
Uniform Wrappers: Bridging Concave to Quadratizable Functions in Online Optimization

Mohammad Pedramfar

Mila - Quebec AI Institute/McGill University
`mohammad.pedramfar@mila.quebec`

Christopher J. Quinn

Iowa State University
`cjquinn@iastate.edu`

Vaneet Aggarwal

Purdue University
`vaneet@purdue.edu`

Abstract

This paper presents novel contributions to the field of online optimization, particularly focusing on the adaptation of algorithms from concave optimization to more challenging classes of functions. Key contributions include the introduction of uniform wrappers, a class of meta-algorithms that could be used for algorithmic conversions such as converting algorithms for convex optimization into those for quadratizable optimization. Moreover, we propose a guideline that, given a base algorithm \mathcal{A} for concave optimization and a uniform wrapper \mathcal{W} , describes how to convert a proof of the regret bound of \mathcal{A} in the concave setting into a proof of the regret bound of $\mathcal{W}(\mathcal{A})$ for quadratizable setting. Through this framework, the paper demonstrates improved regret guarantees for various classes of DR-submodular functions under zeroth-order feedback. Furthermore, the paper extends zeroth-order online algorithms to bandit feedback and offline counterparts, achieving notable improvements in regret/sample complexity compared to existing approaches.

1 Introduction

The optimization of continuous DR-submodular functions has become increasingly prominent in recent years. This form of optimization represents an important subset of non-convex optimization problems at the forefront of machine learning and statistics. These challenges have numerous real-world applications like revenue maximization, mean-field inference, and recommendation systems, among others [Bian et al., 2019, Hassani et al., 2017, Mitra et al., 2021, Djolonga and Krause, 2014, Ito and Fujimaki, 2016, Gu et al., 2023, Li et al., 2023].

A natural starting point for DR-submodular maximization is to begin with a convex optimization algorithm and adapt it to the setting of DR-submodular functions. Online Convex Optimization (OCO) is extensively utilized across various fields due to its numerous practical applications and robust theoretical underpinnings. The tools from the area of online convex optimization have been applied to many online non-concave optimization algorithms, e.g., to converge to stationary points in online non-concave optimization [Yang et al., 2018], or algorithms with approximation guarantees for DR-submodular optimization [Chen et al., 2018, Niazadeh et al., 2020, Zhang et al., 2022, Pedramfar et al., 2023].

In this paper, we focus on a large class of functions, namely the class of quadratizable functions, first introduced in [Pedramfar and Aggarwal, 2024a]. Quadratizable functions includes special subclasses of non-convex/non-concave functions where the offline constrained optimization problem is NP-hard to solve but we can find an α -approximation of the optimal value in polynomial time. Indeed, it is shown that this class of online upper quadratizable optimization includes up-concave

optimization (a generalization of DR-submodular and concave optimization) in the following cases: (i) monotone γ -weakly μ -strongly DR-submodular functions with curvature c over general convex sets, (ii) monotone γ -weakly DR-submodular functions over convex sets containing the origin, and (iii) non-monotone DR-submodular optimization over general convex sets.

Even though the tools from OCO have proven effective in more challenging classes, much of the past work along these lines take inspiration from OCO and manually designing new algorithms and analyzing them specific to each problem setting. This raises the following question

When and how can we adapt algorithms from the (simpler) setup of online convex optimization into algorithms for online optimization over more general classes of functions?

In this paper, we try to provide partial solutions to this question for adapting OCO algorithms to algorithms for online quadratizable optimization. The notion of quadratizability is built upon a generalization of the defining condition $f(\mathbf{x}) - f(\mathbf{y}) \geq \langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle$ of convex functions. This similarity with convex functions is a starting point which allows us to define a class of meta-algorithms called “uniform wrappers”. Uniform wrappers provide a straightforward way to convert OCO algorithms into algorithms that can handle quadratizable functions. We also develop a guideline to convert the existing proofs for regret bounds of the base algorithms in the convex setting into regret bounds of the new algorithms over quadratizable functions.

We note that for a specific class of algorithms, this question was partly addressed in [Pedramfar and Aggarwal, 2024a]. Specifically, as we will discuss in Appendix E, their result can be formulated as a special case of ours, where they assume that the starting algorithm is a deterministic first order online algorithm with semi-bandit feedback. This condition is too restrictive to facilitate adapting many of the ideas in the OCO literature. For example, their result could not be applied to Follow-The-Perturbed-Leader type algorithms, or more generally any algorithm that is not deterministic. Another important setting where their result could not be applied is the setting with zeroth-order feedback. Without using a zeroth-order algorithm as the base algorithm, we can obtain zeroth-order results by converting first-order algorithms to zeroth-order ones. However, such conversion comes at a cost. For example, in the convex case, taking an optimal first-order algorithm such as Online Gradient Descent and converting it into a zeroth-order algorithm will result in sub-optimal $O(T^{3/4})$ regret [Flaxman et al., 2005]. In contrast, using our framework, starting from a zeroth-order base algorithm, we obtain $O(T^{2/3})$ regret for zeroth-order feedback. In particular, our approach recovers the result of [Wan et al., 2023] while their approach could not. (See Table 1.)

In this paper, we take a step further and can handle broader classes of algorithms, including the more challenging setting of zeroth order feedback. Note that, without limiting ourselves to classes of algorithms such as the one discussed in [Pedramfar and Aggarwal, 2024a], there is not much hope of obtaining a single theorem that captures conversions of any OCO algorithm (and proof) into one for online quadratizable optimization. Instead, we provide a guideline that captures the process of converting algorithms and proofs for convex optimization into ones for quadratizable optimization.

As an application of our framework, we propose a variant of a bandit convex optimization algorithm that was introduced in [Saha and Tewari, 2011] as the base algorithm, namely Zeroth Order Follow the Regularized Leader (ZO-FTRL) and demonstrate how it can be converted using uniform wrappers (denoted by \mathcal{W}) to obtain 3 algorithms for function classes (i)-(iii) mentioned above. See Tables 1 and 2 for details. Note that ZO-FTRL and $\mathcal{W}(\text{ZO-FTRL})$ are zeroth order, but they are not bandit feedback algorithms. We also extend the results to those with bandit feedback, as well as derive sample complexity guarantees for the offline algorithm.

We consider 3 classes of weakly DR-submodular functions, specifically (i) monotone γ -weakly μ -strongly DR-submodular functions with curvature c over general convex sets, (ii) monotone γ -weakly DR-submodular functions over convex sets containing the origin, and (iii) non-monotone DR-submodular optimization over general convex sets. The main contributions in this work include:

1. We develop a general framework for converting algorithms and their regret guarantees from online convex concave optimization to online quadratizable optimization. Conversion of the algorithm could be applied to any online optimization algorithm, and the conversion of the proof is described using a general guideline.

Table 1: Online up-concave maximization

F	Set	Feedback		Reference	Appx.	# of queries	$\log_T(\alpha\text{-regret})$
Monotone	$0 \in \mathcal{K}$	∇F	Full Information	stoch.	$1 - e^{-\gamma}$ [Zhang et al., 2022] (*) [Pedramfar et al., 2024a]	1	$1/2$
			Semi-bandit	stoch.	$1 - e^{-1}$ [Pedramfar et al., 2024a] [Pedramfar and Aggarwal, 2024a] (*)	$T^\theta(\theta \in [0, 1/2])$	$2/3 - \theta/3$
		F	Full Information	det.	$1 - e^{-1}$ [Pedramfar and Aggarwal, 2024a] (*)	1	$1/2$
				stoch. \dagger	$1 - e^{-1}$ Theorem 5	1	$2/3$
			Bandit	stoch.	$1 - e^{-1}$ [Pedramfar et al., 2024a] [Pedramfar and Aggarwal, 2024a] (*)	$T^\theta(\theta \in [0, 1/4])$	$4/5 - \theta/5$
				det.	$1 - e^{-1}$ [Wan et al., 2023] (*) [Zhang et al., 2024] (*)	1	$3/4$
		∇F	Semi-bandit	stoch.	$1 - e^{-1}$ [Chen et al., 2018] (*) [Pedramfar et al., 2024a]	1	$4/5$
				stoch.	$1 - e^{-1}$ [Pedramfar and Aggarwal, 2024a] (*)	1	$3/4$
			Full Information	det.	$1/2$ [Pedramfar and Aggarwal, 2024a] (*)	$T^\theta(\theta \in [0, 1/2])$	$2/3 - \theta/3$
				stoch.	$\gamma^2/(1 + \gamma^2)$ [Pedramfar et al., 2024a]	1	$1/2$
Non-Monotone	general	F	Full Information	det.	$1/2$ [Pedramfar and Aggarwal, 2024a] (*)	2	$1/2$
				stoch.	$\gamma^2/(1 + c\gamma^2)$ Theorem 5	1	$2/3$
			Bandit	stoch.	$1/2$ [Pedramfar et al., 2024a] [Pedramfar and Aggarwal, 2024a] (*)	1	$5/6$
				stoch.	$\gamma^2/(1 + c\gamma^2)$ Theorem 6	1	$3/4$
		∇F	Full Information	stoch.	$(1 - h)/4$ [Pedramfar et al., 2024a] [Zhang et al., 2024] (*) [Pedramfar and Aggarwal, 2024a] (*)	$T^\theta(\theta \in [0, 1/2])$	$2/3 - \theta/3$
			Semi-bandit	stoch.	$(1 - h)/4$ [Pedramfar et al., 2024a]	1	$1/2$
			Full Information	stoch.	$(1 - h)/4$ [Pedramfar and Aggarwal, 2024a] (*)	1	$1/2$
				stoch.	$(1 - h)/4$ [Pedramfar and Aggarwal, 2024a] (*)	1	$3/4$
			Bandit	det.	$(1 - h)/4$ [Zhang et al., 2024] (*)	2	$1/2$
				stoch. \dagger	$(1 - h)/4$ Theorem 5	1	$2/3$
		F	Full Information	stoch.	$(1 - h)/4$ [Pedramfar et al., 2024a] [Pedramfar and Aggarwal, 2024a] (*)	$T^\theta(\theta \in [0, 1/4])$	$4/5 - \theta/5$
				stoch.	$(1 - h)/4$ [Pedramfar and Aggarwal, 2024a] (*)	1	$3/4$
			Bandit	det.	$(1 - h)/4$ [Pedramfar et al., 2024a]	1	$4/5$
				stoch.	$(1 - h)/4$ [Pedramfar and Aggarwal, 2024a] (*)	1	$3/4$

This table compares different static regret results for the online up-concave maximization. The logarithmic terms in regret are ignored. Here $h := \min_{\mathbf{z} \in \mathcal{K}} \|\mathbf{z}\|_\infty$. **Rows marked with (*) are results in the literature that are special cases of the results stated here and therefore fit within the framework described in this paper.** The rows describing results with stochastic feedback that are marked with \dagger assume that the random query oracle is contained with a cone, as detailed in Theorem 5. Note that such oracles cover deterministic oracles as a special case.

2. To the best our knowledge, in all cases considered, our framework recovers all algorithms in the literature for continuous DR-submodular maximization that are not Frank-Wolfe type. (See Appendix B for more details.) This includes all results marked with (*) in Tables 1 and 2.
3. In all online optimization settings considered, our framework either beats the SOTA or successfully recovers it (see Table 1). Note that our framework also recovers all known results for non-stationary DR-submodular maximization. (See Remark 11 and Table 3 in [Pedramfar and Aggarwal, 2024a].)
4. In all offline optimization settings considered—except for deterministic first-order feedback and the special case of γ -weakly non-monotone functions with $\gamma < 1$ —our framework either beats the SOTA or successfully recovers it. (See Table 2.)
5. When considering either deterministic or fully stochastic query oracles, we obtain superior regret guarantees for 3 settings. Specifically, for online or offline zeroth order feedback, we improve the SOTA for (ii) and for bandit feedback, we improve the SOTA for (iii). (See Tables 1 and 2)
6. We define a new notion of query oracles that are stochastic, but contained within a cone. (See Section 8) When considering these types of oracles (together with deterministic and fully stochastic query oracles), we improve SOTA for 6 more settings. Specifically, for online zeroth order, offline zeroth order and bandit feedback settings for (i) and (iii). (See Tables 1 and 2)

To simplify the notation and statements, we define regret for maximization problems and focus on concave maximization and DR-submodular maximization.

2 Background and Notation

For a set $\mathcal{D} \subseteq \mathbb{R}^d$, we define its *affine hull* $\text{aff}(\mathcal{D})$ to be the set of $\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}$ for all \mathbf{x}, \mathbf{y} in \mathcal{D} and $\alpha \in \mathbb{R}$. The *relative interior* of \mathcal{D} is defined as $\text{relint}(\mathcal{D}) := \{\mathbf{x} \in \mathcal{D} \mid \exists r > 0, \mathbb{B}_r(\mathbf{x}) \cap \text{aff}(\mathcal{D}) \subseteq \mathcal{D}\}$. All convex functions are continuous on any point in the relative interior of their domains. In this work, we will only focus on (everywhere) continuous functions. If $\mathbf{x} \in \text{relint}(\mathcal{K})$ and f is convex

and is differentiable at \mathbf{x} , then we have $f(\mathbf{y}) - f(\mathbf{x}) \geq \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$, for all $\mathbf{y} \in \mathcal{K}$. More generally, given $\mu \geq 0$, we say a vector $\mathbf{o} \in \mathbb{R}^d$ is a μ -subgradient of f at \mathbf{x} if $f(\mathbf{y}) - f(\mathbf{x}) \geq \langle \mathbf{o}, \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2$. for all $\mathbf{y} \in \mathcal{K}$. Given a convex set \mathcal{K} , a function $f : \mathcal{K} \rightarrow \mathbb{R}$ is μ -strongly convex if and only if it has a μ -subgradient at all points $\mathbf{x} \in \mathcal{K}$. A function $F : \mathcal{D} \rightarrow \mathbb{R}^+$ is G -Lipschitz continuous if for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$, $\|F(\mathbf{x}) - F(\mathbf{y})\| \leq G \|\mathbf{x} - \mathbf{y}\|$. A differentiable function $F : \mathcal{D} \rightarrow \mathbb{R}^+$ is L -smooth if for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$, $\|\nabla F(\mathbf{x}) - \nabla F(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|$. Given a continuous monotone function $f : \mathcal{K} \rightarrow \mathbb{R}$, its curvature is defined as the smallest number $c \in [0, 1]$ such that $f(\mathbf{y} + \mathbf{z}) - f(\mathbf{y}) \geq (1 - c)(f(\mathbf{x} + \mathbf{z}) - f(\mathbf{x}))$, for all $\mathbf{x}, \mathbf{y} \in \mathcal{K}$ and $\mathbf{z} \geq 0$ such that $\mathbf{x} + \mathbf{z}, \mathbf{y} + \mathbf{z} \in \mathcal{K}$. We define the curvature of a function class \mathbf{F} as the supremum of the curvature of functions in \mathbf{F} .

We say $\tilde{\nabla}f : \mathcal{K} \rightarrow \mathbb{R}^d$ is a μ -strongly γ -weakly up-super-gradient of f if for all $\mathbf{x} \leq \mathbf{y}$ in \mathcal{K} , we have $\gamma(\langle \tilde{\nabla}f(\mathbf{y}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2) \leq f(\mathbf{y}) - f(\mathbf{x}) \leq \frac{1}{\gamma}(\langle \tilde{\nabla}f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2)$. Then we say f is μ -strongly γ -weakly up-concave if it is continuous and it has a μ -strongly γ -weakly up-super-gradient. When $\gamma = 1$ and the above inequality holds for all $\mathbf{x}, \mathbf{y} \in \mathcal{K}$, we say f is μ -strongly concave. A differentiable function $f : \mathcal{K} \rightarrow \mathbb{R}$ is called continuous DR-submodular if for all $\mathbf{x} \leq \mathbf{y}$, we have $\nabla f(\mathbf{x}) \geq \nabla f(\mathbf{y})$. More generally, we say f is γ -weakly continuous DR-submodular if for all $\mathbf{x} \leq \mathbf{y}$, we have $\nabla f(\mathbf{x}) \geq \gamma \nabla f(\mathbf{y})$. It follows that any γ -weakly continuous DR-submodular functions is γ -weakly up-concave.

3 Problem setup

Online optimization problems can be formalized as a repeated game between an agent and an adversary. The game lasts for T rounds on a convex domain \mathcal{K} where T and \mathcal{K} are known to both players. In the t -th round, the agent chooses an action \mathbf{x}_t from an action set $\mathcal{K} \subseteq \mathbb{R}^d$, then the adversary chooses a loss function $f_t \in \mathbf{F}$ and a query oracle for the function f_t . Then, for some $k_t \geq 0$, and $1 \leq i \leq k_t$, the agent chooses a points $\mathbf{y}_{t,i}$ and receives the outputs of the query oracle.

To be more precise, an agent consists of a tuple $(\Omega^{\mathcal{A}}, \mathcal{A}^{\text{action}}, \mathcal{A}^{\text{query}})$, where $\Omega^{\mathcal{A}}$ is a probability space that captures all the randomness of \mathcal{A} . We assume that, before the first action, the agent samples $\omega \in \Omega$. The next element in the tuple, $\mathcal{A}^{\text{action}} = (\mathcal{A}_1^{\text{action}}, \dots, \mathcal{A}_T^{\text{action}})$ is a sequence of functions such that \mathcal{A}_t that maps the history $\Omega^{\mathcal{A}} \times \mathcal{K}^{t-1} \times \prod_{s=1}^{t-1} (\mathcal{K} \times \mathcal{O})^{k_s}$ to $\mathbf{x}_t \in \mathcal{K}$ where we use \mathcal{O} to denote range of the query oracle. The last element in the tuple, $\mathcal{A}^{\text{query}}$, is the query policy. For each $1 \leq t \leq T$ and $1 \leq i \leq k_t$, $\mathcal{A}_{t,i}^{\text{query}} : \Omega^{\mathcal{A}} \times \mathcal{K}^t \times \prod_{s=1}^{t-1} (\mathcal{K} \times \mathcal{O})^{k_s} \times (\mathcal{K} \times \mathcal{O})^{i-1}$ is a function that, given previous actions and observations, either selects a point $\mathbf{y}_t^i \in \mathcal{K}$, i.e., query, or signals that the query policy at this time-step is terminated. We may drop ω as one of the inputs of the above functions when there is no ambiguity. We say the agent query function is *trivial* if $k_t = 1$ and $\mathbf{y}_{t,1} = \mathbf{x}_t$ for all $1 \leq t \leq T$. In this case, we simplify the notation and use the notation $\mathcal{A} = \mathcal{A}^{\text{action}} = (\mathcal{A}_1, \dots, \mathcal{A}_T)$ to denote the agent action functions and assume that the domain of \mathcal{A}_t is $\Omega^{\mathcal{A}} \times (\mathcal{K} \times \mathcal{O})^{t-1}$.

