Bandit Convex Optimization with Biased Noisy Gradient Oracles

by

Xiaowei Hu

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

Optimizing an objective function over convex sets is a key problem in many different machine learning models. One of the various kinds of well studied objective functions is the convex function, where any local minimum must be the global minimum over the domain. To find the optimal point that minimize the objective convex function, a natural choice for the search direction is the negative gradient. The resulting algorithm, usually called the gradient descent method, is widely used to approach convex optimization problems. Given the entire objective function or the first-order information (gradient), it is straightforward to do the line search following the chosen direction. However, another scenario exists where the algorithm has no access to such entire function or the first-order information, except evaluation of some queried points. Then there comes the bandit optimization problem, which is also known as zeroth-order or derivative-free optimization.

Algorithms for bandit convex optimization often rely on constructing noisy gradient estimates, which are then used in appropriately adjusted first-order algorithms, like gradient descent, replacing actual gradients. Depending on the properties of the function to be optimized and the nature of "noise" in the bandit feedback, the bias and variance of gradient estimates exhibit various tradeoffs. For example, the gradient estimate with a small bias tends to have a large variance, while the estimate with a small variance could have a large bias. Considering that both the bias and variance of the gradient estimate might jeopardize the optimization algorithm. It is worthwhile to measure their influences in a quantitative pattern, and study if the

optimization error basing on a certain gradient estimate can be further improved.

This thesis proposes a novel framework that replaces the specific gradient estimation methods with an abstract oracle. The oracle directly interacts with the algorithm, outputting biased, noisy gradient estimates satisfying some predefined properties. In this way, we abstract tradeoffs of the bias and variance, skipping the details of constructing gradient estimates. With the help of the new framework we unify previous works, reproducing their results in a clean and concise fashion, proving the upper bound of the optimization error with the Mirror Descent algorithm. Meanwhile, perhaps more importantly, the framework also allows us to show a lower bound of the optimization error, which can match the corresponding upper bound under certain conditions. This formally demonstrates that, to achieve the optimal root-n rate for the bandit convex optimization, either the algorithms that use existing gradient estimators, or the proof techniques used to analyze them, have to go beyond what exists today.

Preface

This thesis is an original work by Xiaowei Hu. Parts of it were published in the Proceedings of the Nineteenth International Conference on Artificial Intelligence and Statistics (AISTATS), volume 51 of JMLR: W&CP, Cadiz, Spain, May 9–11 (Hu, Prashanth L.A., György, and Szepesvári, 2016).

Acknowledgements

I would like to sincerely thank my great supervisors Csaba Szepesvári and András György for their insightful advice and patient support. Their enthusiasm in research has always encouraged me to move forward and explore my interests. I am also grateful for all my friends and colleagues, who have made my experiences in Edmonton extremely worthwhile and memorable.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Bandit optimization schemes, also known as derivative-free or zeroth-order optimization, have a long history in machine learning. In the earlier work (Nemirovskii and Yudin, 1983), an overview of both first-order and zeroth-order methods was given for convex optimization problems. In the zeroth-order setting, only functional (zeroth-order) information is available — rather than first-order gradient information. Such procedures are desirable when explicit gradient calculations may be impossible or computationally unfeasible. Applications of bandit problems include online auction, controlling an unknown system, and many examples provided by simulation-based optimization (Spall, 2005). Additionally, in graphical model inference (Wainwright and Jordan, 2008), the objective function can be defined in a variational way so that the explicit differentiation is difficult.

Despite the long history and abundant study in bandit optimization problems, a precise understanding of their convergence behavior remains elusive. This thesis presents and analyzes a novel framework with biased, noisy gradient oracles, which enable us to study the performance of a certain kind of methods, where the algorithms observe noisy point-evaluations of the objective function and use these to construct gradient estimates. We will see how the performance will be influenced and whether the optimal rate can be achieved under this framework.

In this chapter, we state our problem in Section 1.1. Relevant work is reviewed in Section 1.2, which also explains the motivation of our work. Our contributions are presented in Section 1.3, as well as a brief discussion of the gradient oracle model. Section 1.4 summarizes the organization of the thesis.

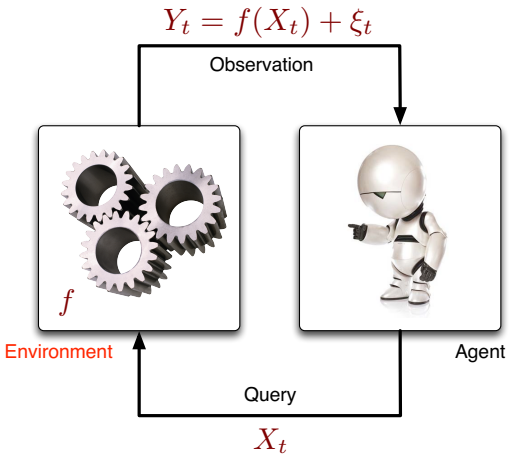


Figure 1.1: The interaction of the algorithm and the environment in bandit optimization.

1.1 Problem Statement

We consider bandit convex optimization in the stochastic setting as well as online setting.

In the stochastic bandit convex optimization problem, the environment chooses a single fixed objective function $f : \mathcal{K} \to \mathbb{R}$, where $\mathcal{K} \subset \mathbb{R}^d$ is a non-empty closed convex set. In each round t, the algorithm queries at the point $X_t \in \mathcal{K}$, and observes the noisy evaluation of $f(X_t)$. The goal of the algorithm is to minimize the *optimization error*

$$\Delta_n = \mathbb{E}\left[f(\hat{X}_n)\right] - \inf_{x \in \mathcal{K}} f(x)\,,$$

where $\hat{X}_n \in \mathcal{K}$ is chosen by the algorithm after n rounds, and n is given to the algorithm at the beginning of the game.

In the online bandit convex optimization problem, a sequence of loss functions $f_1, \ldots, f_n$ are chosen by the environment. In round t, the algorithm queries at $Y_t \in \mathcal{K}$[1], and suffers the loss $f_t(Y_t)$. The goal in online BCO is to minimize the expected *regret*

$$R_n = \mathbb{E}\left[\sum_{t=1}^{n} f_t(Y_t)\right] - \inf_{x \in \mathcal{K}} \sum_{t=1}^{n} f_t(x)\,.$$

[1]For simplicity, in some cases we allow f_t to be defined outside of $\mathcal{K}$ and allow Y_t to be in a small vicinity of $\mathcal{K}$.

Note that in bandit optimization, the algorithm can only observe noisy samples from the objective function, instead of accessing the full gradient or function.

In this work, we would like to bound the optimization error (or regret) in a minimax fashion. The study of bounds under certain types of assumptions is not unprecedented. In fact, both upper bounds and lower bounds are widely studied under different models and environment settings. It is known that the optimization error of a convex Lipschitz function after n rounds of queries scales as $O(\sqrt{d^2/n})$ (Shamir, 2012). Yet, to the best of our knowledge, at present time there is no algorithm that comes even close to obtaining the desired dependence on the dimension while simultaneously having the $O(\sqrt{1/n})$ convergence rate. Among various kinds of algorithms, we are particularly interested in "gradient"-based algorithms (Flaxman et al., 2005), where the gradient estimate is constructed basing on noisy bandit feedback, and then substitutes the true gradient in some gradient descent algorithms. In Section 1.2 we will discuss why we choose to study the "gradient"-based algorithm, and give an overview of the background and state of the art of the bandit convex optimization problem.

1.2 Motivation and Related Work

For a general convex objective function, the minimax rate for bandit convex optimization in both stochastic settings (optimization error) and online settings (expected regret) is known to be $\Theta(\sqrt{n})$. The lower bound is given by Shamir, 2012. The existence of such a upper bound is proved by Bubeck et al., 2015 with a non-constructive algorithm.

To achieve the optimal rate, many algorithms have been proposed, which mainly fall into two categories: One is ellipsoid methods, the other is "gradient"-based methods. Ellipsoid methods are able to achieve the optimal square-root rate in terms of time complexity, whereas has very large dependency on the dimensions, such as $O(\sqrt{d^{33}/n})$ (Agarwal et al., 2013) and $O(\sqrt{d^{14}/n})$ (Liang et al., 2014). These methods are generally based on a random walk on the epigraph of the function. They suffer from the high dimension and are not actually used in practice. As

to the "gradient"-based method, the gradient here is quoted because the algorithm actually uses an estimate of the gradient, rather than getting the first-order information directly from the environment. Not surprisingly, the performance of this kind of algorithms depends heavily upon the bias and variance of the gradient estimate.

Recall that in each round the algorithm queries at X and receives a noisy function value $Z = F(X, \xi)$, where ξ is the noise from a given set or distribution. Nemirovskii and Yudin, 1983 (Chapter 9.3) developed a randomized sampling strategy that estimate the gradient $\nabla F(X, \xi)$ via randomized evaluations of function values at points on the surface of an L_2-sphere center at X. Flaxman et al., 2005 built on this approach and established the upper bound as $O(\sqrt[4]{d^2/n})$ for bandit convex optimization in the online setting.

Although convergence rates of "gradient" methods for general convex functions are sub-optimal, they can be improved under certain assumptions. For the smooth function, the upper bound is improved to $O(\sqrt[3]{d^2/n})$ (Saha and Tewari, 2011). For the smooth and strongly convex function, the upper bound is shown to be $O(\sqrt{d^2/n})$ (Hazan and Levy, 2014), which already matches the lower bound. The difficulties inherent from estimating gradient using only a single function evaluation can also be alleviated when the function $F(\cdot, \xi)$ can be evaluated at multiple points, as noted by Agarwal et al., 2010 and Nesterov and Spokoiny, 2011. In particular, when the noise can be kept fixed between queries, the optimization error can be optimal (Duchi et al., 2015). This is because the gradient estimation can benefit from the controlled noise, and present better bias-variance tradeoff. [2]

Now we see that there remains a big gap for bandit convex optimization in the general sense, without assuming the strong convexity of objective functions or controlled noise in the feedback. One of the appealing questions may rise up: Can we do better with a clever gradient method? Current work has demonstrated the sub-optimal upper bounds of gradient methods, whereas the lower bounds are still missing. One may assume to get the optimal rate by designing an algorithm that makes better use of the gradient estimate. However, if we could find the matching lower bounds with regards to upper bounds, this assumption must be invalided. In

[2] We will discuss gradient estimation with controlled noise later in Chapter 4.

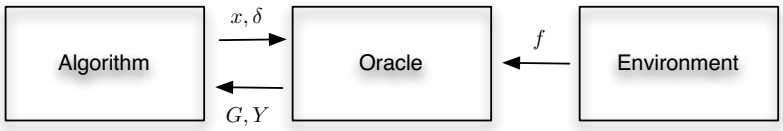


Figure 1.2: The Gradient Oracle Model: interaction of algorithms with the gradient oracle and the environment.

the next section, we will introduce how we figure out the lower bound with the help of the gradient oracle model, which is one of the main contributions of this work.

1.3 Contributions and the Gradient Oracle Model

This thesis studies the convex optimization problem in a novel framework of *Gradient Oracle Models*, for both stochastic and online settings. In the oracle-based framework, the algorithm, upon selecting point X_t, receives a noisy and potentially biased estimate $G_t \in \mathbb{R}^d$ of the gradient[3] of the loss function f from the gradient oracle. To control the bias and the variance, the algorithm can choose a *tolerance parameter* $\delta_t > 0$ (in particular, we allow the algorithms to choose the tolerance parameter sequentially). A smaller δ_t results in a smaller "bias" [4], while typically with a smaller δ_t, the "variance" of the gradient estimate increases. In the online setting, the oracle also gives a response point Y_t in the vicinity of the query point X_t, which serves as the point where the cost is incurred.

The main feature of the model is that the information flow between the algorithm and the environment (holding f, or $f_{1:n}$) is mediated by a stochastic gradient estimation oracle. In this way, we extract the bias-variance tradeoff from gradient estimation techniques extensively used in the literature, mostly for the case when the gradient is estimated only based on noisy observations of the objective function (Katkovnik and Kulchitsky, 1972; Kushner and Clark, 1978; Spall, 1992, 1997; Dippon, 2003; Bhatnagar et al., 2013; Duchi et al., 2015). As we shall see, numerous "gradient" approaches to bandit optimization and online learning essentially

[3] More generally, an estimate of a subgradient of f, in case f is not differentiable at X_t

[4] For the precise meaning of bias, we will consider two definitions, see Section 2.2.

using gradient estimates and first-order methods fit in this framework (Polyak and Tsybakov, 1990; Flaxman et al., 2005; Abernethy et al., 2008; Agarwal et al., 2010; Nesterov and Spokoiny, 2011; Agarwal et al., 2013; Hazan and Levy, 2014).

In addition to reproducing existing results in a unified approach, perhaps more importantly, we provide lower bounds on the minimax optimization error (or regret) for several oracle models. In particular, for optimizing smooth, convex functions, we have matching lower and upper bounds. For instance, under the type-I oracle[5], the minimax optimization error for L-smooth, convex functions is

$$\Delta^{*,\text{type}-\text{I}}_{\mathcal{F}_{L,0},n}(c_1, c_2) = \Theta\left(\sqrt{d}\, C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}} n^{-\frac{p}{2p+q}}\right) ,$$

where the bias of the gradient estimate is bounded by $c_1 = C_1\delta^p$, the variance is bounded by $c_2 = C_2\delta^{-q}$, C_1, C_2, p, q are some constants relative to the properties of gradient estimates. In Chapter 4, we will see using the state-of-the-art gradient estimation techniques, we can obtain $p = 2, q = 0$ for controlled noise, and $p = 2, q = 2$ for uncontrolled noise. This reproduce the optimal rate $\Theta(\sqrt{1/n})$ for controlled noise (Duchi et al., 2015), and shows that if the noise is uncontrolled, gradient methods can not surpass the sub-optimal rate $\Theta(\sqrt[3]{1/n})$.

Note that our oracle model does not capture the full strength of the gradient estimates used in previous work, but it fully describes the properties of the estimates that *so far have been used in their analysis*. As a consequence, our lower bounds show that the known minimax regret of $\sqrt{n}$ (Bubeck et al., 2015; Bubeck and Eldan, 2015; Shamir, 2012) of online and stochastic bandit convex optimization cannot be shown to hold for any algorithm that uses current gradient estimation procedures, unless the proof exploited finer properties of the gradient estimators than used in prior works. In particular, our lower bounds even invalidate the claimed (weaker) upper bound of Dekel et al. (2015) (see Section 3.5).

1.4 Organization

The thesis is organized as follows: We formally define the biased, noisy gradient oracle model in Chapter 2. Upper and lower bounds under this framework are

[5]Detailed definitions will be given in Section 2.2

provided in Chapter 3. Theorems and proofs are presented for different settings, including stochastic versus online optimization, smooth versus strongly convex functions, and type-I versus type-II oracles. Chapter 4 is devoted to describing gradient estimation methods. In addition to general propositions, some widely used examples of estimates are illustrated, which can apply to stochastic and online BCO in Chapter 5 and 6. We close the thesis by summarizing conclusions in Chapter 7.

Chapter 2

Gradient Oracle Models

This chapter gives a detailed description to the biased, noisy gradient oracle model, which will be used to derive the optimization error and regret later. Section 2.1 defines all the notations and concepts we will use in the following chapters. Section 2.2 proposes two types of oracles to catch properties of different gradient estimation methods listed in Chapter 4. The relationship of the two types of oracle are explained in Section 2.3, where we will see one type of oracles actually can be reduced to another under some mild assumptions. In Section 2.4, we review previous work on gradient oracle models and highlight the novel features of our work.

2.1 Notations

Capital letters will denote random variables. For $i \leq j$ positive integers, we use the notation $a_{i:j}$ to denote the sequence $(a_i, a_{i+1}, \ldots, a_j)$.

We let $\|\cdot\|$ denote some norm on $\mathbb{R}^d$, whose dual is denoted by $\|\cdot\|_*$. Let $\mathcal{K} \subset \mathbb{R}^d$ be a non-empty closed convex set.

Given the function $f : \mathcal{K} \to \mathbb{R}$ which is differentiable in $\mathcal{K}^\circ$, [1] f is said to be μ-**strongly convex** w.r.t. a *norm* $\|\cdot\|$ ($\mu \geq 0$) if

$$\mathcal{D}_f(x, y) \geq \tfrac{\mu}{2}\|x - y\|^2$$

for all $x \in \mathcal{K}, y \in \mathcal{K}^\circ$, where $\mathcal{D}_f(x, y) \doteq f(x) - f(y) - \langle \nabla f(y), x - y\rangle$ is the Bregman divergence associated with f between points x and y. Similarly, f is μ-strongly convex w.r.t. a *function* $\mathcal{R}$ if $\mathcal{D}_f(x, y) \geq \frac{\mu}{2}\mathcal{D}_\mathcal{R}(x, y)$ for all $x \in \mathcal{K}, y \in \mathcal{K}^\circ$,

[1] For $A \subset \mathbb{R}^d$, A° denotes the interior of A.

where $\mathcal{K}^\circ \subseteq \mathrm{dom}(\mathcal{R})$ and $\mathcal{R}$ is differentiable over $\mathcal{K}^\circ$. A function f is L-**smooth** w.r.t. a *norm* $\|\cdot\|$ for some $L > 0$ if

$$\mathcal{D}_f(x, y) \leq \tfrac{L}{2}\|x - y\|^2$$

for all $x \in \mathcal{K}, y \in \mathcal{K}^\circ$. This condition is equivalent to that ∇f is L-Lipschitz, that is, $\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|$ (Nesterov, 2004, Theorem 2.1.5).

We let $\mathcal{F}_{L,\mu,\mathcal{R}}(\mathcal{K})$ denote the class of functions that are μ-strongly convex w.r.t. $\mathcal{R}$ and L-smooth w.r.t. some norm $\|\cdot\|$ on the set $\mathcal{K}$ (typically, we will assume that $\mathcal{R}$ is also strongly convex w.r.t. $\|\cdot\|$). Note that $\mathcal{F}_{L,\mu,\mathcal{R}}(\mathcal{K})$ includes functions whose domain is larger than or equal to $\mathcal{K}$. We also let $\mathcal{F}_{L,\mu}(\mathcal{K})$ be $\mathcal{F}_{L,\mu,\mathcal{R}}(\mathcal{K})$ with $\mathcal{R}(\cdot) = \frac{1}{2}\|\cdot\|_2^2$. Then, the set of convex and L-smooth functions with domain including $\mathcal{K}$ is $\mathcal{F}_{L,0}(\mathcal{K})$.

Besides the standard $O(\cdot)$ notation, we will also use $\tilde{O}(\cdot)$: For a positive valued function $f : \mathbb{N} \to \mathbb{R}_+$, $\tilde{O}(f)$ contains any $g : \mathcal{N} \to \mathbb{R}_+$ such that $g = O(\log^p(n) f(n))$ for some $p > 0$. (As usual, we abuse notation by writing $g = O(f)$ instead of $g \in O(f)$.)

Finally, we will denote the indicator function of an event E by $\mathbb{I}\{E\}$, that is $\mathbb{I}\{E\} = 1$ if E holds and equals zero otherwise.

2.2 Type-I and Type-II Oracles

We will use two classes of oracles. In both cases, the oracles are specified by two functions $c_1, c_2 : [0, \infty) \to [0, \infty)$, which will be assumed to be continuous, monotonously increasing (resp., decreasing) with

$$\lim_{\delta \to 0} c_1(\delta) = 0 \text{ and } \lim_{\delta \to 0} c_2(\delta) = +\infty\,.$$

Typical choices for c_1, c_2 are $c_1(\delta) = C_1\delta^p$, $c_2(\delta) = C_2\delta^{-q}$ with $p, q > 0$.

Our type-I oracles are defined as follows:

Definition 1 ((c_1, c_2) type-I oracle) We say that γ is a (c_1, c_2) type-I oracle for $\mathcal{F}$, if for any function $f \in \mathcal{F}$, $x \in \mathcal{K}, 0 < \delta \leq 1$, γ returns $G \in \mathbb{R}^d$ and $Y \in \mathcal{K}$ random elements such that $\|x - Y\| \leq \delta$ almost surely (a.s.) and the following hold:

1. $\|\mathbb{E}[G] - \nabla f(x)\|_* \leq c_1(\delta)$ (bias); and

2. $\mathbb{E}\left[\|G - \mathbb{E}[G]\|_*^2\right] \leq c_2(\delta)$ (variance). □

The upper bound on δ is arbitrary: by changing the norm, any other value can also be accommodated. Also, the upper bound only matters when $\mathcal{K}$ is bounded and the functions in $\mathcal{F}$ are defined only in a small vicinity of $\mathcal{K}$.

The second type of oracles considered is as follows:

Definition 2 ((c_1, c_2) type-II oracle) We say that γ is a (c_1, c_2) type-II oracle for $\mathcal{F}$, if for any function $f \in \mathcal{F}$, $x \in \mathcal{K}, 0 < \delta \leq 1$, γ returns $G \in \mathbb{R}^d$ and $Y \in \mathcal{K}$ random elements such that $\|x - Y\| \leq \delta$ a.s. and the following hold:

1. There exists $\tilde{f} \in \mathcal{F}$ such that $\left\|\tilde{f} - f\right\|_\infty \leq c_1(\delta)$ and $\mathbb{E}[G] = \nabla \tilde{f}(x)$ (bias); and

2. $\mathbb{E}\left[\|G - \mathbb{E}[G]\|_*^2\right] \leq c_2(\delta)$ (variance). □

We will denote the set of type-I (type-II) oracles satisfying the (c_1, c_2)-requirements given a function $f \in \mathcal{F}$ by $\Gamma_1(f, c_1, c_2)$ (resp., $\Gamma_2(f, c_1, c_2)$).

Note that while a type-I oracle returns a biased, noisy gradient estimate for f, a type-II oracle returns an unbiased, noisy gradient estimate for some function $\tilde{f}$ which is close to f. Note that $\tilde{f}$ is allowed to change with the inputs (not only by f, but also with x and δ) in the definition. In fact, the oracles (in both cases) can have a memory of previous queries and depending on the memory can respond to the same inputs (x, δ, f) with a differently constructed pair.[2] The oracles that we use will nevertheless be memoryless.

As noted above, even a memoryless type-II oracle can respond such that $\tilde{f}$ depends on x or δ. A type-II oracle is called a *uniform* type-II oracle if $\tilde{f}$ only depends on f (and possibly the history of previous queries), but not on x and δ. The type-II oracles that will be explicitly constructed will all be uniform.

We call an oracle (type-I or II) *unbiased* if $\mathbb{E}[Y] = x$ in the above definitions. Note that if the oracle is unbiased and the loss function is smooth, an algorithm

[2]For oracles with memory, in the definition (and in the proofs provided later in the paper) the expectation should be replaced with an expectation that is conditioned on the past.

does not loose too much from suffering loss at Y instead of the query point x since in this case $\mathbb{E}[f(Y)] - f(x) \leq \mathbb{E}\left[\langle \nabla f(x), Y - x\rangle + \frac{L}{2}\|Y - x\|^2\right] \leq L\delta^2/2$.

