Quasi-Newton Steps for Efficient Online Exp-Concave Optimization

Zakaria Mhammedi MHAMMEDI@MIT.EDU
Khashayar Gatmiry GATMIRY@MIT.EDU
Massachusetts Institute of Technology

Editors: Gergely Neu and Lorenzo Rosasco

Abstract

The aim of this paper is to design computationally-efficient and optimal algorithms for the online and stochastic exp-concave optimization settings. Typical algorithms for these settings, such as the Online Newton Step (ONS), can guarantee a $O(d \ln T)$ bound on their regret after T rounds, where d is the dimension of the feasible set. However, such algorithms perform so-called *generalized projections* whenever their iterates step outside the feasible set. Such generalized projections require $\Omega(d^3)$ arithmetic operations even for simple sets such a Euclidean ball, making the total runtime of ONS of order d^3T after T rounds, in the worst-case. In this paper, we side-step generalized projections by using a self-concordant barrier as a regularizer to compute the Newton steps. This ensures that the iterates are always within the feasible set without requiring projections. This approach still requires the computation of the inverse of the Hessian of the barrier at every step. However, using stability properties of the Newton iterates, we show that the inverse of the Hessians can be efficiently approximated via Taylor expansions for most rounds, resulting in a $\widetilde{O}(d^2T + d^\omega\sqrt{T})$ total computational complexity, where $\omega \in (2, 3]$ is the exponent of matrix multiplication. In the stochastic setting, we show that this translates into a $\widetilde{O}(d^3/\varepsilon)$ computational complexity for finding an ε-optimal point, answering an open question by Koren 2013. We first prove these new results for the simple case where the feasible set is a Euclidean ball. Then, to move to general convex sets, we use a reduction to Online Convex Optimization over the Euclidean ball. Our final algorithm for general convex sets can be viewed as a more computationally-efficient version of ONS.

1. Introduction

We consider the problem of Online Convex Optimization (OCO) with exp-concave loss functions. In this setting, an algorithm outputs vectors in a closed convex set $\mathcal{C} \subset \mathbb{R}^d$ in rounds: At the beginning of round t, the algorithm outputs $\boldsymbol{w}_t \in \mathcal{C}$ based on the past history, then observes an exp-concave function $\ell_t : \mathcal{C} \to \mathbb{R}_{\geq 0}$, which can be chosen by an adversary based on $\boldsymbol{w}_t$ and the history. The algorithm suffers loss $\ell_t(\boldsymbol{w}_t)$ and proceeds to the next round $t+1$. The performance of the algorithm is measured by its *regret* against the best comparator vector $\boldsymbol{w} \in \mathcal{C}$ in hindsight after $T \in \mathbb{N}$ rounds:

$$\mathrm{Reg}_T := \sum_{t=1}^{T} \ell_t(\boldsymbol{w}_t) - \min_{\boldsymbol{w} \in \mathcal{C}} \sum_{t=1}^{T} \ell_t(\boldsymbol{w}).$$

Our goal in this paper is to design computationally-efficient algorithms that achieve (up to poly-log-factors) the optimal $O(d \ln T)$ regret for the online exp-concave setting (Mahdavi et al., 2015).

Many Machine Learning (ML) problems can be reduced to Online/Stochastic Exp-Concave Optimization. Of most relevance are certain *online supervised learning* problems (Rakhlin et al., 2015) that include, for example, Online Linear Regression (OLR), where at each round t, the algorithm receives a feature vector $\boldsymbol{x}_t \in \mathbb{R}^d$ (that may be chosen adversarially), then outputs $\hat{y}_t \in \mathbb{R}$