A query oracle is a function that provides the observation to the agent. Formally, a query oracle for a function f is a map \mathcal{Q} defined on \mathcal{K} such that for each $\mathbf{x} \in \mathcal{K}$, the $\mathcal{Q}(\mathbf{x})$ is a random variable taking value in the observation space \mathcal{O} . The query oracle is called a *stochastic value oracle* or *stochastic zeroth order oracle* if $\mathcal{O} = \mathbb{R}$ and $f(\mathbf{x}) = \mathbb{E}[\mathcal{Q}(\mathbf{x})]$. Similarly, it is called a *stochastic up-super-gradient oracle* or *stochastic first order oracle* if $\mathcal{O} = \mathbb{R}^d$ and $\mathbb{E}[\mathcal{Q}(\mathbf{x})]$ is a up-super-gradient of f at \mathbf{x} . In all cases, if the random variable takes a single value with probability one, we refer to it as a *deterministic oracle*. Note that, given a function, there is at most a single deterministic gradient oracle, but there may be many deterministic up-super-gradient oracles. We will use ∇ to denote the deterministic gradient oracle. We say an oracle is bounded by B if its output is always within the Euclidean ball of radius B centered at the origin. We say the agent takes *semi-bandit feedback* if the oracle is first-order and the agent query function is trivial. Similarly, it takes *bandit feedback* if the oracle is zeroth-order and the agent query function is trivial. If the agent query function is non-trivial, then we say the agent requires *full-information feedback*.

An adversary Adv is a set such that each element $\mathcal{B} \in \text{Adv}$, referred to as a *realized adversary*, is a sequence $(\mathcal{B}_1, \dots, \mathcal{B}_T)$ of functions where each \mathcal{B}_t maps a tuple $(\mathbf{x}_1, \dots, \mathbf{x}_t) \in \mathcal{K}^t$ to a tuple (f_t, \mathcal{Q}_t) where $f_t \in \mathbf{F}$ and \mathcal{Q}_t is a query oracle for f_t . We say an adversary Adv is *oblivious* if for any realization $\mathcal{B} = (\mathcal{B}_1, \dots, \mathcal{B}_T)$, all functions \mathcal{B}_t are constant, i.e., they are independent of $(\mathbf{x}_1, \dots, \mathbf{x}_t)$. In this case, a realized adversary may be simply represented by a sequence of func-

tions $(f_1, \dots, f_T) \in \mathbf{F}^T$ and a sequence of query oracles $(\mathcal{Q}_1, \dots, \mathcal{Q}_T)$ for these functions. In this work we also consider adversaries that are *fully adaptive*, i.e., adversaries with no restriction. Given a function class \mathbf{F} and $i \in \{0, 1\}$, we use $\text{Adv}_i^f(\mathbf{F})$ to denote the set of all possible realized adversaries with deterministic i -th order oracles. If the oracle is instead stochastic and bounded by B , we use $\text{Adv}_i^f(\mathbf{F}, B)$ to denote such an adversary. Finally, we use $\text{Adv}_i^o(\mathbf{F})$ and $\text{Adv}_i^o(\mathbf{F}, B)$ to denote all oblivious realized adversaries with i -th order deterministic and stochastic oracles, respectively.

In order to handle different notions of regret with the same approach, for an agent \mathcal{A} , adversary Adv , compact set $\mathcal{U} \subseteq \mathcal{K}^T$, approximation coefficient $0 < \alpha \leq 1$ and $1 \leq a \leq b \leq T$, we define *regret* as

$$\mathcal{R}_{\alpha, \text{Adv}}^{\mathcal{A}}(\mathcal{U})[a, b] := \sup_{\mathcal{B} \in \text{Adv}} \mathbb{E} \left[\alpha \max_{\mathbf{u}=(\mathbf{u}_1, \dots, \mathbf{u}_T) \in \mathcal{U}} \sum_{t=a}^b f_t(\mathbf{u}_t) - \sum_{t=a}^b f_t(\mathbf{x}_t) \right],$$

where the expectation in the definition of the regret is over the randomness of the algorithm and the query oracle. We use the notation $\mathcal{R}_{\alpha, \mathcal{B}}^{\mathcal{A}}(\mathcal{U})[a, b] := \mathcal{R}_{\alpha, \text{Adv}}^{\mathcal{A}}(\mathcal{U})[a, b]$ when $\text{Adv} = \{\mathcal{B}\}$ is a singleton. We may drop α when it is equal to 1. When $\alpha < 1$, we often assume that the functions are non-negative. *Static adversarial regret* or simply *adversarial regret* corresponds to $a = 1, b = T$ and $\mathcal{U} = \mathcal{K}_*^T := \{(\mathbf{x}, \dots, \mathbf{x}) \mid \mathbf{x} \in \mathcal{K}\}$. When $a = 1, b = T$ and \mathcal{U} contains only a single element then it is referred to as the *dynamic regret* [Zinkevich, 2003, Zhang et al., 2018]. *Adaptive regret*, is defined as $\max_{1 \leq a \leq b \leq T} \mathcal{R}_{\alpha, \text{Adv}}^{\mathcal{A}}(\mathcal{K}_*^T)[a, b]$ [Hazan and Seshadhri, 2009]. We drop a, b and \mathcal{U} when the statement is independent of their value or their value is clear from the context.

4 Uniform wrappers

We next introduce a class of meta-algorithms that will be a central element of our proposed framework for adapting algorithms. At a high level, the meta-algorithms we consider wrap around the base algorithm and translate each action and feedback signal between the base algorithm and the adversary. The qualifier “uniform” highlights that the translations are one-to-one and independent of time.

Definition 1. Given a function class \mathbf{F} and a family of query oracles \mathcal{Q} over \mathbf{F} , we say a *uniform wrapper* $\mathcal{W} = (\mathcal{W}^{\text{action}}, \mathcal{W}^{\text{function}}, \mathcal{W}^{\text{query}})$ is a tuple of maps where $\mathcal{W}^{\text{action}} : \mathcal{K} \rightarrow \mathcal{K}$, $\mathcal{W}^{\text{function}} : \mathbf{F} \rightarrow \mathbf{H}$ for a function class \mathbf{H} and for any $f \in \mathbf{F}$ and any query oracle $\mathcal{Q}_f \in \mathcal{Q}$, $\mathcal{W}^{\text{query}}(\mathcal{Q}_f)$ is a query oracle for $\mathcal{W}^{\text{function}}(f) \in \mathbf{H}$.¹ Given an oblivious adversary Adv choosing functions in \mathbf{F} and query oracles in \mathcal{Q} , we define $\mathcal{W}(\text{Adv})$ to be the adversary over \mathbf{H} where the selected function and query by the adversary are transformed according to $\mathcal{W}^{\text{function}}$ and $\mathcal{W}^{\text{query}}$.² We say $\mathcal{W} = \text{Id}$ if all the maps in \mathcal{W} are identity.

In Section C we will discuss several examples of uniform wrappers for up-concave optimization. We drop the superscripts and use $\mathcal{W}(\mathbf{x})$, $\mathcal{W}(f)$ and $\mathcal{W}(\mathcal{Q}_f)$ to denote $\mathcal{W}^{\text{action}}(\mathbf{x})$, $\mathcal{W}^{\text{function}}(f)$ and $\mathcal{W}^{\text{query}}(\mathcal{Q}_f)$, respectively, when there is no ambiguity.

Meta-algorithm 1 details the pseudo-code for $\mathcal{W}(\mathcal{A})$ for a uniform wrapper \mathcal{W} and an online optimization algorithm \mathcal{A} . Note that, when $\mathcal{W} = \text{Id}$, the meta-algorithm also reduces to the identity meta-algorithm and we see that $\mathcal{W}(\mathcal{A}) = \mathcal{A}$. Note that in the special case where \mathcal{A} is an online algorithm with semi-bandit feedback, Meta-algorithm 1 reduces to Algorithm 1

Meta-algorithm 1: Application of a uniform wrapper to the base algorithm - $\mathcal{W}(\mathcal{A})$

Input : horizon T , algorithm \mathcal{A} , uniform wrapper \mathcal{W}

for $t = 1, 2, \dots, T$ **do**

Play $\mathcal{W}^{\text{action}}(\mathbf{x}_t)$ where \mathbf{x}_t is the action chosen by $\mathcal{A}^{\text{action}}$

The adversary selects f_t and a query oracle \mathcal{Q}_t for f_t

for i starting from 1, while $\mathcal{A}^{\text{query}}$ is not terminated for this time-step **do**

Let $\mathbf{y}_{t,i}$ be the query chosen by $\mathcal{A}^{\text{query}}$

Return $\mathbf{o}_{t,i} = \mathcal{W}^{\text{query}}(\mathcal{Q}_t)(\mathbf{y}_{t,i})$ as the output of the query oracle to $\mathcal{A}^{\text{query}}$

end

end

¹In general $\mathcal{W}^{\text{action}}$ may map a point to a random variable taking values in \mathcal{K} . In this work, we only consider the cases where $\mathcal{W}^{\text{action}}$ is deterministic.

²Recall that, in the case of non-oblivious adversary, each realized adversary $\mathcal{B} \in \text{Adv}$ is a function mapping the history to the next choice of the adversary. In this case, given a history \mathbf{h}_t , we define $\mathcal{W}(\mathcal{B})(\mathbf{h}_t)$ to be $(\mathcal{W}^{\text{function}}(\hat{f}_t), \mathcal{W}^{\text{query}}(\hat{\mathcal{Q}}_t))$ where $(\hat{f}_t, \hat{\mathcal{Q}}_t) = \mathcal{B}(\mathcal{W}^{\text{action}}(\mathbf{h}_t))$. When the adversary is oblivious, the output of any realized adversary \mathcal{B} does not depend on the history of actions \mathbf{h}_t and we may define $\mathcal{W}(\text{Adv})$ only using $\mathcal{W}^{\text{function}}$ and $\mathcal{W}^{\text{query}}$.

in [Pedramfar and Aggarwal, 2024a]. See Appendix A for more details.

In this paper, we will design uniform wrappers that could convert algorithms for concave optimization into algorithms for more general class of functions that contains many DR-submodular functions. Specifically, we consider upper-quadratizable/linearizable functions which we will discuss in the following section.

5 Linearizable and quadratizable functions classes

We next define an important function class significantly generalizes concavity but preserves enough structure that will enable us to obtain improved regret bounds for various problems.

Definition 2 ([Pedramfar and Aggarwal, 2024a]). Let $\mathcal{K} \subseteq \mathbb{R}^d$ be a convex set, \mathbf{F} be a function class over \mathcal{K} . We say the function class \mathbf{F} is *upper quadratizable* if there are maps $\mathbf{g} : \mathbf{F} \times \mathcal{K} \rightarrow \mathbb{R}^d$ and $h : \mathcal{K} \rightarrow \mathcal{K}$ and constants $\mu \geq 0$, $0 < \alpha \leq 1$ and $\beta > 0$ such that

$$\alpha f(\mathbf{y}) - f(h(\mathbf{x})) \leq \beta \left(\langle \mathbf{g}(f, \mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right). \quad (1)$$

As a special case, when $\mu = 0$, we say \mathbf{F} is *upper linearizable*. By setting $\mathbf{g}(f, \mathbf{x}) = \nabla f(\mathbf{x})$, $h = \text{Id}_{\mathcal{K}}$ and $\alpha = \beta = 1$, we see that the notion of upper linearizability generalizes concavity and upper quadratizability generalizes strong concavity. It was shown in [Pedramfar and Aggarwal, 2024a] that several classes of DR-submodular (and up-concave) functions are upper quadratizable. (see Lemmas 1, 2 and 3) A similar notion of *lower-quadratizable/linearizable* may be similarly defined for minimization problems such as convex minimization³.

Definition 3. We say \mathbf{F} is *upper quadratizable with a uniform wrapper* \mathcal{W} if $\mathcal{W}(\mathbf{F})$ is defined and differentiable over \mathcal{K} and, for all $f \in \mathbf{F}$, we have

$$\alpha f(\mathbf{y}) - f(\mathcal{W}(\mathbf{x})) \leq \beta \left(\langle \nabla \mathcal{W}(f)(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right). \quad (2)$$

Note that a uniform wrapper is not uniquely determined by h and \mathbf{g} in the definition of upper quadratizable functions as it also needs to describe transformations of query oracles. The special case with $\alpha = \beta = 1$, $\mathcal{W} = \text{Id}$ reduces to the definition of (strong) concavity. In Appendix C, we will construct uniform wrappers for several classes of upper quadratizable functions, specifically, monotone μ -strongly γ -weakly up-concave functions with bounded curvature (\mathbf{F}^M), monotone γ -weakly up-concave functions over convex sets containing the origin (\mathbf{F}^{M0}), and non-monotone up-concave functions over general convex sets (\mathbf{F}^{NM}).

6 When Is Concave Optimization Enough?

As can be seen in Meta-algorithm 1, we may apply a uniform wrapper \mathcal{W} to any online optimization algorithm \mathcal{A} . However, even if the original algorithm has a sublinear regret over concave functions and \mathbf{F} is a function class that is upper quadratizable with \mathcal{W} , this does not guarantee that the resulting algorithm $\mathcal{W}(\mathcal{A})$ has a sublinear regret over \mathbf{F} . In this section we discuss how we might convert the proofs of the regret bound for \mathcal{A} over concave functions into a proof of a similar regret bound for $\mathcal{W}(\mathcal{A})$ over \mathbf{F} . That starting point is to notice that a game played between $\mathcal{W}(\mathcal{A})$ and Adv is mathematically equivalent to one played between \mathcal{A} and $\mathcal{W}(\text{Adv})$. Therefore we need to convert the regret guarantees of \mathcal{A} against a concave adversary into guarantees for \mathcal{A} against $\mathcal{W}(\text{Adv})$. See Appendix A for more details.

The guideline for converting proof for concave optimization into proofs for upper-quadratizable optimization can be informally summarized in a few steps:

- (0) Sometimes, if the algorithm \mathcal{A} is the result of application of a meta-algorithm to another algorithm \mathcal{B} , e.g. $\mathcal{A} = \text{SFTT}(\mathcal{B})$ (the meta-algorithm SFTT converts algorithms that require full-information feedback to ones that work with (semi)-bandit feedback; see Appendix M), we may

³We say \mathbf{F} is lower quadratizable if $\alpha f(\mathbf{y}) - f(h(\mathbf{x})) \geq \beta \left(\langle \mathbf{g}(f, \mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right)$. This generalizes the notion of convexity and strong convexity.

need to consider the base algorithm instead. For example, in the example of SFTT, we might want to consider SFTT($\mathcal{W}(\mathcal{B})$) instead of $\mathcal{W}(\text{SFTT}(\mathcal{B})) = \mathcal{W}(\mathcal{A})$.

- (1) Rewrite the parts of proof (after possibly adapting the algorithm) of the original regret bound without assuming that the function class is concave, in order to isolate the use on concavity in the proof. In this step, we hope to obtain a result that would only require a single use of an inequality of the type $f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2$ to complete the proof for the concave case. See Theorems 1, 3 (as an example of a family of zeroth order results), and 11 (as an example of a family of first order results) for examples of this step.
- (2) Verify that the results of the previous step could be adapted to upper-quadratizable setting. See the proof of Theorems 2, 4, and 12 for examples of this step.

In the following section, we discuss a version of Follow The Regularized Leader (FTRL) algorithm for concave optimization and adapt it to fit the guidelines discussed above. As another application of the guideline, we refer to Appendix E for a discussion of applying this guideline to recover some previous results in the literature, including all the results in Tables 1 and 2 that are marked with (*).

7 Examples

In this section, we provide examples of applying uniform wrappers and the guideline described earlier. The first example is illustrative and recovers some existing results, while the second example provides new results, as we will discuss in Section 8.

7.1 Online Gradient Ascent

Here we clarify the guideline in Section 6 by applying it to a simple base algorithm, namely Online Gradient Ascent (OGA).

Note that this is only an illustrative example and a much more general result is proven in Appendix E.

In Algorithm 2, we use \mathcal{Q}_t to denote the stochastic first order query oracle for f_t which returns unbiased estimates of ∇f_t . The effect of Meta-algorithm 1 is that it transforms the action of the base algorithm using $\mathcal{W}^{\text{action}}$ and query oracle of the base algorithm using $\mathcal{W}^{\text{query}}$. If we apply the first-order to first-order uniform wrapper $\mathcal{W}_1^{\text{NM}}$ (for non-monotone up-concave functions over general convex sets; see Appendix C.3 for additional details), we get a first order algorithm for online optimization of non-monotone DR-submodular function over general convex sets. The resulting algorithm is detailed in Algorithm 3.

By plugging in the definitions of $(\mathcal{W}_1^{\text{NM}})^{\text{action}}$ and $(\mathcal{W}_1^{\text{NM}})^{\text{query}}$ as per Appendix C.3, we get the Algorithm 4 which is Algorithm 1 in [Zhang et al., 2024] for the case of non-monotone functions.

The conversion of the regret bound is the focus of Section 6, in the form of a three-step guideline. Since that is a guideline that depends on the structure and the details of the original proof, any explanation of the application of the guideline requires an understanding of the original proof. For OGA specifically, we may simplify the proofs and the guideline in the following manner:

Algorithm 2: Online Gradient Ascent (OGA)

Input : horizon T , convex set \mathcal{K} , $\mathbf{x}_1 \in \mathcal{K}$,
step-sizes $\{\eta_t\}_{t=1}^T$
for $t = 1, 2, \dots, T$ **do**
| Play \mathbf{x}_t
| The adversary selects f_t and a query oracle \mathcal{Q}_t for f_t
| Observe \mathbf{o}_t , a sample of $\mathcal{Q}_t(\mathbf{x}_t)$, an unbiased estimate of $\nabla f_t(\mathbf{x}_t)$
| $\mathbf{x}_{t+1} \leftarrow P_{\mathcal{K}}(\mathbf{x}_t + \eta_t \mathbf{o}_t)$
end

Algorithm 3: Application of $\mathcal{W}_1^{\text{NM}}$ to Online Gradient Ascent ($\mathcal{W}_1^{\text{NM}}(\text{OGA})$)

Input : horizon T , convex set \mathcal{K} , $\mathbf{x}_1, \underline{\mathbf{x}} \in \mathcal{K}$,
step-sizes $\{\eta_t\}_{t=1}^T$
for $t = 1, 2, \dots, T$ **do**
| Play $(\mathcal{W}_1^{\text{NM}})^{\text{action}}(\mathbf{x}_t)$
| The adversary selects f_t and a query oracle \mathcal{Q}_t for f_t
| Observe \mathbf{o}_t , a sample of $(\mathcal{W}_1^{\text{NM}})^{\text{query}}(\mathcal{Q}_t)(\mathbf{x}_t)$
| $\mathbf{x}_{t+1} \leftarrow P_{\mathcal{K}}(\mathbf{x}_t + \eta_t \mathbf{o}_t)$
end

As per Step 1 of the guideline, we first rewrite a part of the proof of the OGA, in a way that does not require concavity, but the proof of the regret bound for the concave case can be recovered using the inequality $f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \nabla f_t(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$.