Examples of specific oracle constructions will be given in Chapter 4. We also note that for type-II oracles we only need properties of the function class which the surrogate function $\tilde{f}$ belongs to, the assumption $f \in \mathcal{F}$ is only included to simplify the definition (e.g., some oracles work for non-convex functions f for which a suitable convex surrogate and the associated oracle exists).

2.3 Reduction Between Two Types of Oracles

As the next result shows, type-I and II oracles are closely related. In particular, a type-I oracle is also a type-II oracle (although not a uniform type-II oracle). On the other hand, type-II oracles need to satisfy an alternative condition to become type-I oracles as the closeness of $\tilde{f}$ and f is insufficient to conclude anything about the distance of their gradients:

Proposition 1 *Definition 1 is a sufficient condition for Definition 2, given a bounded $\mathcal{K}$. In particular, letting $R = \sup_{y\in\mathcal{K}} \|y\|$, for any f, c_1, c_2 such that $f + \langle c, \cdot\rangle \in \mathcal{F}$ for any $\|c\|_* \leq c_1(1)$, it holds that $\Gamma_1(f, c_1, c_2) \subset \Gamma_2(f, Rc_1, c_2)$. Furthermore, if $\left\|\tilde{f} - f\right\|_\infty \leq c_1(\delta)$ is replaced by*

$$\left\|\nabla\tilde{f} - \nabla f\right\|_* \leq c_1(\delta) \tag{2.1}$$

in Definition 2 (for all $x \in \mathcal{K}$ and $0 < \delta \leq 1$), then any oracle satisfying this modified definition is also a (c_1, c_2) type-I oracle. □

PROOF The second part of the claim is immediate from the definitions, hence it remains to prove the first part. Let γ be a (c_1, c_2) type-I oracle. Fix x, δ, f and let the oracle's response be G, Y. Define $\tilde{f} : \mathcal{K} \to \mathbb{R}$ by

$$\tilde{f}(y) = \mathbb{E}[f(y) + \langle G - \nabla f(x), y\rangle],$$

where the expectation is over the randomness of G (note that $\tilde{f}$ depends on x and δ). Then, $\nabla\tilde{f}(y) = \nabla f(y) - \nabla f(x) + \mathbb{E}[G]$ and thus substituting x for y we get

that $\nabla \tilde{f}(x) = \mathbb{E}[G]$. Further, using $\|\mathbb{E}[G] - \nabla f(x)\|_* \le c_1(\delta)$, we have, for any $y \in \mathcal{K}$,

$$|\tilde{f}(y) - f(y)| = |\mathbb{E}[\langle G - \nabla f(x), y\rangle]| \le \|\mathbb{E}[G] - \nabla f(x)\|_* \, \|y\| \le R\, c_1(\delta)\,.$$

showing that γ is also an (Rc_1, c_2) Type-II oracle, since $\tilde{f} \in \mathcal{F}$ by the conditions of the proposition. ∎

While in the online convex optimization setting algorithms are compared based on their minimax *regret* in the stochastic convex optimization setting, they are compared based on their minimax *error* (sometimes, also called as the "simple regret"). Both are defined with respect to a class of loss functions $\mathcal{F}$, and the bias/variance control functions c_1, c_2. The *worst-case regret* of algorithm $\mathcal{A}$ interacting with (c_1, c_2) type-I oracles for the function class $\mathcal{F}$ is defined as

$$R^{\mathcal{A}}_{\mathcal{F},n}(c_1, c_2) = \sup_{f_{1:n} \in \mathcal{F}^n} \sup_{\substack{\gamma_t \in \Gamma_1(f_t, c_1, c_2) \\ 1 \le t \le n}} R^{\mathcal{A}}_n(f_{1:n}, \gamma_{1:n})$$

where $R^{\mathcal{A}}_n(f_{1:n}, \gamma_{1:n})$ denotes the expected regret of $\mathcal{A}$ (against $f_{1:n}, \gamma_{1:n}$), and the *minimax expected regret* for $(\mathcal{F}, c_1, c_2)$ with type-I oracles is defined as

$$R^*_{\mathcal{F},n}(c_1, c_2) = \inf_{\mathcal{A}} R^{\mathcal{A}}_{\mathcal{F},n}(c_1, c_2),$$

where $\mathcal{A}$ ranges through all algorithms that interact with the loss sequence $f_{1:n} = (f_1, \ldots, f_n)$ through the oracles $\gamma_{1:n}$ (in round t, oracle γ_t is used). The minimax regret for type-II oracles is defined analogously.

In the stochastic BCO setting, the *worst case error* is defined through

$$\Delta^{\mathcal{A}}_{\mathcal{F},n}(c_1, c_2) = \sup_{f \in \mathcal{F}} \sup_{\gamma \in \Gamma_1(f, c_1, c_2)} \Delta^{\mathcal{A}}_n(f, \gamma)\,, \tag{2.2}$$

where $\Delta^{\mathcal{A}}_n(f, \gamma)$ is the optimization error that $\mathcal{A}$ suffers after n rounds of interaction with f through (a single) γ as described earlier, and the *minimax error* is defined as

$$\Delta^*_{\mathcal{F},n}(c_1, c_2) = \inf_{\mathcal{A}} \Delta^{\mathcal{A}}_{\mathcal{F},n}(c_1, c_2),$$

where, again, $\mathcal{A}$ ranges through all algorithms that interact with f through an oracle. The minimax error for type-II oracles is defined analogously.

Consider now the case when the set $\mathcal{K}$ is bounded and, in particular, assume that $\mathcal{K}$ is included in the unit ball w.r.t. $\|\cdot\|$. Assume further that the function set $\mathcal{F}$ is invariant to linear shifts (that is for any $f \in \mathcal{F}$, $w \in \mathbb{R}^d$, $x \mapsto f(x) + \langle x, w \rangle$ is also in $\mathcal{F}$). Let $\Delta_n^{\text{type-I}}$ and $\Delta_n^{\text{type-II}}$ denote the appropriate minimax errors for the two types of oracles. Then, by the construction in Proposition 1,

$$\Delta_{\mathcal{F},n}^{\text{type-I}}(c_1, c_2) \leq \Delta_{\mathcal{F},n}^{\text{type-II}}(Rc_1, c_2)\,. \tag{2.3}$$

Note that R may depend on the dimension d, e.g., for $\mathcal{K} = [-1, 1]^d$, $R = \sqrt{d}$ when using the Euclidean norm. To clarify the different c_1 used by type-I and II oracles, we will present the upper and lower bounds separately for the two oracle types, although the type-I upper bound can actually be derived from type-II (and the type-II lower bound can be derived from type-I). Also note that for either type of oracles, $\Delta^*_{\mathcal{F},n}(c_1, c_2) \leq R^*_{\mathcal{F},n}(c_1, c_2)/n$. This follows by the well known construction that turns an online convex optimization method $\mathcal{A}$ for regret minimization into an optimization method by running the method and at the end choosing $\hat{X}_n$ as the average of the points $X_1, \ldots, X_n$ queried by $\mathcal{A}$ during the n rounds. Indeed, then $f(\hat{X}_n) \leq \frac{1}{n}\sum_{t=1}^n f(X_t)$ by Jensen's inequality, hence the average regret of $\mathcal{A}$ will upper bound the error of choosing $\hat{X}_n$ at the end. A consequence of this relation is that a lower bound for $\Delta^*_{\mathcal{F},n}(c_1, c_2)$ will also be a lower bound for $R^*_{\mathcal{F},n}(c_1, c_2)/n$ and an upper bound on $R^*_{\mathcal{F},n}(c_1, c_2)$ leads to an upper bound on $\Delta^*_{\mathcal{F},n}(c_1, c_2)$. This explains why we allowed taking supremum over time-varying oracles in the definition of the regret and why we used a static oracle for the optimization error: to maximize the strength of the bounds we obtain.

2.4 Previous Work

Gradient oracles have been considered in a number of papers before: Several previous works assume that the accuracy requirements hold with probability one (d'Aspremont, 2008; Baes, 2009; Devolder et al., 2014) or consider adversarial noise (Schmidt et al., 2011). Gradient oracles with stochastic noise, which is central to our development, were also considered (Juditsky and Nemirovski, 2011; Honorio, 2012; Dvurechensky and Gasnikov, 2015); however, these papers assume that the bias

and the variance are controlled separately, and consider the performance of special algorithms (in some cases in special setups). A full comparison between these oracle models is given by Devolder et al. (2014). For illustration, here we only review the model of this latter paper as a typical example of these previous works.

The model of Devolder et al. (2014) assumes a first-order approximation to the function with parameters (δ, L). In particular, given (x, δ, L) and the convex function f, the oracle gives a pair $(t, g) \in \mathbb{R} \times \mathbb{R}^d$ such that $t + \langle g, \cdot - x \rangle$ is a linear lower approximation to $f(\cdot)$ in the sense that

$$0 \leq f(y) - \{t + \langle g, y - x \rangle\} \leq \frac{L}{2} \|y - x\|^2 + \delta \,.$$

Devolder et al. (2014) argue that this notion appears naturally in several optimization problems and study whether the so-called accelerated gradient techniques are still superior to their non-accelerated counterparts (and find a negative answer). The authors study both lower and upper rates of convergence, similarly to our paper.

A major difference between the previous and our settings is that we allow stochastic noise (and bias), which the algorithms can control, while the oracle in these previous paper must guarantee that the accuracy requirements hold in each time step with probability one. This is a much stronger requirement, which may be impossible to satisfy in some problems, such as when the only information available about the functions is noise contaminated. Some works, such as Schmidt et al. (2011) allow arbitrary sequences of errors and show error bounds as a function of the accumulated errors.

Our proof technique is actually essentially the same (as can be expected). However, the noisy case requires special care. For example, Proposition 3 of Schmidt et al. (2011) bounds the optimization error for the smooth, convex case by

$$O(1/n^2(\|x_1 - x^*\|^2 + A_n^2)) \,,$$

where $A_n = O(\sum_{t=1}^n t \|e_t\|)$, e_t being the error of the approximate gradient. This expression becomes $\Theta(\frac{1}{n^2} \sum_{t=1}^n t^2) \approx n$ assuming that errors' noise level is a positive constant (in all our result, this holds). This clearly shows that the noisy case requires (somewhat) special treatment. Similar, but simpler noisy oracle models

were introduced (Juditsky and Nemirovski, 2011; Honorio, 2012; Dvurechensky and Gasnikov, 2015), but these models lack the bias-variance tradeoff central to this paper (i.e., they assume the variance and bias can be controlled independently of each other). The results in these papers are upper bounds on the error of certain gradient methods (also to some very specific problem for Honorio (2012)), and they correspond to the bounds we obtained with $q = 0$.

Chapter 3

Main Results

This chapter has the most essential contributions of our work. Upper and lower bounds are presented for both stochastic BCO (optimization error) and online BCO (expected regret). We have structured the analysis in such a way that the role of objective functions and gradient estimates becomes clearer in our results, i.e., $\mathcal{F}$ specifies the class of objective functions, the type-I or II oracle illustrates the properties and bias-variance tradeoff of gradient estimates. Table 3.1 summarizes the upper and lower bounds[1] for two specific choices of p and q (relevant to applications in Chapter 5 and Chapter 6). These bounds can be inferred from the results in Theorems 1 and 2.[2] In Section 3.5, we apply our gradient oracle model to give a

[1]Note that when $\mathcal{R}$ is the squared norm and $\mathcal{K}$ is the hypercube (as in the lower bounds), $D = \Theta(d)$ in the upper bounds and also that C_1, C_2 may hide dimension-dependent quantities for the common gradient estimators, as will be discussed later.

[2]While it appears that for the strongly convex case the error becomes smaller with a larger dimension, in most applications C_1, C_2 will hide dimension dependent constants, and the lower bound

Type-I Oracle	**Convex + Smooth**		**Strongly Convex + Smooth**	
	Upper bound	**Lower bound**	**Upper bound**	**Lower bound**
δ-bias, δ^{-2}-variance ($p = 1$, $q = 2$)	$\left(\frac{C_1^2 C_2 D^2}{n}\right)^{1/4}$	$\left(\frac{C_1^2 C_2 d^2}{n}\right)^{1/4}$	$\left(\frac{C_1^2 C_2 D}{n}\right)^{1/3}$	$\left(\frac{C_1^2 C_2}{n}\right)^{1/2}$
δ^2-bias, δ^{-2}-variance ($p = 2$, $q = 2$)	$\left(\frac{C_1 C_2 \sqrt{D^3}}{n}\right)^{1/3}$	$\left(\frac{C_1 C_2 \sqrt{d^3}}{n}\right)^{1/3}$	$\left(\frac{C_1 C_2 \sqrt{D}}{n}\right)^{1/2}$	$\left(\frac{C_1 C_2}{n}\right)^{2/3}$

Table 3.1: Summary of upper and lower bounds on the minimax optimization error for different smooth function classes and gradient oracles for the settings of Theorem 1 and Theorem 2.

Algorithm 1 Mirror Descent with Type-I/II Oracle.

Input: Closed convex set $\mathcal{K} \neq \emptyset$, regularization function $\mathcal{R} : \mathrm{dom}(\mathcal{R}) \to \mathbb{R}$, $\mathcal{K}^\circ \subset \mathrm{dom}(\mathcal{R})$, tolerance parameter δ, learning rates $\{\eta_t\}_{t=1}^{n-1}$.
Initialize $X_1 \in \mathcal{K}$ arbitrarily.
for $t = 1, 2, \cdots, n-1$ **do**
  Query the oracle at X_t to receive G_t, Y_t.
  Set $X_{t+1} = \mathrm{argmin}_{x \in \mathcal{K}} \left[\eta_t \langle G_t, x\rangle + D_{\mathcal{R}}(x, X_t)\right]$.
Return: $\hat{X}_n = \frac{1}{n}\sum_{t=1}^{n} X_t$.

lower bound to the algorithm proposed in Dekel et al., 2015, which invalidates their upper bounds and enhanced our statement about gradient methods.

3.1 Upper Bounds of the Minimax Error

First we give an upper bound for the mirror-descent algorithm shown as Algorithm 1. In the algorithm, we assume that the regularizer function $\mathcal{R}$ is α-strongly convex and the target function f is smooth or smooth and strongly convex. We give results for polynomial oracles, that is, when c_1 and c_2 are polynomial functions (in particular, monomial functions) of their argument. The reason, as we will see, is that existing oracle constructions give rise to polynomial oracles for the function classes that we consider.

Theorem 1 (*Upper bound*) *Consider the class $\mathcal{F} = \mathcal{F}_{L,0}$ of convex, L-smooth functions whose domain includes the bounded, convex set $\mathcal{K} \neq \emptyset$, $\mathcal{K} \subset \mathbb{R}^d$. Assume that the regularization function $\mathcal{R}$ is α-strongly convex with respect to (w.r.t.) some norm $\|\cdot\|$, and $\mathcal{K}^\circ \subseteq \mathrm{dom}(\mathcal{R})$. For any (c_1, c_2) type-I or any memoryless uniform (c_1, c_2) type-II oracle with $c_1(\delta) = C_1\delta^p$, $c_2(\delta) = C_2\delta^{-q}$, $p, q > 0$, the worst-case error (and hence the minimax error) of Algorithm 1 run with an appropriate parameter setting can be bounded as*

$$\Delta^{MD,\mathrm{type-I}}_{\mathcal{F}_{L,0},n}(c_1, c_2) \leq K_1 D^{\frac{1}{2}} C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}} n^{-\frac{p}{2p+q}} + \frac{1}{n}\left(\mathbb{E}\left[f(X_1) - \inf_{x\in\mathcal{K}} f(x)\right] + \frac{DL}{\alpha}\right), \tag{3.1}$$

actually increases with the dimension increasing.

$$\Delta^{MD,\text{type-II}}_{\mathcal{F}_{L,0},n}(c_1,c_2) \leq K_1' D^{\frac{p}{2p+q}} C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}} n^{-\frac{p}{2p+q}} + \frac{1}{n}\left(\mathbb{E}\left[f(X_1) - \inf_{x\in\mathcal{K}} f(x)\right] + \frac{DL}{\alpha}\right), \tag{3.2}$$

where $D = \sup_{x,y\in\mathcal{K}} \mathcal{D}_{\mathcal{R}}(x,y)$. *For the class* $\mathcal{F} = \mathcal{F}_{L,\mu,\mathcal{R}}$ *of* μ*-strongly convex (w.r.t.* $\mathcal{R}$*) and* L*-smooth functions, with* $\alpha > 2L/\mu$*, we have*

$$\Delta^{MD,\text{type-I}}_{\mathcal{F}_{L,\mu,\mathcal{R}},n}(c_1,c_2) \leq K_2 D^{\frac{q}{2(p+q)}} C_1^{\frac{q}{p+q}} C_2^{\frac{p}{p+q}} \left(\frac{\log n + 1 + \frac{\alpha\mu}{\alpha\mu - 2L}}{n}\right)^{\frac{p}{p+q}} + \frac{1}{n}\mathbb{E}\left[f(X_1) - \inf_{x\in\mathcal{K}} f(x)\right]. \tag{3.3}$$

$$\Delta^{MD,\text{type-II}}_{\mathcal{F}_{L,\mu,\mathcal{R}},n}(c_1,c_2) \leq K_2' C_1^{\frac{q}{p+q}} C_2^{\frac{p}{p+q}} \left(\frac{\log n + 1 + \frac{\alpha\mu}{\alpha\mu - 2L}}{n}\right)^{\frac{p}{p+q}} + \frac{1}{n}\mathbb{E}\left[f(X_1) - \inf_{x\in\mathcal{K}} f(x)\right]. \tag{3.4}$$

Above, the constants K_1*,* K_1'*,* K_2 *and* K_2' *depend on* p, q, α, μ*.*[3]

If the oracle is unbiased (but may be non-uniform and may have memory) and either (i) the oracle is of type-I or (ii) the oracle is of type-II and all functions in $\mathcal{F}$ *have bounded gradients*[4] *then, for* $\mathcal{F} \subset \mathcal{F}_{L,0}$*, the regret of Algorithm 1 run with an appropriate parameter setting can be bounded as*

$$\frac{1}{n} R^{MD}_{\mathcal{F}}(c_1,c_2) = O\left(D^{\frac{\hat{p}}{2\hat{p}+q}} \hat{C}_1^{\frac{q}{2\hat{p}+q}} C_2^{\frac{\hat{p}}{2\hat{p}+q}} n^{-\frac{\hat{p}}{2\hat{p}+q}}\right)$$

where $\hat{p} = \min\{p, 2\}$*,* $\hat{C}_1 = C_1 \mathbb{I}\{p \leq 2\} + (L/4)\mathbb{I}\{p \geq 2\}$ *for type-II oracles and* $\hat{C}_1 = RC_1 \mathbb{I}\{p \leq 2\} + (L/4)\mathbb{I}\{p \geq 2\}$ *for type-I oracles where* $R = \sup_{x\in\mathcal{K}} \|x\|$*.*[5] *In the strongly convex case, that is, when* $\mathcal{F} \subset \mathcal{F}_{L,\mu}$*, an appropriate parameter setting of Algorithm 1 yields a regret bound*[6]

$$\frac{1}{n} R^{MD}_{\mathcal{F}}(c_1,c_2) = O\left(\hat{C}_1^{\frac{q}{\hat{p}+q}} C_2^{\frac{\hat{p}}{\hat{p}+q}} n^{-\frac{\hat{p}}{\hat{p}+q}} (1+\log n)^{\frac{\hat{p}}{\hat{p}+q}}\right). \qquad \square$$

[3] In particular, $K_1 = 2^{\frac{q}{2(2p+q)}}\left(\alpha^{-1} + 2\alpha^{-\frac{q}{2(p+q)}}\right)\left(\frac{2p+q}{2p}\right)^{\frac{p}{2p+q}}$, $K_1' = 3\left(2+\frac{2}{n}\right)^{\frac{q}{2p+q}} \alpha^{-\frac{p}{2p+q}} \left(\frac{2p+q}{2p}\right)^{\frac{p}{2p+q}}$, $K_2 = 2^{\frac{q}{2(p+q)}} \alpha^{-\frac{2p+q}{2(p+q)}} \mu^{-\frac{p}{p+q}}$ and $K_2' = 2^{\frac{q}{p+q}} \alpha^{-\frac{p}{p+q}} \mu^{-\frac{p}{p+q}}$.

[4] This follows from the smoothness if, for example, the functions in f are bounded.

[5] The coefficient associated with the dominating term of the bound is $2^{1+\frac{q/2}{2\hat{p}+q}}(2\hat{p}+q)(2\hat{p}\alpha)^{-\frac{\hat{p}}{2\hat{p}+q}} q^{-\frac{q}{2\hat{p}+q}}$.

[6] The coefficient associated with the main term of the bound is $(\hat{p}+q)\hat{p}^{-\frac{\hat{p}}{\hat{p}+q}} q^{-\frac{q}{\hat{p}+q}} (\alpha\mu)^{-\frac{\hat{p}}{\hat{p}+q}}$.

The proof of this theorem follows the steps of the standard analysis of the mirror descent algorithm and is provided in Section 3.2, mainly for completeness and because it is somewhat cumbersome to extract from the existing results what properties of the oracles they use. Comparing the bounds on the optimization error and the regret for the non-strongly convex case, note that $\hat{p}$ plays the same role as p and $\hat{C}_1$ as C_1. The reason for the difference is that the extra loss introduced by using Y_t instead of X_t in the regret minimization case brings in an extra $L\delta^2/2$ term (as discussed at the introduction of unbiased oracles), and this term dominates the $C_1\delta^p$ bias term when $p > 2$, and increases its coefficient for $p = 2$; $\hat{p}$ and $\hat{C}_1$ are obtained as the exponent and the coefficient of the dominating term from these two. On another note, the dependence on D for type-I oracles seems different for the optimization and the regret minimization cases. However, by the strong convexity of $\mathcal{R}$, $R \leq \sqrt{2D/\alpha}$ (when $\mathcal{R}$ is also L'-smooth, $R \geq \sqrt{2D/L'}$, so R is of the same order as $\sqrt{D}$); applying this inequality gives the same dependence on D for both types of oracles (for $p \geq 2$, the main term scales with a smaller power of D for regret minimization due to the approximation issues discussed beforehand).

3.2 Proofs of the Upper Bounds

In this section we prove Theorem 1. First we derive the bounds for the optimization settings and then for the regret.

3.2.1 Stochastic Optimization

The proof for the stochastic optimization scenario is based on Lemma 1 stated below. This is essentially Theorem C.4 of Mahdavi (2014), and also identical to Theorem 6.3 of Bubeck (2014), who cites Dekel et al. (2012) as the source. For completeness, the proof of the lemma is given in Section 3.2.3. The lemma is somewhat more general than what we need (we will only need it for the case when $\beta_t = 0$); the general form is presented because its proof is not significantly different than the simpler form and it may find other applications in the future.

