Faster Projection-free Online Learning

Elad Hazan EHAZAN@CS.PRINCETON.EDU
Princeton University, Google AI Princeton

Edgar Minasyan MINASYAN@PRINCETON.EDU
Princeton University

Editors: Jacob Abernethy and Shivani Agarwal

Abstract

In many online learning problems the computational bottleneck for gradient-based methods is the projection operation. For this reason, in many problems the most efficient algorithms are based on the Frank-Wolfe method, which replaces projections by linear optimization. In the general case, however, online projection-free methods require more iterations than projection-based methods: the best known regret bound scales as $T^{3/4}$. Despite significant work on various variants of the Frank-Wolfe method, this bound has remained unchanged for a decade.

In this paper we give an efficient projection-free algorithm that guarantees $T^{2/3}$ regret for general online convex optimization with smooth cost functions and one linear optimization computation per iteration. As opposed to previous Frank-Wolfe approaches, our algorithm is derived using the Follow-the-Perturbed-Leader method and is analyzed using an online primal-dual framework.

Keywords: Online optimization, Frank-Wolfe method, Follow-the-Perturbed-Leader

1. Introduction

In many machine learning problems the decision set is high dimensional or otherwise complex such that even convex optimization over the set is not practical. Such is the case, for example, in matrix learning problems: performing matrix decomposition for very large problems is computationally intensive and super-linear in the sparsity of the input. This renders common algorithms such as projected gradient descent infeasible.

An alternative methodology which has proven successful in several applications is projection-free online learning. In this model, the access of the learner to the decision set is via a *linear optimization* oracle, as opposed to general convex optimization. As an example, linear optimization over matrices amounts to eigenvector computations, which can be carried out in time proportional to the sparsity of the matrices.

We henceforth consider online algorithms that perform one (or more generally a constant number) linear optimization and/or gradient evaluation per iteration. The reason is that if we do not restrict the number of linear optimizations, we can compute projections up to arbitrary precision and run standard projected online gradient descent which attains optimal regret. We restrict the number of gradient oracle calls since otherwise, in the stochastic setting, one can evaluate the real gradient up to arbitrary precision and run standard offline Frank-Wolfe which attains optimal convergence rate. This defeats the purpose of creating efficient algorithms.

Definition 1 *The* **oracle complexity** *of a projection-free algorithm is the number of linear optimizations and gradient evaluations per iteration on average. We say that a projection-free algorithm is* **oracle-efficient** *if its oracle complexity is constant.*

Oracle-efficient projection-free methods can have a considerable running time advantage for certain structured problems in which linear optimization is more efficient than projections. This has spurred significant research in recent years on projection-free methods in general and the Frank-Wolfe algorithm in particular. However, despite a decade-long search, the best known oracle-efficient projection-free online algorithm attains a regret bound that scales as $T^{3/4}$, where T is the number of iterations [1]. This method, the Online Frank-Wolfe (OFW) algorithm (Hazan and Kale, 2012), attains the lowest oracle complexity over all iterations, even if we include projection-free methods that are not oracle-efficient.

The $T^{3/4}$ bound is particularly striking when compared to stochastic projection-free optimization. In this setting, it is possible to obtain $T^{2/3}$ regret for smooth stochastic projection-free optimization by the so-called blocking technique: grouping several game iterations into one and thereby changing the decision less often, and its variants (Merhav et al., 2002; Chen et al., 2018). The optimal rate of $\sqrt{T}$ is more challenging to obtain as given in Lan and Zhou (2016). We are unaware of an improvement to the $T^{3/4}$ rate in the stochastic non-smooth case.

1.1. Our Results

Our main result is an efficient randomized algorithm that improves the state-of-the-art in general projection-free online optimization with smooth loss functions. The expected regret of this algorithm scales as $T^{2/3}$, with only one linear optimization computation per iteration. We then extend the analysis of this algorithm to show that it attains the same regret bound with high probability. Our main results are summarized by the informal statement below, with the exact dependence on smoothness and other relevant problem parameters detailed in later sections.

Theorem 2 *There exists an efficient algorithm for online convex optimization (see Algorithm 2) with smooth loss functions that is projection-free, performs only one linear optimization computation per iteration, and guarantees an expected regret of $\mathcal{O}(T^{2/3})$. Furthermore, the algorithm guarantees a regret of $\tilde{\mathcal{O}}(T^{2/3} \log \frac{1}{\sigma})$ with probability at least $1 - \sigma$.*

Techniques. Our algorithm is not based on the Frank-Wolfe method, but rather a version of the Follow-the-Perturbed-Leader (FPL) method (Kalai and Vempala, 2005). It was already established in Hazan et al. (2016) that a deterministic version of FPL works for online convex optimization. This version computes the expected point FPL plays at every iteration. In order to convert this algorithm to an efficient projection-free method, two main challenges arise:

1. Estimating the expectation by sampling FPL points via linear optimization creates time dependence between iterations, since the gradient is taken at a point which depends on all previous iterations. This means that a small error in one iteration potentially propagates to all future iterations.

[1] We omit the $\mathcal{O}$-notation in the introduction to make the exposition cleaner. In this case, $T^{3/4}$ hides constants including the norm of the gradients, diameter of the decision set, and more. See Hazan et al. (2016) for more details.

2. The number of linear optimization evaluations to estimate the mean of the FPL algorithm up to ε accuracy is $\mathcal{O}(\frac{1}{\varepsilon^2})$. This dependence is not sufficient to improve the previously best $T^{3/4}$ regret bound with only constantly many linear optimization computations per iteration.

To overcome the above issues we require two tools, that are new to the analysis of randomized online methods. First, we use the online primal-dual methodology of Shalev-Shwartz and Singer (2007a). This allows us to avoid the error-propagation caused by random estimation of the mean, and could be a technique of independent interest.

The second tool is using the smoothness of the loss functions to leverage not only the estimation proximity but also the fact that the estimation is unbiased. This is executed by switching gradients at nearby points which are not too far off due to the Lipschitz property of the gradients of smooth functions.

Paper structure. In the next subsection we discuss related work, and then move to describe preliminaries, including tools necessary for the online primal-dual analysis framework. In section 3, we state the main algorithm and formally state our main theorems including precise constants. In section 4 we state the deterministic FPL algorithm and analyze it using the primal-dual framework to illustrate its versatility in handling error propagation. We then use unbiased estimation and smoothness in section 5 to derive the first main theorem. In section 5.1, we provide a reduction of the algorithm to the setting of one linear optimization step per iteration. High probability bounds are given in detail in section 5.2 and derived in the appendix along with other miscellaneous proofs.

1.2. Related Work

In recent years the projection-free learning and optimization literature has seen a resurgence of results. We separate the related work into the broad categories below.

Projection-free offline optimization. The starting point for our line of work is the seminal paper of Frank and Wolfe (1956), who apply the conditional gradient method for smooth optimization over polyhedral sets. This was extended to semi-definite programming in Hazan (2008), and to general convex optimization in Jaggi (2013). This algorithm requires $\frac{\beta}{\varepsilon}$ linear optimization steps to find an ε-approximate solution for a β-smooth function, optimal with no other assumptions.

A significant advancement in projection-free methods was obtained by Garber and Hazan (2013), who give an algorithm that requires only $\log\frac{1}{\varepsilon}$ linear optimization steps for strongly convex and smooth functions over polyhedral sets. Data-dependent bounds for the spectahedron were obtained by Garber (2016); Allen-Zhu et al. (2017).

Projection-free optimization on non-smooth objective functions is typically performed via various smoothing techniques. The optimal complexity of linear optimization calls in this case is $\frac{1}{\varepsilon^2}$ (Lan, 2013). Several algorithms attain nearly optimal rates as in Lan (2013); Argyriou et al. (2014); Pierucci et al. (2014).

Projection-free online learning. The online variant of the Frank-Wolfe algorithm that applies to general online convex optimization was given in Hazan and Kale (2012). This method attains $T^{3/4}$ regret for the general OCO setting, with only one linear optimization step per iteration [2].

[2] If arbitrarily many linear optimization steps are allowed, the projections can be computed and this regret can be improved to optimal $\sqrt{T}$.

For OCO over polyhedral sets, an implication of the result of Garber and Hazan (2013) is an efficient $\sqrt{T}$ regret algorithm with only one linear optimization step per iteration, as well as $\log T$ regret for strongly convex online optimization. Recently Levy and Krause (2019) give an efficient projection-free online learning algorithm for smooth sets that devises a new fast projection operation for such sets and attains the optimal $\sqrt{T}$ regret for convex and $\log T$ regret for strongly convex online optimization.