Theorem 1. *For any differentiable function class \mathbf{F} , with stochastic first-order query oracles bounded by G and step-sizes $\eta_t = \frac{D}{G\sqrt{t}}$, the algorithm OGA guarantees that $\max_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T \mathbb{E}[\langle \nabla f_t(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle] \leq \frac{3}{2}GD\sqrt{T}$.*

The proof of Theorem 1 is a simple rewriting of the classical results. We include a proof in Appendix D for completeness. Note that the boundedness of the stochastic first-order query oracle by G implies that the gradients of functions in \mathbf{F} are bounded by G , i.e., functions in \mathbf{F} are G -Lipschitz continuous.

In Step 2 of the guideline, we adapt the result of Theorem 1 to the upper-linearizable setting.

Theorem 2. *If \mathbf{F} is a function class over \mathcal{K} that is upper-linearizable with $0 < \alpha \leq 1$ and $\beta \geq 0$ and a first-order uniform wrapper \mathcal{W} , then with stochastic first-order query oracles bounded by G and step-sizes $\eta_t = \frac{D}{G\sqrt{t}}$, the algorithm $\mathcal{W}(\text{OGA})$ guarantees that $\max_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T \mathbb{E}[f_t(\mathbf{x}) - f_t(\mathcal{W}^{\text{action}}(\mathbf{x}_t))] \leq \frac{3}{2}GD\sqrt{T}$.*

We refer to Appendix D for a proof.

7.2 Zeroth Order Follow The Regularized Leader

Follow The Regularized Leader is a popular online optimization algorithm. When applied to a sequence of vectors $\{\mathbf{g}_t\}_{t=1}^T$ in \mathcal{K} , FTRL outputs a sequence of points $\{\mathbf{x}_t\}_{t=1}^T$, where

$$\mathbf{x}_1 = \underset{\mathbf{x} \in \mathcal{K}}{\text{argmin}} \Phi(\mathbf{x}), \quad \mathbf{x}_{t+1} = \underset{\mathbf{x} \in \mathcal{K}}{\text{argmin}} \eta \sum_{s=1}^t \langle -\mathbf{g}_s, \mathbf{x} \rangle + \Phi(\mathbf{x}). \quad (3)$$

Here $\Phi(\mathbf{x})$ is an arbitrary regularizer and η is a parameter. In this paper, we use a self-concordant barrier of \mathcal{K} as the regularizer of FTRL. Self-concordant barriers were first proposed in the convex optimization literature, with [Abernethy et al., 2008] the first use in bandit feedback setting. We refer to Appendix H for an overview of the main ideas present in FTRL, including the definition of self-concordant barrier Φ , the Minkowski set $\mathcal{K}_{\gamma, \mathbf{x}_1}$, and Σ -smoothing of function f to obtain f^{Σ} .

Here we propose a FTRL variant for zeroth-order feedback, based on [Saha and Tewari, 2011], which will be a key base algorithm for our framework. See Algorithm 5 for pseudo-code.

The following theorems demonstrate how to apply the guideline described in the beginning of Section 6 to the results of [Saha and Tewari, 2011]. The first step is to analyze the proof and modify the base algorithm so that we could obtain a result that is valid for non-convex functions and would only require a single use of an inequality similar to $f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$ to obtain a regret bound for concave case. By a small modification in the original algorithm, we get ZO-FTRL which

Algorithm 4: Application of $\mathcal{W}_1^{\text{NM}}$ to Online Gradient Ascent ($\mathcal{W}_1^{\text{NM}}(\text{OGA})$)

Input : horizon T , convex set \mathcal{K} , $\mathbf{x}_1, \underline{\mathbf{x}} \in \mathcal{K}$, step-sizes $\{\eta_t\}_{t=1}^T$
for $t = 1, 2, \dots, T$ **do**
 Play $(\mathbf{x}_t + \underline{\mathbf{x}})/2$
 Sample z according to \mathcal{Z}^{NM} (as defined in Appendix C.3)
 The adversary selects f_t and a query oracle \mathcal{Q}_t for f_t
 Observe \mathbf{o}_t , a sample of $\mathcal{Q}_t(\frac{z}{2}(\mathbf{x}_t - \underline{\mathbf{x}}) + \underline{\mathbf{x}})$, an unbiased estimate of $\nabla f_t(\frac{z}{2}(\mathbf{x}_t - \underline{\mathbf{x}}) + \underline{\mathbf{x}})$
 $\mathbf{x}_{t+1} \leftarrow P_{\mathcal{K}}(\mathbf{x}_t + \eta_t \mathbf{o}_t)$
end

Algorithm 5: Zeroth Order Follow The Regularized Leader - ZO-FTRL

Input : Horizon T , smoothing radius δ , learning rate η , ν -self-concordant barrier Φ
 $\mathbf{x}_1 \leftarrow \underset{\mathbf{x} \in \mathcal{K}}{\text{argmin}} \Phi(\mathbf{x})$
for $t = 1, 2, \dots, T$ **do**
 Play \mathbf{x}_t
 The adversary selects f_t and reveals a zeroth-order query oracle \mathcal{Q}_t for f_t
 $\Sigma_t \leftarrow (\nabla^2 \Phi(\mathbf{x}_t))^{-1/2}$
 Draw \mathbf{v}_t uniformly from \mathbb{S}^{d-1}
 $y_t \leftarrow \text{a sample of } \mathcal{Q}_t \text{ at } \mathbf{x}_t + \delta \Sigma_t \mathbf{v}_t$
 $\mathbf{o}_t \leftarrow \frac{d}{\delta} y_t \Sigma_t^{-1} \mathbf{v}_t$
 $\mathbf{x}_{t+1} \leftarrow \underset{\mathbf{x} \in \mathcal{K}}{\text{argmin}} \sum_{s=1}^t -\eta \langle \mathbf{o}_s, \mathbf{x} \rangle + \Phi(\mathbf{x})$
end

differs from the original in that it is no longer a bandit algorithm. While the agent plays \mathbf{x}_t it queries the oracle at $\mathbf{x}_t + \delta \Sigma_t \mathbf{v}_t \neq \mathbf{x}_t$. This modification allows us to obtain the following result.

Theorem 3. *Let \mathbf{F} be an M_1 -Lipschitz M_2 -smooth function class that is bounded by M_0 and let $B_0 \geq M_0$. Also let $\mathcal{B} \in \text{Adv}_0^o(\mathbf{F}, B_0)$ be a realized adversary that returns f_1, \dots, f_T , let $\mathbf{u}_* \in \text{argmax}_{\mathbf{u} \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{u})$ and $\hat{\mathbf{u}}_* \in \text{argmin}_{\mathbf{x} \in \mathcal{K}_{\gamma, \mathbf{x}_1}} \|\mathbf{u}_* - \mathbf{x}\|$ where $\gamma = T^{-1}$. Then, when running Algorithm 5 against \mathcal{B} , we have*

$$\sum_{t=1}^T \mathbb{E} [f_t(\mathbf{u}_*) - f_t(\mathbf{x}_t)] - O(\delta^2 T) \leq \sum_{t=1}^T \mathbb{E} [f_t^{\delta \Sigma_t}(\hat{\mathbf{u}}_*) - f_t^{\delta \Sigma_t}(\mathbf{x}_t)],$$

and
$$\sum_{t=1}^T \mathbb{E} [\langle \nabla f_t^{\delta \Sigma_t}(\mathbf{x}_t), \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle] \leq O(\eta \delta^{-2} T + \eta^{-1} \log T).$$

See Appendix J for the proof. Note that if f is concave, then we use use Lemma 5 to see that the right hand side of the first inequality is bounded by the left hand side of the second inequality and obtain the regret bound for the concave case. See Appendix K for the proof.

Theorem 4. *Let \mathbf{F} be an M_1 -Lipschitz M_2 -smooth function class over \mathcal{K} that is upper-linearizable with $0 < \alpha \leq 1$, $\beta \geq 0$ and a zeroth-order uniform wrapper \mathcal{W} . Also assume that $\mathcal{W}^{\text{action}}$ is M'_1 -Lipschitz and M'_2 -smooth. If Adv is a zeroth order oblivious adversary over \mathbf{F} such that for for any $f \in \mathbf{F}$ and any query oracle \mathcal{Q}_f returned by Adv , $\mathcal{W}(\mathcal{Q}_f)$ is a stochastic zeroth order query oracle for $\mathcal{W}(f)$ that is bounded by B_0 , then*

$$\mathcal{R}_{\alpha, \text{Adv}}^{\mathcal{W}(\text{ZO-FTRL})} = O(\eta \delta^{-2} T + \eta^{-1} \log T + \delta^2 T),$$

In particular, by setting $\eta = T^{-2/3}$ and $\delta = T^{-1/6}$, we see that $\mathcal{R}_{\alpha, \text{Adv}}^{\mathcal{W}(\text{ZO-FTRL})} = \tilde{O}(T^{2/3})$.

We emphasize that Theorems 3 and 4 correspond to steps 1 and 2 in the guideline discussed in Section 6 respectively. Specifically, for step 1, we note that Theorem 3 is NOT for concave functions. Moreover, if we assume that the functions are concave, then we may use Theorem 3 and the the inequality

$$f_t^{\delta \Sigma_t}(\hat{\mathbf{u}}_*) - f_t^{\delta \Sigma_t}(\mathbf{x}_t) \leq \langle \nabla f_t^{\delta \Sigma_t}(\mathbf{x}_t), \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle$$

to immediately obtain a regret bound. For step 2, we note that Theorem 3 adapts the result of previous step to uniform wrappers to obtain a regret bound for linearizable functions.

8 Applications

We next discuss some specific online and offline non-convex/non-concave optimization problems for which we can use our new framework to derive improved regret and sample complexity bounds respectively by applying uniform wrappers proposed in Section C to the zeroth order feedback OCO base algorithm ZO-FTRL (Algorithm 5). We note that we can also apply our framework to other base algorithms to recover many existing results in the literature. (See Section E for more details).

We start with a definition. For $\mathbf{x} \in \mathcal{K}$ and $C > 0$, we say a zeroth order query oracle \mathcal{Q}_f is contained in a (\mathbf{x}, C) cone if we have $|\mathcal{Q}_f(\mathbf{z}) - f(\mathbf{x})| \leq C \|\mathbf{z} - \mathbf{x}\|$ for all $\mathbf{z} \in \mathcal{K}$. In other words, the randomness of the query oracle approaches to zero at least linearly as we approach the point \mathbf{x} . We use the notation $\text{Adv}_0^o(\mathbf{F}, \text{Cone}(\mathbf{x}, C))$ to denote the oblivious adversary over \mathbf{F} with query oracles that are contained within this cone. Note that $\mathcal{Q}_f \in \text{Adv}_0^o(\mathbf{F}, \text{Cone}(\mathbf{x}, C))$ is equivalent to $\mathcal{W}_0^{\text{NM}}(\mathcal{Q}_f)$ being bounded. See condition (iii) of Definition 5 for details. If \mathcal{Q}_f does not belong to a cone as described above, we can see that the term $(\frac{z}{2})^{-1}$ causes $\mathcal{W}_0^{\text{NM}}(\mathcal{Q}_f)$ to blow up. Similarly, in the special case when $\underline{x} = \mathbf{0}$ and $f(\mathbf{0}) = 0$, it is also equivalent to $\mathcal{W}_0^{\text{M0}}(\mathcal{Q}_f)$ being bounded.

We begin by showing $\tilde{O}(T^{2/3})$ α -regret bounds for online optimization problems for the three function classes discussed in Section C under zeroth order feedback. See Section L for the proof.

Theorem 5. *Let \mathbf{F}_0^{M} , \mathbf{F}_0^{M0} and \mathbf{F}_0^{NM} denote the function classes described in Lemmas 1, 2 and 3 respectively and let α^{M} , α^{M0} and α^{NM} be the values of α . If the function classes are M_1 -Lipschitz and M_2 -smooth, then for any $C > 0$ and $B_0 \geq M_0 = \max_{\mathbf{x} \in \mathcal{K}} f(\mathbf{x})$, the following are $\tilde{O}(T^{2/3})$:*

$$\mathcal{R}_{\alpha^{\text{M}}, \text{Adv}_0^o(\mathbf{F}_0^{\text{M}}, B_0)}^{\mathcal{W}_0^{\text{M}}(\text{ZO-FTRL})}, \quad \mathcal{R}_{\alpha^{\text{M0}}, \text{Adv}_0^o(\mathbf{F}_0^{\text{M0}}, \text{Cone}(\mathbf{0}, C))}^{\mathcal{W}_0^{\text{M0}}(\text{ZO-FTRL})}, \quad \mathcal{R}_{\alpha^{\text{NM}}, \text{Adv}_0^o(\mathbf{F}_0^{\text{NM}}, \text{Cone}(\underline{x}, C))}^{\mathcal{W}_0^{\text{NM}}(\text{ZO-FTRL})}.$$

Remark 1. For each function class, the SOTA for noisy zeroth order feedback achieved $\tilde{O}(T^{3/4})$ α -regret bounds while we achieve $\tilde{O}(T^{2/3})$. For the special case of exact zeroth order feedback, the SOTA is $\tilde{O}(\sqrt{T})$. All the SOTA algorithms mentioned are special cases of our framework.

We next show $\tilde{O}(T^{3/4})$ α -regret bounds for online optimization problems for the three function classes discussed in Section C under bandit feedback. For full information zeroth order algorithms, the query location may differ from the action taken. Here we convert them into bandit algorithms using the meta-algorithm Stochastic Full-information To Trivial query (SFTT) from [Pedramfar and Aggarwal, 2024a] (see Section M for details). The proof is in Section N.

Theorem 6. *Under the assumptions of Theorem 5, the following are $\tilde{O}(T^{3/4})$:*

$$\mathcal{R}_{\alpha^M, \text{Adv}_0^{\alpha}(\mathbf{F}_0^M, B_0)}^{\text{SFTT}(\mathcal{W}_0^M(\text{ZO-FTRL}))}, \quad \mathcal{R}_{\alpha^{M0}, \text{Adv}_0^{\alpha}(\mathbf{F}_0^{M0}, \text{Cone}(\mathbf{0}, C))}^{\text{SFTT}(\mathcal{W}_0^{M0}(\text{ZO-FTRL}))}, \quad \mathcal{R}_{\alpha^{NM}, \text{Adv}_0^{\alpha}(\mathbf{F}_0^{NM}, \text{Cone}(\underline{\mathbf{x}}, C))}^{\text{SFTT}(\mathcal{W}_0^{NM}(\text{ZO-FTRL}))},$$

where SFTT is Algorithm 4 in [Pedramfar and Aggarwal, 2024a] with $L = T^{1/4}$.

Remark 2. Note that Algorithm 3 in [Wan et al., 2023] is in fact SFTT($\mathcal{W}_0^{M0}(\text{ZO-FTRL})$). However, our analysis simplifies the proof and generalizes the result to allow for stochastic feedback.

Remark 3. For the class \mathbf{F}^{NM} of non-monotone up-concave functions over general convex sets, our $\tilde{O}(T^{3/4})$ bound beats the SOTA $\tilde{O}(T^{4/5})$ bounds for exact and for noisy bandit feedback. For the class \mathbf{F}^{M0} of monotone γ -weakly up-concave functions over convex sets containing the origin, our $\tilde{O}(T^{3/4})$ bound beats the SOTA $\tilde{O}(T^{4/5})$ bound for noisy bandit feedback and matches the bound for exact bandit feedback. For the third class \mathbf{F}^M of monotone μ -strongly γ -weakly up-concave functions with bounded curvature, our results match the SOTA. All of the SOTA algorithms mentioned here are special cases of our framework.

Conversions of online algorithms to offline are referred to online-to-batch techniques and are well-known in the literature (See [Shalev-Shwartz, 2012]). A simple approach is to simply run the online algorithm and if the actions chosen by the algorithm are $\mathbf{x}_1, \dots, \mathbf{x}_T$, return \mathbf{x}_t for $1 \leq t \leq T$ with probability $1/T$. We use OTB to denote the meta-algorithm that uses this approach to convert online algorithms to offline algorithms.

We next show that using OTB conversion (on top of $\mathcal{W}(\text{ZO-FTRL})$), we obtain $\tilde{O}(1/\epsilon^3)$ sample complexity for finding an α -approximate solution in each function class under a noisy value oracle model, beating the SOTA $\tilde{O}(1/\epsilon^4)$ sample complexity. The proof is in Section O

Theorem 7. *Under the assumptions of Theorem 5, the following is true.*

- (i) *If the stochastic query oracle is bounded by B_0 , then the sample complexity of the offline algorithm OTB($\mathcal{W}_0^M(\text{ZO-FTRL})$) over \mathbf{F}_0^M is $\tilde{O}(\epsilon^{-3})$.*
- (ii) *If the stochastic query oracle is contained in the cone $\text{Cone}(\mathbf{0}, C)$, then the sample complexity of the offline algorithm OTB($\mathcal{W}_0^{M0}(\text{ZO-FTRL})$) over \mathbf{F}_0^{M0} is $\tilde{O}(\epsilon^{-3})$.*
- (iii) *If the stochastic query oracle is contained in the cone $\text{Cone}(\underline{\mathbf{x}}, C)$, then the sample complexity of the offline algorithm OTB($\mathcal{W}_0^{NM}(\text{ZO-FTRL})$) over \mathbf{F}_0^{NM} is $\tilde{O}(\epsilon^{-3})$.*

9 Conclusion

We proposed a class of meta-algorithms that can be used to convert convex optimization algorithms (with various types of feedback) into algorithms for quadratizable optimization. Using this framework, we improved regret guarantees for various classes of DR-submodular functions under zeroth-order feedback. Furthermore, we extended zeroth-order online algorithms to bandit feedback and offline counterparts, achieving notable improvements in regret/sample complexity compared to existing approaches.

