# (Bandit) Convex Optimization with Biased Noisy Gradient Oracles

Xiaowei Hu[1], Prashanth L.A.[2], András György[3] and Csaba Szepesvári[1]

[1]Department of Computing Science, University of Alberta
[2]Institute for Systems Research, University of Maryland
[3]Department of Electrical and Electronic Engineering, Imperial College London

## Abstract

For bandit convex optimization we propose a model, where a gradient estimation oracle acts as an intermediary between a noisy function evaluation oracle and the algorithms. The algorithms can control the bias-variance tradeoff in the gradient estimates. We prove lower and upper bounds for the minimax error of algorithms that interact with the objective function by controlling this oracle. The upper bounds replicate many existing results (capturing the essence of existing proofs) while the lower bounds put a limit on the achievable performance in this setup. In particular, our results imply that no algorithm can achieve the optimal minimax error rate in stochastic bandit smooth convex optimization.

## 1 Introduction

In stochastic bandit convex optimization (also known as convex optimization with stochastic zeroth order oracles) an algorithm submits queries to an oracle in a sequential manner in $n$ rounds. The oracle returns noisy values of the convex objective function at the submitted points. At the end, the algorithm also produces a guess of the objective's minimizer; the algorithm's performance is measured in terms of the suboptimality of this guess, measured using the objective function. In their seminal work Nemirovskii and Yudin (1983) consider two approaches to this problem: Methods that try to construct gradient information and methods that avoid gradient estimation and rather use geometric principles (the ellipsoid method, essentially). While methods in the second class make the error decay at the $O(1/\sqrt{n})$ rate, the error scales extremely poorly with the number of optimization variables $d$. For example, Liang et al. (2014) proves a bound of the form $\sqrt{d^{14}/n}$, improving the $\sqrt{d^{33}/n}$ bound of Agarwal et al. (2013). A lower bound due to (Shamir, 2012) however scales only with $\sqrt{d^2/n}$. Methods that first construct gradient estimates, which are fed to algorithms that expect gradient-information have a long history, though in early years the focus was either asymptotic convergence, or asymptotic rates (Poljak and Tsypkin, 1973; Kushner and Clark, 1978). The story with these methods is that they get the optimal rate and dimension dependence for "nice" problems, such as when the objective function is strongly convex and smooth (Hazan and Levy, 2014), but their performance, mostly in terms of the rate degrades as one removes constraints from the objective function. For example, for smooth convex problems, the best published results with this technique is $O(n^{1/3})$ (Saha and Tewari, 2011). Our motivation in this paper is to formally study the possible limitations of the gradient-based approach. We do this by precisely defining a new oracle model. The new oracles can be consulted to obtain gradient estimates. They have a tuneable parameter, which controls the bias-variance tradeoff that exists for all known gradient construction methods. We then prove lower bounds and (sometimes) matching upper bounds for algorithms that use these oracles. Our lower bound for stochastic smooth bandit convex optimization gives the rate $\Omega(n^{1/3})$, assuming that the currently available gradient-estimation methods are unimprovable.

## 2 Problem Setup

*Preliminaries:* Capital letters will denote random variables. We let $\|\cdot\|$ denote some norm on $\mathbb{R}^d$ and we denote its dual norm by $\|\cdot\|_*$. A function $f$ is $\mu$-strongly convex w.r.t. $\|\cdot\|$ if $\frac{\mu}{2}\|x-y\|^2 \le \mathcal{D}_f(x,y) \doteq f(x)-f(y)-\langle \nabla f(y), x-y\rangle$, for all $x, y$ in a non-empty convex closed set $\mathcal{K} \in \mathbb{R}^d$. A function $f$ is $\mu$-strongly convex w.r.t. a *function* $\mathcal{R}$ if $\frac{\mu}{2} D_{\mathcal{R}}(x, y) \leq \mathcal{D}_f(x, y)$ for all $x, y \in \mathcal{K}$. A function $f$ is $L$-smooth w.r.t. $\|\cdot\|$ for some $L > 0$ if $D_f(x, y) \leq \frac{L}{2}\|x-y\|^2$, for all $x, y \in \mathcal{K}$. This latter condition is equivalent to that $\nabla f$ is $L$-Lipschitz. We let $\mathcal{F}_{L,\mu}(\mathcal{K})$ denote the class of functions that are $\mu$-strongly convex and $L$-smooth on the set $\mathcal{K}$. The set of $L$-smooth functions is thus $\mathcal{F}_{L,0}(\mathcal{K})$. Furthermore, let $\mathcal{F}_{L,\mu,\mathcal{R}}(\mathcal{K})$ denote the set of functions that are $L$-smooth and $\mu$-strongly convex w.r.t. $\mathcal{R}$.