Without further assumptions, the OFW method in Hazan and Kale (2012) attains the best known bounds for general online convex optimization. To the best of our knowledge, our $T^{2/3}$ regret is the first to improve in this general OCO setting for smooth functions.

Projection-free stochastic optimization. An important application of projection-free optimization is in the context of supervised learning and the optimization problem of empirical risk minimization. In this setting, there are more techniques that can be applied to further accelerate optimization, as compared to the online setting, most notably variance reduction.

This requires more careful accounting of the actual operations that the algorithms perform, including counting the number of full-gradient computations, stochastic gradient evaluations, linear optimizations, and projections. There have been a multitude of algorithms suggested that attain various tradeoffs of the various computations, and have different merits/caveats. The reader is referred to the vast literature on stochastic projection-free methods, including the recent papers of Lan and Zhou (2016); Hazan and Luo (2016); Chen et al. (2018); Hassani et al. (2019); Xie et al. (2019); Yurtsever et al. (2019); Zhang et al. (2019).

2. Problem Setting

We consider the online convex optimization framework as an iterative game between a player and an adversary. At each iteration $t \in \mathbb{N}$, the player chooses an action $\mathbf{x}_t$ from the constraint set $\mathcal{K} \subset \mathbb{R}^d$ of permissible actions while the adversary simultaneously chooses a loss function $f_t : \mathcal{K} \to \mathbb{R}$ that determines the loss the player will occur for the action $\mathbf{x}_t$. The performance metric for such settings is the notion of regret – the difference between the cumulative loss suffered throughout T iterations of the online game and the overall loss for the best fixed action in hindsight:

$$\mathcal{R}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}) \,. \qquad (1)$$

For a given online algorithm $\mathcal{A}$, we denote $\mathcal{R}_T(\mathcal{A})$ to be its regret after T iterations. In this work, the adversary has no computational or information restrictions as long as it chooses f_t simultaneously to the player choosing $\mathbf{x}_t$, i.e. we operate in the *adaptive adversarial* setting.

Before proceeding to the main results, we formalize several notations and assumptions preserved throughout the paper. We discuss our additional/modified assumptions, made for simplicity in the analysis, without touching upon the conventional standards established in the community (e.g. see Hazan et al. (2016)). Throughout this work $\|\cdot\|$ refers to the Euclidean norm, $a_{l:k}$ refers to the sum $a_{l:k} = \sum_{i=l}^{k} a_i$, and $\mathbb{B} = \{v \in \mathbb{R}^d, \; \|v\| \;\le\; 1\}$ denotes the unit ball. The notation $O(\cdot)$ hides absolute constants, and $\mathcal{O}(\cdot)$ hides all irrelevant constants, e.g. everything other than T.

Assumption 3 *The constrained action set $\mathcal{K} \subset \mathbb{R}^d$ is a convex and compact set. Moreover, all the points in the set have bounded norms, i.e. $\|\mathbf{x}\| \;\le\; D$, $\forall \mathbf{x} \in \mathcal{K}$.*

Assumption 4 *For each iteration $t \in \mathbb{N}$, the loss functions $f_t : \mathcal{K} \to \mathbb{R}$ are convex, differentiable and have bounded gradient norms $\forall \mathbf{x} \in \mathcal{K},\ \|\nabla f_t(\mathbf{x})\| \ \le\ G$.*

The convention in OCO is to simply assume a bounded diameter for the set $\mathcal{K}$ instead of the norm bound. It is, however, straightforward to derive the above formulation: given an arbitrary point in the set $\mathbf{x}' \in \mathcal{K}'$ consider the shifted set $\mathcal{K} = \{\mathbf{x} - \mathbf{x}' \,|\, \mathbf{x} \in \mathcal{K}\}$ whose diameter bound implies the bounded norms of the points in $\mathcal{K}$; properties such as convexity and compactness are preserved through shifts. The convexity and bounded gradient norm assumptions for the loss functions are part of the standard throughout literature, while differentiability of the losses is assumed for simplicity and can be avoided by using subgradients instead (see Shalev-Shwartz and Singer (2007b)).

Definition 5 *The Fenchel dual of a function $f : \mathcal{K} \to \mathbb{R}$ with domain $\mathcal{K} \subset \mathbb{R}^d$ is defined as*

$$\forall \mathbf{y} \in \mathbb{R}^d, \quad f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \mathcal{K}} \{\langle \mathbf{y}, \mathbf{x} \rangle - f(\mathbf{x})\}\ . \tag{2}$$

The concept of a Fenchel dual will be crucial in the online primal-dual analysis of the presented algorithms. If the function f is convex, then its Fenchel dual f^* is also convex, and the Fenchel-Moreau theorem gives biconjugacy, i.e. the dual of a dual is equal to the function itself $(f^*)^* = f$. In this case, it is essential to note that $f(\mathbf{x}) = \sup_{\mathbf{y} \in \mathbb{R}^d} \{\langle \mathbf{x}, \mathbf{y} \rangle - f^*(\mathbf{y})\}$ directly implies that $\nabla f(\mathbf{x}) = \arg\max_{\mathbf{y} \in \mathbb{R}^d} \{\langle \mathbf{x}, \mathbf{y} \rangle - f^*(\mathbf{y})\}$, which is well-defined, when f is differentiable.

Linear Optimization Oracle. A linear optimization oracle, along with a value oracle, over the constraint set $\mathcal{K}$ is provided to the player, defined as

$$\forall \mathbf{y} \in \mathbb{R}^d, \quad \mathcal{O}_{\mathcal{K}}(\mathbf{y}) = \arg\max_{\mathbf{x} \in \mathcal{K}} \langle \mathbf{y}, \mathbf{x} \rangle, \quad \mathcal{M}_{\mathcal{K}}(\mathbf{y}) = \max_{\mathbf{x} \in \mathcal{K}} \langle y, x \rangle\ . \tag{3}$$

The reliance on linear optimization is the key motivation of the paper. This work concerns itself with the special case of online convex constrained optimization where the operation of projection to the set $\mathcal{K}$, as a problem of quadratic optimization, has a significantly higher computational cost than the linear optimization. In such cases, the use of the projected Online Gradient Descent (OGD) (Zinkevich, 2003) that achieves an optimal regret bound $\mathcal{O}(\sqrt{T})$ with respect to T is not always preferred to methods that bypass projection and use linear optimization instead. It is worth to notice that the existence of only $\mathcal{O}_{\mathcal{K}}(\cdot)$ is enough since $\mathcal{M}_{\mathcal{K}}(\mathbf{y}) = \langle \mathbf{y}, \mathcal{O}_{\mathcal{K}}(\mathbf{y}) \rangle$ and $\nabla \mathcal{M}_{\mathcal{K}}(\mathbf{y}) = \mathcal{O}_{\mathcal{K}}(\mathbf{y})$. Moreover, the function $\mathcal{M}_{\mathcal{K}}(\cdot)$ is convex and Lipschitz as suggested below.

Lemma 6 *The linear value oracle $\mathcal{M}_{\mathcal{K}} : \mathbb{R}^d \to \mathbb{R}$ is convex and D-Lipschitz, i.e.*

$$\forall \mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^d, \quad |\mathcal{M}_{\mathcal{K}}(\mathbf{y}_1) - \mathcal{M}_{\mathcal{K}}(\mathbf{y}_2)| \ \le\ D\|\mathbf{y}_1 - \mathbf{y}_2\|\ . \tag{4}$$

3. Algorithm and Main Theorem

The algorithm we propose is fairly straightforward, and the main hurdle lies in the analysis. The seminal work of Kalai and Vempala (2005) introduces the Follow-the-Perturbed-Leader (FPL) online algorithm that obtains optimal $\mathcal{O}(\sqrt{T})$ regret for *linear* loss functions. A more general version of FPL that applies expectations over the perturbations at each iteration extends the result to general convex functions (Hazan et al., 2016). Our algorithm mimics the expected FPL replacing the computationally expensive expectations with empirical averages of i.i.d. samples. It is presented in detail in Algorithm 1. The following theorem states the convergence guarantees for both general convex and smooth convex loss functions.