Lemma 1 *Let $(\mathfrak{F}_t)_t$ be a filtration such that X_t is $\mathfrak{F}_t$-measurable. Let $\overline{G}_t = \mathbb{E}\left[G_t|\mathfrak{F}_t\right]$*

and assume that the nonnegative real-valued deterministic sequence $(\beta_t)_{1\le t\le n}$ *is such that* $\left\|\overline{G}_t - \nabla f(X_t)\right\|_* \le \beta_t$ *holds almost surely. Further, assume that* $\mathcal{R}$ *is* α*-strongly convex with respect to* $\|\cdot\|$, $D = \sup_{x,y\in\mathcal{K}} D_{\mathcal{R}}(x,y) < \infty$, *and let* $\eta_t = \frac{\alpha}{a_t+L}$ *for some increasing sequence* $(a_t)_{t=1}^{n-1}$ *of numbers. Then, the cumulative loss of Algorithm 1 for a fixed convex and* L*-smooth function* f *can be bounded as*

$$\mathbb{E}\left[\sum_{t=1}^{n} f(X_t) - f(x)\right] \le \mathbb{E}\left[f(X_1) - f(x)\right] + \sqrt{\tfrac{2D}{\alpha}}\sum_{t=1}^{n-1}\beta_t + \frac{D(a_{n-1}+L)}{\alpha} + \sum_{t=1}^{n-1}\frac{\sigma_t^2}{2a_t}\,,$$

where $\sigma_t^2 = \mathbb{E}\left[\left\|G_t - \overline{G}_t\right\|_*^2\right]$ *is the "variance" of* G_t.

If f *is also* μ*-strongly convex with respect to* $\mathcal{R}$ *with* $\mu > 2L/\alpha$, *then letting* $\eta_t = \dfrac{2}{\mu t}$ *and* $a_t = \alpha\mu t/2 - L > 0$, *the cumulative loss of Algorithm 1 can be bounded as*

$$\mathbb{E}\left[\sum_{t=1}^{n} f(X_t) - f(x)\right] \le \mathbb{E}\left[f(X_1) - f(x)\right] + \sqrt{\tfrac{2D}{\alpha}}\sum_{t=1}^{n-1}\beta_t + \sum_{t=1}^{n-1}\frac{\sigma_t^2}{2a_t}\,. \qquad \square$$

Now we can easily prove the theorem. First we consider the case of smooth and convex functions. We select

$$\eta_t = \alpha/(a_t + L)$$

as in the lemma with $a_t = at^r$ for some $0 < r < 1$. For type-I oracles, the result immediately follows by substituting $\beta_t = C_1\delta^p$, $\sigma_t^2 = C_2\delta^{-q}$, using that $\sum_{t=1}^{n-1} t^{-r} \le 1 + \int_1^n t^{-r} \le n^{1-r}/(1-r)$:

$$\begin{aligned}\frac{1}{n}\mathbb{E}&\left[\sum_{t=1}^{n} f(X_t) - \inf_{x\in\mathcal{K}}\sum_{t=1}^{n} f(x)\right] \\ &\le \frac{1}{n}\left(\mathbb{E}\left[f(X_1) - \inf_{x\in\mathcal{K}} f(x)\right] + \frac{DL}{\alpha}\right) + \sqrt{\frac{2D}{\alpha}}C_1\delta^p + \frac{Da}{\alpha}n^{r-1} + \frac{C_2\delta^{-q}}{2a(1-r)}n^{-r}\,.\end{aligned} \tag{3.5}$$

Choosing

$$\begin{aligned} r &= \tfrac{p+q}{2p+q}, \\ a &= 2^{\frac{q}{2(2p+q)}}\left(\tfrac{2p+q}{2p}\right)^{\frac{p}{2p+q}} D^{-\frac{1}{2}} C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}} \\ \delta &= \alpha^{\frac{1}{2(p+q)}}\left(\tfrac{2p+q}{4p}\right)^{\frac{1}{2p+q}} C_1^{-\frac{2}{2p+q}} C_2^{\frac{1}{2p+q}} n^{-\frac{1}{2p+q}}, \end{aligned}$$

the last 3 terms in (3.5) are optimized to

$$K_1 D^{1/2} C_1^{q/(2p+q)} C_2^{p/(2p+q)} n^{-p/(2p+q)} ,$$

with $K_1 = 2^{\frac{q}{2(2p+q)}} \left(\alpha^{-1} + 2\alpha^{-\frac{q}{2(p+q)}}\right) \left(\frac{2p+q}{2p}\right)^{\frac{p}{2p+q}}$. This implies (3.1).

For type-II oracles, from the bias condition in Definition 2 and using that the oracle is memoryless and uniform, we get

$$\frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n} f(X_t) - \inf_{x\in\mathcal{K}} \sum_{t=1}^{n} f(x)\right] \le \frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n} \tilde{f}(X_t) - \inf_{x\in\mathcal{K}} \sum_{t=1}^{n} \tilde{f}(x)\right] + 2C_1\delta^p .$$

Given $\overline{G}_t = \mathbb{E}[G_t] = \nabla\tilde{f}(X_t)$, where $\tilde{f} \in \mathcal{F}_{L,0}$ is convex and smooth, the result immediately follows by applying Lemma 1 to $\tilde{f}$. Substituting $\beta_t = 0$ (since we have a type-II oracle), $\sigma_t^2 = C_2\delta^{-q}$, respectively, and using the bias condition again, we obtain

$$\begin{aligned}
&\frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n} f(X_t) - \inf_{x\in\mathcal{K}} \sum_{t=1}^{n} f(x)\right] \\
&\quad\le \frac{1}{n}\left(\mathbb{E}\left[\tilde{f}(X_1) - \inf_{x\in\mathcal{K}} \tilde{f}(x)\right] + \frac{DL}{\alpha}\right) + \frac{Da}{\alpha}n^{r-1} + \frac{C_2\delta^{-q}}{2a(1-r)}n^{-r} + 2C_1\delta^p \qquad (3.6)\\
&\quad\le \frac{1}{n}\left(\mathbb{E}\left[f(X_1) - \inf_{x\in\mathcal{K}} f(x)\right] + \frac{DL}{\alpha}\right) + \frac{Da}{\alpha}n^{r-1} + \frac{C_2\delta^{-q}}{2a(1-r)}n^{-r} + \left(2+\frac{2}{n}\right)C_1\delta^p . \qquad (3.7)
\end{aligned}$$

Choosing

$$\begin{aligned}
r &= \tfrac{p+q}{2p+q},\\
a &= \left(2+\tfrac{2}{n}\right)^{\frac{q}{2p+q}} \left(\tfrac{2p+q}{2p}\right)^{\frac{p}{2p+q}} \left(\tfrac{D}{\alpha}\right)^{-\frac{p+q}{2p+q}} C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}}\\
\delta &= \left(2+\tfrac{2}{n}\right)^{-\frac{2}{2p+q}} \left(\tfrac{2p+q}{2p}\right)^{\frac{1}{2p+q}} \left(\tfrac{D}{\alpha}\right)^{\frac{1}{2p+q}} C_1^{-\frac{2}{2p+q}} C_2^{\frac{1}{2p+q}} n^{-\frac{1}{2p+q}},
\end{aligned}$$

the last 3 terms in (3.7) are optimized to

$$K_1' D^{p/(2p+q)} C_1^{q/(2p+q)} C_2^{p/(2p+q)} n^{-p/(2p+q)} ,$$

where $K_1' = 3\left(2+\frac{2}{n}\right)^{\frac{q}{2p+q}} \left(\frac{2p+q}{2p}\right)^{\frac{p}{2p+q}} \alpha^{-\frac{p}{2p+q}}$. This implies (3.2).

When $\tilde{f} \in \mathcal{F}_{L,\mu,\mathcal{R}}$ is L-smooth and μ-strongly convex, for $\eta_t = 2/(\mu t)$ and $\delta^{p+q} = \frac{C_2\left(\log n+1+\frac{\alpha\mu}{\alpha\mu-2L}\right)}{\sqrt{2D\alpha}\mu C_1 n}$, we similarly obtain, for type-I oracle,

$$\begin{aligned}
&\frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n} f(X_t) - \inf_{x\in\mathcal{K}}\sum_{t=1}^{n} f(x)\right] - \frac{1}{n}\mathbb{E}\left[f(X_1) - \inf_{x\in\mathcal{K}} f(x)\right] \\
&\quad\le \sqrt{\frac{2D}{\alpha}}C_1\delta^p + \frac{C_2\delta^{-q}}{\alpha\mu n}\sum_{t=1}^{n-1}\frac{1}{t-\frac{2L}{\alpha\mu}} \\
&\quad\le \sqrt{\frac{2D}{\alpha}}C_1\delta^p + \frac{C_2}{\alpha\mu}\delta^{-q}\frac{\log n + 1 + \alpha\mu/(\alpha\mu-2L)}{n} \\
&\quad\le 2^{\frac{q}{2(p+q)}}\alpha^{-\frac{2p+q}{2(p+q)}}\mu^{-\frac{p}{p+q}}D^{\frac{q}{2(p+q)}}C_1^{\frac{q}{p+q}}C_2^{\frac{p}{p+q}}\left(\frac{\log n + 1 + \frac{\alpha\mu}{\alpha\mu-2L}}{n}\right)^{\frac{p}{p+q}}.
\end{aligned}$$

For type-II oracle, choosing $\delta^{p+q} = \frac{C_2\left(\log n+1+\frac{\alpha\mu}{\alpha\mu-2L}\right)}{2\alpha\mu C_1(n+1)}$, we get

$$\begin{aligned}
&\frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n} f(X_t) - \inf_{x\in\mathcal{K}}\sum_{t=1}^{n} f(x)\right] - \frac{1}{n}\mathbb{E}\left[f(X_1) - \inf_{x\in\mathcal{K}} f(x)\right] \\
&\quad\le (2+\frac{2}{n})C_1\delta^p + \frac{C_2\delta^{-q}}{\alpha\mu n}\sum_{t=1}^{n-1}\frac{1}{t-\frac{2L}{\alpha\mu}} \\
&\quad\le (2+\frac{2}{n})C_1\delta^p + \frac{C_2}{\alpha\mu}\delta^{-q}\frac{\log n + 1 + \alpha\mu/(\alpha\mu-2L)}{n} \\
&\quad\le 2^{\frac{q}{p+q}}\alpha^{-\frac{p}{p+q}}\mu^{-\frac{p}{p+q}}K_2C_1^{\frac{q}{p+q}}C_2^{\frac{p}{p+q}}\left(\frac{\log n + 1 + \frac{\alpha\mu}{\alpha\mu-2L}}{n}\right)^{\frac{p}{p+q}},
\end{aligned}$$

where the bound is optimized in the last step via the choice of δ.

3.2.2 Online Optimization

The proof in this section follows closely the derivation of Saha and Tewari (2011). First we consider the case of type-II oracles.

Let $\mathfrak{F}_t$ denote the σ-algebra of all random events up until and including the selection of X_t. Since the oracle is unbiased, that is, $\mathbb{E}\left[Y_t|\mathfrak{F}_t\right] = X_t$, we have

$$\mathbb{E}\left[\tilde{f}_t(Y_t) - \tilde{f}_t(X_t)|\mathfrak{F}_t\right] \le \mathbb{E}\left[\langle\nabla\tilde{f}_t(X_t), Y_t - X_t\rangle + \tfrac{L}{2}\|X_t - Y_t\|^2|\mathfrak{F}_t\right] \le L\delta^2/2\,. \tag{3.8}$$

This inequality, the definition of type-II oracles, and the convexity of $\tilde{f}_t$ implies, for any $x \in \mathcal{K}$,

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{n} f_t(Y_t)\right] - \sum_{t=1}^{n} f_t(x) &\le \mathbb{E}\left[\sum_{t=1}^{n} \tilde{f}_t(Y_t) - \sum_{t=1}^{n} \tilde{f}_t(x)\right] + 2nC_1\delta^p \\
&\le \mathbb{E}\left[\sum_{t=1}^{n} \tilde{f}_t(X_t) - \sum_{t=1}^{n} \tilde{f}_t(x)\right] + 2nC_1\delta^p + \frac{nL\delta^2}{2} \\
&\le \mathbb{E}\left[\sum_{t=1}^{n} \langle \nabla \tilde{f}_t(X_t), X_t - x\rangle\right] + 2nC_1\delta^p + \frac{nL\delta^2}{2} \qquad (3.9) \\
&= \mathbb{E}\left[\sum_{t=1}^{n} \langle G_t, X_t - x\rangle\right] + 2nC_1\delta^p + \frac{nL\delta^2}{2}\,. \qquad (3.10)
\end{aligned}$$

Instead of Lemma 2 used in the optimization proof, we apply the prox-lemma (see, e.g., Beck and Teboulle, 2003; Nemirovski et al., 2009):

$$\langle G_t, X_t - x\rangle \le \frac{1}{\eta_t}\big(D_{\mathcal{R}}(x, X_t) - D_{\mathcal{R}}(x, X_{t+1})\big) + \eta_t \frac{\|G_t\|_*^2}{2\alpha}\,. \qquad (3.11)$$

Summing up the above bound for all t, the divergence terms telescope, since

$$\begin{aligned}
&\sum_{t=1}^{n-1} \frac{1}{\eta_t}\left(\mathcal{D}_{\mathcal{R}}(x, X_t) - \mathcal{D}_{\mathcal{R}}(x, X_{t+1})\right) \\
=&\mathcal{D}_{\mathcal{R}}(x, X_1)\frac{1}{\eta_1} + \mathcal{D}_{\mathcal{R}}(x, X_2)\left(\frac{1}{\eta_2} - \frac{1}{\eta_1}\right) + \dots \\
&+ \mathcal{D}_{\mathcal{R}}(x, X_{n-1})\left(\frac{1}{\eta_{n-1}} - \frac{1}{\eta_{n-2}}\right) - \frac{1}{\eta_{n-1}}\mathcal{D}_{\mathcal{R}}(x, X_n) \\
\le&\frac{D}{\eta_1} + D\sum_{t=2}^{n-1}\left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right) \\
=&\frac{D}{\eta_{n-1}}\,, \qquad (3.12)
\end{aligned}$$

where the inequality results from the fact that $\{\eta_t\}$ is non-increasing.

To bound the last term in (3.11), we use the assumption $\|\nabla \tilde{f}(x)\|_* \le M$ for all $x \in \mathcal{K}$ to obtain

$$\mathbb{E}\left[\|G_t\|_*^2 \,|\mathfrak{F}_t\right] \le 2\mathbb{E}\left[\left\|G_t - \nabla \tilde{f}_t(X_t)\right\|_*^2 + \left\|\nabla \tilde{f}_t(X_t)\right\|_*^2 \Big|\mathfrak{F}_t\right] \le 2(M^2 + C_2\delta^{-q}), \qquad (3.13)$$

Combining the latter with (3.10), (3.11), and (3.12), we obtain, for any $x \in \mathcal{K}$,

$$\mathbb{E}\left[\sum_{t=1}^{n} f_t(Y_t)\right] - \sum_{t=1}^{n} f_t(x) \leq \frac{D}{\eta_{n-1}} + \sum_{t=1}^{n} \eta_t \frac{M^2 + C_2\delta^{-q}}{\alpha} + 2nC_1\delta^p + \frac{nL\delta^2}{2} . \tag{3.14}$$

Setting the parameters

$$\delta = \left(\frac{q}{2p'}\right)^{\frac{2}{2\hat{p}+q}} \left(\frac{C_2 D}{\alpha \hat{C}_1^2}\right)^{\frac{1}{2\hat{p}+q}} n^{-\frac{1}{2\hat{p}+q}} ,$$

where $\hat{p} = \min\{p, 2\}$, $\hat{C}_1 = C_1 \mathbb{I}\{p \leq 2\} + (L/4)\mathbb{I}\{p \geq 2\}$ (i.e., $\hat{p}$ is the dominating exponent from δ^p and δ^2, and $\hat{C}_1$ is the coefficient of the dominating term),

$$\eta_t = D^{\frac{\hat{p}+q}{2\hat{p}+q}} \left(\frac{q}{2\hat{p}}\right)^{\frac{q}{2\hat{p}+q}} \left(\frac{C_2}{\alpha}\right)^{-\frac{\hat{p}}{2\hat{p}+q}} \hat{C}_1^{-\frac{q}{2\hat{p}+q}} n^{-\frac{\hat{p}+q}{2\hat{p}+q}} .$$

When $f_t \in \mathcal{F}_{L,0}$ for all t, it gives that

$$\frac{1}{n}\left(\mathbb{E}\left[\sum_{t=1}^{n} f_t(Y_t)\right] - \inf_{x\in\mathcal{K}} \sum_{t=1}^{n} f_t(x)\right) = O\left(\hat{C}_1^{\frac{q}{2\hat{p}+q}} (C_2 D)^{\frac{\hat{p}}{2\hat{p}+q}} n^{-\frac{\hat{p}}{2\hat{p}+q}}\right) \tag{3.15}$$

where the coefficient of the main term equals $K = 2^{1+\frac{q/2}{2\hat{p}+q}} (2\hat{p}+q)(2\hat{p}\alpha)^{-\frac{\hat{p}}{2\hat{p}+q}} q^{-\frac{q}{2\hat{p}+q}}$.

When the set of functions is also strongly convex, in (3.9) we can use strong convexity instead of linearization:

$$\tilde{f}(X_t) - \tilde{f}(x) \leq \langle \nabla \tilde{f}_t, X_t - x\rangle - \frac{\mu}{2} D_{\mathcal{R}}(x, X_t) = \mathbb{E}\left[\langle G_t, X_t - x\rangle | \mathfrak{F}_t\right] - \frac{\mu}{2} D_{\mathcal{R}}(x, X_t) .$$

Combining this with (3.11) and (3.13) gives the well-known variant of (3.14) for strongly convex loss functions (Bartlett et al., 2008) for the choice $\eta_t = 2/(t\mu)$:

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{n} f_t(Y_t)\right] - \sum_{t=1}^{n} f_t(x) &\leq \sum_{t=1}^{n} \frac{\mathbb{E}\left[\|G_t\|_*^2\right]}{t\alpha\mu} + 2nC_1\delta^p + \frac{nL\delta^2}{2} \\
&\leq \frac{\max_t \mathbb{E}\left[\|G_t\|_*^2\right]}{\alpha\mu}(1 + \log n) + 2nC_1\delta^p + \frac{nL\delta^2}{2} \\
&\leq \frac{2(M^2 + C_2\delta^{-q})}{\alpha\mu}(1 + \log n) + 2nC_1\delta^p + \frac{nL\delta^2}{2} .
\end{aligned}$$

Setting $\delta = \left(\frac{C_2 q(1+\log n)}{\alpha\mu\hat{C}_1\hat{p}n}\right)^{\frac{1}{\hat{p}+q}}$, we obtain

$$\frac{1}{n}\left(\mathbb{E}\left[\sum_{t=1}^{n} \tilde{f}_t(Y_t)\right] - \inf_{x\in\mathcal{K}} \sum_{t=1}^{n} \tilde{f}_t(x)\right) = O\left(\hat{C}_1^{\frac{q}{\hat{p}+q}} C_2^{\frac{\hat{p}}{\hat{p}+q}} n^{-\frac{\hat{p}}{\hat{p}+q}} (1 + \log n)^{\frac{\hat{p}}{\hat{p}+q}}\right) , \tag{3.16}$$

where the coefficient of the leading term is $K' = (\hat{p}+q)\hat{p}^{-\frac{\hat{p}}{\hat{p}+q}} q^{-\frac{q}{\hat{p}+q}} (\alpha\mu)^{-\frac{\hat{p}}{\hat{p}+q}}$.

For a type-I oracle, we need a slightly different derivation. Using the oracle's definition, similarly to (3.10), we get for evry $x \in \mathcal{K}$,

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{n} f_t(Y_t)\right] - \sum_{t=1}^{n} f_t(x) &\leq \mathbb{E}\left[\sum_{t=1}^{n} f_t(X_t) - \sum_{t=1}^{n} f_t(x)\right] + \frac{nL\delta^2}{2} \\
&\leq \mathbb{E}\left[\sum_{t=1}^{n} \langle \nabla f_t(X_t), X_t - x\rangle\right] + \frac{nL\delta^2}{2} \\
&= \mathbb{E}\left[\sum_{t=1}^{n} \langle G_t, X_t - x\rangle + \langle \nabla f_t(X_t) - G_t, X_t - x\rangle\right] + \frac{nL\delta^2}{2} \\
&\leq \mathbb{E}\left[\sum_{t=1}^{n} \langle G_t, X_t - x\rangle\right] + C_1\delta^p \sum_{t=1}^{n} \mathbb{E}\left[\|X_1 - x\|\right] + \frac{nL\delta^2}{2} \\
&\leq \mathbb{E}\left[\sum_{t=1}^{n} \langle G_t, X_t - x\rangle\right] + 2nRC_1\delta^p + \frac{nL\delta^2}{2}, \quad (3.17)
\end{aligned}$$

where the second to last inequality holds by the Cauchy-Schwarz inequality, and in the last step we used our assumption that $\sup_{x\in\mathcal{K}} \|x\| \leq R$. We now proceed similarly to the type-II case, applying the prox-lemma (3.11), but bound the second moment of G_t differently:

$$\begin{aligned}
\mathbb{E}\left[\|G_t\|_*^2 |\mathfrak{F}_t\right] &\leq 2\mathbb{E}\left[\|G_t - \mathbb{E}\left[G_t|\mathfrak{F}_t\right]\|_*^2\right] + 2\left\|\mathbb{E}\left[G_t|\mathfrak{F}_t\right] - \nabla f_t(X_t)\right\|_*^2 \\
&\leq 2(C_1^2\delta^{2p} + C_2\delta^{-q}), \quad (3.18)
\end{aligned}$$

Combining this with (3.11), (3.12), and (3.17) yields

$$\mathbb{E}\left[\sum_{t=1}^{n} f_t(Y_t)\right] - \sum_{t=1}^{n} f_t(x) \leq \frac{D}{\eta_{n-1}} + \frac{C_1^2\delta^{2p} + C_2\delta^{-q}}{\alpha} \sum_{t=1}^{n} \eta_t + 2nRC_1\delta^p + \frac{nL\delta^2}{2}.$$

Now, the main terms in the above inequality are identical to those of (3.14) except that instead of C_1 we have RC_1 here. Thus, optimizing the parameters of the algorithm for this case, (3.14) holds for non-strongly convex loss functions with $\hat{C}_1 = RC_1\mathbb{I}\{p \leq 2\} + (L/4)\mathbb{I}\{p \geq 2\}$. Similarly, (3.16) holds with the latter choice of $\hat{C}_1$ for μ-strongly convex loss functions and type-I oracles.

3.2.3 The Mirror Descent Lemma

Before the proof, we introduce a well-known bound on the instantaneous linearized "forward-peeking" regret of Mirror Descent.

