h_{\tilde{\varepsilon}} : \mathcal{X} \rightarrow \mathbb{R}$ defined by

$$(3.3) \quad h_{\tilde{\varepsilon}}(x) = \max \left\{ 8\tilde{\varepsilon} - 16\frac{L}{K}\|x - \tilde{x}_1\|, 0 \right\}.$$

First note that both $f - h_{\tilde{\varepsilon}}$ and $f + h_{\tilde{\varepsilon}}$ are L -Lipschitz, since $h_{\tilde{\varepsilon}}$ is $(L - \text{Lip}(f))$ -Lipschitz (by $\frac{16L}{K} = L - \text{Lip}(f)$). Moreover, since $h_{\tilde{\varepsilon}}$ is supported on $\mathcal{X} \cap B(\tilde{x}_1, K\tilde{\varepsilon}/2L)$ and $\|h_{\tilde{\varepsilon}}\|_\infty \leq 8\tilde{\varepsilon} \leq \alpha_t(E)$ for all $t \in \mathcal{T}_1$, the two functions $f - h_{\tilde{\varepsilon}}$ and $f + h_{\tilde{\varepsilon}}$ belong by construction to the set

$$\mathcal{G} := \left\{ g \in \mathcal{F}_L : \forall t = 1, \dots, \tau, g(x_t(E)) \in [f(x_t(E)) - \alpha_t(E), f(x_t(E)) + \alpha_t(E)] \right\}.$$

We now show that $\max(g) - g(x_\tau^*(E)) > \varepsilon$ for $g = f - h_{\tilde{\varepsilon}}$ or $g = f + h_{\tilde{\varepsilon}}$, by distinguishing two subcases. If $x_\tau^*(E) \in B(\tilde{x}_1, K\tilde{\varepsilon}/4L)$, we perturb f “downwards” around \tilde{x}_1 and consider $g = f - h_{\tilde{\varepsilon}}$. In this case, since $h_{\tilde{\varepsilon}}(x_\tau^*(E)) \geq 4\tilde{\varepsilon}$ and $h_{\tilde{\varepsilon}}(\tilde{x}_2) = 0$, we have $\max(g) - g(x_\tau^*(E)) \geq f(\tilde{x}_2) - h_{\tilde{\varepsilon}}(\tilde{x}_2) - (f(x_\tau^*(E)) - h_{\tilde{\varepsilon}}(x_\tau^*(E))) \geq -\tilde{\varepsilon} + 4\tilde{\varepsilon} = 3\tilde{\varepsilon}$.¹¹ In the other case, if $x_\tau^*(E) \notin B(\tilde{x}_1, K\tilde{\varepsilon}/4L)$, we consider $g = f + h_{\tilde{\varepsilon}}$: since $h_{\tilde{\varepsilon}}(x_\tau^*(E)) \leq 4\tilde{\varepsilon}$ and $h_{\tilde{\varepsilon}}(\tilde{x}_1) = 8\tilde{\varepsilon}$, we have $\max(g) - g(x_\tau^*(E)) \geq f(\tilde{x}_1) + h_{\tilde{\varepsilon}}(\tilde{x}_1) - (f(x_\tau^*(E)) + h_{\tilde{\varepsilon}}(x_\tau^*(E))) \geq -\tilde{\varepsilon} + 8\tilde{\varepsilon} - 4\tilde{\varepsilon} = 3\tilde{\varepsilon}$.

In both subcases above, we proved $\max(g) - g(x_\tau^*(E)) \geq 3\tilde{\varepsilon} > \varepsilon$ for some $g \in \{f - h_{\tilde{\varepsilon}}, f + h_{\tilde{\varepsilon}}\} \subset \mathcal{G}$, which entails $\text{err}_\tau(A, E) > \varepsilon$ (by definition of err_τ). This raises a contradiction in (3.2), so that we must have $C_{\text{inf}}(f, \varepsilon) \geq \frac{\Omega_f}{1+m_\varepsilon} S_{16}(f, \varepsilon)$. This concludes the proof. \blacksquare

4. Special Case: Noisy Evaluations of f (a.k.a. Stochastic Setting). Previously all the environments E that we considered were deterministic. We now assume that the algorithm receives noisy (stochastic and unbiased) evaluations of f , but that for all $t \in \mathbb{N}^*$ it can observe several independent noisy evaluations of $f(x_t)$ and decide the number m_t of them.

More formally, we consider the following variant of the online protocol described in Section 1.1. Let $(\zeta_{t,u})_{t,u \in \mathbb{N}^*}$ be a sequence of independent v -subGaussian random variables.¹² The $\zeta_{t,u}$'s are unknown, but the constant $v > 0$ is assumed to be known to the learner. At each

¹¹We used the fact that $f(\tilde{x}_m) - f(x_\tau^*(E)) \geq f(\tilde{x}_m) - \max(f) \geq -\tilde{\varepsilon}$ for all $1 \leq m \leq M$ (since $\tilde{x}_m \in \mathcal{X}_{\tilde{\varepsilon}}$).

¹²A real-valued random variable X is v -subGaussian if $\mathbb{E}[\exp(\lambda X)] \leq \exp(\lambda^2 v/2)$ for all $\lambda \in \mathbb{R}$.

round $t \in \mathbb{N}^*$, the algorithm A chooses a query point $x_t \in \mathcal{X}$, as well as a number $m_t \geq 1$ of noisy evaluations (instead of α_t). The algorithm incurs a cost equal to m_t . In return, the environment outputs a mini-batch $(y_{t,1}, \dots, y_{t,m_t})$ with m_t components (instead of a single inaccurate evaluation y_t), where $y_{t,u} = f(x_t) + \zeta_{t,u}$ for any $1 \leq u \leq m_t$. Then, just as before, A outputs a recommendation $x_t^* \in \mathcal{X}$ for the maximum of f , together with a (tentative) error certificate $\xi_t \geq 0$. In this setting, the goal is (with high probability) to maximize f with an error certifiably below ε , while minimizing the total number of evaluations of f .

To that end, we reduce the problem to the deterministic setting of Section 2. We consider c.MF-StoOO (Certified Multi-Fidelity Stochastic Optimistic Optimization), which is a mini-batch version of c.MF-DOO and whose pseudo-code is given in Algorithm 4.1 below. In the sequel, we use the same identification between nodes and rounds as before (see Footnote 6).