Algorithm 1 Sampled Follow-the-Perturbed-Leader Algorithm, $\mathcal{A}_1$

```
  **Input:** constraint set $\mathcal{K}$, number of rounds $T$, perturbation parameter $\delta$, number of samples $m$, linear optimization oracle $\mathcal{O}_{\mathcal{K}}(\cdot)$
  **for** $t = 1$ **to** $T$ **do**
    sample $\mathbf{v}_t^j \sim \mathbb{B}$ uniformly for $j = 1, \dots, m$
    denote $\mathbf{x}_t^j = \mathcal{O}_{\mathcal{K}}(-\tilde{\nabla}_{1:t-1} + \frac{1}{\delta} \cdot \mathbf{v}_t^j)$ for $j = 1, \dots, m$
    play $\tilde{\mathbf{x}}_t = \frac{1}{m} \sum_{j=1}^m \mathbf{x}_t^j$
    observe $f_t$, denote $\tilde{\nabla}_t = \nabla f_t(\tilde{\mathbf{x}}_t)$
  **end for**
```

Theorem 7 *Given that the Assumptions 3 and 4 hold, Algorithm 1, for general convex loss functions, obtains an expected regret of*

$$\mathbb{E}\left[\sum_{t=1}^{T} f_t(\tilde{\mathbf{x}}_t)\right] \le \min_{\mathbf{x}\in\mathcal{K}}\{\sum_{t=1}^{T} f_t(\mathbf{x})\} + 2D/\delta + \delta D G^2 \cdot dT/2 + \frac{2GDT}{\sqrt{m}} . \tag{5}$$

If the convex loss functions are also β-smooth then the expected regret bound becomes

$$\mathbb{E}\left[\sum_{t=1}^{T} f_t(\tilde{\mathbf{x}}_t)\right] \le \min_{\mathbf{x}\in\mathcal{K}}\{\sum_{t=1}^{T} f_t(\mathbf{x})\} + 2D/\delta + \delta D G^2 \cdot dT/2 + \frac{4\beta D^2 T}{m}, \tag{6}$$

where δ and $m \ge 1$ are parameters of the algorithm.

Algorithm 2 Online Smooth Projection Free (OSPF) Algorithm, $\mathcal{A}_{\text{OSPF}}$

```
  **Input:** constraint set $\mathcal{K}$, number of rounds $T$, perturbation parameter $\delta$, block size $k$, linear optimization oracle $\mathcal{O}_{\mathcal{K}}(\cdot)$
  pick arbitrary $\mathbf{x}_0 \in \mathcal{K}$, denote $\nabla_0 = \mathbf{0}$
  **for** $t = 1$ **to** $T$ **do**
    **if** $t \mod k \neq 0$ **then**
      play $\mathbf{x}_t = \mathbf{x}_{t-1}$
      observe $f_t$, denote $\nabla_t = \nabla f_t(\mathbf{x}_t)$
    **else**
      sample $\mathbf{v}_{t-k+j} \sim \mathbb{B}$ uniformly for $j = 1, \dots, k$
      denote $\mathbf{x}_t^j = \mathcal{O}_{\mathcal{K}}(-\nabla_{0:t-1} + \frac{1}{\delta} \cdot \mathbf{v}_{t-k+j})$ for $j = 1, \dots, k$
      play $\mathbf{x}_t = \frac{1}{k} \sum_{j=1}^k \mathbf{x}_t^j$
      observe $f_t$, denote $\nabla_t = \nabla f_t(\mathbf{x}_t)$
    **end if**
  **end for**
```

Remark 8 *Take the perturbation parameter to be $\delta = 2/(G\sqrt{dT})$. Then Algorithm 1 attains expected regret of $\mathbb{E}[\mathcal{R}_T(\mathcal{A}_1)] = \mathcal{O}(\sqrt{T})$ with $m = T$ over general convex losses and $\mathbb{E}[\mathcal{R}_T(\mathcal{A}_1)] = \mathcal{O}(\sqrt{T})$ with $m = \beta\sqrt{T}$ over smooth convex losses. In particular, this restores the original result of the FPL method attaining $\mathcal{O}(\sqrt{T})$ regret with $m = 1$ sample per iteration for linear, $\beta = 0$, loss functions shown in Kalai and Vempala (2005).*

Algorithm 1 does not adhere to our Definition 1 of oracle-efficient methods. Hence, we additionally provide a reduction based on a blocking technique (see section 5.1) that is oracle-efficient with the following regret guarantees.

Theorem 9 *Algorithm 2 as an oracle-efficient method suffers $O(GD\sqrt{d}T^{3/4})$ expected regret over general convex loss functions with blocking size $k = T^{1/2}$ and $\delta = 2/(G\sqrt{dT})$. In the case of smooth convex loss functions, Algorithm 2, named $\mathcal{A}_{OSPF}$, attains $O(D(G\sqrt{d}+\beta D)T^{2/3})$ expected regret with blocking size $k = T^{1/3}$ and $\delta = 2/(G\sqrt{dT})$.*

4. The Case of Unlimited Computation

In an ideal scenario the player would be given unrestrained computational power along with access to the linear optimization oracle. Then the expected FPL method, as a projection-free online algorithm, attains $\mathcal{O}(\sqrt{T})$ regret bound optimal with respect to T for general convex loss functions. The exact algorithm is spelled out in Algorithm 3. The original analysis follows the standard recipe of online learning literature coined by Kalai and Vempala (2005): no regret for Be-The-Leader – the player suffers no regret if it is hypothetically one step ahead of the adversary, i.e. uses $\mathbf{x}_{t+1}$ for the loss function f_t; stability – the predictions of consecutive rounds $\mathbf{x}_t, \mathbf{x}_{t+1}$ are not too far apart from each other (Hazan et al., 2016). We provide an alternative approach developed by Shalev-Shwartz and Singer (2007b) based on duality and crucial for the further analysis of Algorithm 1.

Algorithm 3 Expected Follow-the-Perturbed-Leader Algorithm, $\mathcal{A}_3$

```
Input: constraint set $\mathcal{K}$, number of rounds $T$, perturbation parameter $\delta$, linear optimization oracle $\mathcal{O}_{\mathcal{K}}(\cdot)$
for $t = 1$ to $T$ do
    compute $\mathbf{x}_t = \mathbb{E}_{\mathbf{v}\sim\mathbb{B}}[\mathcal{O}_{\mathcal{K}}(-\nabla_{1:t-1} + \frac{1}{\delta}\cdot\mathbf{v})]$
    play $\mathbf{x}_t$, observe $f_t$, denote $\nabla_t = \nabla f_t(\mathbf{x}_t)$
end for
```

Theorem 10 *Given that Assumptions 3 and 4 hold, Algorithm 3 with the perturbation parameter value $\delta = 2/(G\sqrt{dT})$ suffers regret $\mathcal{R}_T(\mathcal{A}_3) \leq 2GD\sqrt{dT} = \mathcal{O}(\sqrt{T})$.*

Proof The proof is based on duality when one considers the following optimization problem

$$\min_{\mathbf{x}\in\mathcal{K}}\{h_\delta(\mathbf{x}) + \sum_{t=1}^{T} f_t(\mathbf{x})\}, \tag{7}$$

which resembles the loss suffered by the best-in-hindsight fixed action. The dual objective, that is to be maximized, can be obtained using Lagrange multipliers and is given by (see Shalev-Shwartz and Singer (2007b) for details)

$$\mathcal{D}(\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_T) = -h_\delta^*(-\boldsymbol{\lambda}_{1:T}) - \sum_{t=1}^{T} f_t^*(\boldsymbol{\lambda}_t)\,. \tag{8}$$

The term $h_\delta(\cdot)$ serves as regularization and is defined implicitly through its Fenchel conjugate $h_\delta^*(\mathbf{y}) = \mathbb{E}_{\mathbf{v}\sim\mathbb{B}}[\mathcal{M}_{\mathcal{K}}(\mathbf{y} + \frac{1}{\delta}\cdot\mathbf{v})]$, a stochastic smoothing of the value oracle, that is δdD-smooth according to the following lemma and the fact that $\mathcal{M}_{\mathcal{K}}(\cdot)$ is D-Lipschitz.