Lemma 2 *For any $x \in \mathcal{K}$ and any $t \geq 1$,*

$$\langle G_t, X_{t+1} - x \rangle \leq \frac{1}{\eta_t} \left(\mathcal{D}_{\mathcal{R}}(x, X_t) - \mathcal{D}_{\mathcal{R}}(x, X_{t+1}) - \mathcal{D}_{\mathcal{R}}(X_{t+1}, X_t) \right) ,$$

where X_{t+1} is selected as in Algorithm 1. □

PROOF The point X_{t+1} is the minimizer of $\Psi_{t+1}(x) = \eta_t \langle G_t, x \rangle + \mathcal{D}_{\mathcal{R}}(x, X_t)$ over $\mathcal{K}$. Since the gradient of $\Psi_{t+1}(x)$ is

$$\nabla \Psi_{t+1}(x) = \eta_t G_t + \nabla \mathcal{R}(x) - \nabla \mathcal{R}(X_t),$$

by the optimality condition, for any $x \in \mathcal{K}$,

$$\langle \eta_t G_t + \nabla \mathcal{R}(x) - \nabla \mathcal{R}(X_t), x - X_{t+1} \rangle \geq 0 \,,$$

which is equivalent to the result by substituting the definition of the Bregman divergence $\mathcal{D}_{\mathcal{R}}$. ■

With this, we can turn to the proof of Lemma 1. From the smoothness and convexity of f, and using the strong convexity of $\mathcal{R}$, we get

$$\begin{aligned}
&f(X_{t+1}) - f(x) \\
&\leq f(X_t) + \langle \nabla f(X_t), X_{t+1} - X_t \rangle + \frac{L}{2} \|X_{t+1} - X_t\|^2 - \left\{ f(X_t) + \langle \nabla f(X_t), x - X_t \rangle \right\} \\
&= \langle \nabla f(X_t), X_{t+1} - x \rangle + \frac{L}{2} \|X_{t+1} - X_t\|^2 \\
&\leq \langle \nabla f(X_t), X_{t+1} - x \rangle + \frac{L}{\alpha} D_{\mathcal{R}}(X_{t+1}, X_t) \,. \qquad (3.19)
\end{aligned}$$

Writing $\nabla f(X_t) = (\nabla f(X_t) - \overline{G}_t) + \xi_t + G_t$ where $\xi_t = \overline{G}_t - G_t$ is the "noise", and using the Cauchy-Schwartz inequality and the strong convexity of $\mathcal{R}$, we obtain

$$\begin{aligned}
\langle \nabla f(X_t), X_{t+1} - x \rangle &= \langle (\nabla f(X_t) - \overline{G}_t) + \xi_t + G_t, X_{t+1} - x \rangle \\
&\leq \|X_t - x\| \left\| \nabla f - \overline{G}_t \right\|_* + \langle \xi_t, X_{t+1} - x \rangle + \langle G_t, X_{t+1} - x \rangle \\
&\leq \beta_t \sqrt{\frac{2D}{\alpha}} + \langle \xi_t, X_{t+1} - x \rangle + \langle G_t, X_{t+1} - x \rangle \,.
\end{aligned}$$

After plugging this into (3.19), the plan is to take the conditional expectation of both sides w.r.t. $\mathfrak{F}_t$. As X_t is $\mathfrak{F}_t$-measurable and $\mathbb{E}[\xi_t | \mathfrak{F}_t] = 0$ by the definition of ξ_t and $\overline{G}_t$, we have

$$\mathbb{E}[\langle \xi_t, X_{t+1} - x \rangle | \mathfrak{F}_t] = \underbrace{\mathbb{E}[\langle \xi_t, X_t - x \rangle | \mathfrak{F}_t]}_{=0} + \mathbb{E}[\langle \xi_t, X_{t+1} - X_t \rangle | \mathfrak{F}_t] \,.$$

The second term inside the expectation can be bounded by the Fenchel-Young inequality and the strong convexity of $\mathcal{R}$ as

$$\langle \xi_t, X_{t+1} - X_t \rangle \leq \frac{1}{2}\left(\frac{\|\xi_t\|_*^2}{a_t} + a_t \|X_{t+1} - X_t\|^2 \right) \leq \frac{1}{2}\left(\frac{\|\xi_t\|_*^2}{a_t} + \frac{2a_t}{\alpha} D_{\mathcal{R}}(X_{t+1}, X_t) \right).$$

Applying Lemma 2 to bound $\langle G_t, X_{t+1} - x \rangle$, and putting everything together gives

$$\begin{aligned} \mathbb{E}\left[f(X_{t+1}) - f(x) | \mathfrak{F}_t \right] \leq \quad & \beta_t \sqrt{\frac{2D}{\alpha}} + \frac{1}{2a_t} \mathbb{E}\left[\|\xi_t\|_*^2 | \mathfrak{F}_t \right] + \frac{1}{\eta_t} \left(D_{\mathcal{R}}(x, X_t) - D_{\mathcal{R}}(x, X_{t+1}) \right) \\ & + \underbrace{\left(\frac{a_t + L}{\alpha} - \frac{1}{\eta_t} \right) D_{\mathcal{R}}(X_{t+1}, X_t)}_{=0} . \end{aligned} \tag{3.20}$$

Finally, we sum up these inequalities for $t = 1, \ldots, n-1$. Since the divergence terms telescope, recall (3.12), by the tower rule and using $\sigma_t^2 = \mathbb{E}\left[\|\xi_t\|_*^2 \right]$, we obtain

$$\begin{aligned} \mathbb{E}\left[\sum_{t=1}^{n} f(X_t) - f(x) \right] &\leq \mathbb{E}\left[f(X_1) - f(x) \right] + \sqrt{\tfrac{2D}{\alpha}} \sum_{t=1}^{n-1} \beta_t + \frac{D}{\eta_{n-1}} + \sum_{t=1}^{n-1} \frac{\sigma_t^2}{2a_t} \\ &= \mathbb{E}\left[f(X_1) - f(x) \right] + \sqrt{\tfrac{2D}{\alpha}} \sum_{t=1}^{n-1} \beta_t + \frac{D(a_{n-1} + L)}{\alpha} + \sum_{t=1}^{n-1} \frac{\sigma_t^2}{2a_t} . \end{aligned}$$

When f is L-smooth and μ-strongly convex, we can rewrite (3.19) as

$$\begin{aligned} & f(X_{t+1}) - f(x) \\ & \quad \leq f(X_t) + \langle \hat{G}_t, X_{t+1} - X_t \rangle + \frac{L}{2} \|X_{t+1} - X_t\|^2 - \left\{ f(X_t) + \langle \hat{G}_t, x - X_t \rangle + \frac{\mu}{2} \mathcal{D}_{\mathcal{R}}(x, X_t) \right\} \\ & \quad = \langle \hat{G}_t, X_{t+1} - x \rangle + \frac{L}{2} \|X_{t+1} - X_t\|^2 - \frac{\mu}{2} \mathcal{D}_{\mathcal{R}}(x, X_t) \\ & \quad \leq \langle \hat{G}_t, X_{t+1} - x \rangle + \frac{L}{\alpha} D_{\mathcal{R}}(X_{t+1}, X_t) - \frac{\mu}{2} \mathcal{D}_{\mathcal{R}}(x, X_t) . \end{aligned}$$

Now, similarly to (3.20), we obtain

$$\begin{aligned} \mathbb{E}\left[f(X_{t+1}) - f(x) | \mathfrak{F}_t \right] \leq \beta_t \sqrt{\frac{2D}{\alpha}} + \frac{1}{2a_t} \mathbb{E}\left[\|\xi_t\|_*^2 | \mathfrak{F}_t \right] & \\ + \left(\frac{1}{\eta_t} - \frac{\mu}{2} \right) \mathcal{D}_{\mathcal{R}}(x, X_t) - \frac{1}{\eta_t} \mathcal{D}_{\mathcal{R}}(x, X_{t+1}) & \\ + \left(\frac{L + a_t}{\alpha} - \frac{1}{\eta_t} \right) \mathcal{D}_{\mathcal{R}}(X_{t+1}, X_t) . & \end{aligned}$$

Since $\frac{1}{\eta_t} = \frac{\mu t}{2} = \frac{L + a_t}{\alpha}$ by definition, summing up theses inequalities for $t = 1, 2, \ldots, n-1$, we get

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{n} f(X_t) - f(x)\right] &\leq \mathbb{E}\left[f(X_1) - f(x)\right] + \sqrt{\tfrac{2D}{\alpha}} \sum_{t=1}^{n-1} \beta_t + \sum_{t=1}^{n-1} \frac{\sigma_t^2}{2a_t} - \frac{1}{\eta_{n-1}} \mathcal{D}_{\mathcal{R}}(x, X_n) \\ &\leq \mathbb{E}\left[f(X_1) - f(x)\right] + \sqrt{\tfrac{2D}{\alpha}} \sum_{t=1}^{n-1} \beta_t + \sum_{t=1}^{n-1} \frac{\sigma_t^2}{2a_t}\,,\end{aligned}$$

which finishes the proof.

3.3 Lower Bounds of the Minimax Error

We next state lower bounds for both convex as well as strongly convex function classes. In particular, we observe that for convex and smooth functions the upper bound for the mirror descent scheme matches the lower bound, up to constants, whereas there is a gap for strongly convex+smooth functions. Filling the gap is left for future work.

Theorem 2 (*Lower bound*) *Let $n > 0$ be an integer, $p, q > 0$, $C_1, C_2 > 0$, $\mathcal{K} \subset \mathbb{R}^d$ convex, closed, with $[+1, -1]^d \subset \mathcal{K}$. Then, for any algorithm that observes n random elements from a (c_1, c_2) type-I oracle with $c_1(\delta) = C_1\delta^p$, $c_2(\delta) = C_2\delta^{-q}$, the minimax error (and hence the regret) satisfies the following bounds:*

- *$\mathcal{F}_{L,0}(\mathcal{K})$ (Convex and smooth) w.r.t. the Euclidean norm $\|\cdot\|_2$ with $L \geq \frac{1}{2}$*

$$\Delta^{*,\text{type}-\text{I}}_{\mathcal{F}_{L,0},n}(c_1, c_2) \geq K_3 \sqrt{d}\, C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}} n^{-\frac{p}{2p+q}},$$

$$\Delta^{*,\text{type}-\text{II}}_{\mathcal{F}_{L,0},n}(c_1, c_2) \geq K_3 d^{\frac{p}{2p+q}}\, C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}} n^{-\frac{p}{2p+q}},$$

- *$\mathcal{F}_{L,1}(\mathcal{K})$ (1-strongly convex and smooth) with $L \geq 1$*

$$\Delta^{*,\text{type}-\text{I}}_{\mathcal{F}_{L,1},n}(c_1, c_2) \geq K_4\, C_1^{\frac{2q}{2p+q}} C_2^{\frac{2p}{2p+q}} n^{-\frac{2p}{2p+q}}\,.$$

$$\Delta^{*,\text{type}-\text{II}}_{\mathcal{F}_{L,1},n}(c_1, c_2) \geq K_4\, D^{-\frac{q}{2p+q}} C_1^{\frac{2q}{2p+q}} C_2^{\frac{2p}{2p+q}} n^{-\frac{2p}{2p+q}}\,.$$

Above, the constants K_1 and K_2 depend on p and q only.[7] □

[7] In particular, $K_3 = \frac{(2p+q)^2}{2q^{\frac{q}{2p+q}}(4p+q)^{\frac{4p+q}{2p+q}}}$ and $K_4 = 2^{\frac{2p}{2p+q}} \frac{(2p+q)^3}{q^{\frac{2q}{2p+q}}(6p+q)^{\frac{6p+q}{2p+q}}}$.

By continuity, the above claim can be extended to cover the case of $q = 0$ (constant variance). For the special case of $p = 0$ and $C_1 > 0$, which implies a constant bias, it is possible to derive an $\Omega(1)$ lower bound by tweaking the proof. On the other hand, the case of $p = 0$ and $C_1 = 0$ (no bias) leads to an $\Omega(d/\sqrt{n})$ lower bound. The proof of the lower bound, presented in Section 3.4, is obtained in the usual way by providing a family of functions and a type-I oracle such that any algorithm suffers at least the stated error on one of the functions.

In particular, for $\mathcal{F}_{L,0}$ with $L \geq 1/2$ we use

$$f_{v,\epsilon}(x) = \epsilon\,(x - v) + 2\epsilon^2 \ln\left(1 + e^{-\frac{x-v}{\epsilon}}\right),$$

with $v = \pm 1$, $\epsilon > 0$, and $x \in \mathcal{K} \subset \mathbb{R}$ for appropriate ϵ. Note that for *any* $\epsilon > 0$, $f_{v,\epsilon} \in \mathcal{F}_{1/2,0} \setminus \cup_{0<\lambda<1/2}\mathcal{F}_{\lambda,0}$.

Remark 1 (***Scaling***) For any function class $\mathcal{F}$, by the definition of the minimax error (2.2), it is easy to see that

$$\Delta_n^*(\mu\mathcal{F}, c_1, c_2) = \mu\Delta_n^*\left(\mathcal{F}, c_1/\mu, c_2/\mu^2\right),$$

where $\mu\mathcal{F}$ denotes the function class comprised of functions in $\mathcal{F}$, each scaled by $\mu > 0$. In particular, this relation implies that the bound for μ-strongly convex function class is only a constant factor away from the bound for 1-strongly convex function class. □

3.4 Proofs of the Lower Bounds

In this section we present the proof of Theorem 2. Note that we will only prove lower bounds with the type-I oracle. According to Proposition 1, lower bounds for type-II can be directly attained by replacing C_1 of type-I with $C_1/\sqrt{d}$, given $[+1, -1]^d \subset \mathcal{K}$.

3.4.1 Smooth Convex Functions

We will use a novel technique that will allow us to reduce the d-dimensional case to the 1-dimensional case (see later). Thus, we start with the one-dimensional case.

Proof in one dimension: We first prove the theorem for $\mathcal{F} = \mathcal{F}_{L,0}(\mathcal{K}) \cap \{f : \mathbb{R} \to \mathbb{R} : \mathrm{dom}(f) = \mathcal{K}\}$, where by the assumptions of the theorem, $L \geq 1/2$, $\mathcal{K}$ is convex and $[-1,1] \subset \mathcal{K}$, thereby proving a slightly stronger result than stated. For brevity, let Δ_n^* denote the minimax error $\Delta_n^*(\mathcal{F}, c_1, c_2)$. Throughout the proof, a d-dimensional normal distribution with mean μ and covariance matrix Σ is denoted by $\mathsf{N}(\mu, \Sigma)$.

We follow the standard proof technique of lower bounds: We define two functions $f_+, f_- \in \mathcal{F}$ with associated type-I gradient oracles γ_+, γ_- such that the expected error of any deterministic algorithm can be bounded from below for the case when the environment is chosen uniformly at random from $\{(f_+, \gamma_+), (f_-, \gamma_-)\}$. By Yao's principle (Yao, 1977), the same lower bound applies to the minimax error Δ_n^* even when randomized algorithms are also allowed.

The proof uses (c_1, c_2) type-I oracles which have no memory. In particular, we restrict the class of oracles to those that on input (x, δ) return a random gradient estimate

$$G(x, \delta) = \overline{\gamma}(x, \delta) + \xi \tag{3.21}$$

with some map $\overline{\gamma} : \mathcal{K} \times [0, 1) \to \mathbb{R}$, where ξ is a zero-mean normal random variable with variance $c_2(\delta) := C_2 \delta^{-q}$, satisfying the variance requirement, and drawn independently every time the oracle is queried.[8] The map $\overline{\gamma}$, which will be chosen based on f to satisfy the requirement on the bias. The Y value returned by the oracles is made equal to x.

Next we define the two target functions and their associated oracles. With a slight abuse of notation, we will use interchangeably the subscripts $+$ ($-$) and $+1$ (-1) for any quantities corresponding to these two environments, e.g., f_+ and f_{+1} (respectively, f_- and f_{-1}). For $v \in \{\pm 1\}$, let

$$f_v(x) := \epsilon\,(x - v) + 2\epsilon^2 \ln\left(1 + e^{-\frac{x-v}{\epsilon}}\right), \quad x \in \mathcal{K}\,. \tag{3.22}$$

These functions, with the choice $\epsilon = 0.1$, are shown in Fig. 3.1a. The idea underlying these functions is that they approximate $\epsilon|x - v|$, but with a prescribed

[8] The argument presented below is not hard to extend to the case when all observations are from a bounded set, but this extension is left to the reader.

smoothness. The first and second derivatives of f_v are

$$f'_v(x) = \epsilon \frac{1 - e^{-\frac{x-v}{\epsilon}}}{1 + e^{-\frac{x-v}{\epsilon}}}, \qquad \text{and} \qquad f''_v(x) = \frac{2e^{-\frac{x-v}{\epsilon}}}{\left(1 + e^{-\frac{x-v}{\epsilon}}\right)^2}$$

(the functions were designed by choosing f'_v). From the above calculation, it is easy to see that $0 \le f''(x) \le 1/2$; thus f_v is $\frac{1}{2}$-smooth, and so $f_v \in \mathcal{F}$.

For $f_v, v \in \{-1, +1\}$, the gradient oracle we consider is defined as $\gamma_v(x, \delta) = \overline{\gamma}_v(x, \delta) + \xi_\delta$ with $\xi_\delta \sim \mathsf{N}(0, \frac{C_2}{\delta^q})$ selected independently for every query, where $\overline{\gamma}_v$ is a biased estimate of the gradient f'_v. The derivatives of f_+ and f_- are shown in Fig. 3.1a; we define the "bias" in $\overline{\gamma}_v$ to move the gradients closer to each other: The idea is to shift f'_+ and f'_- towards each other, with the shift depending on the allowed bias $c_1(\delta) = C_1\delta^p$. In particular, since $f'_+ \le f'_-$, f'_+ is shifted up, while f'_- is shifted down. However, the shifted up version of f'_+ is clipped for positive x so that it never goes above the shifted down version of f'_-, cf. Fig. 3.1b. By moving the curves towards each other, algorithms which rely on the obtained oracles will have an increasingly harder time (depending on the size of the shift) to distinguish whether the function optimized is f_+ or f_-. Since

$$0 \le f'_-(x) - f'_+(x) \le \sup_x f'_-(x) - \inf_x f'_+(x) = 2\epsilon,$$

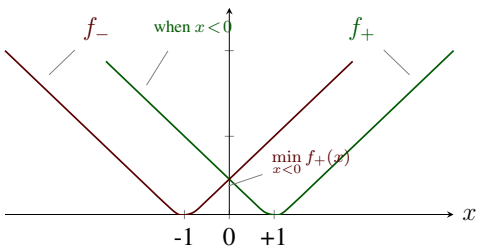
we don't allow shifts larger than ϵ (so no crossing over happens), leading to the following formal definitions:

$$\overline{\gamma}_+(x, \delta) = \begin{cases} f'_+(x) + \min(\epsilon, C_1\delta^p), & \text{if } x < 0; \\ \min\left\{f'_+(x) + \min(\epsilon, C_1\delta^p), f'_-(x) - \min(\epsilon, C_1\delta^p)\right\}, & \text{otherwise}, \end{cases} \tag{3.23}$$

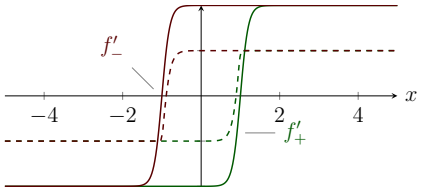
and

$$\overline{\gamma}_-(x, \delta) = \begin{cases} f'_-(x) - \min(\epsilon, C_1\delta^p), & \text{if } x > 0; \\ \max\left\{f'_-(x) - \min(\epsilon, C_1\delta^p), f'_+(x) + \min(\epsilon, C_1\delta^p)\right\}, & \text{otherwise}. \end{cases} \tag{3.24}$$

We claim that the oracle γ_v based on these functions is indeed a (c_1, c_2) type-I oracle, with $c_1(\delta) = C_1\delta^p$ and $c_2(\delta) = \frac{C_2}{\delta^q}$. The variance condition is trivial. To



(a) Plot of f_+ and f_- with $\epsilon = 0.1$



(b) Plot of f'_+ and f'_- with $\epsilon = 0.1$. The dashed lines show $\overline{\gamma}_v(\cdot, \delta)$ for $C_1\delta^p = \epsilon$, $v \in \{\pm 1\}$.

see that $c_1(\delta) = C_1\delta^p$ works, notice that $\gamma_v(x,\delta) = -\gamma_{-v}(-x,\delta)$ and $f'_v(x) = -f'_{-v}(-x)$. Thus, $|\overline{\gamma}_+(x,\delta) - f'_+(x)| = |\overline{\gamma}_-(-x,\delta) - f'_-(-x)|$, hence it suffices to consider $v = +1$. The bias condition trivially holds for $x < 0$. For $x \geq 0$, using that $f'_+(x) \leq f'_-(x)$, we get $f'_+(x) - \min(\epsilon, C_1\delta^p) \leq \overline{\gamma}_+(x,\delta) \leq f'_+(x) + \min(\epsilon, C_1\delta^p)$, showing $|\overline{\gamma}_+(x,\delta) - f'_+(x)| \leq C_1\delta^p$. Thus, γ_v is indeed an oracle with the required properties.

To bound the performance of any algorithm in minimizing $f_v, v \in \{\pm 1\}$, notice that f_v is minimized at $x^*_v = v$, with $f_v(v) = 2\epsilon^2 \ln 2$. Next we show that if x has the opposite sign of v, the difference $f_v(x) - f_v(x^*_v)$ is "large". This will mean that if the algorithm cannot distinguish between $v = +1$ and $v = -1$, it necessarily chooses a highly suboptimal point for either of these cases.

