Math. Stat. Learn. 8 (2025), 165–192
DOI 10.4171/MSL/54

A simple and improved algorithm for noisy, convex, zeroth-order optimisation

Alexandra Carpentier

Abstract. In this paper, we study the problem of noisy, convex, zeroth-order optimisation of a function f over a bounded convex set $\bar{\mathcal{X}} \subset \mathbb{R}^d$. Given a budget n of noisy queries to the function f that can be allocated sequentially and adaptively, our aim is to construct an algorithm that returns a point $\hat{x} \in \bar{\mathcal{X}}$ such that $f(\hat{x})$ is as small as possible. We provide a conceptually simple method inspired by the textbook centre of gravity method, but adapted to the noisy and zeroth-order setting. We prove that this method is such that the $f(\hat{x}) - \min_{x \in \bar{\mathcal{X}}} f(x)$ is of smaller order than $d^2/\sqrt{n}$ up to poly-logarithmic terms. We slightly improve upon literature preceding this work, where the best-known rate was in Lattimore (2019) and was of order $d^{2.5}/\sqrt{n}$, albeit for a more challenging problem – yet in the literature contemporaneous to our work, the remarkable work of Fokkema et al. (2024) attains the faster rate of $d^{1.5}/\sqrt{n}$ under mild conditions on $\bar{\mathcal{X}}$. Our main contribution is, however, conceptual, as we believe that our algorithm and its analysis bring novel ideas and are significantly simpler than the existing approaches.

1. Introduction

We consider in this paper the setting of convex, noisy, zeroth-order optimisation. For $d \geq 1$, consider a bounded convex set $\bar{\mathcal{X}} \subset \mathbb{R}^d$ with non-zero volume, and consider a convex function $f : \bar{\mathcal{X}} \to [0, 1]$.

We consider a sequential setting with fixed horizon $n \in \mathbb{N} \setminus \{0\}$. At each time $t \leq n$, the learner chooses a point $x_t \in \bar{\mathcal{X}}$ and observes a noisy observation $y_t \in [0, 1]$ such that

$$\mathbb{E}[y_t | (x_i, y_i)_{i<t}, x_t] = f(x_t).$$

In this work, we will study the problem of optimising the function f in the sequential game described above; namely, after the budget n has been fully used by the learner, she has to predict a point $\hat{x}$ – based on all her observations $(x_t, y_t)_{t \leq n}$ – and her aim will be to estimate the minimum for the function f. Her performance for this

Mathematics Subject Classification 2020: 62L12.
Keywords: sequential learning, convex optimisation, minimax statistics.

task will be measured through the following (simple) *regret*:

$$f(\hat{x}) - \inf_{x \in \bar{\mathcal{X}}} f(x),$$

namely, the difference between the true infimum of f and f evaluated at $\hat{x}$.

This setting, known as convex, noisy, zeroth-order optimisation, is related to two popular settings: first-order optimisation – where the learner has access to noisy evaluations of the (sub-)gradient of f – and noiseless zeroth-order optimisation – where the noise $\varepsilon_t = y_t - f(x_t)$ is equal to 0. We refer the reader to [7, 10, 14, 18, 23], among others, for books and surveys on these topics. Unfortunately, a naive application of methods crafted for the two aforementioned topics to the problem of noisy, zeroth-order optimisation typically provides poor results, as the noise present in the evaluations of the function perturbs significantly the learning process, and, e.g., makes attempts of computing (sub-)gradients of f difficult – see, e.g., [1] for a precise discussion on this topic.

In the case where $d = 1$, optimal algorithms, however, exist for this problem since a long time – see [23] for a survey – and are related to dichotomic search. The optimal regret in this case is of order $n^{-1/2}$ up to polylogarithmic terms. An important question that remained open for a long time was on whether the minimax regret was also scaling with $n^{-1/2}$ in higher dimension, and on whether its dependence in the dimension d should be exponential or polynomial. A first groundbreaking work on this topic can be found in [23, Chapter 9], where they provide a complex algorithm whose regret can be bounded uniformly, with high probability, as $\frac{\text{poly}(d)}{\sqrt{n}}$, proving that it is possible to have an algorithm whose regret depends actually only polynomially on d, which was revisited in [1]. This gave rise to a sequence of works, mostly in the related, more challenging setting where one aims at minimising the cumulative regret[1] – see sometimes with concrete proposed algorithm as in [11] or sometimes non-constructive but proving the existence of an algorithm with given properties [17]. The exponent of the polynomial in d has been successively reduced through this stream of works. Before our work, the best-known bound was proven in [17], and for a more challenging problem (cumulative regret, adversarial setting). However, their results would translate in our setting in a regret of order (up to logarithmic terms) $\frac{d^{2.5}}{\sqrt{n}}$. However, note that a remarkable contemporaneous work [13] to this paper manages, with a completely different method, to reach the faster rate of $d^{1.5}/\sqrt{n}$, under mild conditions on $\bar{\mathcal{X}}$. This has to be compared to the best lower bound, derived for this problem, which

[1] In these works, the aim is to minimise the sum of collected samples – i.e., sample as often as possible close to the minimum. They also often consider the challenging adversarial setting. Note that upper bounds in this setting yield upper bounds for our simpler setting, which can be proven through straightforward derivations.

is of order $\frac{d}{\sqrt{n}}$, and which can be proven over the smaller class of linear functions using almost the same proof as[2] in [20, Chapter 24, Theorem 24.2]; see also [5] for a proof for the simple regret and [25] for a bound in a slightly different setting (different noise scaling). This highlights the fact that a gap remains in this setting. In parallel, another stream of literature has been devoted to studying the effect of additional shape constraints, in particular strong convexity and smoothness – see, e.g., [3, 5, 6, 15, 25] – under which a regret of order $\frac{d}{\sqrt{n}}$ is achievable[3], and minimax optimal [5]. Note, however, that strong convexity is a very strong assumption that has important consequences – in particular, when combined to a smoothness assumption, it essentially implies that the shape of the level sets of f is close to a ball. To complement this short literature review, see rather [18] for an excellent very recent survey on these topics – see in particular [18, Section 2.3] for a recent overview of the state of the art in these problems.

In this paper, we provide a simple algorithm for the problem described above. We prove that with high probability and up to polylogarithmic terms depending on the probability, the budget, the dimension, and the diameter of $\mathcal{X}$, the regret is uniformly bounded as $\frac{d^2}{\sqrt{n}}$. This slightly improves over the best-known bound for this problem through work anterior to this paper[4], yet worse than the rate of $d^{1.5}/\sqrt{n}$ obtained by [18] in a contemporaneous work. The main strength of our work, though, is the conceptual simplicity of the proposed algorithm and also its simple analysis. Indeed, our algorithm is an adaptation of the textbook centre of gravity method [21, 24], namely, a specific kind of dichotomic search, combined with an estimator of the gradient on a well-chosen proxy of f, at a well-chosen point. The closest related work in the noisy setting can be found in [19], where they adapt the related ellipsoid method – albeit with a worst rate. The surrogate that they use is different from ours, which is more related to the one in [12], and is based on smoothing the function in a small neighbourhood. However, the size of this neighbourhood is not taken the same in our papers, and some fundamental arguments in terms of cutting directions are therefore different, which also explains the improved rate in this paper with respect to [19].

In Section 2, we present additional notation, as well as some preliminary results regarding these proxies of f, and also on estimating their values and gradients. In Section 3, we provide the main algorithm and the upper bound on its regret. All proofs are in the appendix and are significantly commented on for clarity.

[2] They prove it for the cumulative regret, but almost the same proof can be used for the simple regret.

[3] Note that all these papers do not propose estimators in the bandit setting since the queries are allowed to be outside the constraint set. The rate $d/\sqrt{n}$ (up to logarithmic terms) in the bandit setting with noise was obtained in [4].

[4] Yet does not answer the open question on what is the minimax rate in this setting.

2. Preliminary results and notation

Write $(e_1, \ldots, e_d)$ for the canonical basis of $\mathbb{R}^d$. Write also for any Borelian set $\mathcal{S} \subset \mathbb{R}^d$, $\mathrm{vol}(\mathcal{S})$ for the volume of this set (i.e., its measure according to the Lebesgue measure), and $\mathrm{conv}(\mathcal{S})$ for its convex hull. Let $p \geq 1$, for $R \geq 0$ and $x \in \mathbb{R}^d$, and write $\mathbb{B}_p(x, R)$ for the d-th dimensional l_p ball of radius R and centre x. We also write $\mathbb{B}_2(R) = \mathbb{B}_2(0, R)$ and $\mathbb{S}_2(R)$ for the l_2 sphere of centre 0 and radius R. For technical reasons, we will extend the definition of f over $\mathbb{R}^d$, and write that for $x \notin \bar{\mathcal{X}}$, $f(x) = +\infty$ – and we state by convention that when we sample a point $x_t \notin \bar{\mathcal{X}}$, we obtain $y_t = +\infty$. We will say by convention that f is convex on $\mathbb{R}^d$, as it is convex on $\bar{\mathcal{X}}$, and prolongated by $+\infty$ outside $\bar{\mathcal{X}}$.

In what follows, we will consider some well-chosen proxies of f which we will use in our algorithm. These proxies will be such that one can estimate in a "natural" way these proxies, as well as their gradients. We will study conditions under which these proxies have good properties. We follow here the natural idea – see [23] to the best of our knowledge for zeroth-order optimisation, and studied more generally in [3, 12] – of considering a proxy of f through smoothing in a neighbourhood around each point. We will, however, adapt this neighbourhood to some ambient convex set, as discussed below – and this adaptation is key for our algorithm later. In what follows, we first describe the proxies of f that we will consider, and provide a condition under which the gradients of these proxies are informative regarding f itself. We then explain how we can estimate these proxies and their gradient through noisy evaluations of f.

2.1. Smoothed functional notation and results on smoothed convex functions

Consider a Lebesgue measurable subspace $\mathcal{S} \subset \bar{\mathcal{X}}$ of non-zero volume. In what follows, we write $\mathcal{U}_{\mathcal{S}}$ for the uniform distribution on $\mathcal{S}$. For a distribution $\mathcal{L}$, we write $\mathbb{P}_{X\sim\mathcal{L}}$, $\mathbb{E}_{X\sim\mathcal{L}}$, $\mathbb{V}_{X\sim\mathcal{L}}$ for probability, expectation, and variance according to $X \sim \mathcal{L}$. We also write $\mathbb{P}_{\mathcal{L}}$, $\mathbb{E}_{\mathcal{L}}$, $\mathbb{V}_{\mathcal{L}}$ for probability, expectation, and variance according to the distribution $\mathcal{L}$.