Lemma 11 *The function $\hat{g}(\mathbf{y}) = \mathbb{E}_{\mathbf{v}\sim\mathbb{B}}[g(\mathbf{y} + \frac{1}{\delta}\cdot\mathbf{v})]$ is δdL-smooth given $g : \mathbb{R}^d \to \mathbb{R}$ is an L-Lipschitz function.*

Weak duality implies that the objective (8) is upper bounded by (7) for any values of $\boldsymbol{\lambda}_t, t = 1, \dots, T$ hence the goal is to upper bound the online cumulative loss by (8). To achieve this, take $\boldsymbol{\lambda}_t = \nabla_t = \nabla f_t(\mathbf{x}_t)$ for all $t \in [T]$ where the action $\mathbf{x}_t$ is chosen according to Algorithm 3. Denote the incremental difference as $\Delta_t = \mathcal{D}(\nabla_1, \dots, \nabla_t, \mathbf{0}, \dots, \mathbf{0}) - \mathcal{D}(\nabla_1, \dots, \nabla_{t-1}, \mathbf{0}, \dots, \mathbf{0})$ and notice that the dual can be written as $\mathcal{D}(\nabla_1, \dots, \nabla_T) = \sum_{t=1}^T \Delta_t + \mathcal{D}(\mathbf{0}, \dots, \mathbf{0})$. For each $t \in [T]$,

$$\begin{aligned}
\Delta_t &= -\left[h_\delta^*(-\nabla_{1:t}) - h_\delta^*(-\nabla_{1:t-1})\right] - f_t^*(\nabla_t) + f_t^*(\mathbf{0}) \;\geq && \text{smoothness of } h_\delta^*(\cdot)\\
&\geq \;\langle \nabla_t, \nabla h_\delta^*(-\nabla_{1:t-1})\rangle - \frac{\delta d D}{2}\|\nabla_t\|^2 - f_t^*(\nabla_t) + f_t^*(\mathbf{0}) = && \text{definition of } \mathbf{x}_t\\
&= \langle \nabla_t, \mathbf{x}_t\rangle - f_t^*(\nabla_t) - \frac{\delta d D}{2}\|\nabla_t\|^2 + f_t^*(\mathbf{0}) = f_t(\mathbf{x}_t) - \frac{\delta d D}{2}\|\nabla_t\|^2 + f_t^*(\mathbf{0}), && (9)
\end{aligned}$$

where we use the fact that the action $\mathbf{x}_t$ from Algorithm 3 can alternatively be expressed as $\mathbf{x}_t = \nabla h_\delta^*(-\nabla_{1:t-1})$ and the Fenchel dual identity $\langle \nabla_t, \mathbf{x}_t\rangle - f_t^*(\nabla_t) = f_t(\mathbf{x}_t)$ for convex $f_t(\cdot)$. The obtained inequality (9) quantifies how much regret an action contributes at a given iteration $t \in [T]$ detached from the rest of the rounds of the game. Such a property of the analysis ends up being crucial in showing the regret bounds further in this work. Note that by definition $\mathcal{D}(\mathbf{0}, \dots, \mathbf{0}) = -h_\delta^*(\mathbf{0}) - \sum_{t=1}^T f_t^*(\mathbf{0})$ which gives the identity $\sum_{t=1}^T \Delta_t - \sum_{t=1}^T f_t^*(\mathbf{0}) = \mathcal{D}(\nabla_1, \dots, \nabla_T) + h_\delta^*(\mathbf{0})$. Thus, sum up (9) for all $t = 1, \dots, T$ to bound the online cumulative loss is by

$$\sum_{t=1}^T f_t(\mathbf{x}_t) \;\leq\; \mathcal{D}(\nabla_1, \dots, \nabla_T) + h_\delta^*(\mathbf{0}) + \frac{\delta d D}{2}\sum_{t=1}^T \|\nabla_t\|^2 \,. \tag{10}$$

The bound in (10) along with weak duality are enough to conclude the $\mathcal{O}(\sqrt{T})$ regret bound. All that is left are technical details to reach the conclusion using the assumptions of the given setup. First, since $\mathcal{M}_\mathcal{K}(\mathbf{0}) = 0$ it implies, by Lipschitzness of $\mathcal{M}_\mathcal{K}(\cdot)$, that $|\mathcal{M}_\mathcal{K}(\frac{1}{\delta}\cdot\mathbf{v})| \;\leq\; D\|\mathbf{v}\|/\delta \;\leq\; D/\delta$ for any $\mathbf{v} \in \mathbb{B}$ so $h_\delta^*(\mathbf{0}) \;\leq\; D/\delta$. Second, the primal expression in (7) can be related to the best loss in hindsight the following way

$$\min_{\mathbf{x}\in\mathcal{K}}\{h_\delta(\mathbf{x}) + \sum_{t=1}^T f_t(\mathbf{x})\} \;\leq\; h_\delta(\mathbf{x}^*) + \sum_{t=1}^T f_t(\mathbf{x}^*) \;\leq\; \min_{\mathbf{x}\in\mathcal{K}}\sum_{t=1}^T f_t(\mathbf{x}) + \max_{\mathbf{x}\in\mathcal{K}} h_\delta(\mathbf{x}), \tag{11}$$

where $\mathbf{x}^*$ is the optimal action in hindsight, i.e. the minimizer of $\sum_{t=1}^T f_t(\cdot)$ over $\mathcal{K}$. Moreover, notice that for any $\mathbf{x} \in \mathcal{K}, \mathbf{y} \in \mathbb{R}^d$ the expression $\langle \mathbf{x}, \mathbf{y}\rangle - h_\delta^*(\mathbf{y}) = \mathbb{E}_{\mathbf{v}\sim\mathbb{B}}[\langle \mathbf{x}, \mathbf{y}\rangle - \max_{\mathbf{x}'\in\mathcal{K}}\langle \mathbf{x}', \mathbf{y} + \frac{1}{\delta}\cdot\mathbf{v}\rangle]$ can be bounded as follows: for each $\mathbf{v} \in \mathbb{B}$ the expression inside the expectation is bounded $\langle \mathbf{x}, \mathbf{y}\rangle - \max_{\mathbf{x}'\in\mathcal{K}}\langle \mathbf{x}', \mathbf{y} + \frac{1}{\delta}\cdot\mathbf{v}\rangle \;\leq\; \langle \mathbf{x}, \mathbf{y}\rangle - \langle \mathbf{x}, \mathbf{y} + \frac{1}{\delta}\cdot\mathbf{v}\rangle = \langle \mathbf{x}, -\frac{1}{\delta}\cdot\mathbf{v}\rangle \;\leq\; \|\mathbf{x}\|\|\mathbf{v}\|/\delta \;\leq\; D/\delta$, hence for any $\mathbf{x} \in \mathcal{K}$ the bound $h_\delta(\mathbf{x}) \;\leq\; D/\delta$ holds. Finally, according to our assumptions the loss gradients are bounded in norm, i.e. $\|\nabla_t\| \;\leq\; G$. Combining these properties with (10) along with the fact that $\mathcal{D}(\nabla_1, \dots, \nabla_T)$ is upper bounded by (7) due to weak duality, we conclude

$$\sum_{t=1}^T f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{K}}\sum_{t=1}^T f_t(\mathbf{x}) \;\leq\; \max_{\mathbf{x}\in\mathcal{K}} h_\delta(\mathbf{x}) + h_\delta^*(\mathbf{0}) + \frac{\delta d D}{2}\sum_{t=1}^T \|\nabla_t\|^2 \;\leq\; 2D/\delta + \frac{\delta d D}{2}G^2 T \,.$$

This yields the regret bound $\mathcal{R}_T(\mathcal{A}_3) \leq 2GD\sqrt{dT} = \mathcal{O}(\sqrt{T})$ with the perturbation parameter value $\delta = 2/(G\sqrt{dT})$. ■

Remark 12 *It is essential to note how each property given in Assumptions 3 and 4 was used in the proof above. The convexity of the constraint set $\mathcal{K}$ allows the action $\mathbf{x}_t$, as an expectation of points in the set $\mathcal{K}$, to be a permissible action as well. Given compactness of $\mathcal{K}$, we interchange the use of supremum and maximum of bounded expressions at various points throughout. The norm bound D of the set $\mathcal{K}$ is used in showing that $\mathcal{M}_{\mathcal{K}}(\cdot)$ is D-Lipschitz and bounding several regularization terms. In terms of the loss functions, the convexity of $f_t(\cdot)$, as well as $\mathcal{M}_{\mathcal{K}}(\cdot)$, allows us to use the Fenchel-Moreau theorem (continuity is implied by differentiability) while the gradient norm bound is simply used in the last stage of obtaining the regret bound.*

5. Oracle Efficiency via Estimation

The results in section 4 suggest that Algorithm 3 possesses the features desired in this work – it is both online and projection-free – and obtains an optimal regret bound of $\mathcal{O}(\sqrt{T})$. However, it is computationally intractable due to the expectation term given in the definition of the action $\mathbf{x}_t$. In this section, we remedy this issue and explore the scenario where the actions played during the game are random estimators of the mean. In particular, we propose to simply take the empirical average of m i.i.d. samples instead of the expectation itself as described in Algorithm 1.

It is essential to note that Algorithm 1 has a computational efficiency of $m \cdot T$ calls to the linear optimization oracle $\mathcal{O}_{\mathcal{K}}(\cdot)$ as the rest of the computation is negligible in comparison. Given the duality approach to analyzing online algorithms demonstrated in the previous section, the sampled FPL algorithm can now be analyzed to prove the bounds stated in Theorem 7. In particular, the following lemma demonstrates that each estimation from Algorithm 1 contributes to the regret in a disjoint fashion, i.e. there is no error propagation through time.