Acknowledgments

This work was in part supported by IVADO, CIFAR, and the National Science Foundation under grants CCF-2149588 and CCF-2149617.

References

Jacob Abernethy, Elad E Hazan, and Alexander Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *21st Annual Conference on Learning Theory, COLT 2008*, pages 263–273, 2008.

An Bian, Kfir Levy, Andreas Krause, and Joachim M Buhmann. Continuous DR-submodular maximization: Structure and algorithms. In *Advances in Neural Information Processing Systems*, 2017a.

Andrew An Bian, Baharan Mirzasoleiman, Joachim Buhmann, and Andreas Krause. Guaranteed Non-convex Optimization: Submodular Maximization over Continuous Domains. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, April 2017b.

Yatao Bian, Joachim Buhmann, and Andreas Krause. Optimal continuous DR-submodular maximization and applications to provable mean field inference. In *Proceedings of the 36th International Conference on Machine Learning*, June 2019.

Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.

Lin Chen, Hamed Hassani, and Amin Karbasi. Online continuous submodular maximization. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, April 2018.

Lin Chen, Mingrui Zhang, and Amin Karbasi. Projection-free bandit convex optimization. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, pages 2047–2056. PMLR, 2019.

Shengminjie Chen, Donglei Du, Wenguo Yang, Dachuan Xu, and Suixiang Gao. Continuous non-monotone DR-submodular maximization with down-closed convex constraint. *arXiv preprint arXiv:2307.09616*, July 2023.

Josip Djolonga and Andreas Krause. From map to marginals: Variational inference in Bayesian submodular models. *Advances in Neural Information Processing Systems*, 2014.

Maryam Fazel and Omid Sadeghi. Fast first-order methods for monotone strongly dr-submodular maximization. In *SIAM Conference on Applied and Computational Discrete Algorithms (ACDA23)*, 2023.

Yuval Filmus and Justin Ward. A tight combinatorial algorithm for submodular maximization subject to a matroid constraint. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 659–668, 2012.

Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 385–394, 2005.

Dan Garber and Ben Kretzu. New projection-free algorithms for online convex optimization with adaptive regret guarantees. In *Proceedings of Thirty Fifth Conference on Learning Theory*, pages 2326–2359. PMLR, 2022.

Shuyang Gu, Chuangen Gao, Jun Huang, and Weili Wu. Profit maximization in social networks and non-monotone DR-submodular maximization. *Theoretical Computer Science*, 957:113847, 2023.

Hamed Hassani, Mahdi Soltanolkotabi, and Amin Karbasi. Gradient methods for submodular maximization. In *Advances in Neural Information Processing Systems*, 2017.

Hamed Hassani, Amin Karbasi, Aryan Mokhtari, and Zebang Shen. Stochastic conditional gradient++: (non)convex minimization and continuous submodular maximization. *SIAM Journal on Optimization*, 30(4):3315–3344, 2020.

Elad Hazan and C. Seshadhri. Efficient learning algorithms for changing environments. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 393–400. Association for Computing Machinery, 2009.

Elad Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Shinji Ito and Ryohei Fujimaki. Large-scale price optimization via network flow. *Advances in Neural Information Processing Systems*, 2016.

M. Kirschbraun. Über die zusammenziehende und lipschitzsche transformationen. *Fundamenta Mathematicae*, 22:77–108, 1934.

Duksang Lee, Nam Ho-Nguyen, and Dabeen Lee. Non-smooth, hölder-smooth, and robust submodular maximization. *arXiv preprint arXiv:2210.06061*, 2023.

Yuanyuan Li, Yuezhou Liu, Lili Su, Edmund Yeh, and Stratis Ioannidis. Experimental design networks: A paradigm for serving heterogeneous learners under networking constraints. *IEEE/ACM Transactions on Networking*, 2023.

Yucheng Liao, Yuanyu Wan, Chang Yao, and Mingli Song. Improved Projection-free Online Continuous Submodular Maximization. *arXiv preprint arXiv:2305.18442*, May 2023.

Siddharth Mitra, Moran Feldman, and Amin Karbasi. Submodular+ concave. *Advances in Neural Information Processing Systems*, 2021.

Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. Stochastic conditional gradient methods: From convex minimization to submodular maximization. *The Journal of Machine Learning Research*, 21(1):4232–4280, 2020.

Loay Mualem and Moran Feldman. Resolving the approximability of offline and online non-monotone DR-submodular maximization over general convex sets. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, April 2023.

Yurii Nesterov and Arkadii Nemirovskii. *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.

Rad Niazadeh, Tim Roughgarden, and Joshua R Wang. Optimal algorithms for continuous non-monotone submodular and DR-submodular maximization. *The Journal of Machine Learning Research*, 21(1):4937–4967, 2020.

Rad Niazadeh, Negin Golrezaei, Joshua Wang, Francisca Susan, and Ashwinkumar Badanidiyuru. Online learning via offline greedy algorithms: Applications in market design and optimization. *Management Science*, 69(7):3797–3817, July 2023.

Mohammad Pedramfar and Vaneet Aggarwal. From linear to linearizable optimization: A novel framework with applications to stationary and non-stationary DR-submodular optimization. *Advances in Neural Information Processing Systems*, 37:37626–37664, 2024a.

Mohammad Pedramfar and Vaneet Aggarwal. A generalized approach to online convex optimization. *arXiv preprint arXiv:2402.08621*, 2024b.

Mohammad Pedramfar, Christopher John Quinn, and Vaneet Aggarwal. A unified approach for maximizing continuous DR-submodular functions. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Mohammad Pedramfar, Yididiya Y. Nadew, Christopher John Quinn, and Vaneet Aggarwal. Unified projection-free algorithms for adversarial DR-submodular optimization. In *The Twelfth International Conference on Learning Representations*, 2024a.

Mohammad Pedramfar, Christopher Quinn, and Vaneet Aggarwal. A unified approach for maximizing continuous γ -weakly DR-submodular functions. *optimization-online preprint optimization-online:25915*, 2024b.

Ankan Saha and Ambuj Tewari. Improved regret guarantees for online smooth convex optimization with bandit feedback. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 636–642. JMLR Workshop and Conference Proceedings, 2011.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

Zongqi Wan, Jialin Zhang, Wei Chen, Xiaoming Sun, and Zhijie Zhang. Bandit multi-linear DR-submodular maximization and its applications on adversarial submodular bandits. In *International Conference on Machine Learning*, 2023.

Hassler Whitney. Analytic extensions of differentiable functions defined in closed sets. *Transactions of the American Mathematical Society*, 36(1):63–89, 1934.

Bryan Wilder. Equilibrium computation and robust optimization in zero sum games with submodular structure. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Lin Yang, Lei Deng, Mohammad H Hajiesmaili, Cheng Tan, and Wing Shing Wong. An optimal algorithm for online non-convex learning. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 2(2):1–25, 2018.

Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Mingrui Zhang, Lin Chen, Hamed Hassani, and Amin Karbasi. Online continuous submodular maximization: From full-information to bandit feedback. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Qixin Zhang, Zengde Deng, Zaiyi Chen, Haoyuan Hu, and Yu Yang. Stochastic continuous submodular maximization: Boosting via non-oblivious function. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

Qixin Zhang, Zengde Deng, Zaiyi Chen, Kuangqi Zhou, Haoyuan Hu, and Yu Yang. Online learning for non-monotone DR-submodular maximization: From full information to bandit feedback. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, April 2023.

Qixin Zhang, Zongqi Wan, Zengde Deng, Zaiyi Chen, Xiaoming Sun, Jialin Zhang, and Yu Yang. Boosting gradient ascent for continuous DR-submodular maximization. *arXiv preprint arXiv:2401.08330*, 2024.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936, 2003.

Table of Contents

1	Introduction	1
2	Background and Notation	3
3	Problem setup	4
4	Uniform wrappers	5
5	Linearizable and quadratizable functions classes	6
6	When Is Concave Optimization Enough?	6
7	Examples	7
7.1	Online Gradient Ascent	7
7.2	Zeroth Order Follow The Regularized Leader	8
8	Applications	9
9	Conclusion	10
Appendices		13
A	How uniform wrappers work	15
B	Additional Related Works	15
C	Uniform wrappers for up-concave optimization	17
D	Proof of Theorems 1 and 2	19
E	Recovering previous results in the literature	20
F	Proof of Theorem 11	21
G	Proof of Theorem 12	23
H	Follow The Regularized Leader	23
I	Technical Lemmas	24
J	Proof of Theorem 3	26
K	Proof of Theorem 4	27
L	Proof of Theorem 5	29
M	Stochastic Full-information To Trivial query - SFTT	29
N	Proof of Theorem 6	30
O	Proof of Theorem 7	30

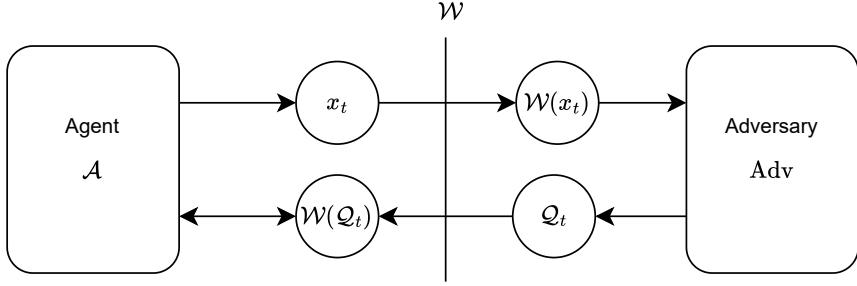


Figure 1: Effect of a uniform wrapper on the interaction between the agent and the adversary.

A How uniform wrappers work

Figure A demonstrates the effect of a uniform wrapper on the interaction of the agent and the adversary. The vertical line depicts the uniform wrapper, acting as a translation layer, where any information passing through is changed accordingly. The action of the agent is changed from \mathbf{x}_t to $\mathcal{W}^{\text{action}}(\mathbf{x}_t)$ and the query oracle selected by the adversary is changed from \mathcal{Q}_t to $\mathcal{W}^{\text{query}}(\mathcal{Q}_t)$.

Remark 4. Note that neither Figure A, nor Meta-Algorithm 1 require $\mathcal{W}^{\text{function}}$. Thus, we define an *abstract uniform wrapper* as a tuple of maps $\mathcal{W} = (\mathcal{W}^{\text{action}}, \mathcal{W}^{\text{query}})$ where $\mathcal{W}^{\text{action}}$ is defined as before. On the other hand, $\mathcal{W}^{\text{query}}$ is defined as a function that, for any query oracle $\mathcal{Q}_f \in \mathcal{Q}$, the value of $\mathcal{W}(\mathcal{Q}_f)$ behaves similar to a query oracle, but is not necessarily one. Specifically, if $i, j \geq 0$ and \mathcal{W} is an i -order to j -order abstract uniform wrapper, then for any i -th order query oracle \mathcal{Q}_f , $\mathcal{W}(\mathcal{Q}_f)$ is just function from \mathcal{K} to \mathbb{R}^{d^j} . In particular, there may not exist a function $F : \mathcal{K} \rightarrow \mathbb{R}$ such that $D^j F(\mathbf{x}) = \mathbb{E}[\mathcal{W}(\mathcal{Q}_f)(\mathbf{x})]$, where D^j denotes the j -th order derivative. The generality of abstract uniform wrappers allow for considering quadratizable function classes where \mathbf{g} , as described in Definition 2, might not satisfy $\mathbf{g}(f, \mathbf{x}) = \nabla F(\mathbf{x})$ for any function $F : \mathcal{K} \rightarrow \mathbb{R}$. Note that, unlike uniform wrappers, we may no longer be able to define $\mathcal{W}(\text{Adv})$ for abstract uniform wrappers.

The figure may be interpreted in two equivalent ways. If we think of \mathcal{W} as a meta-algorithm that wraps around the agent, then we get $\mathcal{W}(\mathcal{A})$, as described in Meta-Algorithm 1, playing against the adversary Adv . Alternatively, we may think of \mathcal{W} as wrapping around the adversary. In this interpretation, the agent \mathcal{A} is playing against $\mathcal{W}(\text{Adv})$. Since these two interpretations are equivalent, we may use one to better understand the other. Specifically, we may modify the proof of the regret bounds of the game between \mathcal{A} and a concave adversary to bound the regret of \mathcal{A} against $\mathcal{W}(\text{Adv})$ and therefore use that to obtain regret bounds for $\mathcal{W}(\mathcal{A})$ against Adv . The guideline in Section 6 describes how this might be achieved.

B Additional Related Works

DR-submodular maximization Two of the main methods for continuous DR-submodular maximization are *Frank-Wolfe type methods* and *Boosting based methods*. This division is based on how the approximation coefficient appears in the proof. To the best of knowledge, the only other algorithms in the literature are those that are designed for specific choices of domain and use techniques such as discretization and rely heavily on the detailed structures on the constraint set.

In Frank-Wolfe type algorithms, the approximation coefficient appears by specific choices of the Frank-Wolfe update rules. (See Lemma 8 in [Pedramfar et al., 2024a]) The specific choices of the update rules for different settings have been proposed in [Bian et al., 2017b,a, Mualem and Feldman, 2023, Pedramfar et al., 2023, Chen et al., 2023]. The momentum technique of [Mokhtari et al., 2020] has been used to convert algorithms designed for deterministic feedback to stochastic feedback setting. [Hassani et al., 2020] proposed a Frank-Wolfe variant with access to a stochastic gradient oracle with *known distribution*. Frank-Wolfe type algorithms been adapted to the online setting using Meta-Frank-Wolfe [Chen et al., 2018, 2019] or using Blackwell approachability [Niazadeh et al., 2023]. Later [Zhang et al., 2019] used a Meta-Frank-Wolfe with random permutation technique to obtain full-information results that only require a single query per function and also bandit results.

Table 2: Offline up-concave maximization

F	Set	Feedback		Reference	Appx.	Complexity
Monotone	0 ∈ \mathcal{K}	∇F	stoch.	[Mokhtari et al., 2020]	$1 - e^{-\gamma}$	$O(1/\epsilon^3)$
				[Hassani et al., 2020] (*)	$1 - e^{-\gamma}$	$O(1/\epsilon^2)$
		F	det.	[Zhang et al., 2022] (*)	$1 - e^{-\gamma}$	$O(1/\epsilon^2)$
			stoch. †	[Pedramfar and Aggarwal, 2024a] (*)	$1 - e^{-\gamma}$	$O(1/\epsilon^2)$
			stoch.	Theorem 7	$1 - e^{-\gamma}$	$\tilde{O}(1/\epsilon^3)$
	general	∇F	stoch.	[Pedramfar et al., 2024b]	$1 - e^{-\gamma}$	$O(1/\epsilon^3)$
				[Pedramfar and Aggarwal, 2024a] (*)	$1 - e^{-\gamma}$	$O(1/\epsilon^2)$
				[Pedramfar and Aggarwal, 2024a] (*)	$1 - e^{-\gamma}$	$O(1/\epsilon^5)$
		F	det.	[Hassani et al., 2017] (*)	$\gamma^2/(1 + \gamma^2)$	$O(1/\epsilon^2)$
			stoch.	[Pedramfar et al., 2024b]	$\gamma^2/(1 + \gamma^2)$	$\tilde{O}(1/\epsilon^3)$
Non-Monotone	general	∇F	stoch.	[Pedramfar and Aggarwal, 2024a] (*)	$\gamma^2/(1 + c\gamma^2)$	$O(1/\epsilon^2)$
				[Pedramfar et al., 2023]	$\gamma^2/(1 + \gamma^2)$	$O(1/\epsilon^3)$
				[Pedramfar and Aggarwal, 2024a] (*)	$\gamma^2/(1 + c\gamma^2)$	$O(1/\epsilon^2)$
		F	stoch.	[Pedramfar et al., 2023]	$\gamma^2/(1 + \gamma^2)$	$O(1/\epsilon^5)$
				[Pedramfar and Aggarwal, 2024a] (*)	$\gamma^2/(1 + c\gamma^2)$	$O(1/\epsilon^4)$
			stoch.	Theorem 7	$\gamma^2/(1 + c\gamma^2)$	$\tilde{O}(1/\epsilon^3)$
			stoch.	[Pedramfar et al., 2024b]	$\frac{\gamma(1-\gamma h)}{\gamma'-1} \left(\frac{1}{2} - \frac{1}{2^{\gamma'}} \right)$	$O(1/\epsilon^3)$
			stoch.	[Zhang et al., 2024] (*)	$(1-h)/4$	$O(1/\epsilon^2)$
			stoch.	[Pedramfar and Aggarwal, 2024a] (*)	$(1-h)/4$	$O(1/\epsilon^2)$
			det.	[Pedramfar et al., 2024b]	$\frac{\gamma(1-\gamma h)}{\gamma'-1} \left(\frac{1}{2} - \frac{1}{2^{\gamma'}} \right)$	$O(1/\epsilon^3)$
			det.	[Pedramfar and Aggarwal, 2024a] (*)	$(1-h)/4$	$O(1/\epsilon^2)$
			stoch. †	Theorem 7	$(1-h)/4$	$\tilde{O}(1/\epsilon^3)$
			stoch.	[Pedramfar et al., 2024b]	$\frac{\gamma(1-\gamma h)}{\gamma'-1} \left(\frac{1}{2} - \frac{1}{2^{\gamma'}} \right)$	$O(1/\epsilon^5)$
			stoch.	[Pedramfar and Aggarwal, 2024a] (*)	$(1-h)/4$	$O(1/\epsilon^4)$

This table compares the different results for the number of oracle calls (complexity) *within the constraint set* for up-concave maximization. We refer to [Pedramfar et al., 2024b] for a more comprehensive table that includes results for deterministic first order feedback. Here $h := \min_{\mathbf{z} \in \mathcal{K}} \|\mathbf{z}\|_\infty$ and $\gamma' := \gamma + 1/\gamma$. **Rows marked with (*) are results in the literature that fit within the framework described in this paper.** The rows describing results with stochastic feedback that are marked with † assume that the random query oracle is contained with a cone, as detailed in Theorem 5.

This was extended to another settings by [Zhang et al., 2023] and generalized to many different settings with improved regret bounds by [Pedramfar et al., 2024a].