In this paper we consider convex optimization in a novel setting, both for stochastic as well as online BCO. In the online BCO setting, the environment chooses a sequence of loss functions $f_1, \ldots, f_n$ belonging to a set $\mathcal{F}$ of convex functions over a common, non-empty convex closed domain $\mathcal{K} \subset \mathbb{R}^d$. In the stochastic BCO setting, a single fixed loss function $f \in \mathcal{F}$ is chosen. As usual, the algorithms choose a sequence of points $X_1, \ldots, X_n \in \mathcal{K}$ in a sequential way. The novelty of our setting is that the algorithm, upon selecting point $X_t$, receives a noisy and potentially biased estimate $G_t \in \mathbb{R}^d$ of the gradient of the loss function $f$ (more generally, an estimate of a subgradient of $f$ in case $f$ is non-differentiable at $X_t$). 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 $\delta_t$ results in a smaller "bias" (for the precise meaning of bias, we will consider two definitions below), while typically with a smaller $\delta_t$ the "variance" of the gradient estimate increases. The algorithm also receives a point $Y_t \in \mathcal{K}$, which is guaranteed to be in the $\delta_t$-vicinity of $X_t$. While this point is not relevant in the optimization setting, in the online learning setting, the algorithm suffers a loss at $Y_t$. As usual, the goal in the online BCO setting is to keep the expected *regret*, $R_n = \mathbb{E}\left[\sum_{t=1}^n f_t(Y_t)\right] - \inf_{x \in \mathcal{K}} \sum_{t=1}^n f_t(x)$, small. In the stochastic BCO setting, the algorithm is also required to select a point $\hat{X}_n \in \mathcal{K}$ once the $n$th round is over (in both settings, $n$ is given to the algorithms) and the algorithm's performance is quantified using the *optimization error*, $\Delta_n = \mathbb{E}\left[f(\hat{X}_n)\right] - \inf_{x \in \mathcal{K}} f(x)$.

The main novelty of the model is that the information flow between the algorithm and the environment is controlled by the gradient estimation oracle. As we shall see, numerous existing approaches to online learning and optimization based on noisy pointwise information fit this framework.

In what follows, the functions $c_1, c_2 : [0, \infty) \to [0, \infty)$ will be assumed to be continuous, monotonously increasing (resp., decreasing) with $\lim_{\delta \to 0} c_1(\delta) = 0$ 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 $p, q > 0$. We define two classes of oracles. For any function $f \in \mathcal{F}$, $x \in \mathcal{K}$, $0 < \delta \leq 1$, we say that $\gamma$ is a $(c_1, c_2)$ Type-I or Type-II oracle if it returns $G \in \mathbb{R}^d$ and $Y \in \mathcal{K}$ random elements such that:

| **Type-I Oracle** | **Type-II Oracle** |
|---|---|
| 1. $\|x - Y\| \leq \delta$ almost surely (a.s.);<br>2. $\|\mathbb{E}[G] - \nabla f(x)\|_* \leq c_1(\delta)$; and<br>3. $\mathbb{E}\left[\|G - \mathbb{E}[G]\|_*^2\right] \leq c_2(\delta)$. | 1. $\|x - Y\| \leq \delta$ a.s.;<br>2. There exists $\tilde{f} \in \mathcal{F}$ such that $\|\tilde{f} - f\|_\infty \leq c_1(\delta)$ and $\mathbb{E}[G] = \nabla \tilde{f}(x)$;<br>3. $\mathbb{E}\left[\|G - \mathbb{E}[G]\|_*^2\right] \leq c_2(\delta)$. |

In the above, the upper bound on $\delta$ 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 a small vicinity of $\mathcal{K}$. Type-II oracle is most useful when $\tilde{f}$ is known to belong to $\mathcal{F}$, too. However, we prefer to state these conditions separately. An alternative to the bias condition for Type-II oracles, which will also be considered later, is to replace the condition $\|\tilde{f} - f\|_\infty \leq c_1(\delta)$ with[1]

$$\|\nabla \tilde{f} - \nabla f\|_* \leq c_1(\delta) \tag{1}$$

We call the resulting oracles Type-IIb. It can be shown that every Type-I oracle is a Type-II oracle for bounded $\mathcal{K}$, while the converse holds under condition (1). Finally note that for optimization it is more natural to drop $Y$ from the requirements; this is indeed what we do later.