of the form $\hat{y}_t = \boldsymbol{w}_t^\top \boldsymbol{x}_t$ for some parameter vector $\boldsymbol{w}_t \in \mathcal{C}$ that is updated at every round. Then, an outcome y_t is revealed and the algorithm suffers a loss $\ell(\hat{y}_t, y_t)$. The special case where ℓ is the square loss $\ell(\hat{y}_t, y_t) = (\hat{y}_t - y_t)^2$—where $\boldsymbol{w} \mapsto \ell(\boldsymbol{w}^\top \boldsymbol{x}, y)$ is exp-concave—has been extensively studied in the ML literature; see e.g. (Foster, 1991; Vovk, 1997, 2001; Azoury and Warmuth, 2001; Bartlett et al., 2015; Gaillard et al., 2019), and (Cesa-Bianchi and Lugosi, 2006, Chapter 11) for a thorough introduction to the topic. Other exp-concave losses that have also been extensively studied in the context of online learning include Cover's loss $\ell(\boldsymbol{w}, \boldsymbol{x}) = -\ln(\boldsymbol{w}^\top \boldsymbol{x})$ for the portfolio selection problem (Cover, 1991; Luo et al., 2018; Mhammedi and Rakhlin, 2022; Jézéquel et al., 2022) and the logistic loss $\ell((\boldsymbol{w}, b), \boldsymbol{x}) = \ln(1 + \exp(-b\boldsymbol{w}^\top \boldsymbol{x}))$ for classification (Cover, 1991; Foster et al., 2018; Agarwal et al., 2022; Mayo et al., 2022). Given the prevalence of ML problems that can be reduced to Online/Stochastic Exp-Concave Optimization, it is crucial to have efficient algorithms for the latter as echoed by the COLT open problem by Koren (2013) that specifically asks for efficient algorithms in the stochastic setting.

In constrained Online eXp-Concave Optimization (OXO) ensuring that the iterates $(\boldsymbol{w}_t)$ are within the feasible set $\mathcal{C}$, while guaranteeing the optimal $O(d \ln T)$ regret, typically requires a special type of projections, which constitute the main computational challenge behind the design of efficient OXO algorithms. For example, the Online Newton Step (ONS) (Hazan et al., 2007), one of the most popular algorithms for OXO, requires a *Mahalanobis projection*, a.k.a. generalized projection, at each round t where the iterate $\boldsymbol{w}_t$ steps outside of the set $\mathcal{C}$. Such projections typically require $\Omega(d^3)$ arithmetic operations (to perform an SVD decomposition) even for simple convex sets such as when $\mathcal{C}$ is a Euclidean ball. This means that in the stochastic setting, ONS-like algorithms can require $O(d^4/\varepsilon)$ arithmetic operations to find an ε-optimal point; see (Koren, 2013) for details. Koren (2013) asks for an algorithm that requires fewer than $O(d^4/\varepsilon)$ arithmetic operations to find such a point.

Contributions. Our main contribution is a new, more efficient version of ONS, which does not require generalized projections thanks to the use of a self-concordant barrier. For the case where $\mathcal{C}$ is a Euclidean ball, our algorithm essentially outputs online Newton iterates with a log-barrier regularizer. The use of this barrier ensures that the iterates are always within $\mathcal{C}$ thanks to self-concordance properties of the barrier. The algorithm achieves the same regret bound as that of ONS (up to log-factors) and performs $\widetilde{O}(d^2)$ arithmetic operations per round, except for a $T^{-1/2}$ fraction of the rounds where the algorithm performs a matrix inversion. In contrast, standard ONS requires $\Omega(d^3)$ arithmetic operations (to perform an SVD decomposition) every round, in the worst-case. We are able to improve on the computational complexity of ONS by leveraging stability properties of the Newton iterates (which are conferred by our choice of regularizer) to efficiently approximate the inverse of certain Hessian matrices via Taylor expansions; for this reason, our approach falls under the category of *Quasi-Newton Methods*[1] (Gill and Murray, 1972).

For general convex feasible sets, we use a reduction to OCO over the Euclidean ball. Using this reduction comes only with an additive $\widetilde{O}(\mathcal{T}_{\mathsf{sep}}(\mathcal{C}))$ [resp. $O(\mathcal{T}_{\mathsf{proj}}(\mathcal{C}))$] computational cost per round for a centrally-symmetric [resp. general] convex set $\mathcal{C}$, where $\mathcal{T}_{\mathsf{sep}}(\mathcal{C})$ denotes the cost of performing Separation [resp. Euclidean projection] with respect to the set $\mathcal{C}$. In general, $\mathcal{T}_{\mathsf{proj}}(\mathcal{C})$ and $\mathcal{T}_{\mathsf{sep}}(\mathcal{C})$ can be much smaller than the cost of a Mahalanobis projection, which is required by ONS. We note that even projected Online Gradient Descent (OGD) requires $\mathcal{T}_{\mathsf{proj}}(\mathcal{C})$ arithmetic operations per round

1. We note, however, that our new algorithm is distinct from the existing BFGS-based stochastic Quasi-Newton methods (Schraudolph et al., 2007; Byrd et al., 2016; Indrapriyadarsini et al., 2019; Mokhtari and Ribeiro, 2020), which do not lead to an improved computational complexity for the stochastic exp-concave problem compared to ONS.

in the worst-case. We discuss the computational complexity of our algorithm in more detail in the sequel (Remark 11).