Some techniques construct an alternative function such that maximization of this function results in approximate maximization of the original function. Given this definition, we may consider the result of [Hassani et al., 2017, Chen et al., 2018, Fazel and Sadeghi, 2023] as the first boosting based results. However, in the case of monotone DR-submodular functions over general convex sets, the alternative function is identical to the original function. The term boosting in this context was first used in [Zhang et al., 2022], based on ideas presented in [Filimur and Ward, 2012, Mitra et al., 2021], for monotone functions over convex sets containing the origin. This idea has been used later in [Wan et al., 2023, Liao et al., 2023] in bandit and projection-free full-information settings. Finally, in [Zhang et al., 2024] a boosting based method was introduced for non-monotone functions over general convex sets.

Up-concave maximization Not all continuous DR-submodular functions are concave and not all concave functions are continuous DR-submodular. [Mitra et al., 2021] considers functions that are the sum of a concave and a continuous DR-submodular function. It is well-known that continuous DR-submodular functions are concave along positive directions [Calinescu et al., 2011, Bian et al., 2017b]. Based on this idea, [Wilder, 2018] defined an up-concave function as a function that is concave along positive directions. Up-concave maximization has been considered in the offline

setting before, e.g. in [Lee et al., 2023] and [Pedramfar and Aggarwal, 2024a]. In this work, we focus on up-concave maximization which is a generalization of DR-submodular maximization.

C Uniform wrappers for up-concave optimization

In this section, we study three classes of up-concave functions and show that they are upper-quadratizable with appropriate uniform wrappers.

Remark 5. As we will see in the following, all of these uniform wrappers fit into the following pattern. There is a random variable \mathcal{Z} taking values in $[0, 1]$ and a parametrized family of functions $h_z : \mathcal{K} \rightarrow \mathcal{K}$ for $z \in [0, 1]$, such that, for some $\alpha \in (0, 1]$ and $\beta, \mu \geq 0$, we have

$$\alpha f(\mathbf{y}) - f(h_1(\mathbf{x})) \leq \beta \left(\langle \nabla F(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right),$$

where $F(\mathbf{x}) = \mathbb{E}_{z \sim \mathcal{Z}}[f(h_z(\mathbf{x})) - f(h_0(\mathbf{x}))]$. If such h_z and \mathcal{Z} exist, we define $\mathcal{W}^{\text{action}} = h_1$, and $\mathcal{W}^{\text{function}}(f) = F$. We also need to define $\mathcal{W}^{\text{query}}(Q_f)$ such that it converts a query oracle for f into a query oracle for F using the following equality.

$$\nabla F(\mathbf{x}) = \mathbb{E}_{z \sim \mathcal{Z}}[\nabla f(h_z(\mathbf{x})) J_{h_z}(\mathbf{x}) - \nabla f(h_0(\mathbf{x})) J_{h_0}(\mathbf{x})].$$

Note that the identity wrapper, where $F = f$, can be recovered by choosing $h_z(x) := z * \mathbf{x}$ and $\mathcal{Z} = \delta_1$, the Dirac delta distribution with mass at 1.

C.1 Monotone μ -strongly γ -weakly up-concave functions with bounded curvature (\mathbf{F}^M)

For differentiable DR-submodular functions, the following lemma is proved for the case $\gamma = 1$ in [Fazel and Sadeghi, 2023] and for the case $\mu = 0$ in [Hassani et al., 2017]. The general form we use here is proved in Lemma 1 in [Pedramfar and Aggarwal, 2024a].

Lemma 1. *Let $f : [0, 1]^d \rightarrow \mathbb{R}$ be a non-negative monotone μ -strongly γ -weakly up-concave function with curvature bounded by c . Then, for all $\mathbf{x}, \mathbf{y} \in [0, 1]^d$, we have*

$$\frac{\gamma^2}{1 + c\gamma^2} f(\mathbf{y}) - f(\mathbf{x}) \leq \frac{\gamma}{1 + c\gamma^2} \left(\langle \tilde{\nabla} f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right),$$

where $\tilde{\nabla} f$ is an up-super-gradient for f .

Lemma 1, together with Definition 1 of uniform wrappers, immediately imply the following.

Theorem 8. *Let \mathbf{F}^M be the class of functions over \mathcal{K} where every $f \in \mathbf{F}^M$ may be extended to a non-negative differentiable monotone μ -strongly γ -weakly up-concave function with curvature bounded by c defined over $[0, 1]^d$. Then \mathbf{F}^M is upper-quadratizable with uniform wrapper $\mathcal{W}^M = \text{Id}$.*

If \mathcal{A} is one of the algorithm described in Theorem 12, using the above theorem recovers Theorem 2 in [Pedramfar and Aggarwal, 2024a] which itself is a generalization of Theorem 2 in [Chen et al., 2018] and Theorem 3 in [Fazel and Sadeghi, 2023].

C.2 Monotone γ -weakly up-concave functions over convex sets containing the origin (\mathbf{F}^{M0})

For differentiable monotone DR-submodular functions, the following lemma is proved in [Zhang et al., 2022]. The general form here is proved in Lemma 2 in [Pedramfar and Aggarwal, 2024a].

Lemma 2. *Let $f : [0, 1]^d \rightarrow \mathbb{R}$ be a non-negative monotone γ -weakly up-concave differentiable function and let $F : [0, 1]^d \rightarrow \mathbb{R}$ be the function defined by $F(\mathbf{x}) := \int_0^1 \frac{\gamma e^{\gamma(z-1)}}{(1-e^{-\gamma})z} (f(z * \mathbf{x}) - f(\mathbf{0})) dz$. Then F is differentiable and $(1 - e^{-\gamma})f(\mathbf{y}) - f(\mathbf{x}) \leq \frac{1-e^{-\gamma}}{\gamma} \langle \nabla F(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$.*

Let the random variable $\mathcal{Z}^{M0} \in [0, 1]$ be defined by the law $\forall z \in [0, 1], \mathbb{P}(\mathcal{Z}^{M0} \leq z) = \int_0^z \frac{\gamma e^{\gamma(u-1)}}{1-e^{-\gamma}} du$. Then we have $\mathbb{E}_{z \sim \mathcal{Z}^{M0}}[z^{-1}(f(z * \mathbf{x}) - f(\mathbf{0}))] = F(\mathbf{x})$. Moreover, for $i \geq 1$, if f is i times differentiable then we also have $\mathbb{E}_{z \sim \mathcal{Z}^{M0}}[z^{i-1} \nabla^i f(z * \mathbf{x})] = \nabla^i F(\mathbf{x})$.

Definition 4. Let $\mathcal{K} \subseteq [0, 1]^d$ be a convex set containing the origin and, for any $i \geq 0$, let \mathbf{F}_i^{MO} be the class of functions over \mathcal{K} that are $\max\{i, 1\}$ times differentiable and where every $f \in \mathbf{F}_i^{\text{MO}}$ may be extended to a non-negative monotone γ -weakly up-concave function defined over $[0, 1]^d$. We also assume that $f(\mathbf{0}) = 0$ for all $f \in \mathbf{F}_0^{\text{MO}}$. We define $\mathcal{W}_i^{\text{MO}} := ((\mathcal{W}_i^{\text{MO}})^{\text{action}}, (\mathcal{W}_i^{\text{MO}})^{\text{function}}, (\mathcal{W}_i^{\text{MO}})^{\text{query}})$ to be the uniform wrapper with (i) $(\mathcal{W}_i^{\text{MO}})^{\text{action}} := \text{Id}_{\mathcal{K}}$; (ii) for any $f \in \mathbf{F}_i^{\text{MO}}$, $(\mathcal{W}_i^{\text{MO}})^{\text{function}}(f) := \mathbf{x} \mapsto \mathbb{E}_{z \sim \mathcal{Z}^{\text{MO}}} [z^{-1}(f(z * \mathbf{x}) - f(\mathbf{0}))] : \mathcal{K} \rightarrow \mathbb{R}$; and (iii) for any $f \in \mathbf{F}_i^{\text{MO}}$ and any i -th order query oracle \mathcal{Q}_f for f , we have $(\mathcal{W}_i^{\text{MO}})^{\text{query}}(\mathcal{Q}_f)(\mathbf{x}) := z^{i-1} * Q_f(z * \mathbf{x})$, where z is sampled according to $\mathbb{P}(\mathcal{Z}^{\text{MO}} \leq z)$.

Remark 6. This wrapper fits into the pattern described in Remark 5 by defining $h_z(\mathbf{x}) := z * \mathbf{x}$ and $\mathcal{Z} = \mathcal{Z}^{\text{MO}}$.

Theorem 9. For any $i \geq 0$, the function class \mathbf{F}_i^{MO} defined above is upper-linearizable with the uniform wrapper $\mathcal{W}_i^{\text{MO}}$.

Remark 7. The meta-algorithm $\mathcal{A} \mapsto \text{OMBQ}(\mathcal{A}, \text{BQM0}, \text{Id})$, described in [Pedramfar and Aggarwal, 2024a], is identical to $\mathcal{A} \mapsto \mathcal{W}_1^{\text{MO}}(\mathcal{A})$. In other words, the results of Theorem 3 in [Pedramfar and Aggarwal, 2024a] are about the first order uniform wrapper $\mathcal{W}_1^{\text{MO}}$. Here we consider a more general case where we are not necessarily limited to first order.

C.3 Non-monotone up-concave functions over general convex sets (\mathbf{F}^{NM})

For differentiable monotone DR-submodular functions, the following lemma is proved in [Zhang et al., 2024]. The general form we use is proven in Lemma 3 in [Pedramfar and Aggarwal, 2024a].

Lemma 3. Let $f : [0, 1]^d \rightarrow \mathbb{R}$ be a non-negative continuous up-concave differentiable function and let $\underline{\mathbf{x}} \in \mathcal{K}$. Define $F : [0, 1]^d \rightarrow \mathbb{R}$ as the function $F(\mathbf{x}) := \int_0^1 \frac{2}{3z(1-\frac{z}{2})^3} (f(\frac{z}{2} * (\mathbf{x} - \underline{\mathbf{x}}) + \underline{\mathbf{x}}) - f(\underline{\mathbf{x}})) dz$, then F is differentiable and we have

$$\frac{1 - \|\underline{\mathbf{x}}\|_\infty}{4} f(\mathbf{y}) - f\left(\frac{\mathbf{x} + \underline{\mathbf{x}}}{2}\right) \leq \frac{3}{8} \langle \nabla F(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle.$$

Let the random variable $\mathcal{Z}^{\text{NM}} \in [0, 1]$ be defined by the law $\forall z \in [0, 1], \mathbb{P}(\mathcal{Z}^{\text{NM}} \leq z) = \int_0^z \frac{1}{3(1-\frac{u}{2})^3} du$. Then we have $\mathbb{E}_{z \sim \mathcal{Z}^{\text{NM}}} [(\frac{z}{2})^{-1} * (f(\frac{z}{2} * (\mathbf{x} - \underline{\mathbf{x}}) + \underline{\mathbf{x}}) - f(\underline{\mathbf{x}}))] = F(\mathbf{x})$. Moreover, if $i \geq 1$ and f is i times differentiable, then $\mathbb{E}_{z \sim \mathcal{Z}^{\text{NM}}} [(\frac{z}{2})^{i-1} * \nabla^i f(\frac{z}{2} * (\mathbf{x} - \underline{\mathbf{x}}) + \underline{\mathbf{x}})] = \nabla^i F(\mathbf{x})$.

Definition 5. Let $\mathcal{K} \subseteq [0, 1]^d$ be a convex set and, for any $i \geq 0$, let \mathbf{F}_i^{NM} be the class of functions over \mathcal{K} where every $f \in \mathbf{F}_i^{\text{NM}}$ may be extended to a non-negative up-concave function defined over $[0, 1]^d$. We also assume that \mathbf{F}_i^{NM} is $\max\{i, 1\}$ times differentiable for all $i \geq 0$ and, for some known constant $c \geq 0$ and all $f \in \mathbf{F}_0^{\text{NM}}$, $f(\underline{\mathbf{x}}) = c$. For $i \geq 0$, we define $\mathcal{W}_i^{\text{NM}} = ((\mathcal{W}_i^{\text{NM}})^{\text{action}}, (\mathcal{W}_i^{\text{NM}})^{\text{function}}, (\mathcal{W}_i^{\text{NM}})^{\text{query}})$ to be the uniform wrapper with

(i) $(\mathcal{W}_i^{\text{NM}})^{\text{action}} := \mathbf{x} \mapsto \frac{\mathbf{x} + \underline{\mathbf{x}}}{2} : \mathcal{K} \rightarrow \mathcal{K}$; (ii) for any $f \in \mathbf{F}_i^{\text{NM}}$, $(\mathcal{W}_i^{\text{NM}})^{\text{function}}(f) := \mathbf{x} \mapsto \mathbb{E}_{z \sim \mathcal{Z}^{\text{NM}}} [(\frac{z}{2})^{-1} * (f(\frac{z}{2} * (\mathbf{x} - \underline{\mathbf{x}}) + \underline{\mathbf{x}}) - f(\underline{\mathbf{x}}))] : \mathcal{K} \rightarrow \mathbb{R}$; and (iii) for any $f \in \mathbf{F}_i^{\text{NM}}$ and any i -th order query oracle \mathcal{Q}_f for f ,

$$(\mathcal{W}_i^{\text{NM}})^{\text{query}}(\mathcal{Q}_f)(\mathbf{x}) := \begin{cases} (\frac{z}{2})^{i-1} * Q_f(\frac{z}{2} * (\mathbf{x} - \underline{\mathbf{x}}) + \underline{\mathbf{x}}) & \text{if } i \geq 1 \\ (\frac{z}{2})^{-1} * (Q_f(\frac{z}{2} * (\mathbf{x} - \underline{\mathbf{x}}) + \underline{\mathbf{x}}) - c) & \text{if } i = 0 \end{cases}$$

where z is sampled according to $\mathbb{P}(\mathcal{Z}^{\text{NM}} \leq z)$.

Remark 8. This wrapper fits into the pattern described in Remark 5 by defining $h_z(\mathbf{x}) := \frac{z}{2} * (\mathbf{x} - \underline{\mathbf{x}}) + \underline{\mathbf{x}}$ and $\mathcal{Z} = \mathcal{Z}^{\text{NM}}$.

Theorem 10. For any $i \geq 0$, the function class \mathbf{F}_i^{NM} defined above is upper-linearizable with the uniform wrapper $\mathcal{W}_i^{\text{NM}}$.

Remark 9. The meta-algorithm $\mathcal{A} \mapsto \text{OMBQ}(\mathcal{A}, \text{BQN}, \mathbf{x} \mapsto \frac{\mathbf{x} + \underline{\mathbf{x}}}{2})$, described in [Pedramfar and Aggarwal, 2024a], is identical to $\mathcal{A} \mapsto \mathcal{W}_1^{\text{NM}}(\mathcal{A})$. In other words, the results of Theorem 4 in [Pedramfar and Aggarwal, 2024a] are about the first order uniform wrapper $\mathcal{W}_1^{\text{NM}}$. Here we consider a more general case where we are not necessarily limited to first order.

D Proof of Theorems 1 and 2

The proof of Theorem 1 is a simple rewriting of the classical results. We start with a lemma which contains the main idea commonly used in the proof of the regret bound of Online Gradient Descent.

Lemma 4. *Let $\mathbf{x}, \mathbf{x}_1 \in \mathcal{K}$, let $G > 0$ and let $(\mathbf{g}_t)_{t=1}^T$ be a sequence of vectors such that $\|\mathbf{g}_t\| \leq G$. Define the sequence $(\mathbf{x}_t)_{t=1}^T$ such that $\mathbf{x}_{t+1} \leftarrow P_{\mathcal{K}}(\mathbf{x}_t - \eta_t \mathbf{g}_t)$, where $\eta_t = \frac{D}{G\sqrt{t}}$. Then we have*

$$\sum_{t=1}^T \langle \mathbf{g}_t, \mathbf{x}_t - \mathbf{x} \rangle \leq \frac{3}{2} GD\sqrt{T}.$$

The following proof is copied from the proof of Theorem 3.1 in [Hazan et al., 2016].

Proof. Using Pythagorean theorem, we have

$$\|\mathbf{x}_{t+1} - \mathbf{x}\|^2 = \|P_{\mathcal{K}}(\mathbf{x}_t - \eta_t \mathbf{g}_t) - \mathbf{x}\|^2 \leq \|\mathbf{x}_t - \eta_t \mathbf{g}_t - \mathbf{x}\|^2.$$

Hence,

$$\begin{aligned} \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 &\leq \|\mathbf{x}_t - \mathbf{x}\|^2 + \eta_t^2 \|\mathbf{g}_t\|^2 - 2\eta_t \langle \mathbf{g}_t, \mathbf{x}_t - \mathbf{x} \rangle. \\ 2\eta_t \langle \mathbf{g}_t, \mathbf{x}_t - \mathbf{x} \rangle &\leq \frac{1}{\eta_t} \left(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 \right) + \eta_t G^2. \end{aligned}$$

Summing this from $t = 1$ to T , and setting $\eta_t = \frac{D}{G\sqrt{t}}$, we see that

$$\begin{aligned} 2 \sum_{t=1}^T \eta_t \langle \mathbf{g}_t, \mathbf{x}_t - \mathbf{x} \rangle &\leq \sum_{t=1}^T \left[\frac{1}{\eta_t} \left(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 \right) + \eta_t G^2 \right] \\ &= \frac{1}{\eta_1} \|\mathbf{x}_1 - \mathbf{x}\|^2 + \sum_{t=2}^T \|\mathbf{x}_t - \mathbf{x}\|^2 \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} \right) + G^2 \sum_{t=2}^T \eta_t \\ &\leq D^2 \left[\frac{1}{\eta_1} + \sum_{t=2}^T \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} \right) \right] + G^2 \sum_{t=2}^T \eta_t \\ &= D^2 \frac{1}{\eta_T} + G^2 \sum_{t=2}^T \eta_t \\ &\leq 3DG\sqrt{T}, \end{aligned}$$

where the second inequality follows from the facts that $\text{diam}(\mathcal{K}) \leq D$ and η_t is decreasing and therefore $\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}$ is always positive; and the last inequality follows from the facts that $\eta_t = \frac{D}{G\sqrt{t}}$ and $\sum_{t=1}^T \frac{1}{\sqrt{t}} \leq 2\sqrt{T}$. \square

Using this lemma, we can easily prove Theorem 1.