Since vf_v is decreasing on $\{x : xv \leq 0\}$, we have

$$M_v := \min_{x:xv\leq 0} f_v(x) - f_v(v) = f_v(0) - f_v(v) = \epsilon\left(-v + 2\epsilon \ln \frac{1 + e^{\frac{v}{\epsilon}}}{2}\right).$$

Let $h(v) = -v + 2\epsilon \ln \dfrac{1 + e^{\frac{v}{\epsilon}}}{2}$. Simple algebra shows that h is an even function, that is, $h(v) = h(-v)$. Indeed,

$$h(v) = -v + 2\,\epsilon\,\ln\left(e^{\frac{v}{\epsilon}}\frac{1 + e^{-\frac{v}{\epsilon}}}{2}\right) = -v + 2\,\epsilon\,\frac{v}{\epsilon} + 2\,\epsilon\,\ln\frac{1 + e^{-\frac{v}{\epsilon}}}{2} = h(-v)\,.$$

Specifically, $h(1) = h(-1)$ and thus

$$M_+ = M_- = \epsilon\left(-1 + 2\epsilon \ln \frac{1 + e^{\frac{1}{\epsilon}}}{2}\right).$$

From the foregoing, when $xv \leq 0$ and $\epsilon < \dfrac{1}{4\ln 2}$, we have

$$f_v(x) - f_v(x^*_v) \geq \epsilon\left(-1 + 2\epsilon \ln \frac{1 + e^{\frac{1}{\epsilon}}}{2}\right) > \frac{\epsilon}{2}.$$

Hence,

$$f_v(x) - f_v(x_v^*) \geq \frac{\epsilon}{2}\mathbb{I}\{xv < 0\}\,. \tag{3.25}$$

Given the above definitions and (3.25), by Yao's principle, the minimax error (2.2) is lower bounded by

$$\Delta_n^* \geq \inf_{\mathcal{A}} \mathbb{E}[f_V(\hat{X}_n) - \inf_{x\in X} f_V(x)] \geq \inf_{\mathcal{A}} \frac{\epsilon}{2}\mathbb{P}(\hat{X}_n V < 0)\,, \tag{3.26}$$

where $V \in \{\pm 1\}$ is a random variable, $\hat{X}_n$ is the estimate of the algorithm after n queries to the oracle γ_V for f_V, the infimum is taken over all deterministic algorithms, and the expectation is taken with respect to the randomness in V and the oracle. More precisely, the distribution above is defined as follows:

Consider a fixed (c_1, c_2) type-I oracle γ satisfying (3.21) and a deterministic algorithm $\mathcal{A}$. Let $x_t^{\mathcal{A}}$ (respectively, $\delta_t^{\mathcal{A}}$) denote the map from the algorithm's past observations that picks the point (respectively, accuracy parameter δ), which are sent to the oracle in round t. Define the probability space $(\Omega, \mathcal{B}, P_{\mathcal{A},\gamma})$ with $\Omega = \mathbb{R}^n \times \{-1, 1\}$, its associated Borel sigma algebra $\mathcal{B}$, where the probability measure $P_{\mathcal{A},\gamma}$ takes the form $P_{\mathcal{A},\gamma} := p_{\mathcal{A},\gamma} d(\lambda \times m)$, where λ is the Lebesgue measure on $\mathbb{R}^n$, m is the counting measure on $\{\pm 1\}$ and $p_{\mathcal{A},\gamma}$ is the density function defined by

$$\begin{aligned} p_{\mathcal{A},\gamma}(g_{1:n}, v) &= \frac{1}{2}\Big(p_{\mathcal{A},\gamma}(g_n \mid g_{1:n-1}) \cdot \ldots \cdot p_{\mathcal{A},\gamma}(g_{n-1} \mid g_{1:n-2}) \cdot \ldots \cdot p_{\mathcal{A},\gamma}(g_1)\Big) \\ &= \frac{1}{2}\Big(p_{\mathcal{N}}\big(g_n - \overline{\gamma}(x_n^{\mathcal{A}}(g_{1:n-1}), \delta_n^{\mathcal{A}}(g_{1:n-1})), c_2(\delta_n^{\mathcal{A}}(g_{1:n-1}))\big) \\ &\qquad \times \ldots\ldots \times p_{\mathcal{N}}\big(g_1 - \overline{\gamma}(x_1^{\mathcal{A}}, \delta_1^{\mathcal{A}}), c_2(\delta_1^{\mathcal{A}})\big)\Big), \end{aligned}$$

where $v \in \{-1, 1\}$ and $p_{\mathcal{N}}(\cdot, \sigma^2)$ is the density function of a $\mathsf{N}(0, \sigma^2)$ random variable. Then the expectation in (3.26) is defined w.r.t. the distribution

$$\mathbb{P} := \frac{1}{2}\left(P_{\mathcal{A},\gamma_+}\mathbb{I}\{v = +1\} + P_{\mathcal{A},\gamma_-}\mathbb{I}\{v = -1\}\right)$$

and $V : \Omega \to \{\pm 1\}$ is defined by $V(g_{1:n}, v) = v$.[9] Define $\mathbb{P}_+(\cdot) := \mathbb{P}(\cdot \mid V = 1)$,

[9]Here, we are slightly abusing the notation as $\mathbb{P}$ depends on $\mathcal{A}$, but the dependence is suppressed. In what follows, we will define several other distributions derived from $\mathbb{P}$, which will all depend on $\mathcal{A}$, but for brevity this dependence will also be suppressed. The point where the dependence on $\mathcal{A}$ is eliminated will be called to the reader's attention.

$\mathbb{P}_-(\cdot) := \mathbb{P}(\cdot \mid V = -1)$. From (3.26), we obtain

$$\Delta_n^* \geq \inf_{\mathcal{A}} \frac{\epsilon}{4} \left(\mathbb{P}_+(\hat{X}_n < 0) + \mathbb{P}_-(\hat{X}_n > 0) \right), \tag{3.27}$$

$$\geq \inf_{\mathcal{A}} \frac{\epsilon}{4} \left(1 - \|\mathbb{P}_+ - \mathbb{P}_-\|_{\mathrm{TV}} \right), \tag{3.28}$$

$$\geq \inf_{\mathcal{A}} \frac{\epsilon}{4} \left(1 - \left(\frac{1}{2} D_{\mathrm{kl}} \left(P_+ \| P_- \right) \right)^{\frac{1}{2}} \right), \tag{3.29}$$

where (3.27) uses the definitions of $\mathbb{P}_+$ and $\mathbb{P}_-$, $\|\cdot\|_{\mathrm{TV}}$ denotes the total variation distance, (3.28) follows from its definition, while (3.29) follows from Pinsker's inequality. It remains to upper bound $D_{\mathrm{kl}} \left(P_+ \| P_- \right)$.

Define G_t to be the tth observation of $\mathcal{A}$. Thus, $G_t : \Omega \to \mathbb{R}$, with $G_t(g_{1:n}, v) = g_t$. Let $P_+^t(g_1, \ldots, g_t)$ denote the joint distribution of $G_1, \ldots, G_t$ conditioned on $V = +1$. Let $P_+^t(\cdot \mid g_1, \ldots, g_{t-1})$ denote the distribution of G_t conditional on $V = +1$ and $G_1 = g_1, \ldots, G_{t-1} = g_{t-1}$. Define $P_{-j}^t(\cdot \mid g_1, \ldots, g_{t-1})$ in a similar fashion. Then, by the chain rule for KL-divergences, we have

$$D_{\mathrm{kl}} \left(P_+ \| P_- \right) = \sum_{t=1}^{n} \int_{\mathbb{R}^{t-1}} D_{\mathrm{kl}} \left(P_+^t(\cdot \mid g_{1:t-1}) \| P_-^t(\cdot \mid g_{1:t-1}) \right) dP_+^t(g_{1:t-1}). \tag{3.30}$$

By the oracle's definition on $V = +1$ we have

$$G_t \sim \mathsf{N}(\overline{\gamma}_+(x_t^{\mathcal{A}}(G_{1:t-1}), \delta_t^{\mathcal{A}}(G_{1:t-1})), c_2(\delta_t^{\mathcal{A}}(G_{1:t-1}))),$$

i.e., $P_+^t(\cdot \mid g_{1:t-1})$ is the normal distribution with mean $\overline{\gamma}_+(x_t^{\mathcal{A}}(G_{1:t-1}), \delta_t^{\mathcal{A}}(G_{1:t-1}))$ and variance $c_2(\delta_t^{\mathcal{A}}(G_{1:t-1}))$. Using the shorthands

$$x_t^{\mathcal{A}} := x_t^{\mathcal{A}}(g_{1:t-1}), \qquad \delta_t^{\mathcal{A}} := \delta_t^{\mathcal{A}}(g_{1:t-1}),$$

we have

$$D_{\mathrm{kl}} \left(P_+^t(\cdot \mid g_{1:t-1}) \| P_-^t(\cdot \mid g_{1:t-1}) \right) = \frac{(\overline{\gamma}_+(x_t^{\mathcal{A}}, \delta_t^{\mathcal{A}}) - \overline{\gamma}_-(x_t^{\mathcal{A}}, \delta_t^{\mathcal{A}}))^2}{2c_2(\delta_t^{\mathcal{A}})},$$

as the KL-divergence between normal distributions $\mathsf{N}(\mu_1, \sigma^2)$ and $\mathsf{N}(\mu_2, \sigma^2)$ is equal to $\dfrac{(\mu_1 - \mu_2)^2}{2\sigma^2}$.

It remains to upper bound the numerator. For $(x, \delta) \in \mathbb{R} \times (0, 1]$, first note that

$\gamma_+(x,\delta) \le \gamma_-(x,\delta)$. Hence,

$$\begin{aligned}
|\gamma_+(x,\delta) - \gamma_-(x,\delta)| &= \gamma_-(x,\delta) - \gamma_+(x,\delta) \\
&< \sup_x \gamma_-(x,\delta) - \inf_x \gamma_+(x,\delta) \\
&= \lim_{x\to\infty} \gamma_-(x,\delta) - \lim_{x\to-\infty} \gamma_+(x,\delta) \\
&= \epsilon - \epsilon \wedge C_1\delta^p - (-\epsilon + \epsilon \wedge C_1\delta^p) \\
&= 2\epsilon - 2\epsilon \wedge C_1\delta^p \\
&\le 2(\epsilon - C_1\delta^p)^+ , \qquad (3.31)
\end{aligned}$$

where $(u)^+ = \max(u, 0)$ is the positive part of u.

From the above, using the abbreviations $x_t^{\mathcal{A}} = x_t^{\mathcal{A}}(g_{1:t-1})$ and $\delta_t^{\mathcal{A}} = \delta_t^{\mathcal{A}}(g_{1:t-1})$ (effectively fixing $g_{1:t-1}$ for this step),

$$\begin{aligned}
D_{\mathrm{kl}}\left(P_+^t(\cdot \mid g_{1:t-1}) \| P_-^t(\cdot \mid g_{1:t-1})\right) &< \frac{2\{(\epsilon - C_1(\delta_t^{\mathcal{A}})^p)^+\}^2 \, (\delta_t^{\mathcal{A}})^q}{C_2} \qquad (3.32) \\
&\le \sup_{\delta>0} \frac{2\{(\epsilon - C_1\delta^p)^+\}^2 \, \delta^q}{C_2} , \qquad (3.33)
\end{aligned}$$

where inequality (3.32) follows from (3.31). Notice that the right-hand side of the above inequality does not depend on the algorithm anymore.

Now, observe that $\sup_{\delta>0}\{(\epsilon - C_1\delta^p)^+\}^2\delta^q = \sup_{(\epsilon/C_1)^{1/p}\ge\delta>0}(\epsilon - C_1\delta^p)^2\delta^q$. From this we obtain

$$\delta_* = \left(\frac{\epsilon q}{C_1(2p+q)}\right)^{1/p} . \qquad (3.34)$$

Note that $C_1\delta_*^p \le \epsilon$, hence $\max_{\delta>0}\{(\epsilon - C_1\delta^p)^+\}^2\delta^q = (\epsilon - C_1\delta_*^p)^2\delta_*^q$. Plugging (3.33) into (3.30) and using this last observation we obtain

$$D_{\mathrm{kl}}\left(P_+ \| P_-\right) \le \frac{2n}{C_2} \left(\epsilon - C_1\delta_*^p\right)^2 \delta_*^q . \qquad (3.35)$$

Note that the above bound holds uniformly over all algorithms $\mathcal{A}$. Substituting the above bound into (3.29), we obtain

$$\Delta_n^* \ge \frac{\epsilon}{4}\left(1 - \sqrt{n}\frac{(\epsilon - C_1\delta_*^p)\delta_*^{q/2}}{\sqrt{C_2}}\right) = \frac{\epsilon}{4}\left(1 - \sqrt{n}K_1\epsilon^{\frac{2p+q}{2p}}\right) , \qquad (3.36)$$

where $K_1 = \frac{2p}{\sqrt{C_2}(2p+q)}\left(\frac{q}{C_1(2p+q)}\right)^{\frac{q}{2p}}$.

By choosing $\epsilon = \left(\frac{2p}{\sqrt{n}K_1(4p+q)}\right)^{\frac{2p}{2p+q}}$, we see that

$$\Delta_n^* \geq \frac{2p+q}{4(4p+q)} \left(\frac{2p}{\sqrt{n}K_1(4p+q)}\right)^{\frac{2p}{2p+q}} = \frac{(2p+q)^2}{4q^{\frac{q}{2p+q}}\,(4p+q)^{\frac{4p+q}{2p+q}}} C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}} n^{-\frac{p}{2p+q}}\,. \tag{3.37}$$

Now, when $p = 1$ and $q = 2$, the lower bound in (3.37) simplifies to

$$\Delta_n^* \geq \frac{1}{3\sqrt{3}} C_1^{1/2} C_2^{1/4} n^{-1/4}\,.$$

On the other hand, for $p = q = 2$, we obtain

$$\Delta_n^* \geq \frac{9}{20} \left(\frac{1}{25}\right)^{1/3} C_1^{1/3} C_2^{1/3} n^{-1/3}\,.$$

Generalization to d dimensions: To prove the d-dimensional result, we introduce a new device which allows us to relate the minimax error of the d-dimensional problem to that of the 1-dimensional problem. The main idea is to use separable d-dimensional functions and oracles and show that if there exists an algorithm with a small loss for a rich set of separable functions and oracles, then there exists good one-dimensional algorithms for the one-dimensional components of the functions and oracles.

This device works as follows: First we define one-dimensional functions. For $1 \leq i \leq d$, let $\mathcal{K}_i \subset \mathbb{R}$ be nonempty sets, and for each $v_i \in V := \{\pm 1\}$, let $f_v^{(i)} : \mathcal{K}_i \to \mathbb{R}$. Let $\mathcal{K} = \times_{i=1}^d \mathcal{K}_i$ and for $v = (v_1, \ldots, v_d) \in V^d$, let $f_v : \mathcal{K} \to \mathbb{R}$ be defined by

$$f_v(x) = \sum_{i=1}^d f_{v_i}^{(i)}(x_i), \qquad x \in \mathcal{K}\,. \tag{3.38}$$

Without the loss of generality, we assume that $\inf_{x_i \in \mathcal{K}_i} f_{v_i}^{(i)}(x_i) = 0$, and hence $\inf_{x \in \times_{i=1}^d \mathcal{K}_i} f_v(x) = 0$, so that the optimization error of the algorithm producing $\hat{X}_n \in \mathcal{K}$ as the output is $f_v^{(i)}(\hat{X}_{n,i})$ and $f_v(\hat{X}_n)$, respectively. We also define a d-dimensional *separable* oracle γ_v as follows: The oracle is obtained from "composing" the d one-dimensional oracles, $(\gamma_{v_i}^{(i)})_i$. In particular, the ith component of the response of γ_v given the history of queries $(x_t, \delta_t, \ldots, x_1, \delta_1) \in (\mathcal{K} \times [0,1))^t$ is

defined as the response of $\gamma_{v_i}^{(i)}$ given the history of queries $(x_{t,i}, \delta_t, \ldots, x_{1,i}, \delta_1) \in (\mathcal{K}_i \times [0,1))^t$. This definition is so far unclear about the randomization of the oracles. In fact, it turns out that the one-dimensional oracles can even use the same randomization (i.e., their output can depend on the same single uniformly distributed random variable U), but they could also use separate randomization: our argument will not depend on this. Let $\Gamma^{(i)}(f_{v_i}^{(i)}, c_1, c_2)$ denote a non-empty set of (c_1, c_2) type-I oracles for objective function $f_{v_i}^{(i)} : \mathcal{K}_i \to \mathbb{R}$, and let us denote by $\Gamma_{\text{sep}}(f_v, c_1, c_2)$ the set of separable oracles for the function f_v defined above. We also define $\mathcal{F}_{\text{sep}} = \{f : f(x) = \sum_{i=1}^d f_{v_i}^{(i)}(x_i), x \in \mathcal{K}, v_i \in V_i\}$, the set of componentwise separable functions. Note that when $\|\cdot\| = \|\cdot\|_2$ is used in the definition of type-I oracles then $\Gamma_{\text{sep}}(f_v, c_1/\sqrt{d}, c_2/d) \subset \Gamma(f_v, c_1, c_2)$.

Let an algorithm $\mathcal{A}$ interact with an oracle γ. We will denote the distribution of the output $\hat{X}_n$ of $\mathcal{A}$ at the end of n rounds by $F_{\mathcal{A},\gamma}$ (we fix n, hence the dependence of F on n is omitted). Thus, the expected optimization error of $\mathcal{A}$ on a function f with zero optimal value is

$$L^{\mathcal{A}}(f, \gamma) = \int f(x) F_{\mathcal{A},\gamma}(dx) \,.$$

Note that this definition applies both in the one and the d-dimensional cases. For $v \in V^d$, we introduce the abbreviation

$$L^{\mathcal{A}}(v) = L^{\mathcal{A}}(f_v, \gamma_v) \,.$$

We also define

$$\tilde{L}_i^{\mathcal{A}}(v) = \int f_{v_i}^{(i)}(x_i) F_{\mathcal{A},\gamma_v}(dx)$$

so that

$$L^{\mathcal{A}}(v) = \sum_{i=1}^d \tilde{L}_i^{\mathcal{A}}(v) \,.$$

Also, for $v_i \in V$ and a one-dimensional algorithm $\mathcal{A}$, we let

$$L_i^{\mathcal{A}}(v_i) = L^{\mathcal{A}}(f_{v_i}^{(i)}, \gamma_{v_i}^{(i)}) \,.$$

Note that while the domain of $\tilde{L}_i^A$ is V^d, the domain of $L_i^{\mathcal{A}}$ is V, while both express an expected error measured against $f_{v_i}^{(i)}$. In fact, $\tilde{L}_i^A$ depends on v because the

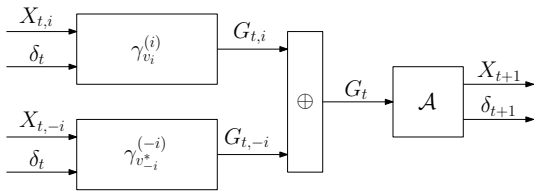


Figure 3.2: The construction of algorithm $\mathcal{A}_i^*$ used in the proof of Lemma 3.

algorithm $\mathcal{A}$ uses the d-dimensional oracle γ_v, which depends on v (and not only on v_i) and thus algorithm $\mathcal{A}$ could use information returned by $\gamma_{v_j}^{(j)}$, $j \neq i$. In a way our proof shows that using this information cannot help a d-dimensional algorithm on a separable problem, a claim that we find rather intuitive, and which we now formally state and prove.

Lemma 3 ("Cross-talk" does not help in separable problems) *Let $(f_v)_{v\in V^d}$, $f_v \in \mathcal{F}_{\text{sep}}$, $(\gamma_v)_{v\in V^d}$, $\gamma_v \in \Gamma_{\text{sep}}(f_v, c_1, c_2)$ be separable for some arbitrary functions c_1, c_2, and let $\mathcal{A}$ be any d-dimensional algorithm. Then there exist d one-dimensional algorithms, $\mathcal{A}_i^*$, $1 \le i \le d$ (using only one-dimensional oracles), such that*

$$\max_{v\in V} L^{\mathcal{A}}(v) \ge \max_{v_1\in V_1} L_1^{\mathcal{A}_1^*}(v_1) + \cdots + \max_{v_d\in V_d} L_d^{\mathcal{A}_d^*}(v_d)\,. \tag{3.39}$$

□

PROOF We will explicitly construct the one-dimensional algorithms, using $\mathcal{A}$. The difficulty is that $\mathcal{A}$ is d-dimensional, and the ith one-dimensional algorithms can only interact with the one-dimensional oracle that depends on v_i but does not depend on $v_{-i} := (v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_d)$. Hence, to use $\mathcal{A}$ we need to supply some values v_{-i}^* replacing v_{-i} so that we can use the full d-dimensional oracle, which $\mathcal{A}$ needs.

Before the construction, we need one more notational convention: Slightly abusing notation, we let $v = (v_i, v_{-i})$ and when writing (v_i, v_{-i}) as the argument of some function g, instead of $g((v_i, v_{-i}))$ we will write $g(v_i, v_{-i})$. The decomposition of a vector into one component and all the others will also be used for other d-dimensional vectors (not only for $v \in V$).

To define $\mathcal{A}_i^*$, consider the solution of the following max-min problem:

$$\max_{v_i} \min_{v_{-i}} \tilde{L}_i^{\mathcal{A}}(v_i, v_{-i}))\,.$$

Let the optimal solution of this problem be denoted by $(\hat{v}_i^*, v_{-i}^*)$; we will use v_{-i}^* replacing the missing values v_{-i} when we create a one-dimensional oracle from a d-dimensional. We also collect $(\hat{v}_i^*)_i$ into the vector $\hat{v}^* \in V^d$.

Now, algorithm $\mathcal{A}_i^*$ is constructed as illustrated on Fig. 3.2. Fix $v_i \in V_i$. Then, algorithm $\mathcal{A}_i^*$ interacts with oracle $\gamma_{v_i}^{(v_i)}$ as follows: In each round t, algorithm $\mathcal{A}_i^*$ produces a pair $(X_t, \delta_t) \in \mathcal{K} \times [0, 1)$. In particular, in the first round, X_1, δ_1 is the output of $\mathcal{A}$ in the first round. In round $t+1$, given the pair X_t, δ_t produced in the previous round, the ith component of X_t and δ_t are fed to oracle $\gamma_{v_i}^{(i)}$ (the ith component of oracle γ_v), whose output we name $G_{t,i}$. The other components of X_t, namely $X_{t,-i}$, together with δ_t are fed to oracle $\gamma_{v_{-i}^*}^{(-i)}$ which produces a $d-1$-dimensional vector of all but the ith component of $\gamma_{(v_i, v_{-i}^*)}$, which we call $G_{t,-i}$. The values $G_{t,i}$, $G_{t,-i}$ are put together to form the d-dimensional vector $G_t = (G_{t,i}, G_{t,-i})$, which is fed to algorithm $\mathcal{A}$. We then set (X_{t+1}, δ_{t+1}) to be equal to the output of $\mathcal{A}$. At the end of the n rounds, $\mathcal{A}$ is queried to produce $\hat{X}_n$, whose ith component, $\hat{X}_{n,i}$, is returned as the output of $\mathcal{A}_i^*$.