Consider a convex subspace $\mathcal{X} \subset \bar{\mathcal{X}}$ of non-zero volume. We can define its barycenter as

$$\mu_{\mathcal{X}} = \mathbb{E}_{X\sim\mathcal{U}_{\mathcal{X}}} X,$$

and its covariance matrix as

$$\Sigma_{\mathcal{X}} = \mathbb{V}_{X\sim\mathcal{U}_{\mathcal{X}}} X.$$

Since $\mathcal{X}$ has non-zero volume, note that $\Sigma_{\mathcal{X}}$ is invertible.

Write $F_{\mathcal{X}}$ for the linear transformation

$$F_{\mathcal{X}} : x \to \frac{1}{\sqrt{d}} \Sigma_{\mathcal{X}}^{-1/2}(x - \mu_{\mathcal{X}}).$$

Note that the convex set $\mathcal{Z}^{\mathcal{X}} = F_{\mathcal{X}}(\mathcal{X})$ is in isotropic position[5] renormalised by $d^{-1/2}$. Write also $\bar{\mathcal{Z}}^{\mathcal{X}} = F_{\mathcal{X}}(\bar{\mathcal{X}})$, and $z^* = F_{\mathcal{X}}(x^*)$.

Define for any $z \in \mathbb{R}^d$

$$g^{\mathcal{X}}(z) = f(F_{\mathcal{X}}^{-1}(z)) = f(\sqrt{d}\Sigma_{\mathcal{X}}^{1/2}(z + \mu_{\mathcal{X}})).$$

Note that $g^{\mathcal{X}}$ is convex on $\mathbb{R}^d$ and that also in particular the function f is the same up to a linear transformation as the function $g^{\mathcal{X}}$ – and this linear transformation transforms $\bar{\mathcal{X}}$ in $\bar{\mathcal{Z}}^{\mathcal{X}}$ and x^* into $(z^*)^{\mathcal{X}}$. When no ambiguity arises, we write g for $g^{\mathcal{X}}$, z^* for $(z^*)^{\mathcal{X}}$, $\mathcal{Z}$ for $\mathcal{Z}^{\mathcal{X}}$ and $\bar{\mathcal{Z}}$ for $\bar{\mathcal{Z}}^{\mathcal{X}}$ – and note that $g(z^*) = f^*$.

Define for $c > 0, z \in \mathbb{R}^d$

$$g_c^{\mathcal{X}}(z) = \mathbb{E}_{Z \sim \mathcal{U}_{\mathbb{B}_2(c)}} g(z + Z),$$

with the convention $g_0^{\mathcal{X}}(\cdot) = g^{\mathcal{X}}$. Again, when no ambiguity arises, we write g_c for $g_c^{\mathcal{X}}$. Note that g_c is convex on $\mathbb{R}^d$, and that $g_c \geq g_{c'}$ for any $0 \leq c' \leq c$ here[6]. Note also that, for $c > 0$, g_c is differentiable on $\mathbb{R}^d$, and that by Stokes' theorem, for any $z \in \mathbb{R}^d$,

$$\nabla g_c(z) = \frac{d}{c^2} \mathbb{E}_{Z \sim \mathcal{U}_{\mathbb{S}_2(c)}}[Z g(z + Z)]; \tag{1}$$

see [2, Theorem 5] for a precise reference.

A fundamental property of convex functions is that, for any $z, \tilde{x} \in \mathbb{R}^d$ and any sub-gradient $\nabla g(z)$ at this point, if $g(z) - g(\tilde{z})$ is large, the sub-gradient correlates significantly with $z - \tilde{z}$. Namely,

$$\langle \nabla g(z), z - \tilde{z} \rangle \geq g(z) - g(\tilde{z}).$$

The following lemma is a simple, yet key result for this paper and extends this property to the smoothed function g_c – namely, that if $g(z) - g(\tilde{z})$ is large, the sub-gradient $\nabla g_c(z)$ correlates significantly with $z - \tilde{z}$ – in fact, it holds under the relaxed condition that $g_c(z) - g(\tilde{z})$ is large.

[5]Here, under the convention that $\mathcal{Z}^{\mathcal{X}}$ in isotropic position means that a uniform distribution on it has mean 0 and an identity covariance matrix.

[6]Since by convexity, for any $e \in \mathbb{R}^d : \|e\|_2 \leq 1$,

$$f(\mu + ce) + f(\mu - ce) \geq f(\mu + c'e) + f(\mu - c'e).$$

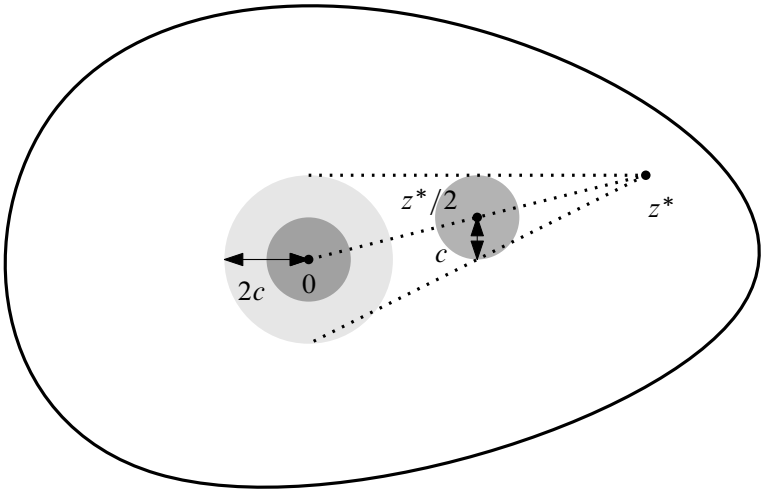

Figure 1. By convexity of f, we know that for any point $z \in \mathbb{B}_2(2c)$: $f(z) - f(z^*) \geq 2[f[(z^* + z)/2] - f(z^*)]$. Integrating both sides over the uniform measure on $\mathbb{B}_2(2c)$ leads to $g_{2c}(z) - g(z^*) \geq 2[g_c[(z^* + z)/2] - g(z^*)]$. If $g_{2c}(0) - g(z^*)$ is not too large when compared to $g_c(0) - g(z^*)$, then $g_c[(z^* + z)/2]$ will be suitably upper bounded by something of order $g_c(0) - g(z^*)$.

Lemma 2.1. *Let $c > 0$ and $z, \tilde{z} \in \mathbb{R}^d$. If $g_{2c}(z) - g_c(z) \leq 2^{-2}[g_c(z) - g(\tilde{z})]$, then*

$$\langle \nabla g_c(z), z - \tilde{z} \rangle \geq \frac{3}{4}[g_c(z) - g(\tilde{z})].$$

The proof of this lemma is in Appendix 6.4 and illustrated in Figure 1. It implies in particular that if $g_{2c}(z) - g(z) \leq 2^{-2}[g(z) - f^*]$ – i.e., if the distance between the proxy $g_{2c}(z)$ and the function $g(z)$ is of smaller order than the optimality gap of $g(z)$ (compared to the minimum f^* of g), then the gradient of the proxy is interesting; namely, $\nabla g_c(z)$ is correlated to $z - z^*$, with a correlation larger than said optimality gap. In other words, the properties of $\nabla g_c(z)$ are similar to those of a sub-gradient $\nabla g(z)$, when it comes to the minimal correlation to the direction of the minimum.

2.2. Estimators of the function and of the gradient of smoothed convex functions

Consider now $z \in \overline{\mathcal{X}}$ and, resp., $Z_1^{(b)}, \ldots, Z_N^{(b)} \sim_{i.i.d.} \mathcal{U}_{\mathbb{B}_2(c)}$ and $Z_1^{(s)}, \ldots, Z_N^{(s)} \sim_{i.i.d.} \mathcal{U}_{\mathbb{S}_2(c)}$ for points sampled, respectively, uniformly in the ball of centre 0 and radius c, and in the sphere of centre 0 and radius c. Assume that we observe independent noisy observations of the function f at the points $F_{\mathcal{X}}^{-1}(z + Z_1^{(k)}), \ldots, F_{\mathcal{X}}^{-1}(z + Z_N^{(k)})$, where $k \in \{b, s\}$ – i.e., equivalently, we observe independent noisy observations of

the function g at the points $z + Z_1^{(k)}, \dots, z + Z_N^{(k)}$ – that is, we write

$$(\tilde{y}_t^{(k)})_{t \le N},$$

where the $\tilde{y}_t^{(k)} \in [0,1]$ are such that $\mathbb{E}[\tilde{y}_t^{(k)} | F_{\mathcal{X}}^{-1}(z + Z_t^{(k)}) = x] = f(x)$ and such that $\tilde{y}_t^{(k)}$, knowing that $F_{\mathcal{X}}^{-1}(z + Z_t^{(k)})$ is independent of the past observations.

Define

$$\hat{g}_c(z) = \frac{1}{N} \sum_{i=1}^{N} \tilde{y}_i^{(b)}, \tag{2}$$

and

$$\widehat{\nabla g_c}(z) = \frac{d}{c^2 N} \sum_{i=1}^{N} Z_i^{(s)} \tilde{y}_i^{(s)}. \tag{3}$$

Set now

$$\eta_{\mathrm{conc}}(1/\delta) = 4\sqrt{\log(2/\delta)};$$

the following lemma provides a concentration result for both the estimator of the function and the estimator of the gradient.

Lemma 2.2. *Let $c \ge 0$, $z \in \overline{\mathcal{Z}}$ such that $\mathbb{B}_2(z,c) \subset \overline{\mathcal{Z}}$ and $u \in \mathbb{R}^d$. With probability larger than $1-\delta$,*

$$|\hat{g}_c(z) - g_c(z)| \le \frac{\eta_{\mathrm{conc}}(1/\delta)}{\sqrt{N}},$$

and if $N \ge d \log(2/\delta)$,

$$|\langle \widehat{\nabla g_c}(z) - \nabla g_c(z), u \rangle| \le \eta_{\mathrm{conc}}(1/\delta) \|u\|_2 \frac{\sqrt{d}}{c\sqrt{N}}.$$

The proof of this lemma is in Appendix 6.4 and is based on very standard concentration arguments. The study of related estimators was first formulated to the best of our knowledge in [23] and then refined in [2, 6] (among others). Note, however, that, in these works, the proximity of these estimators to g or its gradient is controlled, under smoothness assumptions. This is not the approach that we take here, as we do not work under additional smoothness assumptions – so that the proxies g_c can be arbitrarily far from g and its gradient in many points.