Finally, instantiating our OXO results in the stochastic exp-concave setting via a standard application of online-to-batch conversion (see e.g. Cesa-Bianchi and Lugosi (2006)) leads to an algorithm that finds an ε-optimal point using $\widetilde{O}(d^3/\varepsilon)$ arithmetic operations. This answers one of the questions posed in the COLT open problem by Koren (2013). We conjecture that this number of arithmetic operations is the best one can hope for if one insists on a computational complexity that scales with $1/\varepsilon$ (instead of $1/\varepsilon^2$, for example). See App. E for more detail.

Related Works. The idea of using Newton steps with a barrier regularizer to build efficient online learning/optimization algorithms originated in (Abernethy et al., 2012). There, the authors show that such online Newton iterates can approximate the Follow-The-Regularized-Leader (FTRL) iterates well enough to essentially inherit the regret guarantee of the latter. More recently, Mhammedi and Rakhlin (2022) used online Newton iterates with a log-barrier for the simplex to approximate the iterates of the Mirror Descent-based algorithm BARRONS (Luo et al., 2018) and build an efficient algorithm for the portfolio selection problem (Cover, 1991; van Erven et al., 2020). However, the algorithms in (Abernethy et al., 2012; Mhammedi and Rakhlin, 2022) still require $\Omega(d^\omega)$ operations per round due to matrix inversion, where ω is the exponent of matrix multiplication. Reducing this computational cost to essentially $\widetilde{O}(d^2)$ per round is the main challenge we overcome in this paper by leveraging the stability of the Newton iterates when using the log-barrier for the Euclidean ball.

Our technique is inspired by one initially used in (Vaidya, 1987) for efficiently solving linear programs by avoiding the computation of the full inverse of certain Hessian matrices at every iteration. However, their approach crucially relies on the feasible set being a polytope, and provides no computational advantage when the set is a Euclidean ball (we will reduce general OXO to OXO over a ball). Extending some of the ideas in (Vaidya, 1987) to our setting is non-trivial and relies on recent results by Mhammedi and Rakhlin (2022) on the stability of the Newton iterates with a particular choice of regularizer. We also note that our approach is different from previous ones that use, for example, sketching techniques to reduce the per-round computational complexity of ONS (Luo et al., 2016). Such techniques do not lead to a logarithmic regret in the OXO setting without additional assumptions.

Extending our new efficient algorithm for OXO over the Euclidean ball to general convex sets by simply adapting the barrier to the set of interest fails because, beyond the Euclidean ball and polytopes, the barriers of other convex bodies are typically hard to compute. So, instead of taking this path, we leverage recent techniques in OCO (Cutkosky and Orabona, 2018; Mhammedi et al., 2019; Cutkosky, 2020; Mhammedi, 2022) to reduce the exp-concave optimization problem to one over a Euclidean ball. In particular, we use the algorithm of (Mhammedi, 2022) that reduces OCO over an arbitrary convex set to one over a ball for the purpose of efficient projection-free OCO. Though we use the same algorithm as Mhammedi (2022) for the reduction to OCO over a Euclidean ball, we need to extend their analysis to exp-concave losses; using their analysis directly leads to a $O(\sqrt{T})$ regret guarantee, which is sub-optimal in the OXO setting. We show that the reduction in (Mhammedi, 2022) is naturally well suited to exp-concave losses, allowing our final algorithm to achieve near-optimal regret.

Outline. In Section 2, we present the notation and definitions we require in the main body of the paper. In Section 3, we present our efficient algorithm for OXO over a Euclidean ball. In Section 4, we extend our results beyond the Euclidean ball using a reduction to online optimization on the latter.

In Section 4, we also instantiate our results in the stochastic exp-concave setting, where we answer one of the questions posed by Koren (2013).