The intuition behind c.MF-StoOO is the following: for a mini-batch of size m_t , the average $y_t := \frac{1}{m_t} \sum_{u=1}^{m_t} y_{t,u}$ is an unbiased estimate of $f(x_t)$ with a variance bounded by v/m_t . To be in a special case of Section 2, we make the standard deviation $\sqrt{v/m_t}$ comparable to the evaluation accuracy α_t that c.MF-DOO would request, by choosing $m_t \approx v/\alpha_t^2$ and then applying Theorem 2.4 with a cost $c(\alpha) \approx v/\alpha^2$. More precisely, we use a careful weighted union bound on the nodes of the hierarchical partitioning tree. For some desired risk level $\gamma \in (0, 1)$, writing (h_t, i_t) for the node evaluated at time $t \geq 1$ (Line 9 if $t \geq 2$), we take

$$(4.1) \quad m_t = \left\lceil \frac{2v}{\alpha_t^2} \ln \left(\frac{2}{\gamma_{h_t, i_t}} \right) \right\rceil, \text{ with } \gamma_{h,i} = \frac{\gamma}{(h+1)(h+2)K^h} \text{ for } h \in \mathbb{N} \text{ and } i \in \{0, \dots, K^h - 1\}.$$

Note that the weights sum up to $\sum_{h=0}^{+\infty} \sum_{i=0}^{K^h-1} \gamma_{h,i} = \gamma$.

Define the stopping time

$$\tau(f, \varepsilon) := \inf\{t \geq 1 : \xi_t \leq \varepsilon\}.$$

We now bound the total number $\sum_{t=1}^{\tau(f, \varepsilon)} m_t$ of evaluations of f that c.MF-StoOO requests before certifying an ε -maximizer of f . The next high-probability bound (Proposition 4.1) is in terms of the quantity $S_\beta(f, \varepsilon)$ defined in (1.2) with the cost function $c = c_\gamma$ given by

$$(4.2) \quad c_\gamma(\alpha) = \left\lceil \frac{2v}{\alpha^2} \ln \left(\frac{2(h(\alpha) + 1)(h(\alpha) + 2)K^{h(\alpha)}}{\gamma} \right) \right\rceil, \text{ where } h(\alpha) = \frac{\ln(LR/\alpha)}{\ln(1/\delta)}.$$

We recall that $\varepsilon_0 = L \operatorname{diam}(\mathcal{X})$, $m_\varepsilon = \lceil \log_2(\varepsilon_0/\varepsilon) \rceil$, $\varepsilon_{m_\varepsilon} = \varepsilon$ and $\varepsilon_k = \varepsilon_0 2^{-k}$ for $1 \leq k \leq m_\varepsilon - 1$.

Proposition 4.1. *Suppose that $\mathcal{X} \subset \mathbb{R}^d$ is compact, that Assumptions 2.1 and 2.2 hold, and denote by $a > 0$ the same constant as in Theorem 2.4. Then, in the stochastic setting described above, for any known constants $L, v > 0$ and $\gamma \in (0, 1)$, for any L -Lipschitz function $f : \mathcal{X} \rightarrow \mathbb{R}$ with maximizer $x^* \in \mathcal{X}$, and any $\varepsilon \in (0, \varepsilon_0)$, c.MF-StoOO (Algorithm 4.1) satisfies*

$$\mathbb{P} \left(\left[\forall t \geq 1, f(x^*) - f(x_t^*) \leq \xi_t \right] \text{ and } \sum_{t=1}^{\tau(f, \varepsilon)} m_t \leq a S_{\frac{\delta}{3}}(f, \varepsilon) + \left\lceil \frac{2v}{(LR)^2} \ln \left(\frac{4}{\gamma} \right) \right\rceil \right) \geq 1 - \gamma,$$

where the probability is taken over the noise sequence $(\zeta_{t,u})_{t,u \in \mathbb{N}^*}$, and where $S_{\delta/3}(f, \varepsilon)$ is the quantity defined in (1.2) with $\beta = \delta/3$ and $c = c_\gamma$ (see (4.2) above).

Algorithm 4.1 c.MF-StoOO (Certified Multi-Fidelity Stochastic Optimistic Optimization)

-
- Inputs:** \mathcal{X} , K , $(X_{h,i})_{h \in \mathbb{N}, i \in \{0, \dots, K^h - 1\}}$, $(x_{h,i})_{h \in \mathbb{N}, i \in \{1, \dots, K^h - 1\}}$, δ , R , L , v , and γ
- Initialization** Let $t \leftarrow 1$, $(h_1, i_1) \leftarrow (0, 0)$, and $\mathcal{L}_1 \leftarrow \{(0, 0)\}$
- 1: Pick the first query point $x_1 \leftarrow x_{0,0}$, accuracy $\alpha_1 \leftarrow LR$, prior value $\gamma_{0,0} \leftarrow \gamma/2$, and evaluation number $m_1 \leftarrow \left\lceil \frac{2v}{(LR)^2} \ln \left(\frac{4}{\gamma} \right) \right\rceil$
 - 2: Observe the noisy evaluations $(y_{1,u})_{1 \leq u \leq m_1} = (f(x_1) + \zeta_{1,u})_{1 \leq u \leq m_1}$
 - 3: Compute $y_1 = \frac{1}{m_1} \sum_{u=1}^{m_1} y_{1,u}$
 - 4: Output the recommendation $x_1^* \leftarrow x_1$ and certificate $\xi_1 \leftarrow LR$
 - 5: Pick the first node $(h^*, i^*) \leftarrow (0, 0)$
 - 6: **for** $iteration = 1, 2, \dots$ **do**
 - 7: **for each** child $(h^* + 1, j)$ of (h^*, i^*) **do**
 - 8: **if** $\mathcal{X}_{h^*+1,j} \cap \mathcal{X} \neq \emptyset$ **then**
 - 9: Let $t \leftarrow t + 1$, $(h_t, i_t) \leftarrow (h^* + 1, j)$ and $\mathcal{L}_t \leftarrow \mathcal{L}_{t-1} \cup \{(h_t, i_t)\}$
 - 10: Pick the next query point $x_t \leftarrow x_{h_t, i_t}$ and accuracy $\alpha_t \leftarrow LR\delta^{h_t}$
 - 11: Compute the prior value $\gamma_{h_t, i_t} = \gamma / ((h_t + 1)(h_t + 2)K^{h_t})$
 - 12: Pick the number of evaluations $m_t \leftarrow \left\lceil \frac{2v}{\alpha_t^2} \ln \left(\frac{2}{\gamma_{h_t, i_t}} \right) \right\rceil$
 - 13: Observe the noisy evaluations $(y_{t,u})_{1 \leq u \leq m_t} = (f(x_t) + \zeta_{t,u})_{1 \leq u \leq m_t}$
 - 14: Compute $y_t = \frac{1}{m_t} \sum_{u=1}^{m_t} y_{t,u}$
 - 15: Output the recommendation $x_t^* = x_{\tilde{t}}$, with $\tilde{t} \in \arg \max_{1 \leq s \leq t} \{y_s - \alpha_s\}$
 - 16: Output the certificate $\xi_t = y_{h^*, i^*} + LR\delta^{h^*} + \alpha_{h^*, i^*} - (y_{\tilde{t}} - \alpha_{\tilde{t}})$
 - 17: Remove (h^*, i^*) from \mathcal{L}_t
 - 18: Let $(h^*, i^*) \in \arg \max_{(h,i) \in \mathcal{L}_t} \{y_{h,i} + LR\delta^h + \alpha_{h,i}\}$
 - 19: Update the last certificate $\xi_t = y_{h^*, i^*} + LR\delta^{h^*} + \alpha_{h^*, i^*} - (y_{\tilde{t}} - \alpha_{\tilde{t}})$
-