Proof. By applying Lemma 4 to $\mathbf{o}_t = -\mathbf{g}_t$, (to account for the fact, unlike OGD, that we update according to $\mathbf{x}_{t+1} \leftarrow P_{\mathcal{K}}(\mathbf{x}_t + \eta_t \mathbf{o}_t)$) we see that

$$\sum_{t=1}^T \langle \mathbf{o}_t, \mathbf{x} - \mathbf{x}_t \rangle \leq \frac{3}{2} GD\sqrt{T}.$$

If we take the expectation, we see that

$$\begin{aligned} \sum_{t=1}^T \mathbb{E}[\langle \nabla f_t(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle] &= \sum_{t=1}^T \mathbb{E}[\langle \mathbb{E}[\mathbf{o}_t | \mathbf{x}_t], \mathbf{x} - \mathbf{x}_t \rangle] \\ &= \sum_{t=1}^T \mathbb{E}[\mathbb{E}[\langle \mathbf{o}_t, \mathbf{x} - \mathbf{x}_t \rangle | \mathbf{x}_t]] \\ &= \sum_{t=1}^T \mathbb{E}[\langle \mathbf{o}_t, \mathbf{x} - \mathbf{x}_t \rangle] \leq \frac{3}{2} GD\sqrt{T}. \end{aligned} \quad \square$$

Next we prove Theorem 2.

Proof. At first glance, it may seem that moving from Theorem 1 to Theorem 2 might be non-trivial. However, as elaborated in Section A, whether we apply $\mathcal{W}(\text{OGA})$ to an adversary Adv that selects functions from \mathbf{F} or apply OGA to the adversary $\mathcal{W}(\text{Adv})$, the sequence \mathbf{x}_t remains the same. Thus, by using Theorem 1, we immediately see that

$$\max_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T \mathbb{E}[\langle \nabla \mathcal{W}(f_t)(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle] \leq \frac{3}{2} GD\sqrt{T}.$$

Another way to arrive at this inequality is to follow the proof of Theorem 2 and adapt it here. Specifically, for all $\mathbf{x} \in \mathcal{K}$, we have

$$\begin{aligned} \sum_{t=1}^T \mathbb{E}[\langle \nabla \mathcal{W}(f_t)(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle] &= \sum_{t=1}^T \mathbb{E}[\langle \mathbb{E}[\mathbf{o}_t | \mathbf{x}_t], \mathbf{x} - \mathbf{x}_t \rangle] \\ &= \sum_{t=1}^T \mathbb{E}[\mathbb{E}[\langle \mathbf{o}_t, \mathbf{x} - \mathbf{x}_t \rangle | \mathbf{x}_t]] \\ &= \sum_{t=1}^T \mathbb{E}[\langle \mathbf{o}_t, \mathbf{x} - \mathbf{x}_t \rangle] \leq \frac{3}{2} GD\sqrt{T}, \end{aligned}$$

where the last inequality follows from Lemma 4. Now we may use the definition of uniform wrappers to complete the proof of Theorem 2. \square

E Recovering previous results in the literature

As mentioned in Remark 2, Algorithm 3 in [Wan et al., 2023] is in fact SFTT($\mathcal{W}_0^{\text{MO}}(\text{ZO-FTRL})$) and therefore their result fits within our framework. The way the remaining results in the tables that are marked with $(*)$ is discussed in the following.

We demonstrate how to apply the guideline described in the beginning of Section 6 to Theorem 2 in [Pedramfar and Aggarwal, 2024b]. This allows us to obtain a generalized version of Theorems 1 in [Pedramfar and Aggarwal, 2024a]. As we will discuss below, this will allow us to recover all the remaining results in Tables 1 and 2 that are marked with $(*)$ and all the results of [Pedramfar and Aggarwal, 2024a]. Note that the results of [Pedramfar and Aggarwal, 2024a] in non-stationary setting are not discussed in this paper, but they are also recovered.

We start with some definitions. Given a function class \mathbf{F} , we use the notation $\mathbf{F}_{\mu, \mathbf{g}}$ to denote the class of functions $q(\mathbf{y}) := \langle \mathbf{g}(f, \mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 : \mathcal{K} \rightarrow \mathbb{R}$, for all $f \in \mathbf{F}$ and $\mathbf{x} \in \mathcal{K}$. This is the class of quadratic (or linear, when $\mu = 0$) functions that form the upper bound in Equation 1. Similarly, for any $B_1 > 0$, we use the notation $\mathbf{Q}_\mu[B_1]$ to denote the class of functions $q(\mathbf{y}) := \langle \mathbf{o}, \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 : \mathcal{K} \rightarrow \mathbb{R}$, for all $\mathbf{x} \in \mathcal{K}$ and $\mathbf{o} \in \mathbb{B}_{B_1}(\mathbf{0})$. In the following theorems, we will obtain results that allow us to reduce the problem of online optimization over \mathbf{F} to the problem of online optimization over the quadratic (or linear) function class $\mathbf{F}_{\mu, \mathbf{g}}$.

Theorem 11. *Let \mathcal{A} be algorithm for online optimization with semi-bandit feedback. Also let \mathbf{F} be a differentiable function class over \mathcal{K} and $\mu \geq 0$. Then the following are true.*

- *If query oracles in Adv are deterministic gradient oracles, then we have*

$$\sup_{\mathcal{B} \in \text{Adv}} \mathbb{E} \left[\max_{\mathbf{u} \in \mathcal{U}} \sum_{t=a}^b \left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\|^2 \right) \right] \leq \mathcal{R}_{1, \text{Adv}_1^f(\mathbf{F}_{\mu, \nabla})}^{\mathcal{A}}.$$

- *On the other hand, if \mathbf{F} is M_1 -Lipschitz and query oracles in Adv are stochastic gradient oracles that are bounded by $B_1 \geq M_1$, then we have*

$$\sup_{\mathcal{B} \in \text{Adv}} \max_{\mathbf{u} \in \mathcal{U}} \mathbb{E} \left[\sum_{t=a}^b \left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\|^2 \right) \right] \leq \mathcal{R}_{1, \text{Adv}_1^f(\mathbf{Q}_\mu[B_1])}^{\mathcal{A}}.$$

See Appendix F for proof. Note that if f_t are μ -strongly concave, then this result reduces to Theorem 2 in [Pedramfar and Aggarwal, 2024b]. Next, we follow step (2) in the guideline to obtain the following result.

Theorem 12. *Let \mathbf{F} be function class over \mathcal{K} that is upper-quadratizable with $\mu \geq 0$, $0 < \alpha \leq 1$ and $\beta \geq 0$ and a first-order uniform wrapper \mathcal{W} .*

- If $\mathcal{W}(\nabla) = \nabla$, i.e., it maps deterministic gradient oracles into deterministic gradient oracles, then we have $\mathcal{R}_{\alpha, \text{Adv}_1^f(\mathbf{F})}^{\mathcal{W}(\mathcal{A})} \leq \beta \mathcal{R}_{1, \text{Adv}_1^f(\mathbf{F}_{\mu, \nabla})}^{\mathcal{A}}$.
- If, for any $f \in \mathbf{F}$ and any query oracle \mathcal{Q}_f bounded by B_1 , $\mathcal{W}(\mathcal{Q}_f)$ is a stochastic query oracle for $\mathcal{W}(f)$ that is bounded by B'_1 , then we have $\mathcal{R}_{\alpha, \text{Adv}_1^o(\mathbf{F}, B_1)}^{\mathcal{W}(\mathcal{A})} \leq \beta \mathcal{R}_{1, \text{Adv}_1^f(\mathbf{Q}_\mu[B'_1])}^{\mathcal{A}}$.

See Appendix G for proof. In this theorem, by using the uniform wrappers described in Section C, in the special case of $i = 1$, we recover Theorems 2, 3 and 4 in [Pedramfar and Aggarwal, 2024a]. (See Remarks 7 and 9) In other words, we recover all meta-algorithms in [Pedramfar and Aggarwal, 2024a] that are used to convert concave optimization algorithms into up-concave optimization algorithms.

Remark 10. Online Gradient Ascent requires semi-bandit feedback. Thus, Theorems 1 and 2 are special cases of Theorems 11 and 12 respectively.

Remark 11. By applying these uniform wrappers to base algorithms SO-OGA ([Garber and Kretzu, 2022]) or IA ([Zhang et al., 2018]), we recover all the results of [Pedramfar and Aggarwal, 2024a]. In particular, we also recover the results for non-stationary regret described in Table 3 in [Pedramfar and Aggarwal, 2024a].

F Proof of Theorem 11

Proof.

Deterministic oracle:

For any realization $\mathcal{B} = (\mathcal{B}_1, \dots, \mathcal{B}_T) \in \text{Adv} \subseteq \text{Adv}_1^f(\mathbf{F})$, we define $\mathcal{B}'_t(\mathbf{x}_1, \dots, \mathbf{x}_t)$ to be the tuple (q_t, ∇) where

$$\mathcal{B}'_t(\mathbf{x}_1, \dots, \mathbf{x}_t) := q_t := \mathbf{y} \mapsto \langle \nabla f_t(\mathbf{x}_t), \mathbf{y} - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}_t\|^2,$$

and $\mathcal{B}' = (\mathcal{B}'_1, \dots, \mathcal{B}'_T)$. Note that each \mathcal{B}'_t is a deterministic function of $\mathbf{x}_1, \dots, \mathbf{x}_t$ and therefore $\mathcal{B}' \in \text{Adv}_1^f(\mathbf{F}_{\mu, \nabla})$. Since the algorithm uses semi-bandit feedback, the sequence of random vectors $(\mathbf{x}_1, \dots, \mathbf{x}_T)$ chosen by \mathcal{A} is identical between the game with \mathcal{B} and \mathcal{B}' . Hence

$$\begin{aligned} \sup_{\mathcal{B} \in \text{Adv}} \mathbb{E} \left[\max_{\mathbf{u} \in \mathcal{U}} \sum_{t=a}^b \left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\|^2 \right) \right] \\ = \sup_{\mathcal{B} \in \text{Adv}} \mathbb{E} \left[\max_{\mathbf{u} \in \mathcal{U}} \left(\sum_{t=a}^b q_t(\mathbf{u}_t) - \sum_{t=a}^b q_t(\mathbf{x}_t) \right) \right] \\ \leq \sup_{\mathcal{B}' \in \text{Adv}_1^f(\mathbf{F}_{\mu, \nabla})} \mathcal{R}_{1, \mathcal{B}'}^{\mathcal{A}} = \mathcal{R}_{1, \text{Adv}_1^f(\mathbf{F}_{\mu, \nabla})}^{\mathcal{A}}. \end{aligned}$$

Stochastic oracle:

Let $\Omega^{\mathcal{Q}} = \Omega_1^{\mathcal{Q}} \times \dots \times \Omega_T^{\mathcal{Q}}$ capture all sources of randomness in the query oracles of $\text{Adv}_1^o(\mathbf{F}, B_1)$, i.e., for any choice of $\theta \in \Omega^{\mathcal{Q}}$, the query oracle is deterministic. Hence for any $\theta \in \Omega^{\mathcal{Q}}$ and realized adversary $\mathcal{B} \in \text{Adv} \subseteq \text{Adv}_1^f(\mathbf{F}, B_1)$, we may consider \mathcal{B}_θ as an object similar to an adversary with a deterministic oracle. However, note that \mathcal{B}_θ does not satisfy the unbiasedness condition of the oracle, i.e., the returned value of the oracle is not necessarily the gradient of the function at that point. Recall that \mathcal{B}_t maps a tuple $(\mathbf{x}_1, \dots, \mathbf{x}_t)$ to a tuple of f_t and a stochastic query oracle for f_t . We will use $\mathbb{E}_{\Omega^{\mathcal{Q}}}$ to denote the expectation with respect to the randomness of query oracle and

$\mathbb{E}_{\Omega_t^{\mathcal{Q}}}[\cdot] := \mathbb{E}_{\Omega^{\mathcal{Q}}}[\cdot | f_t, \mathbf{x}_t]$ to denote the expectation conditioned on the action of the agent and the adversary. Similarly, let $\mathbb{E}_{\Omega^{\mathcal{A}}}$ denote the expectation with respect to the randomness of the agent. Let \mathbf{o}_t be the random variable denoting the output of \mathcal{Q} at time-step t and let

$$\bar{\mathbf{o}}_t := \mathbb{E}[\mathbf{o}_t | f_t, \mathbf{x}_t] = \mathbb{E}_{\Omega_t^{\mathcal{Q}}}[\mathbf{o}_t] = \nabla f_t(\mathbf{x}_t).$$

Similar to the deterministic case, for any realization $\mathcal{B} = (\mathcal{B}_1, \dots, \mathcal{B}_T) \in \text{Adv}$ and any $\theta \in \Omega^{\mathcal{Q}}$, we define $\mathcal{B}'_{\theta,t}(\mathbf{x}_1, \dots, \mathbf{x}_t)$ to be the pair (q_t, ∇) where

$$q_t := \mathbf{y} \mapsto \langle \mathbf{o}_t, \mathbf{y} - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}_t\|^2.$$

We also define $\mathcal{B}'_{\theta} := (\mathcal{B}'_{\theta,1}, \dots, \mathcal{B}'_{\theta,T})$. Note that a specific choice of θ is necessary to make sure that the function returned by $\mathcal{B}'_{\theta,t}$ is a deterministic function of $\mathbf{x}_1, \dots, \mathbf{x}_t$ and not a random variable and therefore \mathcal{B}'_{θ} belongs to $\text{Adv}_1^f(\mathbf{F}_{\mu}[B_1])$.

Since the algorithm uses (semi-)bandit feedback, given a specific value of θ , the sequence of random vectors $(\mathbf{x}_1, \dots, \mathbf{x}_T)$ chosen by \mathcal{A} is identical between the game with \mathcal{B}_{θ} and \mathcal{B}'_{θ} . Therefore, for any $\mathbf{u} \in \mathcal{U}$, we have

$$\begin{aligned} & \mathbb{E} \left[\sum_{t=a}^b \left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\|^2 \right) \right] \\ &= \mathbb{E} \left[\sum_{t=a}^b \left(\langle \mathbb{E}[\mathbf{o}_t | f_t, \mathbf{x}_t], \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\|^2 \right) \right] \\ &= \mathbb{E} \left[\sum_{t=a}^b \left(\mathbb{E} \left[\langle \mathbf{o}_t, \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\|^2 | f_t, \mathbf{x}_t \right] \right) \right] \\ &= \mathbb{E} \left[\sum_{t=a}^b \left(\mathbb{E} \left[q_t(\mathbf{u}_t) - q_t(\mathbf{x}_t) | f_t, \mathbf{x}_t \right] \right) \right] \\ &= \mathbb{E} \left[\sum_{t=a}^b (q_t(\mathbf{u}_t) - q_t(\mathbf{x}_t)) \right]. \end{aligned}$$

Hence we have

$$\begin{aligned} \max_{\mathbf{u} \in \mathcal{U}} \mathbb{E} \left[\sum_{t=a}^b \left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\|^2 \right) \right] &= \max_{\mathbf{u} \in \mathcal{U}} \mathbb{E} \left[\sum_{t=a}^b (q_t(\mathbf{u}_t) - q_t(\mathbf{x}_t)) \right] \\ &\leq \mathbb{E} \left[\max_{\mathbf{u} = (\mathbf{u}_1, \dots, \mathbf{u}_T) \in \mathcal{U}} \sum_{t=a}^b (q_t(\mathbf{u}_t) - q_t(\mathbf{x}_t)) \right] \\ &= \mathcal{R}_{\mathcal{B}'_{\theta}}^{\mathcal{A}}(\mathcal{U})[a, b] \end{aligned}$$

where the inequality follows from Jensen's inequality. Therefore

$$\begin{aligned} \sup_{\mathcal{B} \in \text{Adv}} \max_{\mathbf{u} \in \mathcal{U}} \mathbb{E} \left[\sum_{t=a}^b \left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\|^2 \right) \right] \\ &\leq \sup_{\mathcal{B} \in \text{Adv}, \theta \in \Omega^{\mathcal{Q}}} \mathcal{R}_{\mathcal{B}'_{\theta}}^{\mathcal{A}} \\ &\leq \sup_{\mathcal{B}' \in \text{Adv}_1^f(\mathbf{F}_{\mu}[B_1])} \mathcal{R}_{\mathcal{B}'}^{\mathcal{A}} \\ &= \mathcal{R}_{\text{Adv}_1^f(\mathbf{F}_{\mu}[B_1])}^{\mathcal{A}} \end{aligned} \quad \square$$

G Proof of Theorem 12

Proof.

(i):

We have

$$\begin{aligned}
\mathcal{R}_{\alpha, \text{Adv}_1^f(\mathbf{F})}^{\mathcal{W}(\mathcal{A})} &= \sup_{\mathcal{B} \in \text{Adv}_1^f(\mathbf{F})} \mathbb{E} \left[\max_{\mathbf{u}=(\mathbf{u}_1, \dots, \mathbf{u}_T) \in \mathcal{U}} \sum_{t=a}^b (\alpha f_t(\mathbf{u}_t) - f_t(\mathcal{W}(\mathbf{x}_t))) \right] \\
&\leq \sup_{\mathcal{B} \in \text{Adv}_1^f(\mathbf{F})} \max_{\mathbf{u}=(\mathbf{u}_1, \dots, \mathbf{u}_T) \in \mathcal{U}} \mathbb{E} \left[\sum_{t=a}^b \beta \left(\langle \nabla \mathcal{W}(f_t)(\mathbf{x}_t), \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\| \right) \right] \\
&= \beta \mathcal{R}_{1, \text{Adv}_1^f(\mathbf{H}_\mu, \nabla)}^{\mathcal{A}}.
\end{aligned}$$

(ii):

Since Adv is oblivious, the sequence of functions (f_1, \dots, f_T) is not random and we have

$$\begin{aligned}
\mathcal{R}_{\alpha, \text{Adv}_1^o(\mathbf{F}, B_1)}^{\mathcal{W}(\mathcal{A})} &= \sup_{\mathcal{B} \in \text{Adv}_1^o(\mathbf{F}, B_1)} \mathbb{E} \left[\max_{\mathbf{u}=(\mathbf{u}_1, \dots, \mathbf{u}_T) \in \mathcal{U}} \sum_{t=a}^b (\alpha f_t(\mathbf{u}_t) - f_t(\mathcal{W}(\mathbf{x}_t))) \right] \\
&= \sup_{\mathcal{B} \in \text{Adv}_1^o(\mathbf{F}, B_1)} \max_{\mathbf{u}=(\mathbf{u}_1, \dots, \mathbf{u}_T) \in \mathcal{U}} \mathbb{E} \left[\sum_{t=a}^b (\alpha f_t(\mathbf{u}_t) - f_t(\mathcal{W}(\mathbf{x}_t))) \right] \\
&\leq \sup_{\mathcal{B} \in \text{Adv}_1^o(\mathbf{F}, B_1)} \max_{\mathbf{u}=(\mathbf{u}_1, \dots, \mathbf{u}_T) \in \mathcal{U}} \mathbb{E} \left[\sum_{t=a}^b \beta \left(\langle \nabla \mathcal{W}(f_t)(\mathbf{x}_t), \mathbf{u}_t - \mathbf{x}_t \rangle - \frac{\mu}{2} \|\mathbf{u}_t - \mathbf{x}_t\| \right) \right] \\
&= \beta \mathcal{R}_{1, \text{Adv}_1^f(\mathbf{Q}_\mu[B'_1])}^{\mathcal{A}}.
\end{aligned}$$

□

H Follow The Regularized Leader

We start by defining the notion of self-concordant barrier.