By construction, $L_i^{\mathcal{A}_i^*}(v_i) = \tilde{L}_i^{\mathcal{A}}(v_i, v_{-i}^*)$. Now, notice that

$$\max_{v_i \in V_i} \tilde{L}_i^{\mathcal{A}}(v_i, v_{-i}^*) = \tilde{L}_i^{\mathcal{A}}(\hat{v}_i^*, v_{-i}^*) \le \tilde{L}_i^{\mathcal{A}}(\hat{v}_i^*, \hat{v}_{-i}^*) = \tilde{L}_i^{A}(\hat{v}^*)\,,$$

where the equality uses the definition of $\hat{v}_i^*$, while the inequality uses the definition of v_{-i}^*. Thus,

$$\sum_{i=1}^{d} \max_{v_i \in V_i} L_i^{\mathcal{A}_i^*}(v_i) \le \sum_{i=1}^{d} \tilde{L}_i^{\mathcal{A}}(\hat{v}^*) = L^{\mathcal{A}}(\hat{v}^*) \le \max_{v \in V} L^{\mathcal{A}}(v)\,,$$

which was the claim to be proven. ∎

Now, let

$$\mathcal{F}^{(i)} = \{f_{v_i} \,:\, v_i \in V\}, \qquad i = 1, \ldots, d\,.$$

The next result follows easily from the previous lemma:

Lemma 4 *Let $\|\cdot\| = \|\cdot\|_2$ in the definition of the type-I oracles. Then, we have that*

$$\Delta^*_{\mathcal{F}_{\text{sep}},n}(c_1, c_2) \ge \sum_{i=1}^{d} \Delta^*_{\mathcal{F}^{(i)},n}(c_1/\sqrt{d}, c_2/d)\,.$$

□

PROOF By our earlier remark, $\Gamma_{\text{sep}}(f_v, c_1/\sqrt{d}, c_2/d) \subset \Gamma(f, c_1, c_2)$. Hence,

$$\Delta^*_{\mathcal{F}_{\text{sep}},n}(c_1, c_2) = \inf_{\mathcal{A}} \sup_{v\in V} \sup_{\gamma\in\Gamma(f_v,c_1,c_2)} \Delta^{\mathcal{A}}_n(f_v, \gamma) \geq \inf_{\mathcal{A}} \sup_{v\in V} \sup_{\gamma\in\Gamma_{\text{sep}}(f_v,c_1/\sqrt{d},c_2/d)} \Delta^{\mathcal{A}}_n(f_v, \gamma)\,. \tag{3.40}$$

For each $i = 1, \ldots, d$, pick $\gamma^{(i)}_{v_i} \in \Gamma(f_{v_i}, c_1/\sqrt{d}, c_2/d)$ such that $\Delta^*_n(\mathcal{F}^{(i)}, c_1/\sqrt{d}, c_2/d) = \inf_{\mathcal{A}} \sup_{v_i\in V_i} \Delta^{\mathcal{A}}_n(f_{v_i}, \gamma^{(i)}_{v_i})$. For $v \in V$, let $\gamma_v \in \Gamma_{\text{sep}}(f_v, c_1/\sqrt{d}, c_2/d)$ be the oracle whose "components" are $\gamma^{(i)}_{v_i}$, $i = 1, \ldots, d$. Now, by Lemma 3,

$$\sup_{v\in V} \Delta^{\mathcal{A}}_n(f_v, \gamma_v) \geq \sum_{i=1}^{d} \inf_{\mathcal{A}} \sup_{v_i\in V_i} \Delta^{\mathcal{A}}_n(f^{(i)}_{v_i}, \gamma^{(i)}_{v_i}) = \sum_{i=1}^{d} \Delta^*_{\mathcal{F}^{(i)},n}(c_1/\sqrt{d}, c_2/d)\,.$$

This, together with $\sup_{v\in V} \sup_{\gamma\in\Gamma_{\text{sep}}(f_v,c_1/\sqrt{d},c_2/d)} \Delta^{\mathcal{A}}_n(f_v, \gamma) \geq \sup_{v\in V} \Delta^{\mathcal{A}}_n(f_v, \gamma_v)$ and (3.40) gives the desired result. ∎

Main proof: Let $\mathcal{K} \subset \mathbb{R}^d$, such that $\times_i \mathcal{K}_i \subset \mathcal{K}$, $\{\pm 1\} \subset \mathcal{K}_i \subset \mathbb{R}$, $\mathcal{F}_d = \mathcal{F}_{L,0}(\mathcal{K})$, where recall that $L \geq 1/2$. For any $1 \leq i \leq d$, $x_i \in \mathcal{K}_i$,

$$f^{(i)}_{v_i}(x_i) := \epsilon\,(x_i - v_i) + 2\epsilon^2 \ln\left(1 + e^{-\frac{x_i - v_i}{\epsilon}}\right)\,. \tag{3.41}$$

i.e., $f^{(i)}_{v_i}$ is like in the one-dimensional lower bound proof (cf. equation 3.22). Note that $f_v \in \mathcal{F}_d$ since f_v is separable, so its Hessian is diagonal and from our earlier calculation we know that $0 \leq \frac{\partial^2}{\partial x_i^2} f^{(i)}_{v_i}(x_i) \leq 1/2$. Let $\Delta^{(d)*}_n$ denote the minimax error $\Delta^*_{\mathcal{F}_d,n}\left(C_1\delta^p, \frac{C_2}{\delta^q}\right)$ for the d-dimensional family of functions $\mathcal{F}_d$. Let $\mathcal{F}^{(i)} = \{f^{(i)}_{-1}, f^{(i)}_{+1}\}$. As it was noted above, $f_v \in \mathcal{F}_d$ for any $v \in \{\pm 1\}^d$. Hence, by Lemma 4,

$$\Delta^{(d)*}_n \geq \sum_{i=1}^{d} \Delta^*_{\mathcal{F}^{(i)},n}\left(\frac{C_1}{\sqrt{d}}\,\delta^p, \frac{C_2}{d}\delta^{-q}\right)\,. \tag{3.42}$$

Derivation of rates:

Plugging the lower bound derived in (3.37) for the one-dimensional setting into the bound in (3.42), we obtain a $\sqrt{d}$-times bigger lower bound for the d-dimensional case for any $p, q > 0$:

$$\Delta^{(d)*}_n \geq \sqrt{d}\frac{(2p+q)^2}{2q^{\frac{q}{2p+q}}\,(4p+q)^{\frac{4p+q}{2p+q}}} C_1^{\frac{q}{2p+q}} C_2^{\frac{p}{2p+q}} n^{-\frac{p}{2p+q}}\,. \tag{3.43}$$

The above bound simplifies to the following for the case where $p = 1$ and $q = 2$:

$$\Delta_n^{(d)*} \geq \frac{2(C_1^2 C_2)^{1/4}}{3\sqrt{3}} \sqrt{d} n^{-1/4}.$$

On the other hand, for the case $p = q = 2$, we obtain

$$\Delta_n^{(d)*} \geq \frac{9}{10} \left(\frac{C_1 C_2}{25} \right)^{1/3} \sqrt{d} n^{-1/3}.$$

3.4.2 Strongly Convex + Smooth Functions

We follow the notational convention used earlier for convex functions in one dimension. Let $\mathcal{F} = \mathcal{F}_{L,1}(\mathcal{K})$, where $L \geq 1$ and $\mathcal{K}$ contains ± 1. We consider functions f_v, for $v \in \{-1, +1\}$, defined as

$$f_v(x) := \frac{1}{2}x^2 - v\epsilon x\,, \quad x \in \mathcal{K}\,. \tag{3.44}$$

It is easy to see that $\{f_+, f_-\} \subset \mathcal{F}$.

Clearly, f_v is minimized at $x_v^* = v\epsilon$. By the definition of f_v, we have

$$f_v(x) - f_v(x_v^*) \geq \frac{\epsilon^2}{2} \mathbb{I}\left\{xv < 0\right\}. \tag{3.45}$$

We will consider the oracles γ_v defined as

$$\gamma_v(x) = x - v\epsilon + v \min(\epsilon, C_1 \delta^p) + \xi, \tag{3.46}$$

where $\xi \sim \mathsf{N}(0, \frac{C_2}{\delta^q})$; as with f_v, we will also use γ_+ (γ_-) to denote γ_{+1} (resp., γ_{-1}). The oracle is indeed a (c_1, c_2) type-I oracle, with $c_1(\delta) = C_1\delta^p$ and $c_2(\delta) = \frac{C_2}{\delta^q}$.

Using arguments similar to those in the proof of lower bound for convex functions, we obtain

$$\Delta_n^{(1)*} := \Delta_n^* \geq \inf_{\mathcal{A}} \frac{\epsilon^2}{2} \left(1 - \left(\frac{1}{2} D_{\mathrm{kl}}\left(P_+ \| P_-\right) \right)^{\frac{1}{2}} \right), \tag{3.47}$$

Note that P_+ (resp. P_-) is $\mathbb{P}$ conditioned on the event $V = +1$ (resp. $V = -1$).

Observe that, for any $x \in \mathbb{R}$, $f_-'(x) - f_+'(x) = 2\epsilon$ and hence

$$|\gamma_+(x) - \gamma_-(x)| = |f_+'(x) - \min(\epsilon, C_1\delta^p) - (f_-'(x) + \min(\epsilon, C_1\delta^p))| = 2(\epsilon - C_1\delta^p)^+. \tag{3.48}$$

From the foregoing,

$$D_{\mathrm{kl}}\left(P_+^t(\cdot \mid g_{1:t-1}) \| P_-^t(\cdot \mid g_{1:t-1})\right) \leq \frac{2\{(\epsilon - C_1\delta_t^p)^+\}^2\delta_t^q}{C_2}, \tag{3.49}$$

where the inequality (3.49) follows from (3.48). Thus, we obtain

$$D_{\mathrm{kl}}\left(P_+ \| P_-\right) \leq 2n \sup_{\delta>0} \frac{\{(\epsilon - C_1\delta^p)^+\}^2\delta^q}{C_2}. \tag{3.50}$$

Substituting the above bound into (3.47), we obtain

$$\Delta_n^{(1)*} \geq \frac{\epsilon^2}{2}\left(1 - \sqrt{n}\sup_{\delta>0}\frac{(\epsilon - C_1\delta^p)^+\delta^{q/2}}{\sqrt{C_2}}\right). \tag{3.51}$$

Derivation of the rates uniformly for all δ: As in the proof of the lower bound for $\mathcal{F}_{L,0}(\mathcal{K})$, we replace the positive part function in (3.51) and optimize over δ to obtain that the right-hand side of (3.50) is optimized by

$$\delta_* = \left(\frac{\epsilon q}{C_1(2p+q)}\right)^{1/p}. \tag{3.52}$$

From the above, we have

$$\Delta_n^{(1)*} \geq \frac{\epsilon^2}{2}\left(1 - \sqrt{n}\frac{(\epsilon - C_1\delta_*^p)\delta_*^{q/2}}{\sqrt{C_2}}\right) = \frac{\epsilon^2}{2}\left(1 - \sqrt{n}K_1\epsilon^{\frac{p+\frac{q}{2}}{p}}\right),$$

where $K_1 = \dfrac{p}{\sqrt{C_2}(p+\frac{q}{2})}\left(\dfrac{q}{2C_1(p+\frac{q}{2})}\right)^{\frac{q}{2p}}$.

Plugging in $\epsilon = \left(\dfrac{4p}{(6p+q)\sqrt{n}K_1}\right)^{\frac{2p}{2p+q}}$, we obtain

$$\Delta_n^{(1)*} \geq 2^{\frac{2p-q}{2p+q}}\frac{(2p+q)^3}{q^{\frac{2q}{2p+q}}(6p+q)^{\frac{6p+q}{2p+q}}}C_1^{\frac{2q}{2p+q}}C_2^{\frac{2p}{2p+q}}n^{-\frac{2p}{2p+q}}. \tag{3.53}$$

Now, when $q=2$ and $p=1$, the lower bound in (3.53) simplifies to

$$\Delta_n^{(1)*} \geq \frac{1}{2}C_1C_2^{1/2}n^{-1/2}.$$

On the other hand, for $p=q=2$, we obtain

$$\Delta_n^{(1)*} \geq 27\left(\frac{2}{77}\right)^{\frac{1}{3}}C_1^{2/3}C_2^{2/3}n^{-2/3}.$$

Generalization to d dimensions: Recall that in this result, $\|\cdot\| = \|\cdot\|_2$. The proof in d dimensions for strongly convex functions is the same as that for the case of smooth convex functions with the difference that we use (3.44) in defining the functions $f_{v_i}^{(i)}$. Then, for any $v \in \{\pm 1\}^d$, $f_v \in \mathcal{F}_{L,1}(\mathcal{K})$. Indeed, $f_v(x) = \sum_{i=1}^d f^{(i)}(x_i)$, hence $\nabla^2 f_v(x) = I_{d\times d}$, where $I_{d\times d}$ is the $d \times d$ identity matrix. Thus, $\lambda_{\min}(\nabla^2 f_v(x)) = \lambda_{\max}(\nabla^2 f_v(x)) = 1$. From (3.42) and (3.53) we get

$$\Delta_n^{(d)*} \geq \Delta_n^{(1)*}. \tag{3.54}$$

3.5 Application to the Averaging Algorithm

As mentioned in Chapter 1, our gradient oracle model can be applied to invalidate the claim that the averaging gradient estimates can be used to improve the bias-variance tradeoff. It is emphasized again that to achieve the optimal rate, algorithms or proofs have to go beyond the current scope.

We consider the problem of iterative optimization of a convex function $f : \mathbb{R} \to [0, \infty)$ using a gradient oracle. In every round, the optimizer can query the gradient oracle g_t at some point x_t, and the goal of the algorithm is to find a point x_T^* after T steps such that x_T^* is a function of $x_1, g_t(x_1), \ldots, x_T, g_T$ and $\mathbb{E}\left[f(x_T^*) - f(x^*)\right]$ is small where x^* is the minimizer of f, that is, $f(x^*) = \min_x f(x)$. Assume $f(x) = \frac{\epsilon}{2}(x-1)^2$ and $g_t(x) = \epsilon(x-1) + C_1\delta^2 + \xi_t$ where ξ_t are zero-mean iid random variables with variance C_2/δ^2. Note that $g_t(x) - f'(x) = C_1\delta^2 + \xi_t$, therefore, the bias of the gradient oracle is

$$\mathbb{E}\left[g_t(x) - f'(x)\right] = C_1\delta^2,$$

while the variance of the oracle is

$$\mathbb{E}\left[g_t(x) - f'(x) - C_1\delta^2\right] = \mathbb{E}\xi^2 = C_2/\delta^2.$$

Note that the above bias-variance bounds (with inequalities instead of equalities) are used in bandit convex optimization in papers that consider gradient methods using estimated gradients, including (Dekel et al., 2015), and they do not use anything else about the gradient estimates.

Next we consider two algorithms, SGD with the gradient estimate g_t and the method using the average gradient estimate $\bar{g}_t$. With a slight abuse of notation, we will write $g_t = g_t(x_t)$, and we define $k^+ = \max\{k, 1\}$.

Algorithm 1: $x_{t+1} = x_t - \eta g_t$;

Algorithm 2: $\bar{g}_t = \dfrac{1}{K+1} \sum_{s=(t-K)^+}^{t} g_s$ and $x_{t+1} = x_t - \eta \bar{g}_t$.

Proposition 2 *Assume Algorithm 1 or 2 is run to produce x_t for $t = 1, 2, \ldots$. Then*

$$x_t = w_0^{(t)} x_1 + 1 - w_0^{(t)} - \frac{C_1 \delta^2}{\epsilon}(1 - w_0^{(t)}) - w_1^{(t)} \xi_1 - w_2^{(t)} \xi_2 - \cdots - w_{t-1}^{(t)} \xi_{t-1}\,, \tag{3.55}$$

for some weights $w_0^{(t)}, \ldots, w_{t-1}^{(t)}$ satisfying $w_0^{(t)} = 1 - \epsilon(w_1^{(t)} + \cdots + w_{t-1}^{(t)})$. □

PROOF Assume Algorithm 1 is used. Then

$$\begin{aligned}
x_{t+1} =& (1 - \eta\epsilon) x_t + \eta\epsilon - \eta C_1 \delta^2 - \eta \xi_t \\
=& \left(1 - \eta\epsilon\right)^t x_1 + \eta\epsilon + \eta\epsilon \left(1 - \eta\epsilon\right) + \cdots + \eta\epsilon \left(1 - \eta\epsilon\right)^{t-1} \\
& - C_1 \delta^2 \left(\eta + \eta \left(1 - \eta\epsilon\right) + \cdots + \eta \left(1 - \eta\epsilon\right)^{t-1}\right) \\
& - \eta\xi_t - \eta \left(1 - \eta\epsilon\right) \xi_{t-1} - \cdots - \eta \left(1 - \eta\epsilon\right)^{t-1} \xi_1 \\
=& \left(1 - \eta\epsilon\right)^t x_1 + 1 - \left(1 - \eta\epsilon\right)^t - C_1 \delta^2 \frac{1}{\epsilon} \left(1 - \left(1 - \eta\epsilon\right)^t\right) \\
& - \eta\xi_t - \eta \left(1 - \eta\epsilon\right) \xi_{t-1} - \cdots - \eta \left(1 - \eta\epsilon\right)^{t-1} \xi_1,
\end{aligned}$$

showing that the proposition holds with $w_0^{(t+1)} = (1-\eta\epsilon)^t$ and $w_s^{(t+1)} = \eta(1-\eta\epsilon)^{t-s}$ for $s = 1, \ldots, t$.

To prove the proposition for Algorithm 2, we use induction. It is obvious that (3.55) holds for $t = 1$ with $w_0^{(1)} = 1$. Assume that (3.55) holds for $t = 1, 2, \ldots, n$, we will prove that it is also true for $t = n + 1$. Given the expression of x_t, we have

$$\begin{aligned}
g_t &= \epsilon(x_t - 1) + C_1 \delta^2 + \xi_t \\
&= \epsilon w_0^{(t)} x_1 - \epsilon w_0^{(t)} + C_1 \delta^2 w_0^{(t)} - \epsilon w_1^{(t)} \xi_1 - \cdots - \epsilon w_{t-1}^{(t)} \xi_{t-1} + \xi_t\,.
\end{aligned}$$

Therefore,

$$\begin{aligned}\bar{g}_n &= \frac{1}{K+1}\sum_{i=(n-K)^+}^{n} g_i \\ &= \frac{\epsilon x_1 - \epsilon + C_1\delta^2}{K+1}\sum_{j=(n-K)^+}^{n} w_0^{(j)} - \frac{\epsilon}{K+1}\sum_{i=1}^{n-1}\sum_{j=\max\{i+1,(n-K)^+\}}^{n} w_i^{(j)}\xi_i \\ &\quad + \frac{1}{K+1}\sum_{i=(n-K)^+}^{n}\xi_i\,.\end{aligned}$$

Then, following the algorithm, by the induction hypothesis we have

$$\begin{aligned}x_{n+1} &= x_n - \eta\bar{g}_n \\ &= \left(w_0^{(n)} - \frac{\eta\epsilon}{K+1}\sum_{j=(n-K)^+}^{n} w_0^{(j)}\right)x_1 + 1 - w_0^{(n)} + \frac{\eta\epsilon}{K+1}\sum_{j=(n-K)^+}^{n} w_0^{(j)} \\ &\quad - \frac{C_1\delta^2}{\epsilon}\left(1 - w_0^{(n)} + \frac{\eta\epsilon}{K+1}\sum_{j=(n-K)^+}^{n} w_0^{(j)}\right) \\ &\quad - \sum_{i=1}^{n-1}\left(w_i^{(n)} - \frac{\eta\epsilon}{K+1}\sum_{j=\max\{i+1,(n-K)^+\}}^{n} w_i^{(j)} + \frac{\eta}{K+1}\mathbb{I}\{i \geq n-K\}\right)\xi_i - \frac{\eta}{K+1}\xi_n.\end{aligned}$$

Letting

$$\begin{aligned}w_0^{(n+1)} &= w_0^{(n)} - \frac{\eta\epsilon}{K+1}\sum_{j=(n-K)^+}^{n} w_0^{(j)}\,, \\ w_i^{(n+1)} &= w_i^{(n)} - \frac{\eta\epsilon}{K+1}\sum_{j=\max\{i+1,(n-K)^+\}}^{n} w_i^{(j)} + \frac{\eta}{K+1}\mathbb{I}\{i \geq n-K\}\,, \quad i = 1, 2, \cdots, n-1\,, \\ w_n^{(n+1)} &= \frac{\eta}{K+1}\,,\end{aligned}$$

we get

$$x_{n+1} = w_0^{(n+1)}x_1 + 1 - w_0^{(n+1)} - C_1\delta^2\frac{1}{\epsilon}(1 - w_0^{(n+1)}) - w_1^{(n+1)}\xi_1 - w_2^{(n+1)}\xi_2 - \cdots - w_n^{(n+1)}\xi_n\,.$$

Now we only need to prove that $w_0^{(n+1)} = 1 - \epsilon\sum_{i=1}^{n} w_i^{(n+1)}$. Given that $w_0^{(j)} =$

$1-\epsilon\sum_{i=1}^{j-1} w_i^{(j)}$ for $j=1,2,\cdots,n$, we have

$$\begin{aligned}\sum_{i=1}^{n} w_i^{(n+1)} &= \sum_{i=1}^{n-1} w_i^{(n)} + \frac{\eta}{K+1}\left(n+1-(n-K)^+ - \epsilon\sum_{i=1}^{n-1}\sum_{j=\max\{i+1,(n-K)^+\}}^{n} w_i^{(j)}\right)\\ &= \frac{1}{\epsilon}\left(1-w_0^{(n)}\right) + \frac{\eta}{K+1}\sum_{j=(n-K)^+}^{n} w_0^{(j)}\\ &= \frac{1}{\epsilon}\left(1-w_0^{(n+1)}\right).\end{aligned}$$

Thereby, (3.55) also holds for $t=n+1$, finishing the proof. ∎

Now we assume that the sequence of estimates x_t satisfies Proposition 2. Then, letting $w_i = w_i^{(T+1)}$, the final estimate x_{T+1} has the form

$$x_{T+1} = w_0 x_1 + 1 - w_0 - \frac{C_1\delta^2}{\epsilon}(1-w_0) - w_1\xi_1 - w_2\xi_2 - \cdots - w_T\xi_T$$

where

$$w_0 = 1-\epsilon(w_1+\cdots+w_T).$$

Since $\{\xi_t\}$ is independent, $\mathbb{E}[\xi_t]=0$, $\mathbb{E}[\xi_t^2]=\frac{C_2}{\delta^2}$, the regret is

$$\begin{aligned}\mathbb{E}[R] &= \mathbb{E}\left[\frac{\epsilon}{2}(x_{T+1}-1)^2\right]\\ &= \mathbb{E}\left[\frac{\epsilon}{2}\left(w_0x_1 - w_0 - \frac{C_1\delta^2}{\epsilon}(1-w_0) - w_1\xi_1 - w_2\xi_2 - \cdots - w_T\xi_T\right)^2\right]\\ &= \frac{\epsilon}{2}\left((x_1-1)-(\epsilon x_1-\epsilon+C_1\delta^2)(w_1+\cdots+w_T)\right)^2 + \frac{\epsilon}{2}\frac{C_2}{\delta^2}\left(w_1^2+\cdots+w_T^2\right) \qquad (3.56)\\ &\geq \frac{\epsilon}{2}\left((x_1-1)-(\epsilon x_1-\epsilon+C_1\delta^2)(w_1+\cdots+w_T)\right)^2 + \frac{\epsilon}{2}\frac{C_2}{\delta^2}\frac{1}{T}\left(w_1+\cdots+w_T\right)^2\\ &= \frac{\epsilon}{2}\Big((x_1-1)^2 - 2(x_1-1)(\epsilon x_1-\epsilon+C_1\delta^2)W + [(\epsilon x_1-\epsilon+C_1\delta^2)^2 + C_2\delta^{-2}T^{-1}]W^2\Big)\end{aligned}$$

where we introduced the shorthand notation $W = w_1+\cdots+w_T$. Using that $aW^2+bW+c \geq c-b^2/(4a)$, we get

$$\begin{aligned}\mathbb{E}[R] &\geq \frac{\epsilon(x_1-1)^2}{2}\frac{C_2}{C_2+\delta^2T(\epsilon x_1-\epsilon+C_1\delta^2)^2}\\ &\geq \frac{\epsilon(x_1-1)^2}{2}\frac{C_2}{C_2+2(x_1-1)^2T\delta^2\epsilon^2+2C_1^2T\delta^6}.\end{aligned}$$

Now, for Algorithm 2, using the parameter choice $\delta = T^{-\frac{3}{16}}$ of Dekel et al. (2015), we can choose $\epsilon = T^{-\frac{5}{16}}$, leading to the lower bound $\mathbb{E}[R] = \Omega(T^{-5/16})$, contradicting the upper bound $O(T^{-3/8})$ of Dekel et al. (2015).