We first comment on the result. Proposition 4.1 implies that, with high probability, the ξ_t 's are valid certificates and the total number of evaluations of f that c.MF-StoOO requests before certifying an ε -maximizer of f is bounded roughly by (combining (1.2) with (4.2), and omitting log factors and some dimension-dependent constants)

$$\frac{v \mathcal{N}(\mathcal{X}_\varepsilon, \frac{\varepsilon}{L})}{\varepsilon^2} + \sum_{k=1}^{m_\varepsilon} \frac{v \mathcal{N}(\mathcal{X}_{(\varepsilon_k, \varepsilon_{k-1}]}, \frac{\varepsilon_k}{L})}{\varepsilon_k^2} \approx v L^d \int_{\mathcal{X}} \frac{dx}{(f(x^*) - f(x) + \varepsilon)^{d+2}},$$

where the sum-integral approximation (which omits multiplicative constants) holds under the following mild geometric condition on \mathcal{X} : there exist constants $r_0 > \text{diam}(\mathcal{X})/2$, $\rho \in (0, 1]$ such that for all $x \in \mathcal{X}$ and $r \in (0, r_0)$, $\text{vol}(B(x, r) \cap \mathcal{X}) \geq \rho \text{vol}(B(x, r))$. This condition roughly states that \mathcal{X} has a non-negligible volume locally everywhere (e.g., we can take $r_0 = 1$ and $\rho = 2^{-d}$ if $\mathcal{X} = [0, 1]^d$ and $\|\cdot\|$ is the sup norm). The proof of this sum-integral approximation follows essentially from [1, Theorem 1], with a direct extension to non-increasing costs.

Therefore, a consequence of Proposition 4.1 is that, under a mild condition on \mathcal{X} , the cost complexity in the stochastic setting is roughly proportional to $\int_{\mathcal{X}} dx / (f(x^*) - f(x) + \varepsilon)^{d+2}$, as conjectured by [1].

To prove Proposition 4.1, we use the following classical lemma, which helps reduce the stochastic setting with mini-batches to the deterministic setting with inaccurate evaluations. (We will later combine this lemma with $y_t - f(x_t) = \frac{1}{m_t} \sum_{u=1}^{m_t} (y_{t,u} - f(x_t)) = \frac{1}{m_t} \sum_{u=1}^{m_t} \zeta_{t,u}$.)

Lemma 4.2. *Let $(\zeta_{t,u})_{t,u \in \mathbb{N}^*}$ be a sequence of independent v -subGaussian random variables for some $v > 0$. Let $\gamma \in (0, 1)$, and let $(\alpha_t)_{t \geq 1}$ and $(h_t, i_t)_{t \geq 1}$ be two predictable¹³ sequences such that, almost surely, $\alpha_t > 0$, $h_t \in \mathbb{N}$ and $i_t \in \{0, \dots, K^{h_t} - 1\}$ for all $t \geq 1$, and $t \geq 1 \mapsto (h_t, i_t)$ being injective. Then, for m_t and $\gamma_{h,i}$ defined as in (4.1), we have*

$$\mathbb{P} \left(\forall t \in \mathbb{N}^*, \left| \frac{1}{m_t} \sum_{u=1}^{m_t} \zeta_{t,u} \right| < \alpha_t \right) \geq 1 - \gamma.$$

Proof. Let \mathcal{F}_t denote the σ -field generated by the random variables $\zeta_{s,u}$, $s \in \{1, \dots, t\}$, $u \in \mathbb{N}^*$. (By convention, \mathcal{F}_0 is the trivial σ -field.) Let $t \geq 1$. Since α_t and m_t are \mathcal{F}_{t-1} measurable, and the $\zeta_{t,u}$, $u \in \mathbb{N}^*$, are independent and v -subGaussian conditionally on \mathcal{F}_{t-1} ,

$$\mathbb{P} \left(\left| \frac{1}{m_t} \sum_{u=1}^{m_t} \zeta_{t,u} \right| \geq \alpha_t \right) = \mathbb{E} \left[\mathbb{P} \left(\left| \frac{1}{m_t} \sum_{u=1}^{m_t} \zeta_{t,u} \right| \geq \alpha_t \mid \mathcal{F}_{t-1} \right) \right] \leq \mathbb{E} \left[2e^{-\frac{m_t \alpha_t^2}{2v}} \right] \leq \mathbb{E} [\gamma_{h_t, i_t}].$$

By a union bound, this yields

$$\mathbb{P} \left(\exists t \in \mathbb{N}^*, \left| \frac{1}{m_t} \sum_{u=1}^{m_t} \zeta_{t,u} \right| \geq \alpha_t \right) \leq \sum_{t=1}^{+\infty} \mathbb{P} \left(\left| \frac{1}{m_t} \sum_{u=1}^{m_t} \zeta_{t,u} \right| \geq \alpha_t \right) \leq \mathbb{E} \left[\sum_{t=1}^{+\infty} \gamma_{h_t, i_t} \right] \leq \gamma,$$

where the last inequality follows by injectivity of $t \geq 1 \mapsto (h_t, i_t)$ and $\sum_{h=0}^{+\infty} \sum_{i=0}^{K^h-1} \gamma_{h,i} = \gamma$. Taking the complementary event concludes the proof. \blacksquare

Proof of Proposition 4.1. We now explain how to treat the problem as a special case of Section 2 (deterministic yet inaccurate evaluations), with the cost function c_γ defined in (4.2) and a well-chosen random environment.

First note that the assumptions of Lemma 4.2 are met, so that with high probability the y_t 's are α_t -close to $f(x_t)$ simultaneously for all $t \geq 1$. More formally, they can be seen as generated by a random environment $(E_t^\omega)_{t \geq 1}$ defined as follows. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be the probability space on which the random variables $(\zeta_{t,u})_{t,u}$ are defined. For any fixed element $\omega \in \Omega$ and any $t \geq 1$, we define the function $E_t^\omega : \mathcal{X} \times \mathbb{R}_+^* \rightarrow \mathbb{R}$ by

$$E_t^\omega(x, \alpha) = \begin{cases} f(x) + \frac{1}{m_t(\omega)} \sum_{u=1}^{m_t(\omega)} \zeta_{t,u}(\omega) & \text{if } \left| \frac{1}{m_t(\omega)} \sum_{u=1}^{m_t(\omega)} \zeta_{t,u}(\omega) \right| \leq \alpha \\ f(x) & \text{otherwise.} \end{cases}$$

Note that the environment $(E_t^\omega)_{t \geq 1}$ lies in $\mathcal{E}(f)$ for each $\omega \in \Omega$. Now, denoting by $\tilde{\Omega} = \left\{ \omega \in \Omega : \forall t \in \mathbb{N}^*, \left| \frac{1}{m_t(\omega)} \sum_{u=1}^{m_t(\omega)} \zeta_{t,u}(\omega) \right| < \alpha_t \right\}$ the event considered in Lemma 4.2, we can see that, for any $\omega \in \tilde{\Omega}$,¹⁴ the value $E_t^\omega(x_t, \alpha_t) = f(x_t) + \frac{1}{m_t} \sum_{u=1}^{m_t} \zeta_{t,u} = \frac{1}{m_t} \sum_{u=1}^{m_t} (f(x_t) + \zeta_{t,u})$

¹³That is, we assume α_t , h_t and i_t to be measurable w.r.t. the subsequence $(\zeta_{s,u})_{1 \leq s \leq t-1, u \in \mathbb{N}^*}$.