Note that our definition allows the same oracle $\gamma$ to respond to the same inputs $(x, \delta, f)$ with a differently constructed pair (e.g., to have memory), though most often the oracles constructed in practice will map the triples $(x, \delta, f)$ to a gradient estimate-point pair using a stochastic mapping. There are several examples of oracles satisfying these definitions and a non-exhaustive list of works is as follows: (Kushner and Clark, 1978, pp. 58–60), (Katkovnik and Kulchitsky, 1972; Spall, 1992; Flaxman et al., 2005; Nesterov, 2011; Polyak and Tsybakov, 1990; Flaxman et al., 2005; Saha and Tewari, 2011; Hazan and Levy, 2014; Duchi et al., 2015). 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., $\Gamma_2(f, c_1, c_2)$).

[1] This definition requires the function $f$ to be differentiable. When the functions in $f$ are not differentiable, it is possible to modify the definitions to use subgradients. For simplicity, we will not consider this here.

In this paper we will study the minimax regret in the online BCO setting, while we study the minimax error in the stochastic BCO setting (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 minimax expected regret for $(\mathcal{F}, c_1, c_2)$ with Type-I oracles is

$$R_n^*(\mathcal{F}, c_1, c_2) = \inf_{\mathcal{A}} \sup_{f_{1:n} \in \mathcal{F}^n} \sup_{\substack{\gamma_t \in \Gamma_1(f_t, c_1, c_2) \\ 1 \le t \le n}} R_n^{\mathcal{A}}(f_{1:n}, \gamma_{1:n}) \tag{2}$$

where $\mathcal{A}$ ranges through all algorithms that interact with the $f_{1:n} = (f_1, \dots, f_n)$ loss sequence as described earlier through the oracles $\gamma_{1:n}$, and we use $R_n^{\mathcal{A}}(f_{1:n}, \gamma_{1:n})$ the expected regret of $\mathcal{A}$ against $(f_{1:n}, \gamma_{1:n})$. The minimax regret for Type-II (and other similar) oracles is defined analogously (in what follows we only define these quantities for Type-I oracles, as the extension to other types of oracles is immediate).

In the stochastic BCO setting, the minimax error is defined through

$$\Delta_n^*(\mathcal{F}, c_1, c_2) = \inf_{\mathcal{A}} \sup_{f \in \mathcal{F}} \sup_{\gamma \in \Gamma_1(f, c_1, c_2)} \Delta_n^{\mathcal{A}}(f, \gamma)\,, \tag{3}$$

where, again, $\mathcal{A}$ ranges through all algorithms that interact with the loss $f$ through $\gamma$ and $\Delta_n^{\mathcal{A}}(f, \gamma)$ is the optimization error that $\mathcal{A}$ suffers after $n$ rounds of interaction with $f$ through $\gamma$ as described earlier.

When the set $\mathcal{K}$ is bounded and the function set $\mathcal{F}$ is invariant to linear shifts, every $(c_1, c_2)$ type-I oracle is also a $(Rc_1, c_2)$ type-II oracle, where $R = \sup_{x \in \mathcal{K}} \|x\|$: Simply consider $\tilde{f}(y) = f(y) + (\mathbb{E}[G] - \nabla f(x))^\top y$, where $G$ is the gradient estimate returned by the oracle. As a result, if for some set $\Delta_n^{\text{Type-I}}$, $\Delta_n^{\text{Type-II}}$ denote the appropriate minimax errors and $R = 1$ then $\Delta_n^{\text{Type-I}}(\mathcal{F}, c_1, c_2) \le \Delta_n^{\text{Type-II}}(\mathcal{F}, c_1, c_2)$. As a result, when proving lower bounds, we shall consider Type-I, while when proving upper bounds we will consider Type-II oracles. Further note that for either type of oracles, $\Delta_n^*(\mathcal{F}, c_1, c_2) \le R_n^*/n$. 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 as $\hat{X}_n$ the average of the points $X_1, \dots, X_n$ queried by $\mathcal{A}$ during the $n$ rounds. Indeed then $f(\hat{X}_n) \le \frac{1}{n} \sum_{t=1}^n f(X_t)$ 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 is that a lower bound for $\Delta_n^*(\mathcal{F}, c_1, c_2)$ will also be a lower bound for $R_n^*/n$.

## 3 Main Results

In this section we provide our main results in forms of upper and lower bounds on the minimax error.