Lemma 13 *Suppose the Assumptions 3 and 4 hold and denote $\hat{\mathbf{x}}_t = \mathbb{E}_{\mathbf{v}\sim\mathbb{B}}\left[\mathcal{O}_{\mathcal{K}}(-\tilde{\nabla}_{1:t-1} + \frac{1}{\delta}\cdot\mathbf{v})\right]$ for all $t \in [T]$ with $\tilde{\mathbf{x}}_t$ and $\tilde{\nabla}_t$ as defined in Algorithm 1. Then, the regret is bounded as follows*

$$\sum_{t=1}^{T} f_t(\tilde{\mathbf{x}}_t) \leq \min_{\mathbf{x}\in\mathcal{K}}\{\sum_{t=1}^{T} f_t(\mathbf{x})\} + 2GD\sqrt{dT} + \sum_{t=1}^{T}\langle\tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\rangle \ . \tag{12}$$

Proof Follow the same proof structure as in the proof of Theorem 10 by considering (7), (8) as the primal and dual objectives. Consider $\boldsymbol{\lambda}_t = \tilde{\nabla}_t = \nabla f_t(\tilde{\mathbf{x}}_t)$ and denote the incremental difference as $\Delta_t = \mathcal{D}(\tilde{\nabla}_1, \ldots, \tilde{\nabla}_t, \mathbf{0}, \ldots, \mathbf{0}) - \mathcal{D}(\tilde{\nabla}_1, \ldots, \tilde{\nabla}_{t-1}, \mathbf{0}, \ldots, \mathbf{0})$. The main component of the proof is showing that Δ_t can be roughly seen as an upper bound on the loss $f_t(\tilde{\mathbf{x}}_t)$ suffered at iteration t.

First note that the played actions $\tilde{\mathbf{x}}_t$ are not, in fact, unbiased estimators of the original $\mathbf{x}_t$; instead denote the expectations by $\hat{\mathbf{x}}_t = \mathbb{E}_{\xi_t}[\tilde{\mathbf{x}}_t] = \mathbb{E}_{\mathbf{v}_t^j}[\mathbf{x}_t^j]$ where $\xi_t = \{\mathbf{v}_t^1, \ldots, \mathbf{v}_t^m\}$ comprises the randomness used at iteration $t \in [T]$. For all $t > 1$, the quantity $\hat{\mathbf{x}}_t$ is different from $\mathbf{x}_t$ in that it uses the gradients at the points $\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_{t-1}$ instead of $\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}$ and such difference can potentially increase with t. In other words, one is defined as $\hat{\mathbf{x}}_t = \nabla h_\delta^*(-\tilde{\nabla}_{1:t-1})$ while the other is equal to $\mathbf{x}_t = \nabla h_\delta^*(-\nabla_{1:t-1})$. Hence, the action sequences of $\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_T$ and $\mathbf{x}_1, \ldots, \mathbf{x}_T$ can behave quite differently and one cannot analyze the former based on results about the latter. However, the duality approach enables us to directly analyze the actions of Algorithm 1. In particular, lower bound the quantity Δ_t using the smoothness of $h_\delta^*(\cdot)$, as done in the proof of Theorem 10:

$$\Delta_t \ \geq\ \langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t \rangle - \frac{\delta d D}{2}\|\tilde{\nabla}_t\|^2 - f_t^*(\tilde{\nabla}_t) + f_t^*(\mathbf{0}) = f_t(\tilde{\mathbf{x}}_t) - \frac{\delta d D}{2}\|\tilde{\nabla}_t\|^2 + f_t^*(\mathbf{0}) + \langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle\,. \quad (13)$$

The obtained inequality (13) resembles the analogous bound (9) in the unlimited computation case with the extra term $\langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle$ that can be seen as accounting for the estimation error. This shows that at a given iteration $t \in [T]$ the additional regret is suffered *only* at the expense of the current action choice, $\tilde{\mathbf{x}}_t$ instead of $\hat{\mathbf{x}}_t$, while *ignoring* the optimality of the previous choices $\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_{t-1}$. We proceed with the proof by summing up (13) for all $t = 1, \ldots, T$ and using the following facts: by definition $\sum_{t=1}^T \Delta_t = \mathcal{D}(\tilde{\nabla}_1, \ldots, \tilde{\nabla}_T) - \mathcal{D}(\mathbf{0}, \ldots, \mathbf{0})$ and $\mathcal{D}(\mathbf{0}, \ldots, \mathbf{0}) = -h_\delta^*(\mathbf{0}) - \sum_{t=1}^T f_t^*(\mathbf{0})$; as shown before $h_\delta^*(\mathbf{0}) \ \leq\ D/\delta$ and $\forall \mathbf{x} \in \mathcal{K}, h_\delta^*(\mathbf{x}) \ \leq\ D/\delta$; according to Assumption 4, for all $t \in [T], \forall \mathbf{x} \in \mathcal{K}, \|\nabla f_t(\mathbf{x})\| \ \leq\ G$. Combining all these properties and choosing the same optimal value of the regularization parameter $\delta = 2/(G\sqrt{dT})$ concludes the stated bound (12). The use of all the assumptions is identical to that of Theorem 10 and detailed in Remark 12. ■

All that remains to reach the conclusions by Theorem 7 is to use Lemma 13 and handle the additional regret terms $\langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle$ for each $t \in [T]$. The following claims about smooth functions and empirical averages of random vectors are necessary for the latter part.

Lemma 14 *If $f : \mathcal{K} \to \mathbb{R}$ is a β-smooth function, then the following bound holds for any $\mathbf{x}, \mathbf{y} \in \mathcal{K}$, $\langle \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \ \leq\ \langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y} \rangle + \beta\|\mathbf{x} - \mathbf{y}\|^2$.*

Lemma 15 *Let $Z_1, \ldots, Z_m \sim \mathcal{Z}$ be i.i.d. samples of a bounded random vector $Z \in \mathbb{R}^d$, $\|Z\| \ \leq\ D$, with mean $\overline{Z} = \mathbb{E}[Z]$. Denote $\overline{Z}_m = \frac{1}{m}\sum_{j=1}^m Z_j$, then $\mathbb{E}_{\mathcal{Z}}\left[\|\overline{Z}_m - \overline{Z}\|^2\right] \ \leq\ \frac{4D^2}{m}$.*

Proof [Proof of Theorem 7] First, note that according to Lemma 15, the following bound $\mathbb{E}_{\xi_t}[\|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\|] \ \leq\ \sqrt{\mathbb{E}_{\xi_t}[\|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\|^2]} \ \leq\ \frac{2D}{\sqrt{m}}$ holds for all $t \in [T]$. In the case of general convex loss functions, use the Cauchy-Schwartz inequality along with the norm bound on the gradients and take expectation over the whole randomness in the algorithm $\xi_{1:T}$ in the reverse order $\xi_T, \ldots, \xi_1$ to obtain for each $t \in [T]$

$$\mathbb{E}_{\xi_{1:T}}[\langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle] \ \leq\ G\mathbb{E}_{\xi_{1:t}}[\|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\|] = G\mathbb{E}_{\xi_{1:t-1}}\left[\mathbb{E}_{\xi_t}[\|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\| \mid \xi_{1:t-1}]\right] \ \leq\ \frac{2DG}{\sqrt{m}} \quad (14)$$

Ordering the randomness of the iterations in reverse and taking the expectation conditional on $\xi_{1:t-1}$ is necessary in order to use Lemma 15 since $\hat{\mathbf{x}}_t$ is a deterministic quantity over ξ_t only when conditioned on the previous randomness $\xi_{1:t-1}$. Finally, taking expectation over $\xi_{1:T}$ on the bound in (12) and using (14) for all $t = 1, \ldots, T$ concludes the expected regret bound of $\mathbb{E}[\mathcal{R}_T(\mathcal{A}_1)] \ \leq\ 2GD\sqrt{dT} + \frac{2DGT}{\sqrt{m}}$ given in detail in (5) for general convex loss functions. It is worth to mention that this result did not require any assumptions on how the loss function $f_t(\cdot)$ at each iteration $t \in [T]$ is chosen by the adversary: in particular, the result holds for the strongest *adaptive adversarial* setting where the adversary can pick $f_t(\cdot)$ having knowledge of the previous actions by the player, i.e. the randomness $\xi_{1:t-1}$. This is true due to the fact that all the terms containing the function $f_t(\cdot)$ explicitly, e.g. $\tilde{\nabla}_t$, are separated and bound on their own.