3. Algorithm

Our algorithm is an adaptation of the centre of gravity method to the noisy, non-differentiable case. In the classical centre of gravity method, we iteratively refine the convex set where the minimum lies – starting with $\overline{\mathcal{X}}$ – at each step. More precisely,

assume that we are given a convex set $\mathcal{X} \subset \overline{\mathcal{X}}$ at a given iteration. We refine it by computing the gradient $\nabla f(x)$ of f at the centre of gravity x of $\mathcal{X}$ and updating $\mathcal{X}$ to $\mathcal{X} \cap \{u : \langle \nabla f(x), u - x \rangle \leq 0\}$. This method is efficient as

(1) by convexity of f, x^* remains in $\mathcal{X}$ for any iteration, and

(2) a fundamental property of convex sets is that if we separate them into two parts by any hyperplane going through their centre of gravity, both parts of the convex set have approximately the same volume.

In our case, we do not have access to ∇f, but only to noisy evaluations of f. The idea behind our method is to estimate instead the gradient of another function – namely, of $g_c^{\mathcal{X}}$ for a well-chosen c, i.e., a linear transformation of f that is also smoothed. We have seen in Lemma 2.2 that this task can be performed efficiently. However, this gradient might be quite different from any sub-gradient of f. We have, however, seen in Lemma 2.1 that, under the condition that $g_{2c}(0) - g_c(0)$ is small enough, the gradient of g_c has the nice property that it correlates positively to $F_{\mathcal{X}}(x) - F_{\mathcal{X}}(\tilde{x})$ for any $\tilde{x}$ such that $f(\tilde{x})$ is small enough. So that $F_{\mathcal{X}}^{-1}(\nabla g_c(x))$ could be used instead of the gradient of f in the centre of gravity method.

The only problem remaining is that the centre of gravity is not necessarily such that $g_{2c}(0) - g_c(0)$ is small. In order to circumvent this, we find another point z that has this property, and is such that $\|z\|_2$ is small enough so that cutting $\mathcal{X}$ in $F_{\mathcal{X}}^{-1}(z)$ provides similar volume guarantees than cutting it in x.

The main algorithm **BarAlg** described below in Algorithm 3 is therefore using two recursive sub-routines:

- it first calls an iterative sub-routine **Cut** described in Algorithm 2 that *cuts the current set $\mathcal{X}$ in two*, until the budget is elapsed,
- this routine calls another sub-routine **FCP** described in Algorithm 1, which finds a *good cutting point*, as explained above.

In what follows, all our sub-routines sample several points in a small ellipsoid around the barycenter of gravity x This small ellipsoid will be $F_{\mathcal{X}}^{-1}(\mathbb{B}_2(c))$, where c will be taken smaller than $1/\sqrt{d}$. Note that by the classical KLS lemma – see Proposition 6.5 – we know that all these points are in $\mathcal{X}$, and therefore in $\overline{\mathcal{X}}$, by definition of $F_{\mathcal{X}}$ as being the transformation that puts $\mathcal{X}$ in renormalised isotropic position.

3.1. Part 1: Finding a cutting point

We first describe the sub-routine that identifies a good candidate for a cutting point. This sub-routine acts in the linear transformation $\mathcal{Z}^{\mathcal{X}}$ of $\mathcal{X}$ through $F_{\mathcal{X}}$. Starting from z_0, we want to find using a budget of order N – up to multiplicative polylog

terms – a point z such that

- either $g_{2c}(z) - g_c(z) \leq 2^{-3}(g_c(z) - f^*)$, or $g_c(z)$ is small (say, smaller than $1/\sqrt{N}$ up to multiplicative polylog terms),
- $\|z - z_0\|_2$ is of smaller order than c up to multiplicative polylog terms,

provided that such a point exists. In this way, we ensure that this point would satisfy the condition of Lemma 2.1, or be such that $g(z)$ is small enough, and also that it is not too far from z_0.

Assume that we are given a set $\mathcal{X}$ and $c > 0$. For $N \geq 1$, let $I_N = \log_2(N) + 1$ and $M_{\mathbf{FCP}}(N) = \log(2N)/\log(17/16) + 1$. The recursive algorithm **FCP** (for *F*inding a *C*utting *P*oint) takes as parameters a candidate for a cutting point $z \in \mathbb{R}^d$, the current set to be cut $\mathcal{X} \subset \mathbb{R}^d$, a smoothness parameter $c > 0$, a basis number of samples that will be our approximate final budget up to polylog terms $N \in \mathbb{N}$, a counting of the number of recursive rounds performed $s \geq 0$, and a confidence parameter $\delta > 0$. During each run, the algorithm either returns the final cutting point $z \in \mathbb{R}^d$, as well as an estimator of $g(z)$ by $\hat{g}_z$, or calls itself recursively. Note that this sub-routine will require sampling the function f and as it is typically called by another algorithm which operates based on a total budget n, as soon as this budget is elapsed, the algorithm **FCP** terminates returning the current $(z, \hat{g}_z)$. It proceeds in the following steps.

(1) It first samples the function f in $F_{\mathcal{X}}^{-1}(z)$ for N times and estimates $g^{\mathcal{X}}(z)$ by $\hat{g}_z$ as described[7] in equation (2).

(2) For all integers $i \leq I_N$, it samples 2^i points distributed as $z + \mathcal{U}_{\mathbb{B}_2(2c)}$, and we write $(z_j^{(i)})_{i \leq I_N, j \leq 2^i}$ for these points. It samples $2^{-i} N / i^2$ times the function f at $F_{\mathcal{X}}^{-1}(z_j^{(i)})$ and estimate in this way $g^{\mathcal{X}}(z_j^{(i)})$ by $\hat{g}_{z_j^{(i)}}$ as described[8] in equation (2).

(3) If there exists $z_j^{(i)}$ such that

$$\hat{g}_{z_j^{(i)}} - \hat{g}_z \geq \frac{(17/16)^s}{16N} + 4\eta_{\mathrm{conc}}(2^i i^2 M_{\mathbf{FCP}}(N)/\delta)\sqrt{\frac{i^2 2^i}{N}},$$

then it calls $\mathbf{FCP}(z_j^{(i)}, \mathcal{X}, c, N, s+1, \delta)$. Otherwise, it returns $(z, \hat{g}_z)$.

In this way, we evaluate whether, in a radius of $2c$ around z there is a significantly large set of points such that g evaluated in these points is large – i.e., exponentially growing with the number of iterations s. If this is the case, we identify one of these

[7] It is written therein as $\hat{g}_0(z)$.

[8] It is written therein as $\hat{g}_0(z_j^{(i)})$.

points and propose it as the next barycentric candidate. Otherwise, we identify z as a good candidate and return it. The full algorithm is summarized in Algorithm 1

Algorithm 1 FCP

Require: $(z, \mathcal{X}, c, N, s, \delta)$

Ensure: $(z, \hat{g}_z)$ – except if the budget elapses in which case it stops

Sample the function f in $F_{\mathcal{X}}^{-1}(z)$ for N times and estimate in this way $g^{\mathcal{X}}(z)$ by $\hat{g}_z$ as in equation (2)
for all integer $i \leq I_N$ **do**
    Sample 2^i points as $z + \mathcal{U}_{\mathbb{B}_2(2c)}$, and write $(z_j^{(i)})_{i \leq I, j \leq 2^i}$ for these points
    Sample $2^{-i} N / i^2$ times the function f at $F_{\mathcal{X}}^{-1}(z_j^{(i)})$ and estimate in this way $g^{\mathcal{X}}(z_j^{(i)})$ by $\hat{g}_{z_j^{(i)}}$ as in equation (2)
end for
if there exists $z_j^{(i)}$ such that $\hat{g}_{z_j^{(i)}} - \hat{g}_z \geq \frac{(17/16)^s}{16N} + 4\eta_{\text{conc}}(2^i i^2 M_{\mathbf{FCP}}(N)/\delta)\sqrt{\frac{i^2 2^i}{N}}$ **then**
    call $\mathbf{FCP}(z_j^{(i)}, \mathcal{X}, c, N, s+1, \delta)$ for such a (i, j)
else
    return $(z, \hat{g}_z)$
end if

3.2. Part 2: Routine for effectively cutting the space

We now describe the sub-routine that iteratively cuts the space, taking as parameter a convex set $\mathcal{X} \subset \overline{\mathcal{X}}$. It also maintains a current estimation $\hat{x}$ of the minimum. It updates these to $\mathcal{X}', \hat{x}'$. We would like it to satisfy that with high probability:

- the volume of $\mathcal{X}'$ is a fraction of the volume of $\mathcal{X}$, and
- *either* a small ball around the true minimum is in $\mathcal{X}'$, *or* the current estimator of the minimum $\hat{x}$ is already very good, namely, such that $f(\hat{x}) - \min_{x \in \mathcal{X}}$ is smaller than our target simple regret.

Set $N_{\mathbf{Cut}} = \frac{n \log(10/9)}{4d \log(8n^2 d^2)}$, $M_{\mathbf{Cut}} = 5n/N_{\mathbf{Cut}}$, $N_{\mathbf{FCP}}$ such that[9] $N_{\mathbf{FCP}} M_{\mathbf{FCP}}(N_{\mathbf{FCP}}) = N_{\mathbf{Cut}}/4$, and $M_{\mathbf{FCP}} = M_{\mathbf{FCP}}(N_{\mathbf{FCP}})$, and $c = 1/(8e M_{\mathbf{FCP}} \sqrt{d})$. We define the recursive algorithm **Cut** taking as parameters a candidate set $\mathcal{X} \subset \mathbb{R}^d$, a candidate estimator

[9] Looking at the definition of $M_{\mathbf{FCP}}(N_{\mathbf{FCP}})$ and since it is a monotone strictly increasing function, it is clear that such $N_{\mathbf{FCP}}$ exists and is unique and $N_{\mathbf{FCP}}$ is of order $N_{\mathbf{Cut}}$.

of the minimum of f by $\hat{x} \in \mathbb{R}^d$, an estimate of the value of f at this point $\hat{f} \in \mathbb{R}$, and a probability $\delta > 0$. During each run, the algorithm calls itself recursively. Note that this sub-routine will require sampling the function f and as it is typically called by another algorithm which operates based on a total budget n, as soon as this budget is elapsed, the algorithm **Cut** terminates returning the current $\hat{x}$. It proceeds in the following steps.

(1) Run **FCP**$(0, \mathcal{X}, c, N_{\mathbf{FCP}}, 0, \delta)$ and collect $(z, \hat{g}_z)$.