**Theorem 1 (*Upper bound*)** *For any $(c_1, c_2)$ Type-II oracle returning $\tilde{f}$ which inherits the properties of the objective function $f$, with $c_1(\delta) = C_1 \delta^p$, $c_2(\delta) = C_2 \delta^{-q}$, $p, q > 0$, the mirror descent algorithm (Nemirovskii and Yudin, 1983) with a $\alpha$-strongly convex (w.r.t. some norm $\|\cdot\|$) regularizer $\mathcal{R}$ such that $\operatorname{int}(\mathcal{K}) \subseteq \operatorname{dom}(\mathcal{R})$, satisfies the following bounds under appropriate choices for the learning rate $\eta_t$, and the accuracy parameter $\delta$ (these depend on $p, q, C_1, C_2$ and $L$):*

***$\mathcal{F}_{L,0}(\mathcal{K})$ (Convex and $L$-smooth):*** $\quad \Delta_n^*(\mathcal{F}_{L,0}, c_1, c_2) \le K_1 \left( \dfrac{D C_1^{\frac{q}{p}} C_2}{n} \right)^{\frac{p}{2p+q}}$,

***$\mathcal{F}_{L,\mu}(\mathcal{K})$ ($\mu$-strongly convex and $L$-smooth):*** $\quad \Delta_n^*(\mathcal{F}_{L,\mu}, c_1, c_2) \le K_2 \left( \dfrac{C_1^{\frac{q}{p}} C_2}{n} \right)^{\frac{p}{p+q}}$,

*where $D := \sup_{x,y \in \mathcal{K}} D_{\mathcal{R}}(x, y)$, $K_1$ is a constant that depends on oracle parameters $p, q$ and strong convexity constant $\alpha$ of $\mathcal{R}$ and where $K_2$ is a constant that depends on $p, q, \alpha$ and $\mu$.* □

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 and 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 $\{+1, -1\}^d \subset \mathcal{K}$ and let $L > 0$. Assume that $n$ is large enough (the exact condition is given after the theorem). Then, for any algorithm that observes $n$ random elements from a $(c_1, c_2)$ Type-I oracle with $c_1(\delta) = C_1 \delta^p$, $c_2(\delta) = C_2 \delta^{-q}$, $p, q > 0$,* *the minimax error* (3) *satisfies the following bounds:*

$\mathcal{F}_{L,0}(\mathcal{K})$ ***(Convex and** $L$**-smooth):*** $$\Delta_n^*(\mathcal{F}_{L,0}, c_1, c_2) \geq K_3 \left(\frac{d^{\frac{q}{2p}} C_1^{\frac{q}{p}} C_2^2}{n}\right)^{\frac{p}{2p+q}},$$

$\mathcal{F}_{L,1}(\mathcal{K})$ ***(1-strongly convex and** $L$**-smooth):*** $$\Delta_n^*(\mathcal{F}_{L,1}, c_1, c_2) \geq K_3^2 \left(\frac{C_1^{\frac{q}{p}} C_2^2}{dn}\right)^{\frac{2p}{2p+q}},$$

*where* $K_3 = \frac{1}{4}\left(\frac{1}{2K_4}\right)^{\frac{2p}{2p+q}}$ *and* $K_4 = \frac{p}{(p+\frac{q}{2})}\left(\frac{q}{2(p+\frac{q}{2})}\right)^{\frac{q}{2p}}$*. Further, the condition connecting the problem parameters is that* $n$ *should be large enough so that* ***(i)*** $d^{\frac{-(4p+q)}{(4p+2q)}}\left(\frac{C_1^{\frac{q}{2p}} C_2}{2\sqrt{n} K_4}\right)^{\frac{2p}{2p+q}} \leq \min\left(\frac{C_1(2p+q)}{q}, \frac{2z+1}{C_1^{2z-1}(z+1)^z}, L\right)$ *with* $z = p/q$*, for* $\mathcal{F}_{L,0}(\mathcal{K})$ *and* ***(ii)*** $d^{\frac{-(4p+q)}{(2p+q)}}\left(\frac{C_1^{\frac{q}{2p}} C_2}{2\sqrt{n} K_4}\right)^{\frac{2p}{2p+q}} \leq \min\left(\frac{C_1(2p+q)}{q}, \frac{2z+1}{C_1^{2z-1}(z+1)^z}, L\right)$ *for* $\mathcal{F}_{L,1}(\mathcal{K})$*.* □

**Remark 1** (***Scaling***) For any function class $\mathcal{F}$, by the definition of the minimax error (3), it is easy to see that $\Delta_n^*(\mu\mathcal{F}, c_1, c_2) = \mu\Delta_n^*\left(\mathcal{F}, \frac{c_1}{\mu}, \frac{c_2}{\mu^2}\right)$, where $\mu F$ denotes the function class comprised of functions in $\mathcal{F}$, each scaled by $\mu > 0$. In particular, this relation implies that the bound for $\mu$-strongly convex function class is only a constant factor away from the bound for 1-strongly convex function class. □

While it may appear that for the strongly convex case the error becomes smaller with a larger dimension, $C_1, C_2$ hide dimension dependent constants (for all oracles we know of) and thus the lower bound actually increases with the dimension increasing.