Chapter 4

Gradient Estimation Methods

A common popular idea in bandit convex optimization is to use the bandit feedback to construct noisy (and biased) estimates of the gradient. In this chapter, we provide a few examples for oracles that construct gradient estimates for function classes that are increasingly general: from smooth, convex to non-differentiable functions.

Firstly, we will formally define the noise in the feedback. In the bandit setting, the algorithm sequentially chooses the points $X_1, \dots, X_n \in \mathcal{K}$ while observing the loss function at these points in noise. In particular, in round t, the algorithm chooses X_t based on the earlier observations $Z_1, \dots, Z_{t-1} \in \mathbb{R}$ and $X_1, \dots, X_{t-1}$, after which it observes Z_t, where Z_t is the value of $f(X_t)$ (or more generally $f_t(X_t)$) corrupted by "noise".

Previous research considered several possible constraints connecting Z_t and $f(X_t)$. One simple assumption is that $\{Z_t - f(X_t)\}_t$ is an $\{\mathcal{F}_t\}_t = \{\sigma(X_{1:t}, Z_{1:t-1})\}_t$-adapted martingale difference sequence (with favorable tail properties). A specific case is when $Z_t - f(X_t) = \xi_t$, where (ξ_t) is a sequence of independent and identically distributed (i.i.d.) variables. A stronger assumption, common in stochastic programming, is that

$$Z_t = F(X_t, \Psi_t), \quad f(x) = \int F(x, \psi) P_\Psi(d\psi)\,, \tag{4.1}$$

where $\Psi_t \in \mathbb{R}$ is chosen by the algorithm and in particular the algorithm can draw Ψ_t at random from P_Ψ. As in Duchi et al. (2015), we assume that the function $F(\cdot, \psi)$ is L_ψ-smooth P_Ψ-a.s. and the quantity $\overline{L}_\Psi = \sqrt{\mathbb{E}[L_\Psi^2]}$ is finite. Note that the algorithm is aware of P_Ψ, but does not know how different values of ψ affect

the noise $\xi(x,\psi) = F(x,\psi) - f(x)$. Nevertheless, as the algorithm can control ψ and thus ξ, we refer to this as *controlled noise* setting and to the others as the case of *uncontrolled noise*. As we will see, and is well known in the simulation optimization literature (Kleinman et al., 1999; Duchi et al., 2015), this extra structure allows the algorithm to reduce the variance of the noise of its gradient estimates by reusing the same Ψ_t in consecutive measurements, while measuring the gradient at the same point, an instance of the method of the method of common random variables. As creating an estimate from K points (which is equivalent to the so-called "multi-point feedback setup" from the literature where K points are queried in each round) changes the number of rounds from n to n/K, which does not change the convergence rate as long as K is fixed.

4.1 One-point Feedback

Given $x \in \mathcal{K}$, $0 < \delta \leq 1$, common gradient estimates that are based on a single query to the function evaluation oracle (the so-called "one-point feedback") take the form

$$G = \frac{Z}{\delta} V, \text{ where } Z = f(x + \delta U) + \xi\,, \tag{4.2}$$

where $(U, V, \xi) \in \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}$ are jointly distributed random variables, ξ is the function evaluation noise whose distribution may depend on $x+\delta U$ but $\mathbb{E}[\xi|V] = 0$, and G is the estimate of $\nabla f(x)$ ($f : \mathcal{K} \to \mathbb{R}$).

In all oracle constructions we will use the following assumption:

Assumption 1 *Let $\mathcal{K} \subset \mathcal{D}^\circ \subset \mathbb{R}^d$, where $f : \mathcal{D} \to \mathbb{R}$. For any $x \in \mathcal{K}$, $x + \delta U \in \mathcal{D}$ a.s., and $\mathbb{E}\left[\|V\|_*^2\right]$, $\mathbb{E}\left[\|U\|^3\right] < +\infty$.*

Note that here the function domain $\mathcal{D}$ can be larger than or equal to the set $\mathcal{K}$, where the algorithm chooses x. This is to ensure that the oracle will not receive invalid inputs, that is, queries where f is not defined. When the functions are defined over $\mathcal{K}$ only and $\mathcal{K}$ is bounded, the above constructions only work for δ small enough. In this case, the best approach perhaps is to use Dikin ellipsoids to construct the oracles, as done by Hazan and Levy (2014).

The next proposition, whose proof is based on ideas from Spall (1992) shows that the above one-point gradient estimator leads to a type-I (and, hence, also type-II) oracle.

Proposition 3 *Let Assumption 1 hold and let γ be the one-point feedback oracle defined in (4.2). Assume further that U is symmetrically distributed, $V = h(U)$, where $h : \mathbb{R}^d \to \mathbb{R}^d$ is an odd function, $\mathbb{E}[V] = 0$, and $\mathbb{E}[VU^\top] = I$. Then, in the uncontrolled noise case, γ is a $(c_1(\delta), c_2(\delta))$ type-I oracle given in Table 4.1, where $C_2 = 4\mathbb{E}\left[\|V\|_*^2\right](\operatorname{ess\,sup}\mathbb{E}[\xi^2|V] + \sup_{x\in\mathcal{D}} f^2(x))$, and $C_1 = \frac{L}{2}\mathbb{E}[\|V\|_* \|U\|^2]$ when $f \in \mathcal{F}_{L,0}$ and $C_1 = \frac{B_3}{6}\mathbb{E}\left[\|V\|_* \|U\|^3\right]$ for $f \in \mathcal{C}^3$ where $B_3 = \sup_{x\in D} \|\nabla^3 f(x)\|_T$ where $\|\cdot\|_T$ denotes the implied norm for rank-3 tensors.*

Another possibility is to use the so-called smoothing technique (Polyak and Tsybakov, 1990; Flaxman et al., 2005; Hazan and Levy, 2014) to obtain type-II oracles. Following the analysis in Flaxman et al. (2005), one gets the following result, which improves the bias of the previous result from $O(\delta)$ to $O(\delta^2)$ in the smooth+convex case:

Proposition 4 *Let Assumption 1 hold and let γ be the one-point feedback oracle defined in (4.2). Define $V = n_W(U)\frac{|\partial W|}{|W|}$, where $W \subset \mathbb{R}^n$ is a convex body with boundary ∂W, U is uniformly distributed on ∂W, $n_W(U)$ denotes the normal vector of ∂W at U, and $|\cdot|$ denotes the appropriate volume. Let $C_2 > 0$ be defined as in Proposition 3. Then, if f is L_0-Lipschitz, γ is a memoryless, uniform type-II oracle with $c_1(\delta) = C_1\delta$, $c_2(\delta) = C_2/\delta^2$ where $C_1 = L_0 \sup_{w\in W}\|w\|$. Further, assuming W is symmetric w.r.t. the origin, if f is L-smooth, then γ is a type-I (and type-II oracle) with $c_1(\delta) = C_1\delta^2$, $c_2(\delta) = C_2/\delta^2$ where $C_1 = (L/|W|)\int_W \|w\|^2 dw$, and, if in addition f is also convex (i.e., $f \in \mathcal{F}_{L,0}$) then γ is a type-I oracle with $c_1(\delta) = C_1\delta^2/2$ and $c_2(\delta) = C_2/\delta^2$.* □

Note that the improvement did not even require convexity. Also, the bias is smaller for smoother functions, a property that will be enjoyed by all the gradient estimators.

Noise → **Function** ↓	**Controlled** (see (4.1))	**Uncontrolled** (see (4.4))
Convex + Smooth	$(C_1\delta, C_2)$	Props 3,5: $(C_1\delta, \frac{C_2}{\delta^2})$ Prop 4: $(C_1'\delta^2, \frac{C_2}{\delta^2})$
$f \in \mathcal{C}^3$	$(C_1\delta^2, \frac{C_2}{\delta^2})$	Props 3,5: $(C_1\delta^2, \frac{C_2}{\delta^2})$

Table 4.1: Gradient oracles for different function classes and noise categories.

4.2 Two-point Feedback

While the one-point estimators are intriguing, in the optimization setting one can also always group two consecutive observations and obtain similar smoothing-type estimates at the price of reducing the number of rounds by a factor of two only, which does not change the rate of convergence. Next we present an oracle that uses two function evaluations to obtain a gradient estimate. As will be discussed later, this oracle encompasses several simultaneous perturbation methods (see Bhatnagar et al., 2013): Given the inputs $x \in \mathcal{K}$, $0 < \delta \leq 1$, the gradient estimate is

$$G = \frac{Z^+ - Z^-}{2\delta} V \,, \tag{4.3}$$

where $Z^\pm = f(X^\pm) + \xi^\pm$, $X^\pm = x \pm \delta U$, $U, V \in \mathbb{R}^d$, $\xi^\pm \in \mathbb{R}$ are random, jointly distributed random variables, U, V chosen by the oracle in the uncontrolled case and chosen by the algorithm in the controlled case from some fixed distribution characterizing the oracle (depending on F), and $\xi^\pm$ being the noise of the returned feedback $Z^\pm$ at points $X^\pm$. For the following proposition we consider $4 = 2 \times 2$ cases. First, the function is either assumed to be L-smooth and convex (i.e., the derivative of f is L-Lipschitz w.r.t. $\|\cdot\|_*$), or it is assumed to be three times continuously differentiable ($f \in C^3$). The other two options are either the controlled noise setting of (4.1), or, in the uncontrolled case, we make the alternate assumptions

$$\begin{aligned} \mathbb{E}[\xi^+ - \xi^- | \, U, V] &= 0 \quad \text{and} \\ \mathbb{E}[(\xi^+ - \xi^-)^2 | \, V] &\leq \sigma_\xi^2 < \infty \,. \end{aligned} \tag{4.4}$$

The following proposition, whose proof is based on (Spall, 1992, Lemma 1) and (Duchi et al., 2015, Lemma 1), provides conditions under which the bias-variance parameters (c_1, c_2) can be bounded as shown in Table 4.1:

Proposition 5 *Let Assumption 1 hold and let γ be a two-point feedback oracle defined by (4.3). Suppose furthermore that $\mathbb{E}[VU^\intercal] = I$. Then γ is a type-I oracle with the pair $(c_1(\delta), c_2(\delta))$ given by Table 4.1. For uncontrolled noise and for controlled noise with $f \in \mathcal{C}^3$, C_1 is as in Proposition 3 and C_2 is $4C_2$ from Proposition 3. For the controlled noise case with $f \in \mathcal{F}_{L,0}$, $C_1 = \frac{\overline{L}_\Psi}{2}\mathbb{E}[\|V\|_* \|U\|^2]$ and $C_2 = 2B_1^2 + \frac{\overline{L}_\Psi^2}{2}\mathbb{E}\left[\|V\|_*^2 \|U\|^4\right]$, with $B_1 = \sup_{x\in\mathcal{K}} \|\nabla f(x)\|_*$.* □

Popular choices for U and V:

• If we set U_i to be independent, symmetric ± 1-valued random variables and $V_i = 1/U_i$, then we recover the popular SPSA scheme proposed by Spall (1992). It is easy to see that $\mathbb{E}\left[VU^\intercal\right] = I$ holds in this case. When the norm $\|\cdot\|$ is the 2-norm, $C_1 = O(d^2)$ and $C_2 = O(d)$. If we set $\|\cdot\|$ to be the max-norm, $C_1 = O(\sqrt{d})$ and $C_2 = O(d)$.

• If we set $V = U$ with U chosen uniform at random on the surface of a sphere with radius $\sqrt{d}$, then we recover the RDSA scheme proposed by Kushner and Clark (1978, pp. 58–60). In particular, the (U_i) are identically distributed with $\mathbb{E}\left[U_i U_j\right] = 0$ if $i \neq j$ and $\mathbb{E}\left[U^\intercal U\right] = d$, hence $\mathbb{E}\left[U_i^2\right] = 1$. Thus, if we choose $\|\cdot\|$ to be the 2-norm, $C_1 = O(d^2)$ and $C_2 = O(d)$.

• If we set $V = U$ with U the standard d-dimensional Gaussian with unit covariance matrix, we recover the smoothed functional (SF) scheme proposed by Katkovnik and Kulchitsky (1972). Indeed, in this case, by definition, $\mathbb{E}\left[VU^\intercal\right] = \mathbb{E}\left[UU^\intercal\right] = I$. When $\|\cdot\|$ is the 2-norm, $C_1 = O(d^2)$ and $C_2 = O(d)$. This scheme can also be interpreted as a smoothing operation that convolves the gradient of f with a Gaussian density.

4.3 Proofs for Gradient Estimates

In this section we present the proofs of the previous propositions for gradient estimates.

Proof of Proposition 3 **Case 1 ($f \in \mathcal{C}^3$):**
We use the proof technique of Spall (1997). We start by bounding the bias. Since by assumption $\mathbb{E}[\xi|V] = 0$, we have

$$\mathbb{E}\left[V\left(\frac{\xi}{\delta}\right)\right] = 0\,,$$

implying that

$$\mathbb{E}[G] = \mathbb{E}\left[V\left(\frac{f(x+\delta U)}{\delta}\right)\right].$$

By Taylor's theorem, we obtain, a.s.,

$$f(x+\delta U) = f(x) + \delta\, U^\intercal\, \nabla f(x) + \frac{\delta^2}{2}\, U^\intercal \nabla^2 f(x) U + \frac{\delta^3}{2}\, R^+(x,\delta,U)\,(U,U,U),$$

where

$$R^+(x,\delta,U) = \int_0^1 \nabla^3 f(x+s\,\delta U)(1-s)^2 ds. \tag{4.5}$$

In the above, $\nabla^3 f(\cdot)$ is considered as a rank-3 tensor. Letting $B_3 = \sup_{x\in D} \|\nabla^3 f(x)\|$,[1] we have $\|R^+(x,\delta,U)\| \le B_3/3$ a.s. Now,

$$\begin{aligned}\mathbb{E}\left[V\,\frac{f(x+\delta U)}{\delta}\right] &= \mathbb{E}\left[V\frac{f(x)}{\delta}\right] + \mathbb{E}\left[VU^\intercal\,\nabla f(x)\right] + \mathbb{E}\left[\frac{\delta}{2}\,VU^\intercal\nabla^2 f(x)U\right] \\ &\quad + \mathbb{E}\left[\frac{\delta^2}{2}V\,R^+(x,\delta,U)(U\otimes U\otimes U)\right] \\ &= \nabla f(x) + \mathbb{E}\left[\frac{\delta^2}{2}V\,R^+(x,\delta,U)(U\otimes U\otimes U)\right].\end{aligned}$$

The final equality above follows from the facts that $\mathbb{E}[V] = 0$, $\mathbb{E}[VU^\intercal] = I$ and for any $i,j = 1,\ldots,d$, $E[V_i U_j^2] = 0$ since V is a deterministic odd function of U, with U having a symmetric distribution. Using the fact that $|R^+(x,\delta,U)(U\otimes U\otimes U)| \le \|R^+(x,\delta,U)\|\,\|U\|^3$, we obtain

$$\|\mathbb{E}[G] - \nabla f(x)\|_* \le C_1\,\delta^2\,,$$

[1] Here, $\|\cdot\|$ is the implied norm: For a rank-3 tensor T, $\|T\| = \sup_{x,y,z\neq 0} \frac{|T(x,y,z)|}{\|x\|\|y\|\|z\|}$.

where $C_1 = \frac{B_3 \mathbb{E}\left[\|V\|_* \|U\|^3\right]}{6}$.

Let us now bound the variance of G: Using the identity $\mathbb{E}\|X - E[X]\|^2 \leq 4\mathbb{E}\|X\|^2$, which holds for any random variable X,[2] we bound $\mathbb{E}\|G - \mathbb{E}G\|_*^2$ as follows:

$$\begin{aligned}
\mathbb{E}\|G - \mathbb{E}G\|_*^2 &\leq 4\mathbb{E}\|G\|_*^2 \\
&= 4\mathbb{E}\left(\|V\|_*^2\left(\left(\frac{\xi}{\delta}\right)^2 + 2\left(\frac{\xi}{\delta}\right)\left(\frac{f(x+\delta U)}{\delta}\right) + \left(\frac{f(x+\delta U)}{\delta}\right)^2\right)\right) \\
&= 4\mathbb{E}\left(\|V\|_*^2\left(\frac{\xi}{\delta}\right)^2\right) + 4\mathbb{E}\left(\|V\|_*^2\right)\left(\frac{f(x+\delta U)}{\delta}\right)^2 \qquad (4.6) \\
&\leq \frac{C_2}{\delta^2},
\end{aligned}$$

where $C_2 = 4\mathbb{E}\left[\|V\|_*^2\right]\left(\sigma_\xi^2 + B_0^2\right)$, where $\sigma_\xi^2 = \operatorname{ess\,sup}\mathbb{E}\left[\xi^2|V\right]$ and $B_0 = \sup_{x\in\mathcal{D}} f(x)$. The equality in (4.6) follows from $\mathbb{E}\left[\xi \mid V\right] = 0$.

Therefore, for $f \in \mathcal{C}^3$, γ defined by (4.2) is a $(C_1\delta^2, C_2/\delta^2)$ type-I oracle.

Case 2 (f is convex and L-smooth):

Since f is convex and L-smooth, for any $0 < \delta < 1$,

$$0 \leq \frac{f(x+\delta u) - f(x)}{\delta} - \langle \nabla f(x), u\rangle \leq \frac{L\delta\|u\|^2}{2}.$$

Denoting $\phi(x,\delta,u) := \frac{f(x+\delta u)-f(x)}{\delta} - \langle\nabla f(x), u\rangle$, we have $|\phi(x,\delta,u)| \leq \frac{L\delta}{2}\|u\|^2$. Then, given $\mathbb{E}\left[VU^\top\right] = I$, $\mathbb{E}\left[V\right] = 0$, we obtain

$$\begin{aligned}
\|\mathbb{E}\left[G\right] - \nabla f(x)\|_* &= \left\|\mathbb{E}\left[\frac{f(x+\delta U)}{\delta}V\right] - \mathbb{E}\left[VU^\top\nabla f(x)\right]\right\|_* \\
&= \left\|\mathbb{E}\left[V\left(\frac{f(x+\delta U)}{\delta} - -U^\top\nabla f(x)\right)\right]\right\|_* \\
&= \left\|\mathbb{E}\left[V\left(\phi(x,\delta,U) + \frac{f(x)}{\delta}\right)\right]\right\|_* \\
&= \|\mathbb{E}\left[V\phi(x,\delta,U)\right]\|_* \\
&\leq C_1\,\delta, \qquad (4.7)
\end{aligned}$$

[2] When $\|\cdot\|$ is defined from an inner product, $\mathbb{E}\|X - E[X]\|^2 = \mathbb{E}\left[\|X\|^2\right] - \|\mathbb{E}\left[X\right]\|^2 \leq \mathbb{E}\left[\|X\|^2\right]$ also holds, shaving off a factor of four from the inequality below.

where $C_1 = \frac{L}{2}\mathbb{E}\left[\|V\|_* \|U\|^2\right]$. The claim regarding the variance of G follows in a similar manner as in Case 1, i.e., $f \in \mathcal{C}^3$.

Therefore, for f convex and L-smooth, γ defined by (4.2) is a $(C_1\delta, C_2/\delta^2)$ type-I oracle, where C_1 is given by (4.7) and C_2 as defined in Case 1.

Proof of Proposition 4 Before the proof, we introduce a fundamental theorem of vector calculus, which is commonly known as the Gauss-Ostrogradsky theorem or the divergence theorem . A special case of the theorem for real-valued functions in $\mathbb{R}^n$ can be stated as follows.

Lemma 5 *Suppose $W \subset \mathbb{R}^n$ is an open set with the boundary ∂W. At each point of ∂W there is a normal vector n_W such that n_W (i) has unit norm, (ii) is orthogonal to ∂W, (iii) points outward from W. Suppose $f : \mathbb{R}^n \to \mathbb{R}$ is a function of class C^1 defined at least on the closure of W, then we have*

$$\int_W \nabla f \, dW = \int_{\partial W} f n_W \, d\partial W \, . \qquad \square$$

PROOF Given that $\mathbb{E}\left[\|V\|_*^2\right]$ and $\mathbb{E}\left[\xi^2\right]$ are bounded, the variance of G remains the same as stated in Proposition 3.

As to the bias, let $\tilde{f}$ be a smoothed version of f, i.e., $\forall x \in \mathcal{K}$,

$$\tilde{f}(x) = \mathbb{E}\left[f(x + \delta V)\right] = \int_{v \in W} f(x + \delta v) \frac{dw}{|W|} \, ,$$

where the expectation is w.r.t. V, which is a random variable uniformly chosen from W. The second equality interprets the expectation as integral. Now we want to prove that for any given $x \in \mathcal{K}$, G is an unbiased gradient estimate of $\tilde{f}$ at x. Since U is uniformly distributed over ∂W, the expectation of G can be written as

$$\mathbb{E}\left[G\right] = \frac{|\partial W|}{|W|} \int_{\partial W} \frac{1}{\delta} f(x + \delta U) n_W(U) \frac{dU}{|\partial W|} = \int_W \nabla f(x + \delta U) \frac{dU}{|W|} \, ,$$

where the second equality follows from Lemma 5, by replacing the gradient of $\hat{f}(u) = \frac{1}{\delta} f(x + \delta u)$ with $\nabla f(x + \delta u)$. Then, the order of the gradient and the integral can be exchanged, because $\int_W f(x + \delta U) \, dU$ exists. Consequently, we obtain $\mathbb{E}\left[G\right] = \nabla \tilde{f}(x)$.

Moreover, $\tilde{f}$ and f are actually close. In particular, for any $x \in \mathcal{K}$,

$$\tilde{f}(x) - f(x) = \int_W f(x + \delta w) - f(x) \frac{dw}{|W|}\,. \tag{4.8}$$

When f is L_0-Lipschitz, $|f(x + \delta w) - f(x)| \leq L_0 \delta \|w\|$, which combined with (4.8) gives that γ is a type-II oracle with $c_1(\delta) = C_1 \delta$, where $C_1 = L_0 \sup_{w \in W} \|w\|$.

When f is convex and L-smooth, $0 \leq f(x + \delta w) - f(x) - \langle \nabla f(x), \delta w \rangle \leq \frac{L}{2} \delta^2 \|w\|^2$. Given that W is symmetric, $\int_W \langle \nabla f(x), \delta w \rangle \, dw = 0$. Hence, one can easily get that γ is a type-II oracle with $c_1(\delta) = C_1' \delta^2$, where $C_1' = \frac{L}{2|W|} \int_W \|w\|^2 \, dw$.