¹⁴For the sake of readability, we drop some dependencies in ω .

coincides with $y_t = \frac{1}{m_t} \sum_{u=1}^{m_t} y_{t,u}$ for all $t \geq 1$. We can thus apply Lemma 2.3: for any $\omega \in \tilde{\Omega}$, we have $f(x^*) - f(x_t^*) \leq \xi_t$ for all $t \geq 1$ (ξ_t is a valid certificate).

Furthermore, a call to $E_t^\omega(x_t, \alpha_t)$ requires $m_t = \left\lceil \frac{2v}{\alpha_t^2} \ln \left(\frac{2}{\gamma^{h_t, i_t}} \right) \right\rceil = c_\gamma(\alpha_t)$ noisy evaluations of f , with c_γ defined in (4.2). Putting everything together, for any $\omega \in \tilde{\Omega}$, the behavior of c.MF-StoOO coincides with the behavior of c.MF-DOO against the environment $E_\omega \in \mathcal{E}(f)$. In particular the total cost $\sum_{t=1}^{\tau(f, \varepsilon)} m_t = \sum_{t=1}^{\tau(f, \varepsilon)} c_\gamma(\alpha_t)$ coincides with the cost complexity of c.MF-DOO against E_ω , with the non-increasing cost function c_γ . We can thus use Theorem 2.4: for any $\omega \in \tilde{\Omega}$,

$$\sum_{t=1}^{\tau(f, \varepsilon)} m_t = \sigma(\text{c.MF-DOO}, E^\omega, \varepsilon) \leq aS_{\frac{\delta}{3}}(f, \varepsilon) + c_\gamma(LR).$$

Recalling that $\mathbb{P}(\tilde{\Omega}) \geq 1 - \gamma$ concludes the proof. \blacksquare

Acknowledgements. The authors would like to thank François Bachoc for insightful feedback. This work has benefited from the AI Interdisciplinary Institute ANITI, which is funded by the French “Investing for the Future–PIA3” program under the Grant agreement ANR-19-P3IA-0004. The authors gratefully acknowledge the support of the DEEL project.¹⁵

Appendix A. Missing proofs in Section 3.

Proof of Lemma 3.2. Let $\tau \in \mathbb{N}^*$. For the sake of clarity, we explicitly write the dependencies of the iterates $x_t(E), \alpha_t(E), y_t(E), x_t^*(E), \xi_t(E)$ w.r.t. the environment $E \in \mathcal{E}(f)$. Recall that \mathcal{F}_L denotes the set of all L -Lipschitz functions from \mathcal{X} to \mathbb{R} . Let $g \in \mathcal{F}_L$ be such that

$$(A.1) \quad \forall t \leq \tau, g(x_t(E)) \in [y_t(E) - \alpha_t(E), y_t(E) + \alpha_t(E)].$$

Then there exists an environment $E^g \in \mathcal{E}(g)$ whose interactions with algorithm A yield the same decisions $x_t, \alpha_t, x_t^*, \xi_t$ and observations y_t as those generated with environment E , up to time $t = \tau$. More formally, we define $E^g = (E_t^g)_{t \geq 1}$ by $E_t^g(x, \alpha) = y_t(E)$ if $t \leq \tau, x = x_t(E)$ and $\alpha = \alpha_t(E)$, but $E_t^g(x, \alpha) = g(x)$ otherwise. Note that $E^g \in \mathcal{E}(g)$ by (A.1).

First note that $x_1(E^g) = x_1(E)$ and $\alpha_1(E^g) = \alpha_1(E)$ since both terms are independent of the environment. From our definition of E^g , this implies that the approximation $y_1(E^g) = E_1^g(x_1(E^g), \alpha_1(E^g))$ returned by E^g is equal to $y_1(E)$. Because the observation y_1 that A receives is the same as before, A outputs the same values for x_1^*, ξ_1, x_2 and α_2 , which again implies that $y_2(E^g) = y_2(E)$. By a simple induction argument, we then have that, for all $t = 1, \dots, \tau$, $x_t(E^g) = x_t(E)$, $\alpha_t(E^g) = \alpha_t(E)$, $y_t(E^g) = y_t(E)$, $x_t^*(E^g) = x_t^*(E)$ and $\xi_t(E^g) = \xi_t(E)$. In particular,

$$\xi_\tau(E) = \xi_\tau(E^g) \geq \max(g) - g(x_\tau^*(E^g)) = \max(g) - g(x_\tau^*(E)),$$

where the inequality follows from the definition of a certificate. Since the above lower bound on $\xi_\tau(E)$ is true for all $g \in \mathcal{F}_L$ satisfying condition (A.1), it is also true for their supremum, which proves that $\xi_\tau(E) \geq \text{err}_\tau(A, E)$. \blacksquare

¹⁵<https://www.deel.ai/>

Proof of Lemma 3.3. Let A be a certified algorithm. We can assume without loss of generality that $\sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon) < +\infty$. Let $C > \sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon)$. Then, for any $E \in \mathcal{E}(f)$, by definition of $\sigma(A, E, \varepsilon)$, there exists $\tau \in \mathbb{N}^*$ such that $\sum_{t=1}^{\tau} c(\alpha_t(E)) \leq C$ and $\xi_{\tau}(E) \leq \varepsilon$. By Lemma 3.2 and since A is a certified algorithm, this entails

$$\sum_{t=1}^{\tau} c(\alpha_t(E)) \leq C \text{ and } \text{err}_{\tau}(A, E) \leq \varepsilon.$$

By definition of $C_{\text{inf}}(f, \varepsilon)$, this immediately yields $C_{\text{inf}}(f, \varepsilon) \leq C$. We conclude the proof by letting C go to $\sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon)$. \blacksquare

In order to prove Lemma 3.4, we first need the following intuitive lemma.