2. Preliminaries

Throughout, we let $\mathcal{C}$ be a closed convex subset of the Euclidean space $\mathbb{R}^d$. We let $\|\cdot\|$ denote the Euclidean norm and $\mathcal{B}(r) \subset \mathbb{R}^d$ the Euclidean ball of radius $r > 0$. We let $\gamma_{\mathcal{K}}(\boldsymbol{x}) \coloneqq \inf\{\lambda \geq 0 : \boldsymbol{x} \in \lambda\mathcal{K}\}$ be the *Gauge function* of a convex set $\mathcal{K}$, and $\mathcal{K}^\circ \coloneqq \{\boldsymbol{x} \in \mathbb{R}^d : \langle \boldsymbol{x}, \boldsymbol{y}\rangle \leq 1, \forall \boldsymbol{y} \in \mathcal{K}\}$ be its *polar* set (Hiriart-Urruty and Lemaréchal, 2004). Further, we denote by $\operatorname{int}\mathcal{K}$ the interior of a set $\mathcal{K}$. Our final algorithm requires access to either a Separation or a Euclidean projection Oracle for the set $\mathcal{C}$.

Definition 1 (Separation Oracle) *A* Separation Oracle $\mathsf{sep}_{\mathcal{K}}$ *for a set $\mathcal{K}$ is an Oracle that given $\boldsymbol{u} \in \mathbb{R}^d$ either asserts that $\boldsymbol{u} \in \mathcal{K}$ or returns $\boldsymbol{w} \in \mathbb{R}^d$ such that $\langle \boldsymbol{w}, \boldsymbol{u}\rangle > \langle \boldsymbol{w}, \boldsymbol{v}\rangle$, for all $\boldsymbol{v} \in \mathcal{K}$. We denote by $\mathcal{T}_{\mathsf{sep}}(\mathcal{K})$ the computational complexity of one call to this Oracle.*

Definition 2 (Euclidean Projection Oracle) *A* Euclidean Oracle $\mathsf{proj}_{\mathcal{K}}$ *for a set $\mathcal{K}$ is an Oracle that given $\boldsymbol{u} \in \mathbb{R}^d$ either asserts that $\boldsymbol{u} \in \mathcal{K}$ or returns $\boldsymbol{w} \in \mathcal{K}$ such that $\|\boldsymbol{w} - \boldsymbol{u}\| \leq \|\boldsymbol{v} - \boldsymbol{u}\|$, for all $\boldsymbol{v} \in \mathcal{K}$. We denote by $\mathcal{T}_{\mathsf{proj}}(\mathcal{K})$ the computational complexity of one call to this Oracle.*

Our results can easily be extended to the case where only approximate Separation/Euclidean projection Oracles are available (see (Lee et al., 2018) for definitions).

This paper focuses on the online optimizing of exp-concave functions.

Definition 3 *Let $\alpha > 0$ and $\mathcal{K} \subseteq \mathbb{R}^d$ be a convex set. A function $f : \mathcal{K} \to \mathbb{R}$ is α-exp-concave if $\boldsymbol{x} \mapsto e^{-\alpha f(\boldsymbol{x})}$ is concave over $\mathcal{K}$.*

For a twice differentiable function f, we denote by $\nabla^2 f(\boldsymbol{u})$ [resp. $\nabla^{-2} f(\boldsymbol{u})$] its Hessian [resp. inverse Hessian] at $\boldsymbol{u}$. We use the notation $\widetilde{O}(\cdot)$ to hide poly-log-factors in problem parameters. We denote by $\omega \in (2, 3]$ the matrix multiplication constant. Our proofs use properties of self-concordant functions which we include in Appendix B.

3. Efficient Online Exp-Concave Optimization Over a Ball

In this section, we construct an efficient online algorithm (Alg. 1) for exp-concave optimization over the unit Euclidean ball $\mathcal{B}(1)$. We later use Alg. 1 as a subroutine in our main algorithm (Alg. 2) for Online and Stochastic Exp-concave Optimization over general convex sets. The "pseudocode" (or the efficiently implementable version) of Algorithm 1 is displayed in Algorithm 3 in the appendix. We carry out our analysis under the following Lipschitzness assumption on the sequence of losses.