(2) If $\hat{g}_z \leq \hat{f}$, set $\hat{x}' = F_{\mathcal{X}}^{-1}(z)$ and $\hat{f}' = \hat{g}_z$; otherwise, set $\hat{x}' = \hat{x}$ and $\hat{f}' = \hat{f}$.

(3) Compute an estimator $\widehat{\nabla g_c}$ of $\widehat{\nabla g_c^{\mathcal{X}}}(x)$ using $N_{\mathbf{Cut}}$ samples, as described in equation (3).

(4) Set $\mathcal{X}' = \mathcal{X} \cap F_{\mathcal{X}}^{-1}(\{u : \langle u - z, \widehat{\nabla g_c}\rangle \leq 0\})$.

(5) Run **Cut**$(\mathcal{X}', \hat{f}', \hat{x}', \delta)$.

This follows the idea of the centre of gravity method, using a well-chosen cutting point returned by **FCP** and cutting then according to the gradient of a smoothed version of f, and continuing recursively. The full algorithm is summarised in Algorithm 2.

Algorithm 2 Cut

Require: $(\mathcal{X}, \hat{f}, \hat{x}, \delta)$

Ensure: $\hat{x}$ as the budget elapses – after all n samples have been used, it returns the current $\hat{x}$

Run **FCP**$(0, \mathcal{X}, c, N_{\mathbf{FCP}}, 0, \delta)$ and collect $(z, \hat{g}_z)$
if $\hat{g}_z \leq \hat{f}$ **then**
    Set $\hat{x}' = F_{\mathcal{X}}^{-1}(z)$ and $\hat{f}' = \hat{g}_z$
else
    Set $\hat{x}' = \hat{x}$ and $\hat{f}' = \hat{f}$
end if
Compute an estimator $\widehat{\nabla g_c}$ of $\widehat{\nabla g_c^{\mathcal{X}}}(x)$ using $N_{\mathbf{Cut}}$ samples, as in equation (3)
Set $\mathcal{X}' = \mathcal{X} \cap F_{\mathcal{X}}^{-1}\big(\{u : \langle u - z, \widehat{\nabla g_c}\rangle \leq 0\}\big)$
Run **Cut**$(\mathcal{X}', \hat{f}', \hat{x}', \delta)$

3.3. Part 3: Final algorithm

The main algorithm **BarAlg** is finally launched with a total budget n and a confidence parameter $\delta > 0$ and returns an estimator $\hat{x}$ of the minimum. It is basically an application of **Cut** on a reasonable initialisation and proceeds in the following steps.

(1) Sample $N_{\mathbf{FCP}}$ times the function f at $\mu_{\overline{\mathcal{X}}}$ and compute an estimator $\hat{f}$ of $f(\mu_{\overline{\mathcal{X}}})$ as in equation (2) – recalling that $f(\mu_{\overline{\mathcal{X}}}) = g(0)$.

(2) Apply $\mathbf{Cut}(\overline{\mathcal{X}}, \hat{f}, \mu_{\overline{\mathcal{X}}}, \delta)$ and retrieve $\hat{x}$ when the budget is elapsed.

(3) Return $\hat{x}$.

This algorithm is summarised in Algorithm 3.

Algorithm 3 BarAlg

Require: (n, δ)

Ensure: $\hat{x}$ as the budget elapses

1: Sample $N_{\mathbf{FCP}}$ times the function f at $\mu_{\overline{\mathcal{X}}}$ and compute in this way an estimator $\hat{f}$ of $f(\mu_{\overline{\mathcal{X}}})$ as in equation (2)
2: Run $\mathbf{Cut}(\overline{\mathcal{X}}, \hat{f}, \mu_{\overline{\mathcal{X}}}, \delta)$ until the budget is elapsed and retrieve $\hat{x}$
3: **return** $\hat{x}$

The following theorem holds for the output of **BarAlg**.

Theorem 3.1. *Assume that the algorithm* **BarAlg** *launched with a total budget n and a confidence parameter δ returns $\hat{x}$ that is such that with probability larger than $1-\delta$:*

$$
\begin{aligned}
f(\hat{x}) - f^* &\leq \left[2^{16} \eta_{\mathrm{conc}}(M_{\mathbf{FCP}}/\delta) \log(2M_{\mathbf{FCP}}/\delta) \frac{1}{\sqrt{N_{\mathbf{FCP}}}} \right] \\
&\quad \vee \left[32 \eta_{\mathrm{conc}}(10 d M_{\mathbf{Cut}}/\delta) \frac{d}{c\sqrt{N_{\mathbf{Cut}}}} \right] \vee (8/n) \vee [d \log(2/\delta)/N_{\mathbf{Cut}}] \\
&\leq c' \log\left(nd/\delta\right)^2 \times \frac{d^2}{\sqrt{n}} \log(1/\delta)^{3/2},
\end{aligned}
$$

where $c' > 0$ is an absolute constant (independent on $f, \overline{\mathcal{X}}, n, d, \delta$).

This theorem is proved in Section 6.1 and its proof is commented on and explained therein. Up to logarithmic terms, our regret here is of order $d^2/\sqrt{n}$ which slightly improves with respect to an adaptation of the best bound developed before our work in [17] – which is derived for the more challenging problem of adversarial minimisation of the cumulative regret[10], but which could translate in our setting as being of order $d^{2.5}/\sqrt{n}$. Note, however, that a contemporaneous work to ours [13], published approximately at the same time, attains the faster rate of $d^{1.5}/\sqrt{n}$, outperforming our

[10] We believe that our algorithm can be easily modified to accommodate cumulative regret in the stochastic case and have a cumulative regret of order $d^2\sqrt{n}$. We however do not think that it could be easily adapted to the adversarial case.

work. This remarkable work is based on second-order surrogates of the function f, which create successive layers of a lower bound on f, allowing to eliminate large parts of the space. Thus, the improvement in terms of rate of our work is only with respect to past literature, and not with respect to contemporaneous literature, which outperforms us – in a more challenging setting.

In any cases, the main strength of our approach is in terms of our algorithm and proof technique, which are – we believe – significantly simpler than existing results[11]. We hope that these techniques would be refined to develop a tighter understanding of this problem and evolve toward understanding the minimax regret in this problem. Another very interesting open problem is to study the case where the variance of the noise is not considered bounded by 1 allowed to go to 0 – or equivalently if the function is not assumed to be bounded by 1 but by some large constant. In this case, having a bound that depends from the variance of the noise – and interpolating between noisy and noiseless zeroth-order optimisation – is a problem of great interest.

4. Discussion

4.1. Computational complexity and variations around algorithm BarAlg

The computational complexity of the proposed method is an issue here, given that it relies on a centre of gravity method. Each step relies on computing the barycenter of the current convex and on computing the associated variance-covariance matrix – so that it can be put in isotropic position. Interestingly, this can be done in polynomial time – see [22] for a description of an algorithm that enables sampling uniformly over a convex in polynomial time – and was applied to the centre of gravity methods to make them provably polynomial time [8]. Recently, the paper [16] improved the sample complexity of this procedure to about $n^{3.5}$ iterations to compute 2-isotropic position of a well-rounded convex body.

However, there are quite a few variations around the barycenter method which are computationally more efficient: for instance, replacing the barycenter method by a either a classical ellipsoid method or an inscribed ellipsoid method (which is only efficient if $\mathcal{X}$ is a polytope). This has been done very recently for both methods in the new Chapter 9 of [18]; see https://tor-lattimore.com/downloads/cvx-book/cvx.pdf for the up-to-date version containing these additions. The adaptation with a classical ellipsoid method would achieve a regret of order $d^{2.5}/\sqrt{n}$, while the method of inscribed

[11] However, while our algorithm is simple conceptually, it is extensive computationally as it requires an (approximate) computation of barycenters of successive convex sets, which is typically very costly.

ellipsoid achieves a regret of order $d^2/\sqrt{n}$ as in this paper. The main advantage of both of these variations is that their computational complexity is polynomial in d.

5. Modification of algorithm BarAlg that handles the cumulative regret

A not too complicated modification of **BarAlg** can be constructed so that it achieves cumulative regret of order $d^2\sqrt{n}$ up to polylogarithmic terms in the stochastic setting. I sketch the modifications in the construction and in the proof here. First, for simplicity, assume that f^* is known – in which case the gap at any point can be estimated very efficiently by sampling the function.

- The constant $N_{\mathbf{Cut}}$ needs to be changed and cannot be taken fixed anymore – otherwise, we would pay too much cumulative regret if the barycenter or other points we consider as cutting points in **FCP** have a large gap. $N_{\mathbf{Cut}}$ needs to be taken adaptively to the gap Δ between f at the current point considered as being a cutting point and f^*. Intuitively, we will adapt so that $N_{\mathbf{Cut}}$ is of order d^3/Δ^2. $N_{\mathbf{FCP}}$, $M_{\mathbf{FCP}}$ are changed according to their relation to $N_{\mathbf{Cut}}$ while $M_{\mathbf{Cut}}$ remains of order $d\log n$.
- After $d\log n$ cuts are done in the main procedure, the algorithm exploits the point it visited that has smallest measured gap.

In that case, the analysis if the algorithm is not very complicated, as we will only incur at most d^4/Δ regret per possible gap Δ until we have realised the $d\log n$ iterations that are necessary to get a visited point whose gap is smaller than $d^2/\sqrt{n}$. Summing over a logarithmic scale of all possible gaps up to Δ of order $d^2/\sqrt{n}$, we get the result.

Now, assume that f^* is not known. In this case, things are more complicated as we cannot estimate the actual gap in an efficient way. However, we do not need to know f^* exactly, but up to $d^2/\sqrt{n}$ to us the ideas described above, in order to be able to construct a lower bound with precision $d^2/\sqrt{n}$. For this reason, we can use a doubling trick, starting initially with a budget of order d^4, and an initial estimation of f^* as being 0. We then double the budget each time and work with a budget N and set an estimate of f^* as being the estimate of the value of the most sampled point minus $d^2/\sqrt{N}$, until we reach the full budget. This not very elegant algorithm will work using the doubling trick.

Now, of course, everything discussed here only works in the stochastic case, and not at all in the adversarial setting.