Lemma A.1. *Let f be any L -Lipschitz function, $\tau \in \mathbb{N}^*$, and let E^* be the “noiseless” environment $E^* = ((x, \alpha) \mapsto f(x))_{t \geq 1} \in \mathcal{E}(f)$. Then, for any certified algorithm A , the best possible certificate $\text{err}_{\tau}(A, E^*)$ against E^* is bounded from below by $\min\{\min_{t \leq \tau} \alpha_t(E^*), \varepsilon_0/2\}$.*

Proof. Within this proof, we only work with the environment E^* , so we skip all dependencies of $x_t, \alpha_t, y_t, x_t^*, \xi_t$ on E^* . We set $\tilde{\varepsilon} = \min\{\min_{t \leq \tau} \alpha_t, \varepsilon_0/2\}$ and define $g : \mathcal{X} \rightarrow \mathbb{R}$ by $g(x) = \min\{f(x) + \tilde{\varepsilon}, f(x_{\tau}^*) - \tilde{\varepsilon} + L \|x - x_{\tau}^*\|\}$. We will now show that g is compatible with the observations $(y_t)_{t \leq \tau}$ and the accuracies $(\alpha_t)_{t \leq \tau}$, and that $\max(g) - g(x_{\tau}^*) \geq \tilde{\varepsilon}$.

Since g is the minimum of two L -Lipschitz functions, it is also L -Lipschitz. Moreover, for any $t \leq \tau$, on the one hand, $g(x_t) \leq f(x_t) + \tilde{\varepsilon} \leq y_t + \alpha_t$, by definition of g and $\tilde{\varepsilon}$, and the fact that $y_t = f(x_t)$ (recall that we work with the “noiseless” environment E^*).

On the other hand, by L -Lipschitz continuity of f , for all $x \in \mathcal{X}$, $L \|x - x_{\tau}^*\| + f(x_{\tau}^*) - \tilde{\varepsilon} \geq f(x) - \tilde{\varepsilon}$. This implies that for all $x \in \mathcal{X}$, $g(x) \geq \min\{f(x) + \tilde{\varepsilon}, f(x) - \tilde{\varepsilon}\} = f(x) - \tilde{\varepsilon}$. In particular, for all $t \leq \tau$,

$$g(x_t) \geq f(x_t) - \tilde{\varepsilon} \geq y_t - \alpha_t.$$

To sum up, g is an L -Lipschitz function such that $|g(x_t) - y_t| \leq \alpha_t$ for all $1 \leq t \leq \tau$: the algorithm A cannot make the difference between the functions f and g .

Now, let us bound $\max(g) - g(x_{\tau}^*)$ from below to derive a lower bound on $\text{err}_{\tau}(A, E^*)$.

First, $g(x_{\tau}^*) = \min\{f(x_{\tau}^*) + \tilde{\varepsilon}, f(x_{\tau}^*) - \tilde{\varepsilon}\} = f(x_{\tau}^*) - \tilde{\varepsilon}$. Second, let $v \in \mathbb{R}^d$ be such that $\|v\|_2 = 1$ and $x_{\tau}^* + \frac{\tilde{\varepsilon}v}{L} \in \mathcal{X}$. Such a v exists by Lemma B.1 and the facts that $\tilde{\varepsilon} \leq \varepsilon_0/2 = L \cdot \text{diam}(\mathcal{X})/2$ and \mathcal{X} is compact and connected. Then,

$$f\left(x_{\tau}^* + \frac{\tilde{\varepsilon}v}{L}\right) + \tilde{\varepsilon} \geq f(x_{\tau}^*) - L \left\| \frac{\tilde{\varepsilon}v}{L} \right\| + \tilde{\varepsilon} = f(x_{\tau}^*)$$

and

$$f(x_{\tau}^*) - \tilde{\varepsilon} + L \left\| x_{\tau}^* + \frac{\tilde{\varepsilon}v}{L} - x_{\tau}^* \right\| = f(x_{\tau}^*) - \tilde{\varepsilon} + L \left\| \frac{\tilde{\varepsilon}v}{L} \right\| = f(x_{\tau}^*).$$

By definition of g , this entails

$$f(x_{\tau}^*) \leq g\left(x_{\tau}^* + \frac{\tilde{\varepsilon}v}{L}\right) \leq \max(g)$$

Putting everything together, we get

$$\text{err}_\tau(A, E^*) \geq \max(g) - g(x_\tau^*) \geq f(x_\tau^*) - (f(x_\tau^*) - \tilde{\varepsilon}) = \tilde{\varepsilon} = \min \left\{ \min_{t \leq \tau} \alpha_t, \varepsilon_0/2 \right\},$$

which concludes the proof. \blacksquare

Proof of Lemma 3.4. Recall the definition of $C_{\text{inf}}(f, \varepsilon)$, and let $C \in \mathbb{R}$ and $A' \in \mathcal{A}$ be such that for all $E \in \mathcal{E}(f)$, there exists $\tau \in \mathbb{N}^*$ such that $\sum_{t=1}^\tau c(\alpha_t(E)) \leq C$ and $\text{err}_\tau(A', E) \leq \varepsilon$. In particular, for the “noiseless” environment $E^* = ((x, \alpha) \mapsto f(x))_{t \geq 1}$, there exists τ such that $\sum_{t=1}^\tau c(\alpha_t(E^*)) \leq C$ and $\text{err}_\tau(A, E^*) \leq \varepsilon$. Using Lemma A.1, we get that $\min \{ \min_{t \leq \tau} \alpha_t(E^*), \varepsilon_0/2 \} \leq \text{err}_\tau(A, E^*) \leq \varepsilon$, which implies $\min_{t \leq \tau} \alpha_t(E^*) \leq \varepsilon$ (since $\varepsilon_0/2 > \varepsilon$ by assumption). Because c is non-negative and non-increasing, we have:

$$C \geq \sum_{t=1}^\tau c(\alpha_t(E^*)) \geq c \left(\min_{t \leq \tau} \alpha_t(E^*) \right) \geq c(\varepsilon).$$

By definition of $C_{\text{inf}}(f, \varepsilon)$, this entails that $C_{\text{inf}}(f, \varepsilon) \geq c(\varepsilon)$. \blacksquare

Appendix B. Useful lemmas.

We recall two rather classical lemmas, and provide proofs for the convenience of the reader.

Lemma B.1. *Let $\mathcal{X} \subset \mathbb{R}^d$ be a compact connected set with diameter ρ . Then for any $x \in \mathcal{X}$ and $\varepsilon \leq \frac{\rho}{2}$, there exists $v \in \mathbb{R}^d$ with $\|v\| = 1$ such that $x + \varepsilon v \in \mathcal{X}$.*

Proof. Let $x \in \mathcal{X}$ and $0 < \varepsilon \leq \frac{\rho}{2}$ (the result is straightforward if $\varepsilon = 0$). Since ρ is the diameter of \mathcal{X} and the latter is compact, there exist $y, z \in \mathcal{X}$ such that $\|y - z\| = \rho$. Let us show by contradiction that there exists a point $x' \in \mathcal{X}$ for which $\|x - x'\| \geq \varepsilon$. Assume for a moment that for all $x' \in \mathcal{X}$, $\|x - x'\| < \varepsilon$. In that case, we would have $\|y - z\| \leq \|y - x\| + \|x - z\| < 2\varepsilon < \rho$, which is in contradiction with $\|y - z\| = \rho$.