Finally, if f is L-smooth,

$$\begin{aligned}\left\| \nabla \tilde{f}(x) - \nabla f(x) \right\|_* &\leq \int_W \|\nabla f(x + \delta w) - \nabla f(x)\|_* \frac{dw}{|W|} \\ &\leq L \delta^2 \int_W \|w\|^2 \frac{dw}{|W|} = 2 C_1' \delta^2\end{aligned}$$

with the same value of C_1' as before. So γ is also a type-I oracle with $c_1(\delta) = 2 C_1' \delta^2$. ∎

Proof of Proposition 5 **Case 1 ($f \in \mathcal{C}^3$):**

We use the proof technique of Spall (1992) (in particular, Lemma 1 there). We start by bounding the bias. Since by assumption $\mathbb{E}\left[\xi^+ - \xi^- | V\right] = 0$, we have

$$\mathbb{E}\left[V \left(\frac{\xi_n^+ - \xi_n^-}{2\delta} \right) \right] = 0\,,$$

implying that

$$\mathbb{E}[G] = \mathbb{E}\left[V \frac{f(X^+) - f(X^-)}{2\delta} \right]\,.$$

By Taylor's theorem, using that $f \in C^3$, we obtain, a.s.,

$$f(x \pm \delta U) = f(x) \pm \delta\, U^\intercal \nabla f(x) + \frac{\delta^2}{2} U^\intercal \nabla^2 f(x) U \pm \frac{\delta^3}{2} R^\pm(x, \delta, U)\,(U, U, U),$$

where, as in the proof of Proposition 3, $R^\pm(x, \delta, U)$ is defined as follows:

$$R^\pm(x, \delta, U) = \int_0^1 \nabla^3 f(x \pm s\, \delta U)(1 - s)^2 ds. \tag{4.9}$$

Letting $B_3 = \sup_{x\in D} \|\nabla^3 f(x)\|$,we have $\|R^{\pm}(x,\delta,U)\| \le B_3/3$ a.s. Now,

$$\begin{aligned} V\,\frac{f(X^+)-f(X^-)}{2\delta} &= V\,\frac{f(x+\delta U)-f(x-\delta U)}{2\delta} \\ &= VU^{\intercal}\,\nabla f(x) + \frac{\delta^2}{4} V\left(R^+(x,\delta,U)+R^-(x,\delta,U)\right)(U\otimes U\otimes U)\,. \end{aligned} \tag{4.10}$$

and therefore, by taking expectations of both sides, using $\mathbb{E}\left[VU^{\intercal}\right] = I$ and then $|R^{\pm}(x,\delta,U)(U\otimes U\otimes U)| \le \|R^{\pm}(x,\delta,U)\|\,\|U\|^3$, we get that

$$\left\|\mathbb{E}\left[G\right]-\nabla f(x)\right\|_* \le C_1\,\delta^2\,,$$

where $C_1 = \frac{B_3\mathbb{E}\left[\|V\|_*\|U\|^3\right]}{6}$.

Using arguments similar to that in the proof of Proposition 3, the variance of G is bounded as follows:

$$\begin{aligned} \mathbb{E}\,\|G-\mathbb{E}G\|_*^2 &\le 4\mathbb{E}\,\|G\|_*^2 \\ &= 4\mathbb{E}\left(\|V\|_*^2\left(\left(\frac{\xi^+-\xi^-}{2\delta}\right)^2 + 2\left(\frac{\xi^+-\xi^-}{2\delta}\right)\left(\frac{f(X^+)-f(X^-)}{2\delta}\right) + \left(\frac{f(X^+)-f(X^-)}{2\delta}\right)^2\right)\right) \\ &= 4\mathbb{E}\left(\|V\|_*^2\left(\frac{\xi^+-\xi^-}{2\delta}\right)^2\right) + 4\mathbb{E}\left(\|V\|_*^2\right)\left(\frac{f(X^+)-f(X^-)}{2\delta}\right)^2 \qquad (4.11) \\ &\le \frac{C_2}{\delta^2}\,, \end{aligned}$$

where $C_2 = 4\mathbb{E}\left[\|V\|_*^2\right]\left(\sigma_\xi^2 + \mathrm{span}(f)\right)$ and $\mathrm{span}(f) = \sup_{x\in\mathcal{D}} f(x) - \inf_{x\in\mathcal{D}} f(x)$. The equality in (4.11) follows from $\mathbb{E}\left[\xi^+-\xi^- \,|\, U,V\right] = 0$.

Therefore, for $f\in\mathcal{C}^3$, γ defined by (4.3) is a $(C_1\delta^2, C_2/\delta^2)$ type-I oracle.

Case 2 (Controlled noise and F is convex and L_ψ-smooth):

The proof follows by parallel arguments to that used in the proof of Lemma 1 in Duchi et al. (2015) and we give it here for the sake of completeness.

For any convex function f with an L-Lipschitz gradient, for any $\delta > 0$ it holds that

$$\frac{\langle\nabla f(x),\delta u\rangle}{2\delta} \le \frac{f(x+\delta u)-f(x)}{2\delta} \le \frac{\langle\nabla f(x),\delta u\rangle + (L/2)\,\|\delta u\|^2}{2\delta}.$$

Using similar inequalities for $f(x-\delta u)$, we obtain

$$\langle\nabla f(x),u\rangle - \frac{L\delta\,\|u\|^2}{2} \le \frac{f(x+\delta u)-f(x-\delta u)}{2\delta} \le \langle\nabla f(x),u\rangle + \frac{L\delta\,\|u\|^2}{2}.$$

Letting $\phi(x,\delta,u) := \frac{1}{\delta}\left(\frac{f(x+\delta u)-f(x-\delta u)}{2\delta} - \langle \nabla f(x), u\rangle\right)$, we get

$$|\phi(x,\delta,u)| \leq \frac{L}{2}\|u\|^2 .$$

Using $\mathbb{E}\left[VU^\top\right] = I$, we obtain

$$\begin{aligned}\mathbb{E}\left[V\left(\frac{f(x+\delta U)-f(x-\delta U)}{2\delta}\right)\right] =& \mathbb{E}\left[VU^\top \nabla f(x) + \delta\phi(x,\delta,U)V\right] \\ =& \nabla f(x) + \delta\widehat{\phi}(x,\delta),\end{aligned}$$

where $\widehat{\phi}(x,\delta)$ satisfies $\left\|\widehat{\phi}(x,\delta)\right\|_* \leq \frac{L}{2}\mathbb{E}[\|V\|_*\|U\|^2]$.

Applying the above expression to $F(\cdot,\Psi)$ and recalling that $G = V\left(\frac{F(X^+,\psi)-F(X^-,\psi)}{2\delta}\right)$, we have, for P-almost every ψ,

$$\mathbb{E}[G] = \nabla F(x,\psi) + \delta\widehat{\phi}(x,\delta),$$

where, as before, $\widehat{\phi}(x,\delta)$ satisfies $\left\|\widehat{\phi}(x,\delta)\right\|_* \leq \frac{L_\psi}{2}\mathbb{E}[\|V\|_*\|U\|^2]$.

Using the fact that $E[\nabla F(x,\Psi)] = \nabla f(x)$, we obtain

$$\begin{aligned}\|\mathbb{E}[G] - \nabla f(x)\|_* &= \left\|\mathbb{E}\left[V\left(\frac{f(x+\delta U)-f(x-\delta U)}{2\delta}\right) - VU^\top\nabla f(x)\right]\right\|_* \\ &\leq \delta\|\mathbb{E}[V\phi(x,\delta,U)]\|_* \\ &\leq \frac{\delta\overline{L}_\Psi}{2}\mathbb{E}[\|V\|_*\|U\|^2],\end{aligned}$$

and the claim for the bias follows by setting $C_1 = \frac{\overline{L}_\Psi}{2}\mathbb{E}[\|V\|_*\|U\|^2]$.

We now bound $\mathbb{E}\left[\|G\|_*^2\right]$ as follows:

$$\begin{aligned}\mathbb{E}\|G\|^2 &= \mathbb{E}\left\|V\left(\delta\phi(x,\delta,U) + U^\top\nabla f(x)\right)\right\|^2 \\ &\leq \mathbb{E}\left[\left(\|VU^\intercal\nabla f(x)\|_* + \frac{\delta L}{2}\|V\|_*\|U\|^2\right)^2\right] \\ &\leq 2\mathbb{E}\left[\|VU^\intercal\nabla f(x)\|_*^2\right] + \frac{\delta^2\overline{L}_\Psi^2}{2}\mathbb{E}\left[\|V\|_*^2\|U\|^4\right],\end{aligned}$$

and the claim for the variance follows by setting $C_2 = 2B_1^2 + \frac{\overline{L}_\Psi^2}{2}\mathbb{E}\left[\|V\|_*^2\|U\|^4\right]$ with $B_1 = \sup_{x\in\mathcal{K}}\|\nabla f(x)\|_*$.

Therefore, for the case of controlled noise with a convex and L_ψ-smooth F, we have that γ defined by (4.3) is a $(C_1\delta, C_2)$ type-I oracle.

Chapter 5

Application to Stochastic Convex Optimization

The main application of the biased noisy gradient oracle based convex optimization of Chapter 3 is bandit convex optimization. We introduce here briefly the stochastic version of the problem, while online bandit convex optimization will be considered in Chapter 6.

In the *stochastic BCO* setting, there is a single objective function. We now consider stochastic BCO with a L-smooth function over a convex, closed non-empty domain $\mathcal{K}$. Let $\mathcal{F}$ denote the set of these functions. Duchi et al. (2015) proves that the minimax expected optimization error for the functions $\mathcal{F}$ with uncontrolled noise is lower bounded by $\Omega(n^{-1/2})$. They also give an algorithm which uses two-point gradient estimates which matches this lower bound for the case of *controlled noise*. For controlled noise, the constructions in the previous section give that for two-point estimators $c_1(\delta) = C_1\delta^p$ and $c_2(\delta) = C_2\delta^{-q}$ with $p = 1$ and $q = 0$. Plugging this into Theorem 1 we get the rate $O(n^{-1/2})$ (which is unsurprising given that the algorithms and the upper bound proof techniques are essentially the same as that of Duchi et al. (2015)). However, when the noise is uncontrolled, the best that we get is $p = 2$ and $q = 2$. From Theorem 2 we get that with such oracles, no algorithm can get better rate than $\Omega(n^{-1/3})$, while from Theorem 1 we get that these rates are matched by mirror descent. We can summarize these findings as follows:

Theorem 3 *Consider $\mathcal{F}_{L,0}$, the space of convex, L-smooth functions over a convex, closed non-empty domain $\mathcal{K}$. Then, we have the following:*

***Uncontrolled noise**: Take any (δ^2, δ^{-2}) type-I oracle γ. There exists an algorithm that uses γ and achieves the rate $O(n^{-1/3})$. Furthermore, no algorithm using γ can achieve better error than $\Omega(n^{-1/3})$ for every (δ^2, δ^{-2}) type-I oracle γ.*

***Controlled noise**: Take any $(\delta, 1)$ type-I oracle γ. There exists an algorithm that uses γ an achieves the rate $O(n^{-1/2})$. Furthermore, no algorithm using γ can achieve better error than $\Omega(n^{-1/2})$ for every $(\delta, 1)$ type-I oracle γ.* □

For stochastic BCO with uncontrolled noise, Agarwal et al. (2013) analyze a variant of the well-known ellipsoid method and provide regret bounds for the case of convex, 1-Lipschitz functions over the unit ball. Their regret bound implies a minimax error (2.2) bound of order $O\left(\sqrt{d^{32}/n}\right)$. Liang et al. (2014) provide an algorithm based on random walks (and not using gradient estimates) for the setting of convex, bounded functions whose domain is contained in the unit cube and their algorithm results in a bound of the order $\mathcal{O}\left((d^{14}/n)^{1/2}\right)$ for the minimax error. These bounds decrease faster in n than the bound available in Theorem 3, while showing a much worse dependence on the dimension. However, what is more interesting is that our results also shows that an $O(n^{-1/2})$ upper bound *cannot* be achieved solely based on the oracle properties of the gradient estimates considered. Since the analysis of all gradient algorithms for stochastic BCO does this, it is no wonder that the best known upper bound for convex+smooth functions is $O(n^{-1/3})$ (Saha and Tewari, 2011). (We will comment on the recent paper of Dekel et al. (2015) later.)

The above result also shows that the gradient oracle based algorithms are optimal for smooth problems, under a controlled noise setting. While Duchi et al. (2015) suggests that it is the power of two-point gradient estimators that helps to achieve this, we need to add that having controlled noise is also critical.

Finally, let us make some remarks on the early literature on this problem. A finite time lower bound for stochastic, smooth BCO is presented by Chen (1988) for convex functions on the real line. When applied to our setting in the uncontrolled noise case, his results imply that $\mathbb{E}\left[|\hat{X}_n - x^*|\right]$, that is, the distance of the estimate to the optimum, is at least $\Omega(n^{-1/3})$. Note that this is larger than the error achieved by the algorithms of Liang et al. (2014); Bubeck et al. (2015); Bubeck and Eldan

(2015), but the apparent contradiction is easily resolved by noticing the difference in their error measure: distance to the optimum vs. error in the function value (in particular, compressing the range of functions makes locating the minimizer harder). Polyak and Tsybakov (1990), who also considered distance to optimum, proved that mirror descent with gradient estimation achieves asymptotically optimal rates for functions that enjoy high order smoothness.

Chapter 6

Application to Online Convex Optimization

In the *online BCO* setting a learner sequentially chooses the points $X_1, \dots, X_n \in \mathcal{K}$ while observing the losses $f_1(X_1), \dots, f_n(X_n)$. More specifically, in round t, having observed $f_1(X_1), \dots, f_{t-1}(X_{t-1})$ of the previous rounds, the learner chooses $X_t \in \mathcal{K}$, after which it observes $f_t(X_t)$. The learner's goal is to minimize its expected regret $\mathbb{E}\left[\sum_{t=1}^n f_t(X_t) - \inf_{x\in\mathcal{K}} \sum_{t=1}^n f_t(x)\right]$. This problem is also called online convex optimization with one-point feedback. A slightly different problem is obtained if we allow the learner to choose multiple points in every round, at which points the function f_t is observed. The loss is suffered at X_t. The points where the function is observed ("observation points" for short) may or may not be tied to X_t. One possibility is that X_t is one of the observation points. Another possibility is that X_t is the average of the observation points (e.g., Agarwal et al. (2010)). Yet another possibility is that there is no relationship between them.

The oracle constructions from the previous section also apply to the online BCO setting where the algorithm is evaluated at Y_t, though in this case one cannot employ two-point feedback as the functions change between rounds. This also rules out the controlled noise case. Thus, for the online BCO setting, one should consider type-I (and II) oracles with $c_1(\delta) = C_1\delta^p$ and $c_2(\delta) = C_2\delta^{-q}$ with $p = q = 2$. For these type of oracles, the results from Theorem 2 give the following result:

Theorem 4 *Let $\mathcal{F}_{L,0}$ be the space of convex, L-smooth functions over a convex non-empty domain $\mathcal{K}$. No algorithm that relies on (δ^2, δ^{-2}) type-I oracles can achieve*

better regret than $\Omega(n^{2/3})$. □

With a noisy gradient oracle of Proposition 4, Theorem 4 implies that this regret rate is achievable, essentially recovering, and in some sense proving optimality of the result of Saha and Tewari (2011):

Theorem 5 *For zeroth order noisy optimization with smooth convex functions, the gradient estimator of Proposition 4 together with mirror descent (see Algorithm 1) achieve* $\mathcal{O}(n^{2/3})$ *regret.* □

This optimality result shows that with the usual analysis of the current gradient estimation techniques, no gradient method can achieve the optimal regret $O(n^{1/2})$ for online bandit convex optimization, established by Bubeck et al. (2015); Bubeck and Eldan (2015). Note that this shows a contradiction to the recent result of Dekel et al. (2015), who claimed to achieve $\tilde{O}(n^{5/8})$ regret with the same (δ^2, δ^{-2}) type-II gradient oracle as Saha and Tewari (2011), but their proof only used the (δ^2, δ^{-2}) tradeoff in the bias and variance properties of the oracle.

Chapter 7

Conclusions

We presented a novel noisy gradient oracle model for convex optimization. The oracle model covers several gradient estimation methods in the literature designed for algorithms that can observe only noisy function values, while allowing to handle explicitly the bias-variance tradeoff of these estimators. The framework allows to derive sharp upper and lower bounds on the minimax optimization error and the regret in the online case. It not only encompasses "gradient" methods, reproducing the state-of-the-art upper bounds in a unified and clear fashion, but also gives the lower bound, implying the best possible rate the algorithm can achieve only with access to the biased, noisy first-order information.

In particular, we obtain matching upper and lower bounds for optimizing smooth, convex functions under the framework. This result is worthwhile because it claims it impossible to design an algorithm that can make better use of current gradient estimates. Therefore, to achieve the optimal $O(\sqrt{1/n})$ rate for smooth, convex functions with uncontrolled noise, other approaches must be considered. For instance, a gradient oracle with constant second moment bound ($q = 0$) must be designed, or some extra properties of gradient estimates must be exploited beyond the bias-variance tradeoff. It is also possible to design a non-gradient method which achieve optimal rates with a reasonably low complexity.

Back to the big picture of bandit convex optimization, some cases are already well understood, including the linear case, the general convex case with controlled noise, and the strongly convex, smooth case. Nevertheless, for the general convex case with uncontrolled noise, a big gap remains. Our results make a theoretically

progressive step towards bridging this gap. It is pointed out that current gradient methods are essentially sub-optimal in terms of designing or analyzing the gradient estimation properties. To find an optimal algorithm, one must go beyond the scope of bias-variance tradeoff analysis, and find other directions.

Bibliography

Abernethy, J., Hazan, E., and Rakhlin, A. (2008). Competing in the dark: An efficient algorithm for bandit linear optimization. In *COLT*, pages 263–274.

Agarwal, A., Dekel, O., and Xiao, L. (2010). Optimal algorithms for online convex optimization with multi-point bandit feedback. In *COLT*, pages 28–40.

Agarwal, A., Foster, D. P., Hsu, D., Kakade, S. M., and Rakhlin, A. (2013). Stochastic convex optimization with bandit feedback. *SIAM Journal on Optimization*, 23(1):213–240.

Baes, M. (2009). Estimate sequence methods: Extensions and approximations. Technical report, IFOR Internal report, ETH Zurich, Switzerland.

Bartlett, P., Hazan, E., and Rakhlin, A. (2008). Adaptive online gradient descent. In Platt, J., Koller, D., Singer, Y., and Roweis, S., editors, *Advances in Neural Information Processing Systems 20*, pages 65–72. MIT Press, Cambridge, MA.

Beck, A. and Teboulle, M. (2003). Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175.

Bhatnagar, S., Prasad, H. L., and Prashanth, L. A. (2013). *Stochastic Recursive Algorithms for Optimization: Simultaneous Perturbation Methods (Lecture Notes in Control and Information Sciences)*, volume 434. Springer.

Bubeck, S. (2014). Theory of convex optimization for machine learning. Technical report, Microsoft Research.

Bubeck, S., Dekel, O., Koren, T., and Peres, Y. (2015). Bandit convex optimization: $o(\sqrt{T})$ regret in one dimension. In *COLT*, pages 266–278.

Bubeck, S. and Eldan, R. (2015). Multi-scale exploration of convex functions and bandit convex optimization. Technical report, Microsoft Research.

Chen, H. (1988). Lower rate of convergence for locating a maximum of a function. *The Annals of Statistics*, 16(3):1330–1334.

d'Aspremont, A. (2008). Smooth optimization with approximate gradient. *SIAM Journal on Optimization*, 19:1171–1183.

Dekel, O., Eldan, R., and Koren, T. (2015). Bandit smooth convex optimization: Improving the bias-variance tradeoff. In *NIPS*, pages 2926–2934.

Dekel, O., Gilad-Bachrach, R., Shamir, O., and Xiao, L. (2012). Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(1):165–202.

Devolder, O., Glineur, F., and Nesterov, Y. (2014). First-order methods of smooth convex optimization with inexact oracle. *Mathematical Programming*, 146:37–75.

Dippon, J. (2003). Accelerated randomized stochastic optimization. *The Annals of Statistics*, 31(4):1260–1281.

Duchi, J. C., Jordan, M., Wainwright, M. J., and Wibisono, A. (2015). Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806.

Dvurechensky, P. and Gasnikov, A. (2015). Stochastic intermediate gradient method: Convex and strongly convex cases. arXiv:1411.2876.

Flaxman, A. D., Kalai, A. T., and McMahan, H. B. (2005). Online convex optimization in the bandit setting: gradient descent without a gradient. In *SODA*, pages 385–394.

Hazan, E. and Levy, K. (2014). Bandit convex optimization: Towards tight bounds. In *NIPS*, pages 784–792.

Honorio, J. (2012). Convergence rates of biased stochastic optimization for learning sparse ising models. In *ICML*, pages 257–264, New York, NY, USA. Omnipress.

Juditsky, A. and Nemirovski, A. (2011). First-order methods for nonsmooth convex large-scale optimization, i: General purpose methods. In Sra, S., Nowozin, S., and Wright, S., editors, *Optimization for Machine Learning*, pages 121–147. MIT press.

Katkovnik, V. Y. and Kulchitsky, Y. (1972). Convergence of a class of random search algorithms. *Automation Remote Control*, 8:1321–1326.

Kleinman, N. L., Spall, J. C., and Naiman, D. Q. (1999). Simulation-based optimization with stochastic approximation using common random numbers. *Management Science*, 45(11):1570–1578.

Kushner, H. J. and Clark, D. S. (1978). *Stochastic Approximation Methods for Constrained and Unconstrained Systems*. Springer Verlag, New York.

Liang, T., Narayanan, H., and Rakhlin, A. (2014). On zeroth-order stochastic convex optimization via random walks. arXiv preprint1402.2667.

Mahdavi, M. (2014). *Exploiting Smoothness in Statistical Learning, Sequential Prediction, and Stochastic Optimization*. PhD thesis, Michigan State University.

Nemirovski, A., Juditsky, A., Lan, G., and Shapiro, A. (2009). Robust stochastic approximation approach to stochastic programming. *SIAM J. Optimization*, 4:1574—1609.

Nemirovskii, A. and Yudin, D. B. (1983). *Problem complexity and method efficiency in optimization*. Wiley-Interscience series in discrete mathematics. Wiley.

Nesterov, Y. (2004). *Introductory lectures on convex optimization*, volume 87. Springer Science & Business Media.

Nesterov, Y. and Spokoiny, V. (2011). Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, pages 1–40.

Polyak, B. and Tsybakov, A. (1990). Optimal orders of accuracy for search algorithms of stochastic optimization. *Problems in Information Transmission*, pages 126–133.

Saha, A. and Tewari, A. (2011). Improved regret guarantees for online smooth convex optimization with bandit feedback. In *AISTATS*, pages 636–642.

Schmidt, M. W., Roux, N. L., and Bach, F. R. (2011). Convergence rates of inexact proximal-gradient methods for convex optimization. In *NIPS*, pages 1458–1466.

Shamir, O. (2012). On the complexity of bandit and derivative-free stochastic convex optimization. In *COLT*.

Spall, J. C. (1992). Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 37(3):332–341.

Spall, J. C. (1997). A one-measurement form of simultaneous perturbation stochastic approximation. *Automatica*, 33(1):109–112.

Spall, J. C. (2005). *Introduction to stochastic search and optimization: estimation, simulation, and control*, volume 65. John Wiley & Sons.

Wainwright, M. J. and Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1-2):1–305.

Yao, A. C. C. (1977). Probabilistic computations: Toward a unified measure of complexity. In *FOCS*, pages 222–227.