6. Proofs of the results in this paper

6.1. Proof of Theorem 3.1

Assume first that $N_{\mathbf{Cut}} \leq d \log(2/\delta)$. Then, by definition of $N_{\mathbf{Cut}}$, it means that $1 \leq d \log(2/\delta)/N_{\mathbf{Cut}}$ so that the bound in Theorem 3.1 is trivially satisfied for any $\hat{x} \in \bar{\mathcal{X}}$. From now on, we therefore restrict to the converse case where $N_{\mathbf{Cut}} \geq d \log(2/\delta)/N_{\mathbf{Cut}}$ – so that the second part of Lemma 2.2 can be applied to gradients constructed with $N_{\mathbf{Cut}}$ points, as we do in our algorithm.

Step 1: Definition of a near-optimal set and lower bound on its volume. Write

$$\mathcal{X}^* = \big\{F^{-1}[(1-(2n)^{-1})z^* + (2n)^{-2}d^{-1/2}e_i],\\ F^{-1}[(1-(2n)^{-1})z^* - (2n)^{-2}d^{-1/2}e_i],\ \forall i \in \{1,\dots,d\}\big\}.$$

Lemma 6.1. *It holds that*

$$\mathcal{X}^* \subset \bar{\mathcal{X}},$$

and also that for any $u \in \mathrm{conv}(\mathcal{X}^*)$, $f(u) - f^* \leq 1/n$, *and*

$$\frac{\mathrm{vol}(\mathrm{conv}(\mathcal{X}^*))}{\mathrm{vol}(\bar{\mathcal{X}})} \geq (8n^2d^2)^{-d},$$

where $\mathrm{conv}(\mathcal{X}^*)$ *is the convex hull of* $\mathcal{X}^*$.

Step 2: Results on FCP. The following result holds for algorithm **FCP**.

Proposition 6.2. *Assume that* $\mathbb{B}_2(z_0, 2M_{\mathbf{FCP}}c) \subset \bar{\mathcal{Z}}^{\mathcal{X}}$. *With probability larger than* $1-4\delta$, $\mathbf{FCP}(z_0, \mathcal{X}, c, N, 0, \delta)$ *returns* z *such that*

- *either*
$$g_{2c}(z) - g_c(z) \leq 2^{-3}(g_c(z) - f^*),$$
or
$$g(z) - f^* \leq 2^{15}\eta_{\mathrm{conc}}(M_{\mathbf{FCP}}/\delta)\log(2M_{\mathbf{FCP}}/\delta)\frac{1}{\sqrt{N}},$$
- $|g(z) - \hat{g}_z| \leq \eta_{\mathrm{conc}}(M_{\mathbf{FCP}}/\delta)/\sqrt{N}$,
- $\|z - z_0\|_2 \leq 2M_{\mathbf{FCP}}c$,
- *the total budget* $T_{\mathbf{FCP}}$ *used to find* z *is smaller than* $4M_{\mathbf{FCP}}N$ *so that* $N \leq T_{\mathbf{FCP}} \leq 4M_{\mathbf{FCP}}N$.

The main idea behind this result is that, on a high-probability event,

- if a point $z_j^{(i)}$ is selected for being a candidate for a cutting point, then it means that $g(z_j^{(i)})$ is larger than a quantity growing exponentially with the number of

iterations s. As the range of g is bounded on $\mathcal{Z}^{\mathcal{X}}$, this means that the number of recursive calls to **FCP** should be logarithmically bounded – hence the bound on $\|z - z_0\|_2$ and the bound on the number of samples used,

- if none of the $z_j^{(i)}$ is selected for being a candidate for a cutting point, then it either means that (i) they are all small, and as they are representative of the average value of g on $\mathbb{B}_2(z, 2c)$, then $g_{2c}(z)$ will be small enough to satisfy our condition in Lemma 2.1, or (ii) that $g(z)$ is already very small.

Step 3: Results on a single run of Cut. We now state the following lemma that describes the high probability behaviour of **Cut**, provided that it is given a reasonable set of parameters. Set

$$B = \left[2^{16}\eta_{\mathrm{conc}}(M_{\mathbf{FCP}}/\delta)\log(2M_{\mathbf{FCP}}/\delta)\frac{1}{\sqrt{N_{\mathbf{FCP}}}}\right] \vee \left[32\eta_{\mathrm{conc}}(2d/\delta)\frac{d}{c\sqrt{N_{\mathbf{Cut}}}}\right] \vee (8/n).$$

Lemma 6.3. *Assume that* **Cut** *is given a convex set* $\mathcal{X} \subset \overline{\mathcal{X}}$, $\hat{x} \in \overline{\mathcal{X}}$, $\hat{f} \in \mathbb{R}$, $\delta > 0$ *such that*

- $|f(\hat{x}) - \hat{f}| \leq \eta_{\mathrm{conc}}(M_{\mathbf{FCP}}/\delta)/\sqrt{N_{\mathbf{FCP}}}$,
- *either* $\mathcal{X}^* \subset \mathcal{X}$, *or* $f(\hat{x}) - f^* \leq B$.

There exists an event of probability larger than $1 - 5\delta$ *such that*

- $\mathcal{X}' \subset \mathcal{X}$ *is convex,*
- $|f(\hat{x}') - \hat{f}'| \leq \eta_{\mathrm{conc}}(M_{\mathbf{FCP}}/\delta)/\sqrt{N_{\mathbf{FCP}}}$,
- *either* $[\mathcal{X}^* \subset \mathcal{X}' \subset \mathcal{X}$ *and* $\mathrm{vol}(\mathcal{X}') \leq \frac{9}{10}\mathrm{vol}(\mathcal{X})]$, *or* $f(\hat{x}') - f^* \leq B$,
- *the total budget* $T_{\mathbf{Cut}}$ *used to run* **Cut** *until the next recursive call of* **Cut** *is such that*

$$N_{\mathbf{FCP}} + N_{\mathbf{Cut}} \leq T_{\mathbf{Cut}} \leq 4M_{\mathbf{FCP}}N_{\mathbf{FCP}} + N_{\mathbf{Cut}}.$$

This lemma ensures that, provided that **Cut** is initialised properly, the convex set $\mathcal{X}'$ obtained after running **Cut** satisfies the following.

- *either* it contains $\mathcal{X}^*$, and its volume is a fraction of the volume of $\mathcal{X}$,
- *or* f measured at the current estimator of the minimum $\hat{x}'$ is already quite small.

The idea behind the proof of this lemma is that whenever $f(\hat{x}) - f^*$ is not too small, then, by Proposition 6.2, **FCP** will return with high probability a cutting point z that satisfies the requirements in Lemma 2.1 – so that $\nabla g_c(z)$ is negatively correlated with $x^* - z$, and can therefore be used to cut the space $\mathcal{X}$. Also, by Proposition 6.2, with high probability, z is such that $\|z\|_2$ is small so that cutting the space according to this approximate centre of gravity still preserves the nice property about exponentially fast volume reduction.

Step 4: Induction on several runs of Cut. Based on this lemma, we proceed by induction over the repeated recursive runs of **Cut** after being called by **BarAlg**, conditioning over the high-probability event of Lemma 6.3, where the conditions for the next run are ensured. Our induction hypothesis H_t as follows on an event ξ_t of probability larger than $1-5t\delta$, if **Cut** is called for the t time, it takes as parameter a convex set $\mathcal{X} \subset \bar{\mathcal{X}}$, $\hat{x} \in \bar{\mathcal{X}}$, $\hat{f} \in \mathbb{R}$ such that

- $|f(\hat{x}) - \hat{f}| \leq \eta_{\text{conc}}(M_{\mathbf{FCP}}/\delta)/\sqrt{N_{\mathbf{FCP}}}$,
- *either* $\mathcal{X}^* \subset \mathcal{X}$, *or* $f(\hat{x}) - f^* \leq B$,
- the total budget n_t used up to the t-th call of **Cut** is such that

$$(t-1)N_{\mathbf{FCP}} + tN_{\mathbf{Cut}} \leq n_t \leq 4(t-1)M_{\mathbf{FCP}}N_{\mathbf{FCP}} + tN_{\mathbf{Cut}}.$$

We prove this by induction as follows.

- Proof of H_1: note first that, by Lemmas 2.2 and 6.1, the conditions of Lemma 6.3 are satisfied after the initialisation phase of **BarAlg** on an event of probability $1-\delta$. Moreover, the running time of the initialisation is $N_{\mathbf{Cut}}$. So, H_1 holds.
- Proof of H_{t+1} assuming that H_t holds: assuming that H_t holds for a given t, we have by Lemma 6.3 that H_{t+1} holds on an event ξ of probability larger than $1-\delta$, conditional on ξ_t. So, writing $\xi_{t+1} = \xi_t \cap \xi$, we have proven that H_{t+1} holds.

So, for any given $t \geq 0$, on an event of probability larger than $1-5t\delta$ if **Cut** is called for the t time, it takes as parameter a convex set $\mathcal{X} \subset \bar{\mathcal{X}}$, $\hat{x} \in \bar{\mathcal{X}}$, $\hat{f} \in \mathbb{R}$ such that

- $|f(\hat{x}) - \hat{f}| \leq \eta_{\text{conc}}(M_{\mathbf{FCP}}/\delta)/\sqrt{N_{\mathbf{FCP}}}$,
- *either* $[\mathcal{X}^* \subset \mathcal{X}$ and $\text{vol}(\mathcal{X}) \leq (\frac{9}{10})^{t-1}\text{vol}(\bar{\mathcal{X}})]$, *or* $f(\hat{x}) - f^* \leq B$,
- the total budget n_t used up to the t-th call of **Cut** is such that $tN_{\mathbf{Cut}} \leq n_t \leq 2tN_{\mathbf{Cut}}$ – since

$$4M_{\mathbf{FCP}}N_{\mathbf{FCP}} = N_{\mathbf{Cut}}.$$

Step 5: Application of the result of the induction to what happens at the end of the algorithm. The induction from *Step 4* applied to $t = M_{\mathbf{Cut}}/5$ implies that, on an event of probability larger than $1-5(n/N_{\mathbf{Cut}})\delta = 1 - M_{\mathbf{Cut}}\delta$ – that we will write ξ_{term} – the algorithm **BarAlg** terminates after at least $n/(2N_{\mathbf{Cut}})$ rounds, and at most $n/N_{\mathbf{Cut}}$ rounds, and at its termination round, the current convex set $\mathcal{X} \subset \bar{\mathcal{X}}$, and the current value $\hat{x}$ (that **BarAlg** will output as it is the last round) are such that

- *either* $[\mathcal{X}^* \subset \mathcal{X}$ and $\text{vol}(\mathcal{X}) \leq (\frac{9}{10})^{n/(2N_{\mathbf{Cut}})-1}\text{vol}(\bar{\mathcal{X}})]$,
- *or* $f(\hat{x}) - f^* \leq B$.