Now that we know that it exists, let $x' \in \mathcal{X}$ be such that $\|x - x'\| \geq \varepsilon$. Because \mathcal{X} is connected, there exists a continuous path $\gamma : t \in [0, 1] \mapsto \gamma(t) \in \mathcal{X}$ such that $\gamma(0) = x$ and $\gamma(1) = x'$. Let g be the function $g : t \in [0, 1] \mapsto \|\gamma(t) - x\| \in \mathbb{R}$. g is a continuous function from $[0, 1]$ to \mathbb{R} , with $g(0) = 0$ and $g(1) \geq \varepsilon > 0$, so according to the intermediate value theorem, there exists $t^* \in [0, 1]$ such that $g(t^*) = \varepsilon$. Taking $x'' = \gamma(t^*)$ and $v = \frac{x'' - x}{\|x'' - x\|}$ solves the problem, because $x + \varepsilon v = x'' \in \mathcal{X}$. \blacksquare

Lemma B.2. *For any bounded set $E \subset \mathbb{R}^d$, and all $0 < r_1 < r_2$, we have*

$$\mathcal{N}(E, r_1) \leq \left(1 + 2 \frac{r_2}{r_1} \right)^d \mathcal{N}(E, r_2).$$

The above lemma is well known and can be found, e.g., in [1, Appendix A] with a slightly weaker statement. We recall the proof for the convenience of the reader. For any $x \in \mathbb{R}^d$ and $r > 0$, we set $B(x, r) = \{u \in \mathbb{R}^d : \|u - x\| \leq r\}$.

Proof. Fix any bounded set $E \subset \mathbb{R}^d$ and $0 < r_1 < r_2$. Consider an r_1 -packing $F = \{x_1, \dots, x_{N_1}\}$ of E , with cardinality $N_1 := \mathcal{N}(E, r_1)$. Let $F_0 = F$. We define a sequence $F_0, F_1, \dots, F_{k_{\text{end}}-1}$ of subsets of F by induction, as follows.

For $k \geq 1$ let \hat{x}_k be any element of F_{k-1} , and define $B_k = F_{k-1} \cap B(\hat{x}_k, r_2)$ and $F_k = F_{k-1} \setminus B_k$. Repeating this procedure, we get an index $k_{\text{end}} \leq N_1$ such that $F_{k_{\text{end}}-1}$ is non-empty while $F_{k_{\text{end}}}$ is empty.

Then, the set $\{\hat{x}_1, \dots, \hat{x}_{k_{\text{end}}}\}$ is an r_2 -packing of E , so that $k_{\text{end}} \leq \mathcal{N}(E, r_2)$. Let us now upper bound N_1 using k_{end} . By construction, the union of the B_k 's contains F , so for all $i \leq N_1$, there exists $k \leq k_{\text{end}}$ such that $x_i \in B_k$, and thus $B(x_i, r_1/2) \subset B(\hat{x}_k, r_2 + r_1/2)$. Therefore,

$$\bigcup_{1 \leq i \leq N_1} B(x_i, r_1/2) \subset \bigcup_{1 \leq k \leq k_{\text{end}}} B(\hat{x}_k, r_2 + r_1/2).$$

Moreover, the N_1 balls $B(x_i, r_1/2)$ are pairwise disjoint, because F is an r_1 -packing of E . By a volumetric argument, we thus get that $(r_1/2)^d N_1 \leq (r_2 + r_1/2)^d k_{\text{end}} \leq (r_2 + r_1/2)^d \mathcal{N}(E, r_2)$. Rearranging terms concludes the proof. \blacksquare

Appendix C. The special case of constant costs.

In this short section we focus on the case of a constant cost $c(\alpha) = 1$. We formalize and prove the following very intuitive fact: when more accurate evaluations come at no additional cost, choosing the best accuracy available is always optimal. To avoid boundary effects, we slightly extend the setting by allowing the learner to choose α_t identically equal to zero.

Lemma C.1. *Let $f : \mathcal{X} \rightarrow \mathbb{R}$ be an L -Lipschitz function, and assume $c : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is the constant function given by $c(\alpha) = 1$ for all $\alpha \geq 0$. Then,*

- $\sigma(A, E, \varepsilon) = \inf\{t \in \mathbb{N}^* : \xi_t(E) \leq \varepsilon\}$;
- the smallest cost complexity (against the worst environment) is achieved by certified algorithms that choose $\alpha_t = 0$ for all $t \in \mathbb{N}^*$.

Proof of Lemma C.1. Let f be an L -Lipschitz function, and $\varepsilon > 0$. Since $c(\alpha) = 1$ for all $\alpha \geq 0$, the definition of $\sigma(A, E, \varepsilon)$ directly yields $\sigma(A, E, \varepsilon) = \inf\{t \in \mathbb{N}^* : \xi_t(E) \leq \varepsilon\}$.

Recall that \mathcal{A} denotes the set of all certified algorithms, and that $\mathcal{E}(f)$ is the set of all environments associated with the function f . As discussed in the introduction, α_t depends on E only via the inaccurate approximations y_1, \dots, y_{t-1} of f . We write $\alpha_t(y_1, \dots, y_{t-1})$ instead of α_t to clarify this dependency when needed. We denote by \mathcal{A}_0 the set of all certified algorithms such that $\alpha_t(y_1, \dots, y_{t-1}) = 0$ for all $t \in \mathbb{N}^*$ against all possible realizations of y_1, \dots, y_{t-1} .

Formally, what we want to prove is

$$\inf_{A' \in \mathcal{A}_0} \sup_{E \in \mathcal{E}(f)} \sigma(A', E, \varepsilon) \leq \inf_{A \in \mathcal{A}} \sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon).$$

Let A be an algorithm in \mathcal{A} , and let $E^* \in \mathcal{E}(f)$ be the environment for which $E_t^*(x, \alpha) = f(x)$ for all $t \in \mathbb{N}^*$, all $x \in \mathcal{X}$ and all $\alpha \geq 0$. Note that

$$(C.1) \quad \sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon) \geq \sigma(A, E^*, \varepsilon)$$

Now, let \tilde{A} be the same algorithm as A , with one difference: whatever the situation and the environment E , for all $t \in \mathbb{N}$, $\tilde{\alpha}_t(y_1, \dots, y_{t-1}) = 0$. Then $\tilde{A} \in \mathcal{A}_0$. Moreover, because of the particularity of E^* , A and \tilde{A} will behave the same way against E^* : the sequences of query

points $(x_t)_{t \in \mathbb{N}^*}$, recommendations $(x_t^*)_{t \in \mathbb{N}^*}$ and certificates $(\xi_t)_{t \in \mathbb{N}^*}$ are the same for A and for \tilde{A} . From this, we get that $\sigma(A, E^*, \varepsilon) = \sigma(\tilde{A}, E^*, \varepsilon)$. Combining with (C.1) yields

$$(C.2) \quad \inf_{A \in \mathcal{A}} \sup_{E \in \mathcal{E}(f)} \sigma(A, E, \varepsilon) \geq \inf_{A \in \mathcal{A}} \sigma(A, E^*, \varepsilon) \geq \inf_{A \in \mathcal{A}_0} \sigma(\tilde{A}, E^*, \varepsilon)$$

Let us now show that

$$(C.3) \quad \inf_{A' \in \mathcal{A}_0} \sigma(A', E^*, \varepsilon) = \inf_{A' \in \mathcal{A}_0} \sup_{E \in \mathcal{E}(f)} \sigma(A', E, \varepsilon),$$

which, combined with (C.2), will conclude the proof.