In the case of smooth convex loss functions, a more nuanced approach is taken to achieve an improvement on the general result. The key is to replace the gradient $\tilde{\nabla}_t$ at the point $\tilde{\mathbf{x}}_t$ with a quantity independent of ξ_t and leverage the fact that $\tilde{\mathbf{x}}_t$ is an unbiased estimator of $\hat{\mathbf{x}}_t$. Formally, given $f_t(\cdot)$ is β-smooth note that $\langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle \;\leq\; \langle \hat{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle + \beta \|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\|^2$ according to Lemma 14 where $\hat{\nabla}_t \;=\; \nabla f_t(\hat{\mathbf{x}}_t)$ is denoted accordingly. The quantities $f_t(\cdot)$ and $\hat{\mathbf{x}}_t$ both (potentially) depend on previous randomness $\xi_{1:t-1}$ but are deterministic with respect to ξ_t when conditioned on $\xi_{1:t-1}$, hence so is $\hat{\nabla}_t$. Thus, it holds that $\mathbb{E}_{\xi_t}[\langle \hat{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle \,|\, \xi_{1:t-1}]] = 0$. This fact results in the additional regret having a quadratic dependence on the estimation error instead of linear as before:

$$\mathbb{E}_{\xi_{1:T}}[\langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle] \;\leq\; \mathbb{E}_{\xi_{1:t-1}}\left[\mathbb{E}_{\xi_t}[\langle \hat{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle + \beta \|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\|^2 \,|\, \xi_{1:t-1}]\right] \;\leq\; \frac{4\beta D^2}{m}\,. \tag{15}$$

Use the inequality in (15) for all $t \in [T]$ in order to bound the additional regret term in (12) and conclude the expected regret bound of $\mathbb{E}[\mathcal{R}_T(\mathcal{A}_1)] \;\leq\; 2GD\sqrt{dT} + \frac{4\beta D^2 T}{m}$ given in detail in (6) for smooth convex loss functions. Analogous to the general convex case, this regret bound holds in the strongest *adaptive adversarial* setting. ■

5.1. Reduction to OSPF

The results given in Theorem 7 indicate $\mathcal{O}(\sqrt{T})$ optimal regret bounds for both convex and smooth convex loss functions when taking $m = T$, and $m = \beta\sqrt{T}$ respectively, as suggested by Remark 8. However, m is not simply a parameter of the algorithm: it indicates the number of linear optimizations per iteration so in T iterations the regret $\mathcal{O}(\sqrt{T})$ is achieved with an overall linear optimization complexity of $m \cdot T$. To avoid such convoluted claims, we instead provide a reduction of Algorithm 1, named OSPF in the smooth case, to the setting of one linear optimization per iteration that gives $\mathcal{O}(T^{2/3})$ and $\mathcal{O}(T^{3/4})$ expected regret for smooth and general convex losses, respectively.

Proof [Proof of Theorem 9] The reduction follows a simple blocking technique, i.e. grouping several rounds of the game into one as detailed in Algorithm 2. Let $T = nk$ where n, k are assumed to be integers for simplicity and denote $f'_i = \sum_{t=(i-1)\cdot k+1}^{i\cdot k} f_t$, for all $i \in [n]$. Since each $f'_i, i \in [n]$ contains k losses from the original problem, then the player is allowed k linear optimizations to handle a single loss f'_i. Hence, use Algorithm 1 for $T = n$ iterations with $m = k$ samples at each iteration to get actions $\mathbf{x}'_1, \ldots, \mathbf{x}'_n$ and play $\mathbf{x}_t = \mathbf{x}'_i$ for all $(i-1)\cdot k + 1 \;\leq\; t \;\leq\; i \cdot k$ in the original setting – call this algorithm $\mathcal{A}'_1$. The corresponding constants of the constructed game are $D' = D$ and $G' = G \cdot k$ since a new loss function constitutes k original losses together. Thus, the expected regret bound of $\mathcal{A}'_1$ for general convex functions, according to Theorem 7, is given by

$$\mathbb{E}[\mathcal{R}_T(\mathcal{A}'_1)] \;\leq\; 2D/\delta + \delta D (G\cdot k)^2 \cdot dn/2 + \frac{2(G\cdot k)Dn}{\sqrt{k}} = 2DG\sqrt{d}\sqrt{n}k + 2DGn\sqrt{k}, \tag{16}$$

with the parameter choice of $\delta = 2/(G\sqrt{d}\sqrt{n}k)$. Letting $n = k = T^{1/2}$ yields the expected regret bound for the oracle-efficient algorithm $\mathcal{A}'_1$ as $\mathbb{E}[\mathcal{R}_T(\mathcal{A}'_1)] \;=\; \mathcal{O}(\sqrt{n}k + n\sqrt{k}) \;=\; \mathcal{O}(T^{3/4})$ for general convex functions. The case of smooth convex functions is handled analogously. Note that the $\mathcal{A}'_1$ algorithm is equivalent to $\mathcal{A}_{\text{OSPF}}$ given in Algorithm 2. The smoothness parameter of a sum of k functions that are β-smooth equals $\beta' = \beta \cdot k$. Hence, the expected regret bound of $\mathcal{A}_{\text{OSPF}}$ for smooth convex functions is given by

$$\mathbb{E}[\mathcal{R}_T(\mathcal{A}_{\text{OSPF}})] \;\leq\; 2D/\delta + \delta D (G\cdot k)^2 \cdot dn/2 + \frac{4(\beta\cdot k)D^2 n}{k} = 2DG\sqrt{d}\sqrt{n}k + 4\beta D^2 n, \tag{17}$$

with the same choice of $\delta = 2/(G\sqrt{d}\sqrt{n}k)$. In this case let $n = T^{2/3}$ and $k = T^{1/3}$ to attain the expected regret bound $\mathcal{O}(T^{2/3})$ for $\mathcal{A}_{\text{OSPF}}$ with one linear optimization per iteration. ■

5.2. High Probability Bounds

The theoretical guarantees for the main algorithm of this paper, Algorithm 1, have all been in terms of expected regret as the performance metric. Even though expected regret is a widely accepted metric for online randomized algorithms, one might wonder whether the expectation bound holds only due to a balance of large and small chunks of regret or the given result actually holds most of the time. To answer this question, we provide high probability bounds on $\mathcal{R}_T(\mathcal{A}_1)$ asymptotically equivalent (up to logarithmic factors) to the statements from Theorem 7: these results also transfer analogously to the reduction from section 5.1. In particular, the following theorem shows that Algorithm 1 obtains regret $\mathcal{R}_T(\mathcal{A}_1) = \mathcal{O}(\sqrt{T}) + \tilde{\mathcal{O}}(T/\sqrt{m})$ for general convex losses and $\mathcal{R}_T(\mathcal{A}_1) = \mathcal{O}(\sqrt{T}) + \tilde{\mathcal{O}}(\beta T/m)$ for smooth convex losses both holding with high probability.

Theorem 16 *Given that the Assumptions 3 and 4 hold and the parameter value is $\delta = 2/(G\sqrt{dT})$, the regret of Algorithm 1 for general convex loss functions is w.p. $1-\sigma$ for any $\sigma > 0$ bounded by*

$$\sum_{t=1}^{T} f_t(\tilde{\mathbf{x}}_t) \;\leq\; \min_{\mathbf{x}\in\mathcal{K}}\{\sum_{t=1}^{T} f_t(\mathbf{x})\} + 2GD\sqrt{dT} + \frac{2GDT}{\sqrt{m}} \cdot \sqrt{\log 2T/\sigma}\,. \tag{18}$$

If the convex loss functions are also β-smooth, then it is w.p. $1-\sigma$ for any $\sigma > 0$ bounded by

$$\sum_{t=1}^{T} f_t(\tilde{\mathbf{x}}_t) \;\leq\; \min_{\mathbf{x}\in\mathcal{K}}\{\sum_{t=1}^{T} f_t(\mathbf{x})\} + 2GD\sqrt{dT} + 2GD\sqrt{2T\log 4/\sigma} + \frac{8\beta D^2 T}{m} \cdot \log 4T/\sigma\,. \tag{19}$$

6. Discussion

We have presented an efficient projection-free method for online convex optimization with smooth functions that makes only a single linear optimization computation per iteration and achieves regret $T^{2/3}$, improving upon the previous bound of $T^{3/4}$.

Certain algorithms in the literature make more than one linear optimization computation per iteration. To make the comparison to other methods more precise, we need a more refined computational metric. Define the following complexity metric for an online projection-free algorithm: let $\mathcal{A}$ be an online optimization algorithm, and define $T_\varepsilon(\mathcal{A})$ to be the overall number of gradient oracle calls as well as linear optimization calls made until the average regret becomes at most ε.