If $f(\hat{x}) - f^* \leq B$, the proof is finished. So, assume that on ξ_{term}, we have $\mathcal{X}^* \subset \mathcal{X}$ and $\text{vol}(\mathcal{X}) \leq (\frac{9}{10})^{n/(2N_{\mathbf{Cut}})-1}\text{vol}(\bar{\mathcal{X}})$. Note that as $\mathcal{X}$ is convex, we have $\text{conv}(\mathcal{X}^*) \subset \mathcal{X}$ on ξ_{term}.

By definition of $N_{\mathbf{Cut}}$, we have that

$$\left(\frac{9}{10}\right)^{n/(2N_{\mathbf{Cut}})-1} \leq (8n^2d^2)^{-d}.$$

So, by Lemma 6.1, we have a contradiction on ξ_{term}: $\text{conv}(\mathcal{X}^*) \subset \mathcal{X}$, but

$$\text{vol}(\text{conv}(\mathcal{X}^*)) \geq \text{vol}(\mathcal{X}).$$

So, it means that on ξ_{term}, we must have $f(\hat{x}) - f^* \leq B$.

6.2. Proof of Proposition 6.2

In what follows, write $M_{\mathbf{FCP}} := M_{\mathbf{FCP}}(N)$. We first state the following lemma.

Lemma 6.4. *Consider $s \geq 0$ and $z \in \mathbb{R}^d$ such that $\mathbb{B}_2(z, 2c) \in \overline{\mathcal{Z}}^{\mathcal{X}}$. There exists an event of probability larger than $1 - 3\delta$ such that the following hold on it, during a run of* **FCP**$(z, \mathcal{X}, c, N, s, \delta)$.

- *Assume that $g(z) - f^* \leq \frac{1}{N}$ and $s = 0$. For any $z_j^{(i)}$ such that*

$$\hat{g}_{z_j^{(i)}} - \hat{g}_z \geq \frac{1}{16N} + 4\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}},$$

  then since $2\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}} \geq \frac{(17/16)}{N}$, we have

$$g(z_j^{(i)}) - f^* \geq \frac{(17/16)}{N}.$$

- *Assume that $g(z) - f^* \geq \frac{(17/16)^s}{N}$. For any $z_j^{(i)}$ such that*

$$\hat{g}_{z_j^{(i)}} - \hat{g}_z \geq \frac{(17/16)^s}{16N} + 4\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}},$$

  it holds that

$$g(z_j^{(i)}) - f^* \geq \frac{(17/16)^{s+1}}{N}.$$

- *Assume that $g_c(2z) - g_c(z) \geq 2^{-3}(g_c(z) - f^*)$, and $g(z) - f^* > \frac{(17/16)^s}{N} \vee [2^{15}\eta_{\text{conc}}(1/\delta)\log(2/\delta)\frac{1}{\sqrt{N}}]$. Then, there exists $z_j^{(i)}$ such that*

$$\hat{g}_{z_j^{(i)}} - \hat{g}_z \geq \frac{(17/16)^s}{16N} + 4\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}}.$$

Assume that $\mathbb{B}_2(z_0, 2M_{\mathbf{FCP}}c) \subset \bar{\mathcal{Z}}^{\mathcal{X}}$. Then, by construction, we know that even if **FCP** calls itself recursively for $M_{\mathbf{FCP}}$ rounds, then all the parameters z that it will take at each round will be such that $\mathbb{B}_2(z, 2c) \subset \bar{\mathcal{Z}}^{\mathcal{X}}$. Write τ for the random round where the recursive application of $\mathbf{FCP}(z_0, \mathcal{X}, c, N, 0, \delta)$ stops. Applying Lemma 6.4, we know that on an event of probability larger than $1 - 3M_{\mathbf{FCP}}\delta$, for the point z taken as input at round $\tau \wedge M_{\mathbf{FCP}}$.

- If $\tau < M_{\mathbf{FCP}}$, we have either

$$g_{2c}(z) - g_c(z) \leq 2^{-3}(g_c(z) - f^*),$$

or

$$g(z) - f^* \leq 2^{15}\eta_{\mathrm{conc}}(1/\delta)\log(2/\delta)\frac{1}{\sqrt{N}}.$$

This concludes the proof in this case.

- Otherwise, if $\tau \geq M_{\mathbf{FCP}}$, then

$$g(z) - f^* \geq \frac{(17/16)^{\lfloor M_{\mathbf{FCP}} \rfloor}}{N}.$$

Note that by definition of $M_{\mathbf{FCP}}$ this implies $g(z) - f^* \geq 2$ which contradicts the fact that f and therefore g takes value in $[0, 1]$. So, this case cannot happen. This concludes the proof in this case as well.

6.3. Proof of Lemma 6.3

We first recall the following classical results of convex geometry.

Proposition 6.5 (KLS lemma). *Let $\mathcal{C}$ be a convex set in isotropic position. It holds that*

$$\mathbb{B}_2(1) \subset \mathcal{C} \subset \mathbb{B}_2(2d).$$

Proposition 6.6 (Approximate barycentric cutting of an isotropic convex). *Let $\mathcal{C}$ be a convex set in isotropic position. It holds for any $u \in \mathbb{R}^d : u \neq 0$, and any $z \in \mathbb{R}^d$,*

$$\mathrm{vol}(\mathcal{C} \cap \{w : \langle w - z, u\rangle \geq 0\}) \geq (1/e - \|z\|_2)\mathrm{vol}(\mathcal{C}).$$

See, e.g., [8, 26], for references on these two classical lemmas.
An immediate corollary of the last proposition is as follows.

Corollary 6.7 (Approximate barycentric cutting of a convex). *Let $\mathcal{K}$ be a convex set. It holds for any $u \in \mathbb{R}^d : u \neq 0$, and any $z \in \mathbb{R}^d$,*

$$\mathrm{vol}\big(\mathcal{K} \cap F_{\mathcal{X}}^{-1}\big(\{w : \langle w - z, u\rangle \geq 0\}\big)\big) \geq (1/e - \sqrt{d}\,\|z\|_2)\,\mathrm{vol}(\mathcal{K}).$$

From Proposition 6.5, we deduce that

$$\mathbb{B}_2(2M_{\mathbf{FCP}}c) \subset \mathcal{Z}^{\mathcal{X}}.$$

We therefore know that Proposition 6.2 holds for the output $(z, \hat{g}_z)$ of

$$\mathbf{FCP}(0, \mathcal{X}, c, N_{\mathbf{FCP}}, 0, \delta)$$

– and write ξ for the event of probability larger than $1 - 4\delta$, where the proposition holds. Note that it already implies by definition of the algorithm that on ξ

$$|f(\hat{x}') - \hat{f}'| \leq \eta_{\text{conc}}(M_{\mathbf{FCP}}/\delta)/\sqrt{N_{\mathbf{FCP}}},$$

and also that on ξ

$$N_{\mathbf{FCP}} + N_{\mathbf{Cut}} \leq T_{\mathbf{FCP}} \leq 4M_{\mathbf{FCP}}N_{\mathbf{FCP}} + N_{\mathbf{Cut}},$$

and also that $\mathcal{X}' \subset \mathcal{X}$ is convex. Note also that on ξ it implies by definition if c that

$$\|z\|_2 \leq 2M_{\mathbf{FCP}}c = 1/(4e\sqrt{d}),$$

which implies by Corollary 6.7, by construction of the algorithm that on ξ,

$$\text{vol}(\mathcal{X} \setminus \mathcal{X}') \geq \frac{1}{2e}\text{vol}(\mathcal{X}),$$

namely, that on ξ,

$$\text{vol}(\mathcal{X}') \geq (1 - \frac{1}{2e})\text{vol}(\mathcal{X}).$$

Case 1: $g(z)$ is small, or $f(\hat{x})$ is small. We first consider the case where either $f(\hat{x}) - f^* \leq B$, or on ξ, we have that $g(z) - f^* \leq B$. In this case, we will have by definition of the algorithm that on ξ:

$$f(\hat{x}') - f^* \leq B,$$

as $B \geq 2^{16}\eta_{\text{conc}}(M_{\mathbf{FCP}}/\delta)\log(2M_{\mathbf{FCP}}/\delta)\frac{1}{\sqrt{N_{\mathbf{FCP}}}} \geq 8\eta_{\text{conc}}(M_{\mathbf{FCP}}/\delta)\frac{1}{\sqrt{N_{\mathbf{FCP}}}}$. This concludes the proof.

Case 2: $g(z)$ and $f(\hat{x})$ are large. We now consider the converse case on ξ. In this case, we know by Proposition 6.2 that on ξ,

$$g(z) - f^* \geq B.$$

In this case, we know by Proposition 6.2 that on ξ,

$$g_{2c}(z) - g_c(z) \leq 2^{-3}(g_c(z) - f^*),$$

and we also know by assumption that

$$\mathcal{X}^* \subset \mathcal{X}.$$

So that by definition of $\mathcal{X}^*$, and since $N_{\mathbf{FCP}} \leq n$, for any $\tilde{x} \in \mathcal{X}^*$, on ξ,

$$g_{2c}(z) - g_c(z) \leq 2^{-2}(g_c(z) - g(F_{\mathcal{X}}(\tilde{x}))).$$

We can therefore apply Lemma 2.1, and we have that on ξ,

$$\langle \nabla g_c(z), z - F_{\mathcal{X}}(\tilde{x}) \rangle \geq \frac{3}{4}[g_c(z) - g(F_{\mathcal{X}}(\tilde{x}))] \geq \frac{5}{8}[g_c(z) - f^*] \geq \frac{5}{8}B, \quad (4)$$

as $B \geq 8/n$ and $g(F_{\mathcal{X}}(\tilde{x})) - f^* \leq 1/n$.

Also, by Lemma 2.2, for any $u \in \mathbb{R}^d$, conditional to ξ and on an event ξ' of probability larger than $1 - \delta$,

$$|\langle \widehat{\nabla g_c} - \nabla g_c(z), u \rangle| \leq \eta_{\text{conc}}(1/\delta)\|u\|_2 \frac{\sqrt{d}}{c\sqrt{N_{\mathbf{Cut}}}}.$$

Thus, on $\xi' \cap \xi$, for any $\tilde{x} \in \mathcal{X}^*$,

$$|\langle \widehat{\nabla g_c} - \nabla g_c(z), z - F_{\mathcal{X}}(\tilde{x}) \rangle| \leq \eta_{\text{conc}}(2d/\delta)\|z - F_{\mathcal{X}}(\tilde{x})\|_2 \frac{\sqrt{d}}{c\sqrt{N_{\mathbf{Cut}}}}.$$

From Proposition 6.5, this implies on $\xi' \cap \xi$

$$|\langle \widehat{\nabla g_c} - \nabla g_c(z), z - F_{\mathcal{X}}(\tilde{x}) \rangle| \leq 2\eta_{\text{conc}}(2d/\delta) \frac{d}{c\sqrt{N_{\mathbf{Cut}}}} \leq B/16,$$

as $B \geq 32\eta_{\text{conc}}(2d/\delta)\frac{d}{c\sqrt{N_{\mathbf{Cut}}}}$.