Definition 6 ([Hazan et al., 2016]). Let $\mathcal{K} \in \mathbb{R}^d$ be a convex set with non empty interior $\text{int}(\mathcal{K})$. We call a function $\Phi : \text{int}(\mathcal{K}) \rightarrow \mathbb{R}$ a ν -self-concordant barrier of \mathcal{K} if:

- (i) Φ is three-times continuously differentiable, convex, and tends to infinity along any sequence of points approaching the boundary of \mathcal{K} ;
- (ii) For every $\mathbf{h} \in \mathbb{R}^d$ and $\mathbf{x} \in \text{int}(\mathcal{K})$, we have:

$$|\nabla^3 \Phi(\mathbf{x})[\mathbf{h}, \mathbf{h}, \mathbf{h}]| \leq 2(\nabla^2 \Phi(\mathbf{x})[\mathbf{h}, \mathbf{h}])^{3/2}, \quad |\nabla \Phi(\mathbf{x})[\mathbf{h}]| \leq \nu^{1/2}(\nabla^2 \Phi(\mathbf{x})[\mathbf{h}, \mathbf{h}])^{1/2}$$

where the third-order differential is defined as $\nabla^3 \Phi(\mathbf{x})[\mathbf{h}, \mathbf{h}, \mathbf{h}] := \frac{\partial^3}{\partial t_1 \partial t_2 \partial t_3} \Phi(\mathbf{x} + t_1 \mathbf{h} + t_2 \mathbf{h} + t_3 \mathbf{h})|_{t_1=t_2=t_3=0}$.

Next we define the notion of local norm and dual norm with respect to a self-concordant barrier.

Definition 7. For every $\mathbf{x} \in \text{int}(\mathcal{K})$, the Hessian of the self-concordant barrier induces a *local* norm, denoted as $\|\cdot\|_{\Phi, \mathbf{x}}$, and a *dual* norm, denoted as $\|\cdot\|_{\Phi, \mathbf{x}, *}$, where for any $\mathbf{v} \in \mathbb{R}^d$,

$$\|\mathbf{v}\|_{\Phi, \mathbf{x}} = \sqrt{\mathbf{v}^T \nabla^2 \Phi(\mathbf{x}) \mathbf{v}}, \quad \|\mathbf{v}\|_{\Phi, \mathbf{x}, *} = \sqrt{\mathbf{v}^T (\nabla^2 \Phi(\mathbf{x}))^{-1} \mathbf{v}}.$$

An important result for FTRL is the following theorem which was proved in [Abernethy et al., 2008]. It shows that if we set the regularizer to be a self-concordant barrier of \mathcal{K} and the algorithm can access the unbiased estimator of \mathbf{g}_t , then the regret of the generated solution sequence $\{\mathbf{x}_t\}_{t=1}^T$ can be bounded in terms of the local norm of the estimator.

Theorem 13 ([Abernethy et al., 2008]). *Let $\mathcal{K} \subseteq \mathbb{R}^d$ be a convex set, $\Phi(x)$ be a self-concordant barrier on \mathcal{K} , $\{\mathbf{g}_t\}_{t=1}^T$ be a sequence of random vectors in \mathbb{R}^d . Then running FTRL (described in Equation 3) on a vector sequence $\{\mathbf{g}_t\}_{t=1}^T$ in \mathbb{R}^d with $\Phi(x)$ as the regularizer will produce a sequence of point $\{\mathbf{x}_t\}_{t=1}^T$ in \mathcal{K} where*

$$\sum_{t=1}^T \langle \mathbf{g}_t, \mathbf{y} - \mathbf{x}_t \rangle \leq \eta \sum_{t=1}^T \|\mathbf{g}_t\|_{\Phi, \mathbf{x}_t, *}^2 + \frac{\Phi(\mathbf{y}) - \Phi(\mathbf{x}_1)}{\eta},$$

for any $\mathbf{y} \in \mathcal{K}$.

The ellipsoid gradient estimator was proposed in [Abernethy et al., 2008], where the authors use it along with Theorem 13 to design an $\tilde{O}(\sqrt{T})$ regret algorithm for bandit linear optimization. For a continuous function but possibly non-smooth $f : \mathbb{R}^d \rightarrow \mathbb{R}$ and an invertible matrix $\Sigma \in \mathbb{R}^{d \times d}$, we define the Σ -smoothed version of f .

Definition 8. For function $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ and invertible matrix $\Sigma \in \mathbb{R}^{d \times d}$, we call $f^\Sigma(\mathbf{x})$ a Σ -smoothed version of $f(\mathbf{x})$, where $f^\Sigma(\mathbf{x}) = \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} [f(\mathbf{x} + \Sigma \mathbf{v})]$. Here $\mathbf{v} \sim \mathbb{B}^d$ means that \mathbf{v} is sampled from the unit ball \mathbb{B}^d uniformly at random.

There is a surprising fact that there is an unbiased estimator of $\nabla f^\Sigma(\mathbf{x})$ for any \mathbf{x} , and the estimator uses only a single query to the value oracle of f .

Lemma 5 ([Abernethy et al., 2008]). *Let $\Sigma \in \mathbb{R}^{d \times d}$ be an invertible matrix, $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ be an arbitrary function. Then $\nabla f^\Sigma(\mathbf{x}) = d \mathbb{E}_{\mathbf{v} \sim \mathbb{S}^{d-1}} [f(\mathbf{x} + \Sigma \mathbf{v}) \Sigma^{-1} \mathbf{v}]$. Here $\mathbf{v} \sim \mathbb{S}^{d-1}$ means that \mathbf{v} is sampled from the $(d-1)$ -dimensional unit sphere \mathbb{S}^{d-1} uniformly at random.*

If f is a linear function, $f^\Sigma(\mathbf{x}) = f(\mathbf{x})$, so Lemma 5 provides a one-sample unbiased estimator of the gradient of the linear function. The ellipsoid gradient estimator is usually used along with FTRL with a self-concordant regularizer Φ of \mathcal{K} . When the invertible matrix Σ is set to be $(\nabla^2 \Phi(\mathbf{x}))^{-1/2}$ and $\mathbf{x} \in \text{int}(\mathcal{K})$, the sampled action $\mathbf{x} + \Sigma \mathbf{v}$ is located in the surface of a so-called **Dikin ellipsoid** centered at \mathbf{x} , i.e. $\{\mathbf{x}' \mid \|\mathbf{x}' - \mathbf{x}\|_{\Phi, \mathbf{x}} \leq 1\}$. The fact that Dikin ellipsoid is entirely contained in \mathcal{K} allows us to define f^Σ at \mathbf{x} .

We finish this section with quick overview of the concept of the Minkowski function, the Minkowski set and some of their useful properties.

Definition 9. Let \mathcal{K} be a compact convex set, the Minkowski function $\pi_{\mathbf{x}} : \mathcal{K} \rightarrow \mathbb{R}$ parameterized by a pole $\mathbf{x} \in \text{int}(\mathcal{K})$ is defined as $\pi_{\mathbf{x}}(\mathbf{y}) := \inf\{t \geq 0 \mid \mathbf{x} + t^{-1}(\mathbf{y} - \mathbf{x}) \in \mathcal{K}\}$. Given $\delta \in \mathbb{R}^+$ and $\mathbf{x}_1 \in \text{int}(\mathcal{K})$, we define the Minkowski set

$$\mathcal{K}_{\gamma, \mathbf{x}_1} := \{\mathbf{x} \in \mathcal{K} \mid \pi_{\mathbf{x}_1}(\mathbf{x}) \leq (1 + \gamma)^{-1}\}.$$

Lemma 6 ([Abernethy et al., 2008]). *Let \mathcal{K} be a compact convex set, $\mathbf{x} \in \text{int}(\mathcal{K})$ with diameter D , $\mathbf{u}_* \in \mathcal{K}$ and $\hat{\mathbf{u}}_* := \arg\min_{\mathbf{z} \in \mathcal{K}_{\gamma, \mathbf{x}}} \|\mathbf{z} - \mathbf{u}_*\|$ be the projection of \mathbf{u}_* onto the Minkowski set $\mathcal{K}_{\gamma, \mathbf{x}}$, then*

$$\|\mathbf{u}_* - \hat{\mathbf{u}}_*\| \leq \gamma D.$$

The following lemma provides an upper bound of the difference between the function value of a self-concordant barrier at two different points.

Lemma 7 ([Nesterov and Nemirovskii, 1994]). *Let Φ be a ν -self-concordant barrier over a compact convex set \mathcal{K} , then for all $\mathbf{x}, \mathbf{y} \in \text{int}(\mathcal{K})$:*

$$\Phi(\mathbf{y}) - \Phi(\mathbf{x}) \leq \nu \log \frac{1}{1 - \pi_{\mathbf{x}}(\mathbf{y})}.$$

I Technical Lemmas

This section provides some technical lemmas that will be used in the proofs later.

Lemma 8. *Let \mathcal{K} be a compact set and let $f : \mathcal{K} \rightarrow \mathbb{R}^d$ be an M_2 -smooth function. Then f may be extended to an M_2 -smooth function $\tilde{f} : \mathbb{R}^d \rightarrow \mathbb{R}$.*

Proof. The function ∇F is an M_2 -Lipschitz function defined on \mathcal{K} . Therefore, according to Kirschbraun theorem [Kirschbraun, 1934] it may be extended to a function $g : \mathbb{R}^d \rightarrow \mathbb{R}^d$ that is M_2 -Lipschitz. Now the result follows directly from Whitney's extension theorem [Whitney, 1934]. \square

Parts (i)-(iii) of the following lemma are well-known in the literature. (See Lemma A.5 in [Wan et al., 2023] for a proof). Here we provide a proof for part (iv).

Lemma 9. *Following properties hold for Σ -smoothed version of a function $f(\mathbf{x})$ for an invertible matrix Σ .*

- (i) *If $f(\mathbf{x})$ is a monotone function, then so is $f^\Sigma(\mathbf{x})$.*
- (ii) *If $f(\mathbf{x})$ is M_1 -Lipschitz, then so is $f^\Sigma(\mathbf{x})$.*
- (iii) *If $f(\mathbf{x})$ is M_2 -smooth, then so is $f^\Sigma(\mathbf{x})$.*
- (iv) *If f is upper-quadratizable with a uniform wrapper \mathcal{W} and α, β and μ , then we have*

$$\alpha f^\Sigma(\mathbf{y}) - (f \circ \mathcal{W})^\Sigma(\mathbf{x}) \leq \beta \left(\langle \nabla(\mathcal{W}(f))^\Sigma(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right).$$

Proof. We have

$$\begin{aligned} \alpha f^\Sigma(\mathbf{y}) - (f \circ \mathcal{W})^\Sigma(\mathbf{x}) &= \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} [\alpha f(\mathbf{y} + \Sigma \mathbf{v}) - f(\mathcal{W}(\mathbf{x} + \Sigma \mathbf{v}))] \\ &\leq \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} \left[\beta \left(\langle \nabla \mathcal{W}(f)(\mathbf{x} + \Sigma \mathbf{v}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right) \right] \\ &= \beta \left(\langle \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} [\nabla \mathcal{W}(f)(\mathbf{x} + \Sigma \mathbf{v})], \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right) \\ &= \beta \left(\langle \nabla \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} [\mathcal{W}(f)(\mathbf{x} + \Sigma \mathbf{v})], \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right) \\ &= \beta \left(\langle \nabla(\mathcal{W}(f))^\Sigma(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 \right). \end{aligned} \quad \square$$

Lemma 10. *If $f : \mathcal{K} \rightarrow \mathbb{R}$ is M_1 -Lipschitz and M_2 -smooth and $g : \mathcal{K} \rightarrow \mathcal{K}$ is M'_1 -Lipschitz and M'_2 -smooth, then $f \circ g$ is M''_1 -Lipschitz and M''_2 -smooth where $M''_1 := M_1 M'_1$ and $M''_2 := M_1 M'_2 + M_2 M'_1$.*

Proof. We have

$$\|D(f \circ g)(\mathbf{x})\| = \|Df(g(\mathbf{x})) \cdot Dg(\mathbf{x})\| \leq M_1 M'_1,$$

and therefore for all $\mathbf{x}, \mathbf{y} \in \mathcal{K}$, we have

$$\begin{aligned} \|D(f \circ g)(\mathbf{x}) - D(f \circ g)(\mathbf{y})\| &= \|Df(g(\mathbf{x})) \cdot Dg(\mathbf{x}) - Df(g(\mathbf{y})) \cdot Dg(\mathbf{y})\| \\ &\leq \|Df(g(\mathbf{x})) \cdot Dg(\mathbf{x}) - Df(g(\mathbf{x})) \cdot Dg(\mathbf{y})\| \\ &\quad + \|Df(g(\mathbf{x})) \cdot Dg(\mathbf{y}) - Df(g(\mathbf{y})) \cdot Dg(\mathbf{y})\| \\ &= \|Df(g(\mathbf{x}))\| \|Dg(\mathbf{x}) - Dg(\mathbf{y})\| \\ &\quad + \|Df(g(\mathbf{x})) - Df(g(\mathbf{y}))\| \|Dg(\mathbf{y})\| \\ &\leq M_1 M'_2 \|\mathbf{x} - \mathbf{y}\| + M_2 M'_1 \|g(\mathbf{x}) - g(\mathbf{y})\| \\ &\leq (M_1 M'_2 + M_2 M'_1) \|\mathbf{x} - \mathbf{y}\|. \end{aligned} \quad \square$$

J Proof of Theorem 3

Proof. We have

$$\begin{aligned}
& \sum_{t=1}^T \mathbb{E} [f_t(\mathbf{u}_*) - f_t(\mathbf{x}_t)] \\
&= \sum_{t=1}^T \mathbb{E} [f_t^{\delta\Sigma_t}(\hat{\mathbf{u}}_*) - f_t^{\delta\Sigma_t}(\mathbf{x}_t)] + \underbrace{\sum_{t=1}^T \mathbb{E} [f_t^{\delta\Sigma_t}(\mathbf{u}_*) - f_t^{\delta\Sigma_t}(\hat{\mathbf{u}}_*)]}_{(A)} \\
&\quad + \underbrace{\sum_{t=1}^T \mathbb{E} [f_t(\mathbf{u}_*) - f_t^{\delta\Sigma_t}(\mathbf{u}_*)]}_{(B)} + \underbrace{\sum_{t=1}^T \mathbb{E} [f_t^{\delta\Sigma_t}(\mathbf{x}_t) - f_t(\mathbf{x}_t)]}_{(C)}
\end{aligned} \tag{4}$$

Note that, for the terms above to be well-defined, we need to be able to define $f_t^{\delta\Sigma_t}$ over \mathcal{K} which requires computing f_t over a set that is slightly larger than \mathcal{K} . Using Lemma 8, we assume that all functions f_t are well-defined and M_2 -smooth over \mathbb{R}^d .

Bounding (A): Since $f_t(\mathbf{x})$ is M_1 -Lipschitz continuous, $f_t^{\delta\Sigma_t}$ is also M_1 -Lipschitz continuous by Lemma 9. Since $\|\hat{\mathbf{u}}_* - \mathbf{u}_*\| \leq \gamma D$ by Lemma 6,

$$\begin{aligned}
\sum_{t=1}^T \mathbb{E} [f_t^{\delta\Sigma_t}(\mathbf{u}_*) - f_t^{\delta\Sigma_t}(\hat{\mathbf{u}}_*)] &\leq \sum_{t=1}^T \mathbb{E} [|f_t^{\delta\Sigma_t}(\hat{\mathbf{u}}_*) - f_t^{\delta\Sigma_t}(\mathbf{u}_*)|] \\
&\leq \sum_{t=1}^T M_1 \gamma D = M_1 \gamma D T.
\end{aligned} \tag{5}$$

Bounding (B): Since $f_t(\mathbf{x})$ is M_2 -smooth, by Lemma 9, $f_t^{\delta\Sigma_t}$ is M_2 -smooth. Thus,

$$\begin{aligned}
f_t(\mathbf{u}_*) - f_t^{\delta\Sigma_t}(\mathbf{u}_*) &= \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} [f_t(\mathbf{u}_*) - f_t(\mathbf{u}_* + \delta\Sigma_t \mathbf{v})] \\
&\leq \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} \left[-\langle \nabla f_t(\mathbf{u}_*), \delta\Sigma_t \mathbf{v} \rangle + \frac{M_2}{2} \|\delta\Sigma_t \mathbf{v}\|^2 \right] \\
&= \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} [-\langle \nabla f_t(\mathbf{u}_*), \delta\Sigma_t \mathbf{v} \rangle] + \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} \left[\frac{M_2}{2} \|\delta\Sigma_t \mathbf{v}\|^2 \right] \\
&= \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} \left[\frac{M_2}{2} \|\delta\Sigma_t \mathbf{v}\|^2 \right] \\
&\leq \frac{M_2 \delta^2 D^2}{2}.
\end{aligned}$$

Note that in the last inequality, we used the fact that the Dikin ellipsoid centered at \mathbf{x}_t is contained in \mathcal{K} which implies that $\mathbf{x}_t + \Sigma_t \mathbf{v} \in \mathcal{K}$ and therefore $\|\Sigma_t \mathbf{v}\| \leq D$. It follows that,

$$\sum_{t=1}^T \mathbb{E} [f_t(\hat{\mathbf{u}}_*) - f_t^{\delta\Sigma_t}(\hat{\mathbf{u}}_*)] \leq \frac{M_2 \delta^2 D^2 T}{2}. \tag{6}$$

Bounding (C): Similarly,

$$\begin{aligned}
f_t^{\delta\Sigma_t}(\mathbf{x}_t) - f_t(\mathbf{x}_t) &= \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} [f_t(\mathbf{x}_t + \delta\Sigma_t \mathbf{v}) - f_t(\mathbf{x}_t)] \\
&\leq \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} \left[-\langle \nabla f_t(\mathbf{x}_t), \delta\Sigma_t \mathbf{v} \rangle + \frac{M_2}{2} \|\delta\Sigma_t \mathbf{v}\|^2 \right] \leq \frac{M_2 \delta^2 D^2}{2}.
\end{aligned}$$

Therefore,

$$\sum_{t=1}^T \mathbb{E} [f_t^{\delta\Sigma_t}(\mathbf{x}_t) - f_t(\mathbf{x}_t)] \leq \frac{M_2 \delta^2 D^2 T}{2} \tag{7}$$

Putting 5,6,7 in 4, we see that

$$\begin{aligned} \sum_{t=1}^T \mathbb{E} [f_t(\mathbf{u}_*) - f_t(\mathbf{x}_t)] &\leq \sum_{t=1}^T \mathbb{E} [f_t^{\delta\Sigma_t}(\hat{\mathbf{u}}_*) - f_t^{\delta\Sigma_t}(\mathbf{x}_t)] \\ &\quad + \alpha M_1 \gamma DT + \frac{M_2 \delta^2 D^2 T}{2} + \frac{M_2 \delta^2 D^2 T}{2}, \end{aligned}$$

which completes the proof of the first claim.