To see why (C.3) holds, let A' be any certified algorithm with $\alpha_t = 0$ for all $t \in \mathbb{N}^*$, whatever the past observations y_1, \dots, y_{t-1} , and let E , and E' be two environments in $\mathcal{E}(f)$. Then, for all $x \in \mathcal{X}$, $E_t(x, 0) = E'_t(x, 0) = f(x)$, because environments should satisfy $|E_t(x, \alpha) - f(x)| \leq \alpha$ for all $x \in \mathcal{X}$ and $\alpha \geq 0$. Therefore, the behavior of A' against any $E \in \mathcal{E}(f)$ is the same as against E^* which proves (C.3). ■

REFERENCES

- [1] F. BACHOC, T. CESARI, AND S. GERCHINOVITZ, *Instance-dependent bounds for zeroth-order Lipschitz optimization with error certificates*, Advances in Neural Information Processing Systems, 34 (2021), pp. 24180–24192.
- [2] P. L. BARTLETT, V. GABILLON, AND M. VALKO, *A simple parameter-free and adaptive approach to optimization under a minimal local smoothness assumption*, in Algorithmic Learning Theory, PMLR, 2019, pp. 184–206.
- [3] J. F. BONNANS, *Convex and Stochastic Optimization*, Universitext, Springer Cham, 2019.
- [4] C. BOUTTIER, T. CESARI, M. DUCOFFE, AND S. GERCHINOVITZ, *Regret analysis of the Piyavskii-Shubert algorithm for global Lipschitz optimization*, 2020. Preprint available as arXiv:2002.02390.
- [5] S. BOYD AND L. VANDENBERGHE, *Convex optimization*, Cambridge University Press, 2004.
- [6] S. BUBECK, *Convex optimization: Algorithms and complexity*, Foundations and Trends in Machine Learning, 8 (2015), pp. 231–357.
- [7] S. BUBECK, R. MUNOS, G. STOLTZ, AND C. SZEPESVÁRI, *X-armed bandits.*, Journal of Machine Learning Research, 12 (2011).
- [8] S. BUBECK, G. STOLTZ, AND J. Y. YU, *Lipschitz bandits without the Lipschitz constant*, in Algorithmic Learning Theory, J. Kivinen, C. Szepesvári, E. Ukkonen, and T. Zeugmann, eds., Berlin, Heidelberg, 2011, Springer Berlin Heidelberg, pp. 144–158.
- [9] A. D. BULL, *Convergence rates of efficient global optimization algorithms.*, Journal of Machine Learning Research, 12 (2011), pp. 2879–2904.
- [10] E. CONTAL, D. BUFFONI, A. ROBICQUET, AND N. VAYATIS, *Parallel Gaussian process optimization with upper confidence bound and pure exploration*, in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2013, pp. 225–240.
- [11] Y. DANILIN, *Estimation of the efficiency of an absolute-minimum-finding algorithm*, USSR Computational Mathematics and Mathematical Physics, 11 (1971), pp. 261–267.
- [12] K. DEWETTINCK, A. DE VISSCHER, L. DEROO, AND A. HUYGHEBAERT, *Modeling the steady-state thermodynamic operation point of top-spray fluidized bed processing*, Journal of Food Engineering, 39 (1999), pp. 131–143.
- [13] C. FIEGEL, V. GABILLON, AND M. VALKO, *Adaptive multi-fidelity optimization with fast learning rates*, in International Conference on Artificial Intelligence and Statistics, PMLR, 2020, pp. 3493–3502.
- [14] A. I. FORRESTER, A. J. KEANE, AND N. W. BRESSLOFF, *Design and analysis of "noisy" computer experiments*, AIAA journal, 44 (2006), pp. 2331–2339.