Combining this result with equation (4) leads to the fact that on $\xi' \cap \xi$, for any $\tilde{x} \in \mathcal{X}^*$,

$$\langle \widehat{\nabla g_c}, z - F_{\mathcal{X}}(\tilde{x}) \rangle \geq \frac{1}{16}B.$$

Thus, on $\xi' \cap \xi$, we have that $\mathcal{X}^* \subset \mathcal{X}'$. This concludes the proof.

Proof of Lemma 6.4. By Lemma 2.2, it holds on an event of probability larger than $1 - \delta(1 + \sum_k 1/k^2) \geq 1 - 2.7\delta$ that

$$|g(z) - \hat{g}_z| \leq \eta_{\text{conc}}(1/\delta)/\sqrt{N},$$

and for any $i \leq I_N$, $j \leq 2^i$,

$$|g(z_j^{(i)}) - \hat{g}_{z_j^{(i)}}| \leq \eta_{\text{conc}}(2^i i^2/\delta)/\sqrt{N}.$$

Write ξ for this event.

Note that if $g(z) - f^* \leq \frac{1}{N}$ and $s = 0$, then on ξ we have that if there exists $z_j^{(i)}$ such that

$$\hat{g}_{z_j^{(i)}} - \hat{g}_z \geq \frac{1}{16N} + 4\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}},$$

then since $2\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}} \geq \frac{(17/16)}{N}$, we have

$$g(z_j^{(i)}) - f^* \geq \frac{(17/16)}{N}.$$

The first part of the lemma is therefore proven.

Assume now that $g(z) - f^* \geq \frac{(17/16)^s}{N}$. Note first that, on ξ, we have that if there exists $z_j^{(i)}$ such that

$$\hat{g}_{z_j^{(i)}} - \hat{g}_z \geq \frac{(17/16)^s}{16N} + 4\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}},$$

then since $g(z) - f^* \geq \frac{(17/16)^s}{N}$,

$$g(z_j^{(i)}) - f^* \geq \frac{(17/16)^{s+1}}{N}.$$

The second part of the lemma is therefore proven.

Now, assume that z satisfies the conditions of the third part of the lemma, namely, $g_c(2z) - g_c(z) \geq 2^{-3}(g_c(z) - f^*)$, and

$$g(z) - f^* > \frac{(17/16)^s}{N} \vee \left[2^{15}\eta_{\text{conc}}(1/\delta)\log(2/\delta)\frac{1}{\sqrt{N}}\right].$$

Step 1: Establishing condition under which at least a $z_j^{(i)}$ is selected. On ξ, it holds that

$$|g(z_j^{(i)}) - g(z)| \leq 2\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}}.$$

So, if

$$g(z_j^{(i)}) - g(z) \geq \frac{(17/16)^s}{16N} + 6\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}} := \Delta_i, \tag{5}$$

then on ξ it can be selected as it satisfies

$$\hat{g}_{z_j^{(i)}} - \hat{g}_z \geq \frac{(17/16)^s}{16N} + 4\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}}.$$

We now recall the following application of Bernstein inequality (see, e.g., [9, Theorem 2.10]).

Lemma 6.8 (Concentration of binomial random variables). *Let $p \in [0,1]$ and $m \geq 1$. Let $X_1, \dots, X_m \sim_{i.i.d.} \mathcal{B}(p)$. Then, with probability larger than $1-\delta$,*

$$\left| \frac{1}{m} \sum_{i=1}^{m} X_i - p \right| \leq \sqrt{2p \frac{\log(2/\delta)}{m}} + 2\frac{\log(2/\delta)}{m},$$

which implies in particular that, with probability larger than $1-\delta$,

$$\frac{p}{2} - 4\frac{\log(2/\delta)}{m} \leq \frac{1}{m} \sum_{i=1}^{m} X_i \leq 2p + 2\frac{\log(2/\delta)}{m}.$$

Assume that there exists $i \leq I_N$ such that

$$\mathbb{P}_{Z \sim \mathcal{U}_{\mathbb{B}_2(2c)}}(g(z+Z) - g(z) \geq \Delta_i) > 8\frac{\log(2/\delta)}{2^i}.$$

By Lemma 6.8, then we know that with probability larger than $1-\delta$, at least one of the $z_j^{(i)}$ for some j will be such that

$$g(z_j^{(i)}) - g(z) \geq \Delta_i.$$

Using equation (5), we therefore know that, in this case, with probability larger than $1-4\delta$, $z_j^{(i)}$ will be selected, finishing the proof in this case.

Step 2: Converse case where, for any $i \leq I_N$, we have $\mathbb{P}_{Z \sim \mathcal{U}_{\mathbb{B}_2(2c)}}(g(z+Z) - g(z) \geq \Delta_i) \leq 8\frac{\log(2/\delta)}{2^i}$. We recall that

$$\Delta_i = \frac{(17/16)^s}{8N} + 6\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}}.$$

Note that, by assumption, we therefore have that, for any $i \leq I_N$, we have

$$\mathbb{P}_{Z \sim \mathcal{U}_{\mathbb{B}_2(2c)}}\left(g(z+Z) - g(z) - \frac{(17/16)^s}{16N} \geq 6\eta_{\text{conc}}(2^i i^2/\delta)\sqrt{\frac{i^2 2^i}{N}} \right) \leq 8\frac{\log(2/\delta)}{2^i}.$$

Thus, since $4\eta_{\text{conc}}(2^{I_N} I_N^2/\delta)\sqrt{\frac{I_N^2 2^{I_N}}{N}} \geq 2$, by definition of I_N,

$$\begin{aligned}
&\mathbb{E}_{Z \sim \mathcal{U}_{\mathbb{B}_2(2c)}}\left[g(z+Z) - g(z) - \frac{(17/16)^s}{16N} \right] \\
&\leq \sum_{i \leq I_N} 64\eta_{\text{conc}}(2^{i+1}(i+1)^2/\delta)\sqrt{\frac{(i+1)^2 2^{i+1}}{N}} \times \frac{\log(2/\delta)}{2^i},
\end{aligned}$$

leading to

$$g_{2c}(z) - g(z) - \frac{(17/16)^s}{16N} \leq 64\sqrt{2}\eta_{\text{conc}}(1/\delta)\log(2/\delta)\frac{1}{\sqrt{N}}\sum_{i\leq I_N}(i+1)^4 2^{-i/2}$$
$$\leq 2^{11}\eta_{\text{conc}}(1/\delta)\log(2/\delta)\frac{1}{\sqrt{N}}.$$

Since $g(z) - f^* \geq \frac{(17/16)^s}{N}$, and $g(z) - f^* > 2^{15}\eta_{\text{conc}}(1/\delta)\log(2/\delta)\frac{1}{\sqrt{N}}$ by assumption, then we have that

$$g_{2c}(z) - f^* < (g(z) - f^*)(1 + 2^{-3}).$$

This implies

$$g_{2c}(z) - g_c(z) < 2^{-3}(g(z) - f^*) \leq 2^{-3}(g_c(z) - f^*).$$

This contradicts our assumption that $g_{2c}(z) - g_c(z) \geq 2^{-3}(g_c(z) - f^*)$ so that this case cannot happen under our assumption. This concludes the proof. ∎

6.4. Proof of technical lemmas

Proof of Lemma 2.1. Assume without loss of generality that $\tilde{z} = 0$ and $g(\tilde{z}) = 0$. Let $c > 0$ and $z \in \mathbb{R}^d$ such that $g_{2c}(z) - g_c(z) \leq 2^{-2}g_c(z)$. In order to prove the lemma, it suffices to prove that $\langle \nabla g_c(z), z\rangle \geq 3g_c(z)/4$.

By convexity of g on $\mathbb{R}^d$, note that, for any $z' \in \mathbb{R}^d$, we have $g(z'/2) \leq g(z')/2$. Thus, by definition of g_c,

$$g_c(z'/2) \leq g_{2c}(z')/2.$$

Since $g_{2c}(z) \leq (5/4)g_c(z)$, we have apply the formula above to z

$$g_c(z/2) \leq g_{2c}(z)/2 \leq 5g_c(z)/8$$

so that

$$g_c(z) - g_c(z/2) \geq 3g_c(z)/8. \tag{6}$$

Since g_c is convex and differentiable on $\mathbb{R}^d$,

$$g_c(z) - g_c(z/2) \leq \langle \nabla g_c(z), z/2\rangle.$$

Thus, finally by equation (6),

$$3g_c(z)/4 \leq \langle \nabla g_c(z), z\rangle.$$ ∎

Proof of Lemma 2.2. Bound on the deviations of $\hat{g}_c(z)$. Let $\delta > 0$. Note that $\hat{g}_c(z)$ is the empirical mean of the $\tilde{y}_i^{(b)}$, which are by construction i.i.d. random variables such that $\tilde{y}_i^{(b)} \in [0, 1]$ and $\mathbb{E}[\tilde{y}_i^{(b)}] = g_c(z)$. Thus, applying Hoeffding's inequality (see, e.g., [9, Theorem 2.8]), with probability larger than $1 - \delta$,

$$|\hat{g}_c(z) - g_c(z)| \leq \sqrt{\frac{\log(2/\delta)}{2N}},$$

leading to the result.