In these terms, we have shown an algorithm with $T_\varepsilon = \mathcal{O}(\frac{d}{\varepsilon^3})$ for smooth functions, as compared to $\mathcal{O}(\frac{1}{\varepsilon^4})$ which is the previous best. The dimension dependence is undesirable and prevents direct improvements in applications such as Garber and Hazan (2016). However, this dependence is a byproduct of FPL and the randomized smoothing technique used in section 4 that works for general convex sets. Various smoothing techniques can be appropriate for different sets and their role in FPL methods is discussed in Abernethy et al. (2014). We note that our contribution is *not* the cause of the dimension dependence and improving the complexity from ε^{-4} to ε^{-3} has remained unknown for a long time.

It thus remains open to obtain a ε^{-3}-complexity algorithm for general convex sets that does not depend on the dimension, or show that this is impossible. It is also unknown at this time if these dependencies on ε, in both the smooth and non-smooth cases, are tight.

Acknowledgments

This research was supported in part by Elad Hazan's NSF grant #1704860.

References

Jacob Abernethy, Chansoo Lee, Abhinav Sinha, and Ambuj Tewari. Online linear optimization via smoothing. In Maria Florina Balcan, Vitaly Feldman, and Csaba Szepesvri, editors, *Proceedings of The 27th Conference on Learning Theory*, volume 35 of *Proceedings of Machine Learning Research*, pages 807–823, Barcelona, Spain, 13–15 Jun 2014. PMLR. URL http://proceedings.mlr.press/v35/abernethy14.html.

Zeyuan Allen-Zhu, Elad Hazan, Wei Hu, and Yuanzhi Li. Linear convergence of a frank-wolfe type algorithm over trace-norm balls. In *Advances in Neural Information Processing Systems*, pages 6191–6200, 2017.

Andreas Argyriou, Marco Signoretto, and Johan Suykens. Hybrid conditional gradient - smoothing algorithms with applications to sparse and low rank regularization, 2014.

Lin Chen, Christopher Harshaw, Hamed Hassani, and Amin Karbasi. Projection-free online optimization with stochastic gradient: From convexity to submodularity. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 814–823, Stockholmsmssan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/chen18c.html.

Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

Dan Garber. Faster projection-free convex optimization over the spectrahedron. In *Advances in Neural Information Processing Systems*, pages 874–882, 2016.

Dan Garber and Elad Hazan. A linearly convergent conditional gradient algorithm with applications to online and stochastic optimization. *arXiv preprint arXiv:1301.4666*, 2013.

Dan Garber and Elad Hazan. Sublinear time algorithms for approximate semidefinite programming. *Mathematical Programming*, 158(1-2):329–361, 2016.

Hamed Hassani, Amin Karbasi, Aryan Mokhtari, and Zebang Shen. Stochastic conditional gradient++, 2019.

Elad Hazan. Sparse approximate solutions to semidefinite programs. In *Latin American symposium on theoretical informatics*, pages 306–316. Springer, 2008.

Elad Hazan and Satyen Kale. Projection-free online learning. In *Proceedings of the 29th International Conference on Machine Learning*, 2012.

Elad Hazan and Haipeng Luo. Variance-reduced and projection-free stochastic optimization. In *International Conference on Machine Learning*, pages 1263–1271, 2016.

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

Martin Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*, pages 427–435, 2013.

Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2005.

Guanghui Lan. The complexity of large-scale convex programming under a linear optimization oracle, 2013.

Guanghui. Lan and Yi. Zhou. Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization*, 26(2):1379–1409, 2016. doi: 10.1137/140992382. URL https://doi.org/10.1137/140992382.

Kfir Levy and Andreas Krause. Projection free online learning over smooth sets. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 1458–1466. PMLR, 16–18 Apr 2019. URL http://proceedings.mlr.press/v89/levy19a.html.

Neri Merhav, Erik Ordentlich, Gadiel Seroussi, and Marcelo J Weinberger. On sequential strategies for loss functions with memory. *IEEE Transactions on Information Theory*, 48(7):1947–1958, 2002.

Federico Pierucci, Zaid Harchaoui, and Jérôme Malick. A smoothing approach for composite conditional gradient with nonsmooth loss. Research Report RR-8662, INRIA Grenoble, July 2014. URL https://hal.inria.fr/hal-01096630.

Iosif Pinelis. Optimum bounds for the distributions of martingales in banach spaces. *The Annals of Probability*, 22(4):1679–1706, 1994. ISSN 00911798.

Shai Shalev-Shwartz and Yoram Singer. A primal-dual perspective of online learning algorithms. *Machine Learning*, 69(2-3):115–142, 2007a.

Shai Shalev-Shwartz and Yoram Singer. Convex repeated games and fenchel duality. *Machine Learning*, 2007b. URL https://www.cse.huji.ac.il/~shais/papers/ShalevSi06_fench_tech.pdf.

Jiahao Xie, Zebang Shen, Chao Zhang, Boyu Wang, and Hui Qian. Efficient projection-free online methods with stochastic recursive gradient, 2019.

Alp Yurtsever, Suvrit Sra, and Volkan Cevher. Conditional gradient methods via stochastic path-integrated differential estimator. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 7282–7291, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL http://proceedings.mlr.press/v97/yurtsever19b.html.

Mingrui Zhang, Zebang Shen, Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. One sample stochastic frank-wolfe, 2019.

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.

Appendix A. Proof of the high probability bounds

This section focuses on regret bound results for Algorithm 1 that hold with high probability. We use the following Azuma-type concentration inequality for vector-valued martingales derived as an application of the work by Pinelis (1994) to the Euclidean space $\mathbb{R}^d$.

Proposition 17 (Theorem 3.5 in Pinelis (1994)) *Let $\boldsymbol{\nu}_1, \ldots, \boldsymbol{\nu}_K \in \mathbb{R}^d$ be a vector-valued martingale difference with respect to $\{\mathcal{F}_k\}_{k=1}^K$ such that for all $k \in [K]$ it holds that $\mathbb{E}[\boldsymbol{\nu}_k | \mathcal{F}_{k-1}] = \mathbf{0}$ and $\|\boldsymbol{\nu}_k\| \leq c_k$ for some $c_k > 0$. Then for any $\lambda > 0$*

$$\mathbb{P}\left[\left\|\sum_{k=1}^K \boldsymbol{\nu}_k\right\| > \lambda\right] \leq 2\exp\left(-\frac{\lambda^2}{2\sum_{k=1}^K c_k^2}\right) \tag{20}$$

Proof [Proof of Theorem 16] We first obtain proximity of the estimates $\tilde{\mathbf{x}}_t$ to their mean $\hat{\mathbf{x}}_t$ for all $t \in [T]$ that hold with high probability using Proposition 17. Fix an arbitrary $t \in [T]$ and denote $\boldsymbol{\nu}_j = \frac{1}{m}(\hat{\mathbf{x}}_t - \mathbf{x}_t^j)$ for $j = 1, \ldots, m$. Note that $\sum_{j=1}^m \boldsymbol{\nu}_j = \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t$ and for each $j \in [m]$ we have $\mathbb{E}_{\mathbf{v}_t^j}[\boldsymbol{\nu}_j \,|\, \mathbf{v}_t^{1:j-1}] = \mathbf{0}$ given the definition of $\hat{\mathbf{x}}_t$ and i.i.d. uniform samples $\mathbf{v}_t^j \sim \mathbb{B}$, $j \in [m]$. Furthermore, $\|\boldsymbol{\nu}_j\| \leq 2D/m$ using triangle inequality since $\mathbf{x}_t^j, \hat{\mathbf{x}}_t \in \mathcal{K}$ given convexity of the constraint set $\mathcal{K}$. Fix any $\sigma > 0$ and let $\lambda = \frac{2D}{\sqrt{m}} \cdot \sqrt{2\log 2T/\sigma}$, then by Proposition 17

$$\mathbb{P}_{\xi_t}\left[\|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\| \geq \lambda\right] \leq \frac{\sigma}{T} \implies \mathbb{P}_{\xi_{1:T}}\left[\forall t \in [T],\, \|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\| \geq \lambda\right] \leq \sigma \tag{21}$$

where the implication stems from union bound. To conclude the regret bound for general convex functions with high probability, use Lemma 13 and apply $\langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle \leq G\|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\|$ to get that

$$\mathcal{R}_T(\mathcal{A}_1) \leq 2GD\sqrt{dT} + \lambda GT = 2GD\sqrt{dT} + \frac{2GDT}{\sqrt{m}} \cdot \sqrt{2\log 2T/\sigma} \tag{22}$$