## 4 Applications to Stochastic BCO

Let us now consider how our result can be applied to stochastic bandit convex optimization. In particular, we consider the case when in round $t$ the algorithm upon querying at $X_t$ gets the observation $Z_t = f(X_t) + \xi_t$, where (for simplicity) $(\xi_t)$ is an i.i.d. noise sequence (zero mean, subgaussian tail). We consider the case when the objective function is smooth and convex. Using the so-called smoothing technique (Polyak and Tsybakov, 1990; Flaxman et al., 2005; Hazan and Levy, 2014), one can obtain type-II oracles. For example, the oracle described by Flaxman et al. (2005) (and used by many subsequent works) when queried at point $x$ with accuracy parameter $\delta$ queries $f$ at $x + \delta U$ to get $Z = f(x + \delta U) + \xi$, which is used to construct the gradient estimate $G = \frac{dZ}{\delta} U$. One can then show that if $f$ is convex and smooth, this is a type-II oracle, and in fact a type-IIb oracle, as well, with $c_1(\delta) = C_1\delta^2$, $c_2(\delta) = C_2/\delta^2$ with some specific numeric constants $C_1$ and $C_2$. Since every type-IIb oracle is also a type-I oracle, we get that this is also a type-I oracle. Now, plugging in $p = q = 2$ into the lower bound Theorem 2, we get that *no algorithm using this type, or an equivalent oracle can get better rate than* $\Omega(n^{-1/3})$ (our upper bound from Theorem 1 matches this rate, replicating the result of Saha and Tewari (2011)). In particular, the only hope to improve this rate, *while staying within the realm of our oracle model*, is to improve the bias-variance properties of the oracle (either by proving a better bound, or constructing better oracles). One case when this is possible is when the noise in the function evaluation oracle is under control (e.g., (Nesterov, 2011; Duchi et al., 2015)). In this case, for smooth convex objectives, one can use two observations (reducing the sample size by a factor of two only, i.e., in a negligible manner) to "cancel the noise out" to get an oracle with bias $c_1(\delta) = C_1\delta$ and a constant variance (i.e., $p = 1$, $q = 0$). Plugging these values into our upper and lower bounds, we get the optimal $\Theta(n^{-1/2})$ rate, replicating the results Nesterov (2011); Duchi et al. (2015). For strongly and smooth convex functions, our upper bound from Theorem 1 also gives the optimal rate, replicating the result of Hazan and Levy (2014) (when not considering the constraint set). By our earlier remarks, our lower bound also provide lower bounds on the regret. In particular, our $\Omega(n^{-1/3})$ lower bound for the smooth convex case transfers into an $\Omega(n^{2/3})$ lower bound for the regret.

## 5 Conclusions

In this paper, motivated by the quest for improving the rates in smooth stochastic bandit convex optimization, we proposed a new convex optimization framework where the optimization algorithm interacts with the objective function through a gradient evaluation oracle with a controllable bias-variance tradeoff. Our main results establish matching lower and upper bounds for the optimization error in this model. The application of this result suggests that already for the smooth-convex case, the algorithms that fit our model (which is essentially *all* algorithms considered in the literature so far) cannot achieve the optimal rate for this problem, unless either our understanding of existing gradient estimation techniques or the techniques themselves improve substantially, which remains an interesting open question.

## References


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.

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.

Flaxman, A. D., Kalai, A. T., and McMahan, H. B. (2005). Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 385–394. Society for Industrial and Applied Mathematics.

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

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

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 preprint arXiv:1402.2667*.

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

Nesterov, Y. (2011). Random gradient-free minimization of convex functions. CORE Discussion Papers 2011001, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE).

Poljak, B. and Tsypkin, Y. Z. (1973). Pseudogradient adaptation and training algorithms. *Automation and Remote Control*, 34:45–67.

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 *International Conference on Artificial Intelligence and Statistics*, pages 636–642.

Shamir, O. (2012). On the complexity of bandit and derivative-free stochastic convex optimization. In *Proceedings of the Twenty Sixth Annual Conference on Computational Learning Theory*.

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