To prove the second claim, we first use Lemma 5, with $\Sigma = \delta\Sigma_t$, to see that $\mathbb{E} [\mathbf{o}_t | \mathbf{x}_t] = \nabla f_t^{\delta\Sigma_t}(\mathbf{x}_t)$. On the other hand, since \mathcal{Q}_t is bounded by B_0 , we have

$$\|\mathbf{o}_t\|_{\mathbf{x}_t,*}^2 = \left\| \frac{d}{\delta} y_t \Sigma_t^{-1} \mathbf{v}_t \right\|_{\mathbf{x}_t,*}^2 = \frac{d^2}{\delta^2} |y_t|^2 \mathbf{v}_t^T \Sigma_t^{-1} (\nabla^2 \Phi(\mathbf{x}_t))^{-1} \Sigma_t^{-1} \mathbf{v}_t \leq \frac{d^2}{\delta^2} B_0^2 \|\mathbf{v}_t\|^2 \leq \frac{d^2 B_0^2}{\delta^2}$$

Hence, using Theorem 13 with $\mathbf{g}_t = \mathbf{o}_t$ and $\mathbf{y} = \hat{\mathbf{u}}_*$, we see that

$$\begin{aligned} \sum_{t=1}^T \mathbb{E} [\langle \nabla f_t^{\delta\Sigma_t}(\mathbf{x}_t), \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle] &= \sum_{t=1}^T \mathbb{E} [\langle \mathbb{E} [\mathbf{o}_t | \mathbf{x}_t], \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle] \\ &= \sum_{t=1}^T \mathbb{E} [\mathbb{E} [\langle \mathbf{o}_t, \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle | \mathbf{x}_t]] \\ &= \mathbb{E} \left[\sum_{t=1}^T \langle \mathbf{o}_t, \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle \right] \\ &\leq \mathbb{E} \left[\eta \sum_{t=1}^T \|\mathbf{o}_t\|_{\Phi, \mathbf{x}_t,*}^2 + \frac{\Phi(\hat{\mathbf{u}}_*) - \Phi(\mathbf{x}_1)}{\eta} \right] \\ &\leq \eta \sum_{t=1}^T \frac{d^2 B_0^2}{\delta^2} + \frac{\Phi(\hat{\mathbf{u}}_*) - \Phi(\mathbf{x}_1)}{\eta} \\ &\leq \frac{\eta d^2 B_0^2 T}{\delta^2} + \frac{\nu \log(\frac{1}{1-(1+\gamma)^{-1}})}{\eta}, \end{aligned}$$

where we used Lemma 7 in the last inequality. \square

K Proof of Theorem 4

Proof. Let $\mathcal{B} \in \text{Adv}$ be a realized adversary and let f_1, \dots, f_T be the sequence of functions selected by \mathcal{B} . Also let $\mathbf{u}_* \in \text{argmax}_{\mathbf{u} \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{u})$ and $\hat{\mathbf{u}}_* \in \text{argmin}_{\mathbf{x} \in \mathcal{K}_{\gamma, \mathbf{x}_1}} \|\mathbf{u}_* - \mathbf{x}\|$ where $\gamma = T^{-1}$. We have

$$\begin{aligned} \mathcal{R}_{\alpha, \mathcal{B}}^{\mathcal{W}(\text{ZO-FTRL})} &= \sum_{t=1}^T \mathbb{E} [\alpha f_t(\mathbf{u}_*) - f_t(\mathcal{W}(\mathbf{x}_t))] \\ &= \sum_{t=1}^T \mathbb{E} [\alpha f_t^{\delta\Sigma_t}(\hat{\mathbf{u}}_*) - (f_t \circ \mathcal{W})^{\delta\Sigma_t}(\mathbf{x}_t)] + \alpha \underbrace{\sum_{t=1}^T \mathbb{E} [f_t^{\delta\Sigma_t}(\mathbf{u}_*) - f_t^{\delta\Sigma_t}(\hat{\mathbf{u}}_*)]}_{(A)} \\ &\quad + \alpha \underbrace{\sum_{t=1}^T \mathbb{E} [f_t(\mathbf{u}_*) - f_t^{\delta\Sigma_t}(\mathbf{u}_*)]}_{(B)} + \underbrace{\sum_{t=1}^T \mathbb{E} [(f_t \circ \mathcal{W})^{\delta\Sigma_t}(\mathbf{x}_t) - f_t(\mathcal{W}(\mathbf{x}_t))]}_{(C)} \end{aligned}$$

As in the proof of Theorem 3, we use Lemma 8 to extend all functions f_t to M_2 -smooth functions over \mathbb{R}^d and we bound the terms (A) and (B) by $M_1 \gamma DT$ and $\frac{M_2 \delta^2 D^2 T}{2}$, respectively. To bound (C),

we first use Lemma 10 to see that $f_t \circ \mathcal{W}$ is M_2'' -smooth, where $M_2'' = M_1 M_2' + M_2 M_1'$. Hence, we see that

$$\begin{aligned} (f_t \circ \mathcal{W})^{\delta \Sigma_t}(\mathbf{x}_t) - f_t(\mathcal{W}(\mathbf{x}_t)) &= \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} [f_t(\mathcal{W}(\mathbf{x}_t + \delta \Sigma_t \mathbf{v})) - f_t(\mathcal{W}(\mathbf{x}_t))] \\ &\leq \mathbb{E}_{\mathbf{v} \sim \mathbb{B}^d} \left[-\langle \nabla f_t(\mathcal{W}(\mathbf{x}_t)), \delta \Sigma_t \mathbf{v} \rangle + \frac{M_2''}{2} \|\delta \Sigma_t \mathbf{v}\|^2 \right] \leq \frac{M_2'' \delta^2 D^2}{2}. \end{aligned}$$

Therefore,

$$\sum_{t=1}^T \mathbb{E} \left[(f_t \circ \mathcal{W})^{\delta \Sigma_t}(\mathbf{x}_t) - f_t(\mathcal{W}(\mathbf{x}_t)) \right] \leq \frac{M_2'' \delta^2 D^2 T}{2}$$

Putting the bounds for (A), (B) and (C) together, we see that

$$\begin{aligned} \mathcal{R}_{\alpha, \mathcal{B}}^{\mathcal{W}(\text{ZO-FTRL})} &= \sum_{t=1}^T \mathbb{E} [\alpha f_t(\mathbf{u}_*) - f_t(\mathcal{W}(\mathbf{x}_t))] \\ &\leq \sum_{t=1}^T \mathbb{E} \left[\alpha f_t^{\delta \Sigma_t}(\hat{\mathbf{u}}_*) - (f_t \circ \mathcal{W})^{\delta \Sigma_t}(\mathbf{x}_t) \right] + \alpha M_1 \gamma D T + \frac{(\alpha M_2 + M_2'') \delta^2 D^2 T}{2} \\ &\leq \sum_{t=1}^T \mathbb{E} \left[\beta \langle \nabla(\mathcal{W}(f_t))^{\delta \Sigma_t}(\mathbf{x}_t), \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle \right] + \alpha M_1 \gamma D T + \frac{(\alpha M_2 + M_2'') \delta^2 D^2 T}{2}, \end{aligned} \tag{8}$$

where the second inequality follows from Lemma 9.

To bound the remaining term, we use an argument similar to the one used in the proof of Theorem 3 again. Using Lemma 5 with $\Sigma = \delta \Sigma_t$ and the fact that y_t is an unbiased sample of $\mathcal{W}(f_t)$ at $\mathbf{x}_t + \delta \Sigma_t \mathbf{v}_t$, we see that $\mathbb{E} [\mathbf{o}_t \mid \mathbf{x}_t] = \nabla(\mathcal{W}(f_t))^{\delta \Sigma_t}(\mathbf{x}_t)$. On the other hand, since $\mathcal{W}(\mathcal{Q}_t)$ is bounded by B_0 , we have $|y_t| \leq B_0$, which implies that

$$\|\mathbf{o}_t\|_{\mathbf{x}_t, *}^2 = \left\| \frac{d}{\delta} y_t \Sigma_t^{-1} \mathbf{v}_t \right\|_{\mathbf{x}_t, *}^2 = \frac{d^2}{\delta^2} |y_t|^2 \mathbf{v}_t^T \Sigma_t^{-1} \left(\nabla^2 \Phi(\mathbf{x}_t) \right)^{-1} \Sigma_t^{-1} \mathbf{v}_t \leq \frac{d^2}{\delta^2} B_0^2 \|\mathbf{v}_t\|^2 \leq \frac{d^2 B_0^2}{\delta^2}.$$

Hence, using Theorem 13 with $\mathbf{g}_t = \mathbf{o}_t$ and $\mathbf{y} = \hat{\mathbf{u}}_*$, we see that

$$\begin{aligned} \sum_{t=1}^T \mathbb{E} \left[\beta \langle \nabla(\mathcal{W}(f_t))^{\delta \Sigma_t}(\mathbf{x}_t), \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle \right] &= \beta \sum_{t=1}^T \mathbb{E} \left[\langle \mathbb{E} [\mathbf{o}_t \mid \mathbf{x}_t], \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle \right] \\ &= \beta \sum_{t=1}^T \mathbb{E} \left[\mathbb{E} [\langle \mathbf{o}_t, \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle \mid \mathbf{x}_t] \right] \\ &= \beta \mathbb{E} \left[\sum_{t=1}^T \langle \mathbf{o}_t, \hat{\mathbf{u}}_* - \mathbf{x}_t \rangle \right] \\ &\leq \beta \mathbb{E} \left[\eta \sum_{t=1}^T \|\mathbf{o}_t\|_{\Phi, \mathbf{x}_t, *}^2 + \frac{\Phi(\hat{\mathbf{u}}_*) - \Phi(\mathbf{x}_1)}{\eta} \right] \\ &\leq \beta \eta \sum_{t=1}^T \frac{d^2 B_0^2}{\delta^2} + \beta \frac{\Phi(\hat{\mathbf{u}}_*) - \Phi(\mathbf{x}_1)}{\eta} \\ &\leq \frac{\beta \eta d^2 B_0^2 T}{\delta^2} + \frac{\beta \nu \log(\frac{1}{1-(1+\gamma)^{-1}})}{\eta}, \end{aligned}$$

where we used Lemma 7 in the last inequality. Plugging this into Equation 8 and using $M_2'' = M_1 M_2' + M_2 M_1'^2$ and $\gamma = T^{-1}$, we see that

$$\begin{aligned} \mathcal{R}_{\alpha, \mathcal{B}}^{\mathcal{W}(\text{ZO-FTRL})} &\leq \frac{\beta \eta d^2 B_0^2 T}{\delta^2} + \frac{\beta \nu \log(\frac{1}{1-(1+\gamma)^{-1}})}{\eta} \\ &\quad + \alpha M_1 \gamma D T + \frac{(\alpha M_2 + M_1 M_2' + M_2 M_1'^2) \delta^2 D^2 T}{2} \\ &= O\left(\eta \delta^{-2} T + \eta^{-1} \log T + \delta^2 T\right). \end{aligned} \quad \square$$

L Proof of Theorem 5

Proof. Note that in all three cases, $\mathcal{W}^{\text{action}}$ is 1-Lipschitz and 0-smooth. Now the result for the first case follows immediately from the fact that $\mathcal{W}^M = \text{Id}$. Also note that for any zeroth order query oracle \mathcal{Q}_f for a function $f \in \mathbf{F}^{\text{MO}}$ and any $\mathbf{y} \in \mathcal{K}$

$$|\mathcal{W}^{\text{MO}}(\mathcal{Q}_f)(\mathbf{y})| = |z^{-1} \mathcal{Q}_f(z * \mathbf{y})| \leq z^{-1} \cdot C \|z * \mathbf{y}\| = \|\mathbf{y}\| \leq D.$$

Thus the query oracle $\mathcal{W}(\mathcal{Q}_f)$ is bounded by D and the assumptions of Theorem 4 are satisfied. The proof of boundedness of $\mathcal{W}^{\text{NM}}(\mathcal{Q}_f)$ for any $f \in \mathbf{F}^{\text{NM}}$ is similar. \square

M Stochastic Full-information To Trivial query - SFTT

In this section, we discuss the SFTT meta-algorithm (Algorithm 4 in [Pedramfar and Aggarwal, 2024a]) which converts algorithms that require full-information feedback into algorithms that have a trivial query oracle. In particular, it converts algorithms require zeroth-order full-information feedback into bandit algorithms.

We say a function class \mathbf{F} is closed under convex combination if for any $f_1, \dots, f_k \in \mathbf{F}$ and any $\delta_1, \dots, \delta_k \geq 0$ with $\sum_i \delta_i = 1$, we have $\sum_i \delta_i f_i \in \mathbf{F}$.

Theorem 14 (Theorem 7 and Remark 1 and Corollary 6 in [Pedramfar and Aggarwal, 2024a]). *Let \mathcal{A} be an online optimization algorithm with full-information feedback and with K queries at each time-step where $\mathcal{A}^{\text{query}}$ does not depend on the observations in the current round and $\mathcal{A}' = \text{SFTT}(\mathcal{A})$. Then, for any M_1 -Lipschitz function class \mathbf{F} that is closed under convex combination and any $B_1 \geq M_1$, $0 < \alpha \leq 1$ and $1 \leq a \leq b \leq T$, let $a' = \lfloor (a-1)/L \rfloor + 1$, $b' = \lceil b/L \rceil$, $D = \text{diam}(\mathcal{K})$ and let $\{T\}$ and $\{T/L\}$ denote the horizon of the adversary. If we also have $\mathcal{R}_{\alpha, \text{Adv}_i^o(\mathbf{F}, B)}^{\mathcal{A}'}(\mathcal{K}_\star^T)[a, b] = O(BT^\eta)$, $K = O(1)$ and $L = O(T^{\frac{1-\eta}{2-\eta}})$, then*

$$\mathcal{R}_{\alpha, \text{Adv}_i^o(\mathbf{F}, B)}^{\mathcal{A}'}(\mathcal{K}_\star^T)[a, b] = O\left(BT^{\frac{1}{2-\eta}}\right).$$

More generally, the above result holds even if the query oracles are not bounded. Specifically, what we require is that the set of query oracles to be closed under convex combinations.

Algorithm 6: Stochastic Full-information To Trivial query - SFTT(\mathcal{A})

Input : base algorithm \mathcal{A} , horizon T , block size $L > K$.

for $q = 1, 2, \dots, T/L$ **do**

 Let $\hat{\mathbf{x}}_q$ be the action chosen by $\mathcal{A}^{\text{action}}$

 Let $(\hat{\mathbf{y}}_q^i)_{i=1}^K$ be the queries selected by $\mathcal{A}^{\text{query}}$

 Let $(t_{q,1}, \dots, t_{q,L})$ be a random permutation of $\{(q-1)L+1, \dots, qL\}$

for $t = (q-1)L+1, \dots, qL$ **do**

if $t = t_{q,i}$ for some $1 \leq i \leq K$ **then**

 Play the action $\mathbf{x}_t = \hat{\mathbf{y}}_q^i$

 Return the observation to the query oracle as the response to the i -th query

else

 Play the action $\mathbf{x}_t = \hat{\mathbf{x}}_q$

end

end

end

N Proof of Theorem 6

Proof. All three class of functions considered are closed under convex combination. Therefore we may directly apply Theorems 5 and 14 to obtain this result for the first case.

For any sequence of functions f_1, \dots, f_k and query oracles $\mathcal{Q}_1, \dots, \mathcal{Q}_k$ for these functions that are contained within a cone $\text{Cone}(\mathbf{0}, C)$ and non-negative numbers $\delta_1, \dots, \delta_k$ such that $\sum_i \delta_i = 0$, the query oracle $\bar{\mathcal{Q}}$ that uses \mathcal{Q}_i with probability δ_i is trivially a query oracle for $\sum_i \delta_i f_i$ that is also contained within this cone. Therefore, we may apply Theorem 14 to obtain this result for the second case as well. The proof of the last case is similar. \square

O Proof of Theorem 7

First we state the following simple result about OTB.

Theorem 15 (Theorem 8 in [Pedramfar and Aggarwal, 2024a]). *If \mathcal{A} is an online algorithm that queries no more than $K = T^\theta$ times per time-step and obtains an α -regret bound of $O(T^\delta)$, then the sample complexity of $\text{OTB}(\mathcal{A})$ is $\Omega(\epsilon^{-\frac{1+\theta}{1-\delta}})$.*

Proof of Theorem 7. This is an immediate corollary of Theorem 5 and the guarantees for the OTB meta-algorithm stated in Theorem 15. \square

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: **[Yes]** Justification: The abstract and introduction accurately reflect the contributions and scope of the paper.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: **[Yes]**

Justification: The paper includes discussion of limitations.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: **[Yes]**

Justification: Assumptions and complete proofs are included.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [NA]

Justification: There are no experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: There are no experiments.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [NA]

Justification: There are no experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [NA]

Justification: There are no experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer “Yes” if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [NA]

Justification: There are no experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: This paper preserves anonymity and follows the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: The paper focuses on the technical contributions and does not include a specific discussion of broader positive or negative societal impacts.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper poses no such risks as no usage of pretrained models, image generators, or scraped datasets in the paper.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [NA]

Justification: The paper does not use the existing assets.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This paper does not involve research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorosity, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.