holds with probability at least $1-\sigma$. In the smooth convex case, proceed analogously and apply the inequality $\langle \tilde{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle \leq \langle \hat{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle + \beta\|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\|^2$ given by Lemma 14. Fix any $\sigma > 0$ and denote $\zeta_t = \langle \hat{\nabla}_t, \hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t \rangle$ for all $t \in [T]$. Notice that $\{\zeta_t\}_{t=1}^T$ is a martingale difference with respect to $\xi_{1:T}$. Indeed, $\mathbb{E}_{\xi_t}[\tilde{\mathbf{x}}_t \,|\, \xi_{1:t-1}] = \hat{\mathbf{x}}_t$ and the quantities $f_t(\cdot), \hat{\mathbf{x}}_t$, and hence $\hat{\nabla}_t$, are deterministic given $\xi_{1:t-1}$ which means that $\mathbb{E}_{\xi_t}[\zeta_t \,|\, \xi_{1:t-1}] = 0$. Moreover, they are bounded $|\zeta_t| \leq \|\hat{\nabla}_t\| \cdot \|\hat{\mathbf{x}}_t - \tilde{\mathbf{x}}_t\| \leq 2GD = c_t$ using Cauchy-Schwartz inequality, triangle inequality and convexity of $\mathcal{K}$. Letting $\gamma = 2GD\sqrt{2T\log 4/\sigma}$ Azuma's inequality yields

$$\mathbb{P}_{\xi_{1:T}}\left[\left|\sum_{t=1}^T \zeta_t\right| \geq \gamma\right] \leq 2\exp\left(-\frac{\gamma^2}{2\sum_{t=1}^T c_t^2}\right) = \sigma/2 \tag{23}$$

Combine (23) and the already obtained (21) with $\sigma' = \sigma/2$, and corresponding λ', to conclude the regret bound for smooth convex functions using union bound and the Lemma 13 to obtain that

$$\mathcal{R}_T(\mathcal{A}_1) \leq 2GD\sqrt{dT} + \gamma + \beta(\lambda')^2 T = 2GD\sqrt{dT} + 2GD\sqrt{2T\log 4/\sigma} + \frac{8\beta D^2 T}{m} \cdot \log 4T/\sigma$$

holds with probability at least $1-\sigma$. This finishes the proof of Theorem 16. The bound in (19) implies, following the same logic as in section 5.1, that the regret bound $\mathcal{R}_T(\mathcal{A}_{\text{OSPF}}) = \tilde{O}(T^{2/3}\log 1/\sigma)$ holds with high probability $1-\sigma$. ■

Appendix B. Miscellaneous proofs

Proof [Proof of Lemma 6] To show convexity, consider arbitrary $\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^d$ and $\lambda > 0$, denote $\mathbf{y}_{12} = \lambda \mathbf{y}_1 + (1-\lambda)\mathbf{y}_2$. Then

$$\mathcal{M}_{\mathcal{K}}(\mathbf{y}_{12}) = \langle \mathbf{y}_{12}, \mathbf{x}^* \rangle = \lambda \langle \mathbf{y}_1, \mathbf{x}^* \rangle + (1-\lambda)\langle \mathbf{y}_2, \mathbf{x}^* \rangle \; \leq \; \lambda \mathcal{M}_{\mathcal{K}}(\mathbf{y}_1) + (1-\lambda)\mathcal{M}_{\mathcal{K}}(\mathbf{y}_2)$$

Next, fix arbitrary $\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^d$ and suppose w.l.o.g. that $\mathcal{M}_{\mathcal{K}}(\mathbf{y}_1) \; \geq \; \mathcal{M}_{\mathcal{K}}(\mathbf{y}_2)$. Then

$$\begin{aligned}\mathcal{M}_{\mathcal{K}}(\mathbf{y}_1) - \mathcal{M}_{\mathcal{K}}(\mathbf{y}_2) &= \langle \mathbf{y}_1, \mathcal{O}_{\mathcal{K}}(\mathbf{y}_1) \rangle - \langle \mathbf{y}_2, \mathcal{O}_{\mathcal{K}}(\mathbf{y}_2) \rangle \\ &\leq \; \langle \mathbf{y}_1, \mathcal{O}_{\mathcal{K}}(\mathbf{y}_1) \rangle - \langle \mathbf{y}_2, \mathcal{O}_{\mathcal{K}}(\mathbf{y}_1) \rangle \\ &= \langle \mathcal{O}_{\mathcal{K}}(\mathbf{y}_1), \mathbf{y}_1 - \mathbf{y}_2 \rangle \; \leq \; \|\mathcal{O}_{\mathcal{K}}(\mathbf{y}_1)\| \|\mathbf{y}_1 - \mathbf{y}_2\| \; \leq \; D\|\mathbf{y}_1 - \mathbf{y}_2\|\end{aligned}$$

where the first inequality follows from the definition of $\mathcal{O}_{\mathcal{K}}(\mathbf{y}_2)$ while the rest is achieved using the Cauchy-Schwarz inequality and the norm bound of the constraint set $\mathcal{K}$. ■

Proof [Proof of Lemma 11] According to Stokes' theorem, the gradient of the smoothed function $\hat{g}(\cdot)$ can be written as

$$\nabla \hat{g}(\mathbf{y}) = \delta d \mathbb{E}_{\mathbf{v} \sim \mathbb{S}} \left[g(\mathbf{y} + \frac{1}{\delta}\mathbf{v})\mathbf{v} \right]$$

where $\mathbb{S} = \{\mathbf{v} \in \mathbb{R}^d, \|\mathbf{v}\| = 1\}$ denotes the unit sphere, the boundary of $\mathbb{B}$. Then for arbitrary $\mathbf{y}_1, \mathbf{y}_2 \in \mathbb{R}^d$

$$\|\nabla \hat{g}(\mathbf{y}_1) - \nabla \hat{g}(\mathbf{y}_2)\| = \delta d \left\| \mathbb{E}_{\mathbf{v} \sim \mathbb{S}} \left[g(\mathbf{y}_1 + \frac{1}{\delta}\mathbf{v})\mathbf{v} - g(\mathbf{y}_2 + \frac{1}{\delta}\mathbf{v})\mathbf{v} \right] \right\| \; \leq \; \delta d L \|\mathbf{y}_1 - \mathbf{y}_2\|$$

using linearity of expectation, Jensen's inequality and the Lipschitz property of $g(\cdot)$. It follows that $\hat{g}(\cdot)$ is a $\delta d L$-smooth function. ■

Proof [Proof of Lemma 14] The function $f : \mathcal{K} \to \mathbb{R}$ being β-smooth is equivalent to its gradient being β-Lipschitz, hence

$$\langle \nabla f(y) - \nabla f(x), x - y \rangle \; \leq \; \|\nabla f(y) - \nabla f(x)\| \cdot \|x - y\| \; \leq \; \beta \|x - y\|^2$$

The desired inequality follows from the result above. ■

Proof [Proof of Lemma 15] Given that $\|Z\| \; \leq \; D$ and $\overline{Z} = \mathbb{E}[Z]$ we have that $\|\overline{Z}\| \; \leq \; D$ and $\|Z - \overline{Z}\| \; \leq \; \|Z\| + \|\overline{Z}\| \; \leq \; 2D$. Hence, by linearity of expectation and linearity of variance for independent random variables we obtain

$$\begin{aligned}\mathbb{E}_{\mathcal{Z}}[\|\overline{Z}_m - \overline{Z}\|^2] = \mathbb{E}_{\mathcal{Z}} \left[\sum_{i=1}^{d} (\overline{Z}_m(i) - \overline{Z}(i))^2 \right] = \sum_{i=1}^{d} \mathbb{E}_{\mathcal{Z}_i}[(\overline{Z}_m(i) - \overline{Z}(i))^2] = \\ = \sum_{i=1}^{d} \mathrm{Var}_{\mathcal{Z}_i}(\overline{Z}_m(i)) = \sum_{i=1}^{d} \frac{1}{m^2} \sum_{j=1}^{m} \mathrm{Var}_{\mathcal{Z}_i}(Z_j(i)) = \frac{1}{m} \sum_{i=1}^{d} \mathrm{Var}_{\mathcal{Z}_i}(Z(i)) = \\ = \frac{1}{m} \sum_{i=1}^{d} \mathbb{E}_{\mathcal{Z}_i}[(Z(i) - \overline{Z}(i))^2] = \frac{1}{m} \mathbb{E}_{\mathcal{Z}}[\|Z - \overline{Z}\|^2] \; \leq \; \frac{4D^2}{m}\end{aligned}$$

■