- [15] A. I. FORRESTER, A. SÓBESTER, AND A. J. KEANE, *Multi-fidelity optimization via surrogate modelling*, Proceedings of the Royal Society A: mathematical, physical and engineering sciences, 463 (2007), pp. 3251–3269.
- [16] R. GARNETT, *Bayesian optimization*, Cambridge University Press, 2023.
- [17] K. GEISELHART, L. OZOROSKI, J. FENBERT, E. SHIELDS, AND W. LI, *Integration of multifidelity multi-disciplinary computer codes for design and analysis of supersonic aircraft*, in 49th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 2011, p. 465.
- [18] J.-B. GRILL, M. VALKO, AND R. MUNOS, *Black-box optimization of noisy functions with unknown smoothness*, Advances in Neural Information Processing Systems, 28 (2015).
- [19] P. HANSEN, B. JAUMARD, AND S.-H. LU, *On the number of iterations of Piyavskii’s global optimization algorithm*, Mathematics of Operations Research, 16 (1991), pp. 334–350.
- [20] P. HANSEN, B. JAUMARD, AND S.-H. LU, *Global optimization of univariate Lipschitz functions: I. survey and properties*, Mathematical Programming, 55 (1992), pp. 251–272.
- [21] P. HANSEN, B. JAUMARD, AND S.-H. LU, *Global optimization of univariate Lipschitz functions: II. new algorithms and computational comparison*, Mathematical Programming, 55 (1992), pp. 273–292.
- [22] M. HORN, *Optimal algorithms for global optimization in case of unknown Lipschitz constant*, Journal of Complexity, 22 (2006), pp. 50–70.
- [23] D. HUANG, T. T. ALLEN, W. I. NOTZ, AND R. A. MILLER, *Sequential kriging optimization using multiple-fidelity evaluations*, Structural and Multidisciplinary Optimization, 32 (2006), pp. 369–382.
- [24] P. JAIN AND P. KAR, *Non-convex optimization for machine learning*, Foundations and Trends® in Machine Learning, 10 (2017), pp. 142–336.
- [25] D. R. JONES, C. D. PERTUNEN, AND B. E. STUCKMAN, *Lipschitzian optimization without the Lipschitz constant*, Journal of Optimization Theory and Applications, 79 (1993), pp. 157–181.
- [26] D. R. JONES, M. SCHONLAU, AND W. J. WELCH, *Efficient global optimization of expensive black-box functions*, Journal of Global optimization, 13 (1998), pp. 455–492.
- [27] K. KANDASAMY, G. DASARATHY, J. OLIVA, J. SCHNEIDER, AND B. PÓCZOS, *Multi-fidelity Gaussian process bandit optimisation*, Journal of Artificial Intelligence Research, 66 (2019), pp. 151–196.
- [28] K. KANDASAMY, G. DASARATHY, J. B. OLIVA, J. SCHNEIDER, AND B. PÓCZOS, *Gaussian process bandit optimisation with multi-fidelity evaluations*, Advances in Neural Information Processing Systems, 29 (2016).
- [29] K. KANDASAMY, G. DASARATHY, B. PÓCZOS, AND J. SCHNEIDER, *The multi-fidelity multi-armed bandit*, Advances in Neural Information Processing Systems, 29 (2016).
- [30] R. KLEINBERG, A. SLIVKINS, AND E. UPFAL, *Multi-armed bandits in metric spaces*, in Proceedings of the fortieth annual ACM symposium on Theory of computing, 2008, pp. 681–690.
- [31] R. KLEINBERG, A. SLIVKINS, AND E. UPFAL, *Bandits and experts in metric spaces*, Journal of the ACM (JACM), 66 (2019), pp. 1–77.
- [32] H. J. KUSHNER, *A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise*, Journal of Basic Engineering, 86 (1964), pp. 97–106.
- [33] L. LE GRATIET, *Multi-fidelity Gaussian process regression for computer experiments*, PhD thesis, Université Paris-Diderot-Paris VII, 2013.
- [34] A. LOCATELLI AND A. CARPENTIER, *Adaptivity to smoothness in X-armed bandits*, in Conference on Learning Theory, PMLR, 2018, pp. 1463–1492.
- [35] C. MALHERBE AND N. VAYATIS, *Global optimization of Lipschitz functions*, in International Conference on Machine Learning, PMLR, 2017, pp. 2314–2323.
- [36] J. MOČKUS, *On bayesian methods for seeking the extremum*, in Optimization Techniques IFIP Technical Conference Novosibirsk, G. I. Marchuk, ed., Springer Berlin Heidelberg, 1975, pp. 400–404.
- [37] R. MUNOS, *Optimistic optimization of a deterministic function without the knowledge of its smoothness*, Advances in neural information processing systems, 24 (2011).
- [38] R. MUNOS ET AL., *From bandits to Monte-Carlo Tree Search: The optimistic principle applied to optimization and planning*, Foundations and Trends® in Machine Learning, 7 (2014), pp. 1–129.
- [39] Y. NESTEROV, *Introductory lectures on convex optimization: a basic course*, vol. 87 of Applied Optimization, Springer US, 2004.
- [40] B. PEHERSTORFER, K. WILLCOX, AND M. GUNZBURGER, *Survey of multifidelity methods in uncertainty propagation, inference, and optimization*, Siam Review, 60 (2018), pp. 550–591.

- [41] A. G. PEREVOZCHIKOV, *The complexity of the computation of the global extremum in a class of multi-extremum problems*, USSR Computational Mathematics and Mathematical Physics, 30 (1990), pp. 28–33.
- [42] V. PICHENY, D. GINSBOURGER, AND Y. RICHET, *Noisy expected improvement and on-line computation time allocation for the optimization of simulators with tunable fidelity*, in 2nd International Conference on Engineering Optimization, 2010, p. 10 p.
- [43] V. PICHENY, T. WAGNER, AND D. GINSBOURGER, *A benchmark of kriging-based infill criteria for noisy optimization*, Structural and multidisciplinary optimization, 48 (2013), pp. 607–626.
- [44] S. PIYAVSKII, *An algorithm for finding the absolute extremum of a function*, USSR Computational Mathematics and Mathematical Physics, 12 (1972), pp. 57–67.
- [45] P. Z. G. QIAN AND C. F. J. WU, *Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments*, Technometrics, 50 (2008), pp. 192–204.
- [46] R. SEN, K. KANDASAMY, AND S. SHAKKOTTAI, *Multi-fidelity black-box optimization with hierarchical partitions*, in International conference on machine learning, PMLR, 2018, pp. 4538–4547.
- [47] R. SEN, K. KANDASAMY, AND S. SHAKKOTTAI, *Noisy blackbox optimization using multi-fidelity queries: A tree search approach*, in The 22nd international conference on artificial intelligence and statistics, PMLR, 2019, pp. 2096–2105.
- [48] X. SHANG, E. KAUFMANN, AND M. VALKO, *General parallel optimization without a metric*, in Proceedings of the 30th International Conference on Algorithmic Learning Theory, A. Garivier and S. Kale, eds., vol. 98 of Proceedings of Machine Learning Research, PMLR, 22–24 Mar 2019, pp. 762–788.
- [49] B. O. SHUBERT, *A sequential method seeking the global maximum of a function*, SIAM Journal on Numerical Analysis, 9 (1972), pp. 379–388.
- [50] A. SLIVKINS ET AL., *Introduction to multi-armed bandits*, Foundations and Trends® in Machine Learning, 12 (2019), pp. 1–286.
- [51] J. SONG, Y. CHEN, AND Y. YUE, *A general framework for multi-fidelity Bayesian optimization with Gaussian processes*, in The 22nd International Conference on Artificial Intelligence and Statistics, PMLR, 2019, pp. 3158–3167.
- [52] J. C. SPALL, *Introduction to stochastic search and optimization: estimation, simulation, and control*, Wiley-Interscience, 2003.
- [53] N. SRINIVAS, A. KRAUSE, S. M. KAKADE, AND M. W. SEEGER, *Gaussian process optimization in the bandit setting: No regret and experimental design*, in Proceedings of the 27th International Conference on Machine Learning, J. Fürnkranz and T. Joachims, eds., Omnipress, 2010, pp. 1015–1022.
- [54] N. SRINIVAS, A. KRAUSE, S. M. KAKADE, AND M. W. SEEGER, *Information-theoretic regret bounds for Gaussian process optimization in the bandit setting*, IEEE Transactions on information theory, 58 (2012), pp. 3088–3095.
- [55] G. SUN, G. LI, S. ZHOU, W. XU, X. YANG, AND Q. LI, *Multi-fidelity optimization for sheet metal forming process*, Structural and Multidisciplinary Optimization, 44 (2011), pp. 111–124.
- [56] M. VALKO, A. CARPENTIER, AND R. MUNOS, *Stochastic simultaneous optimistic optimization*, in International Conference on Machine Learning, PMLR, 2013, pp. 19–27.
- [57] J. WANG, D. BASU, AND I. TRUMMER, *Procrastinated tree search: Black-box optimization with delayed, noisy, and multi-fidelity feedback*, in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, 2022, pp. 10381–10390.