Bound on the deviations of $\langle \widehat{\nabla g_c}(z) \rangle$. Let $\delta > 0$. Note now that $\mathbb{E}\langle \widehat{\nabla g_c}(z), u \rangle$ is the empirical of the i.i.d. random variables $W_i := \frac{d}{c^2} \tilde{y}_i^{(s)} \langle Z_i^{(s)}, u \rangle$. Note that, by equation (1), we have

$$\mathbb{E} W_i = \langle \nabla g_c(z), u \rangle,$$

and

$$\begin{aligned} \mathbb{E} W_i^2 &= \frac{d^2}{c^4} \mathbb{E}[(\tilde{y}_i^{(s)})^2 \langle Z_i^{(s)}, u \rangle^2] \\ &\leq \frac{d^2}{c^4} \mathbb{E}[\langle Z_i^{(s)}, u \rangle^2] \\ &= \frac{d^2}{c^4} \frac{c^2}{d} \|u\|_2^2 = \frac{d}{c^2} \|u\|_2^2, \end{aligned}$$

and

$$\begin{aligned} |W_i| &= \frac{d}{c^2} \tilde{y}_i^{(s)} |\langle Z_i^{(s)}, u \rangle| \\ &\leq \frac{d}{c} \|u\|_2. \end{aligned}$$

Thus, applying Bernstein's inequality (see, e.g., [9, Theorem 2.10]), with probability larger than $1 - \delta$,

$$\|\langle \widehat{\nabla g_c}(z) - \nabla g_c(z), u \rangle\| \leq \frac{\sqrt{d}}{c} \|u\|_2 \sqrt{2 \frac{\log(2/\delta)}{N}} + 2 \frac{d}{c} \|u\|_2 \frac{\log(2/\delta)}{N}.$$

Since $N \geq d$, this leads to the result. ∎

Proof of Lemma 6.1. In what follows, all quantities $\mathcal{Z}$, g, z^*, F are considered with respect to the set $\overline{\mathcal{X}}$.

By Proposition 6.5 and since $\mathcal{Z}$ is in renormalised isotropic position, we know that

$$\mathbb{B}_2(d^{-1/2}) \subset \mathcal{Z}$$

so that by convexity for any $\lambda \in [0, 1]$

$$(1 - \lambda) z^* + \lambda \mathbb{B}_2(d^{-1/2}) \subset \mathcal{Z}.$$

Since g takes values in $[0,1]$ on $\mathcal{Z}$, we know by convexity of g that

$$g((1-\lambda)z^*) - g(z^*) \leq \lambda .$$

Since $(1-\lambda)z^* + \lambda \mathbb{B}_2(d^{-1/2}) \subset \mathcal{Z}$, we also know in the same way that for any $z \in (1-\lambda)z^* + \lambda^2 \mathbb{B}_2(d^{-1/2})$

$$g(z) - g((1-\lambda)z^*) \leq \lambda$$

so that for any $z \in (1-\lambda)z^* + \lambda^2 \mathbb{B}_2(d^{-1/2})$

$$g(z) - g(z^*) \leq 2\lambda .$$

This concludes the proof by taking $\lambda = (2n)^{-1}$ and since by definition

$$\mathcal{X}^* \subset F^{-1}[(1-\lambda)z^* + \lambda^2 \mathbb{B}_2(d^{-1/2})]$$

and since

$$\frac{\mathrm{vol}(\mathrm{conv}(\mathcal{X}^*))}{\mathrm{vol}(\bar{\mathcal{X}})} \geq \frac{(d^{-1}\lambda^2)^d}{(2d)^d},$$

by Proposition 6.5. ∎

Proof of Corollary 6.7. The corollary follows from Proposition 6.6 using the facts that $F_{\mathcal{X}}(\mathcal{K})$ is in isotropic position rescaled by $d^{-1/2}$ and also that since $F_{\mathcal{X}}^{-1}$ is a linear application of the form $F_{\mathcal{X}}^{-1}(z) = \sqrt{d}\,\Sigma_{\mathcal{X}}^{1/2}(z + \mu_{\mathcal{X}})$, then for any convex $\mathcal{K}'$

$$\mathrm{vol}(F_{\mathcal{X}}^{-1}(\mathcal{K}')) = d^{d/2} \det(\Sigma_{\mathcal{X}})^{1/2} \mathrm{vol}(\mathcal{K}'),$$

where $\det(\Sigma_{\mathcal{X}})$ is the determinant of $\Sigma_{\mathcal{X}}$. ∎

Acknowledgements. The author would like to warmly thank Evgenii Chzhen, Christophe Giraud, and Nicolas Verzelen for many insightful discussions on this problem, for their valuable opinion, and for their support without which this work would not have been written. She would also like to thank the anonymous reviewers as well as the editor Sasha Tsybakov for very helpful feedback and insights. This feedback led to a significant improvement of this paper – in particular by removing an unnecessary assumption, by adding relevant references and perspectives – in particular on variations around the centre of gravity method, and on an efficient way to sample in convex bodies – by adding important discussions, and by improving the readability and more generally the writing of this paper.

Funding. This work was partially supported by the Deutsche Forschungsgemeinschaft (DFG) CRC 1294 "Data Assimilation", Project A03, by the DFG Forschungsgruppe FOR 5381 "Mathematical Statistics in the Information Age – Statistical Efficiency and Computational Tractability", Project TP 02, by the Agence Nationale de

la Recherche (ANR) and the DFG on the French-German PRCI ANR ASCAI CA 1488/4-1 “Aktive und Batch-Segmentierung, Clustering und Seriation: Grundlagen der KI”, and by the Université franco-allemande (UFA) through the college doctoral franco-allemande CDFA-02-25 “Statistisches Lernen für komplexe stochastische Prozesse”.

References

[1] A. Agarwal, D. P. Foster, D. Hsu, S. M. Kakade, and A. Rakhlin, Stochastic convex optimization with bandit feedback. In *Advances in neural information processing systems*, pp. 1215–1223, 24, 2011

[2] A. Akhavan, E. Chzhen, M. Pontil, and A. Tsybakov, A gradient estimator via l1-randomization for online zero-order optimization with two point feedback. In *Advances in neural information processing systems*, pp. 7685–7696, 35, 2022

[3] A. Akhavan, E. Chzhen, M. Pontil, and A. B. Tsybakov, Gradient-free optimization of highly smooth functions: Improved analysis and a new algorithm. *J. Mach. Learn. Res.* **25** (2024), article no. 370 MR 4847434

[4] A. Akhavan, K. Lounici, M. Pontil, and A. B. Tsybakov, A conversion theorem and minimax optimality for continuum contextual bandits. [v1] 2024, [v6] 2025, arXiv:2406.05714v6

[5] A. Akhavan, M. Pontil, and A. Tsybakov, Exploiting higher order smoothness in derivative-free optimization and continuous bandits. In *Advances in neural information processing systems*, pp. 9017–9027, 33, 2020

[6] F. Bach and V. Perchet, Highly-smooth zero-th order online optimization. In *Proceedings of The 29th Annual Conference on Learning Theory*, pp. 257–283, Proc. Mach. Learn. Res. 49, PMLR, 2016

[7] D. P. Bertsekas, Incremental gradient, subgradient, and proximal methods for convex optimization: A survey. Tech. rep., Massachusetts Institute of Technology, Cambridge, MA, 2011

[8] D. Bertsimas and S. Vempala, Solving convex programs by random walks. *J. ACM* **51** (2004), no. 4, 540–556 Zbl 1204.90074 MR 2147847

[9] S. Boucheron, G. Lugosi, and O. Bousquet, Concentration inequalities. In *Summer school on machine learning*, pp. 208–240, Lect. Notes Comput. Sci. 3176, Springer, 2003 Zbl 1120.68427

[10] S. Bubeck, Convex optimization: Algorithms and complexity. *Found. Trends Mach. Learn.* **8** (2015), no. 3–4, 231–357 Zbl 1365.90196

[11] S. Bubeck, Y. T. Lee, and R. Eldan, Kernel-based methods for bandit convex optimization. In *STOC'17—Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 72–85, ACM, New York, 2017 Zbl 1370.90175 MR 3678172

[12] A. D. Flaxman, A. T. Kalai, and H. B. McMahan, Online convex optimization in the bandit setting: Gradient descent without a gradient. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 385–394, ACM, New York, 2005 Zbl 1297.90117 MR 2298287

[13] H. Fokkema, D. van der Hoeven, T. Lattimore, and J. J. Mayo, Online newton method for Bandit convex optimisation. 2024, arXiv:2406.06506v1

[14] E. Hazan, Introduction to online convex optimization. *Found. Trends Optim.* **2** (2016), no. 3–4, 157–325

[15] K. G. Jamieson, R. Nowak, and B. Recht, Query complexity of derivative-free optimization. In *Advances in neural information processing systems*, pp. 2672–2680, 25, Curran Associates, Red Hook, NY, 2012

[16] H. Jia, A. Laddha, Y. T. Lee, and S. S. Vempala, Reducing isotropy and volume to kls: Faster rounding and volume algorithms. [v1] 2020, [v3] 2024, arXiv:2008.02146v3

[17] T. Lattimore, Improved regret for zeroth-order adversarial bandit convex optimisation. *Math. Stat. Learn.* **2** (2019), no. 3-4, 311–334 Zbl 1470.90080 MR 4165267

[18] T. Lattimore, Bandit convex optimisation. [v1] 2024, [v4] 2025, arXiv:2402.06535v4

[19] T. Lattimore and A. Gyorgy, Improved regret for zeroth-order stochastic convex bandits. In *Conference on learning theory*, pp. 2938–2964, 134, PMLR, 2021

[20] T. Lattimore and C. Szepesvári, *Bandit algorithms*. Cambridge University Press, Cambridge, 2020 Zbl 1439.68002

[21] A. J. Levin, An algorithm for minimizing convex functions. *Soviet Math. Dokl.* **6** (1965), 286–290 Zbl 0154.45001

[22] L. Lovász and S. Vempala, Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm. *J. Comput. System Sci.* **72** (2006), no. 2, 392–417 Zbl 1090.68112 MR 2205290

[23] A. S. Nemirovsky and D. B. Yudin, *Problem complexity and method efficiency in optimization*. Wiley-Intersci. Ser. Discrete Math., John Wiley & Sons, New York, 1983 Zbl 0501.90062 MR 0702836

[24] D. J. Newman, Location of the maximum on unimodal surfaces. *J. Assoc. Comput. Mach.* **12** (1965), 395–398 Zbl 0139.10402 MR 0182129

[25] O. Shamir, On the complexity of bandit and derivative-free stochastic convex optimization. In *Conference on learning theory*, pp. 3–24, PMLR, 2013

[26] G. Sonnevend, Applications of analytic centers for the numerical solution of semi-infinite, convex programs arising in control theory. In *System modelling and optimization (Leipzig, 1989)*, pp. 413–422, Lect. Notes Control Inf. Sci. 143, Springer, Berlin, 1990 Zbl 0707.90093 MR 1141674

Received 1 July 2024; revised 8 August 2025.

Alexandra Carpentier
Institut für Mathematik, Universität Potsdam, Karl-Liebknecht-Straße 24-25,
14476 Potsdam OT Golm, Germany; carpentier@uni-potsdam.de