Assumption 1 *For $\mathfrak{B} > 0$, the sub-gradients $(\boldsymbol{g}_t)$ in Algorithm 1 satisfy $\|\boldsymbol{g}_t\| \leq \mathfrak{B}$, for all $t \geq 1$.*

The algorithm of this section, Alg. 1, essentially outputs approximate Newton iterates with respect to objective functions (Φ_t) that consist of the log-barrier for the unit Euclidean ball plus quadratic approximations of the observed losses (ℓ_t). In particular, for some parameters $(B, \eta, \beta) \in \mathbb{R}^3_{>0}$,

$$\Phi_t(\boldsymbol{x}) \coloneqq \Psi(\boldsymbol{x}) + \frac{d + B^2\eta}{2}\|\boldsymbol{x}\|^2 + \frac{\beta}{2}\sum_{s=1}^{t-1}\langle \boldsymbol{g}_s, \boldsymbol{x} - \boldsymbol{w}_s\rangle^2 + \boldsymbol{x}^\top \sum_{s=1}^{t-1} \boldsymbol{g}_s, \tag{1}$$

where $\Psi(\boldsymbol{x}) \coloneqq -\eta d \log(1 - \|\boldsymbol{x}\|^2)$ and $(\boldsymbol{g}_t \in \partial \ell_t(\boldsymbol{w}_t))$ are the observed subgradients at the iterates $(\boldsymbol{w}_t)$ of Algorithm 1. With this, we show that the outputs $(\boldsymbol{w}_t)$ of Algorithm 1 are approximate Newton iterates with respect to (Φ_t) in the sense that:

$$\boldsymbol{w}_{t+1} \approx \boldsymbol{w}_t - \nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)\nabla\Phi_{t+1}(\boldsymbol{w}_t), \quad \text{for all } t \in [T]. \tag{2}$$

The regret analysis of Algorithm 1 then consists of showing that the Newton iterates with respect to (Φ_t) are good approximations of the FTRL iterates $(\boldsymbol{x}_t)$:

$$\boldsymbol{x}_t \in \operatorname*{argmin}_{\boldsymbol{x} \in \mathcal{B}(1)} \Phi_t(\boldsymbol{x}) \tag{3}$$

and that these FTRL iterates guarantee our target regret bound. What makes Algorithm 1 special is that it is able to efficiently approximate Newton iterates in the sense of (2). To do this, the algorithm approximates the inverse of the Hessians $\nabla^2\Phi_{t+1}(\boldsymbol{w}_t)$, for $t \geq 1$; for this reason, our approach falls under the category of Quasi-Newton Methods (Gill and Murray, 1972). Next, we discuss in more detail how Algorithm 1 is able to efficiently approximate Newton iterates in the sense of (2).

Algorithm 1 OQNS: Online Quasi-Newton Steps Over the Euclidean ball. (Pseudocode in Alg. 3)

Require: Parameters $B, \eta, \beta, c > 0$, and Taylor order m as in Alg. 3. //B, β needed for (Φ_t) in (1)

Set $\boldsymbol{u}_1 = \boldsymbol{w}_1 = \mathbf{0}$ and $A_0 = (2\eta d + d + \eta B^2)^{-1} I$.
for $t = 1, \dots, T$ **do**
  Play $\boldsymbol{w}_t$ and observe $\boldsymbol{g}_t \in \partial \ell_t(\boldsymbol{w}_t)$.
  Compute $A_t = \left(\frac{2\eta d I}{1 - \|\boldsymbol{u}_t\|^2} + (d + \eta B^2) I + \beta \sum_{s=1}^{t} \boldsymbol{g}_s \boldsymbol{g}_s^\top \right)^{-1}$. //When $\boldsymbol{u}_t = \boldsymbol{u}_{t-1}$, $A_t^{-1} = A_{t-1}^{-1} + \beta \boldsymbol{g}_t \boldsymbol{g}_t^\top$, and so A_t can be computed in $O(d^2)$ using A_{t-1} and a rank-one update of the inverse.
  Compute $H_t = \left(A_t^{-1} + \frac{4\eta d \boldsymbol{w}_t \boldsymbol{w}_t^\top}{(1 - \|\boldsymbol{w}_t\|^2)^2} \right)^{-1}$ //Computable in $O(d^2)$ with a rank-one update.
  Define $\gamma_t = \frac{2\eta d}{1 - \|\boldsymbol{u}_t\|^2} - \frac{2\eta d}{1 - \|\boldsymbol{w}_t\|^2}$. //The definition is such that $H_t^{-1} = \nabla^2\Phi_{t+1}(\boldsymbol{w}_t) + \gamma_t I$.
  Define $\widetilde{H}_t = \sum_{k=1}^{m+1} \gamma_t^{k-1} H_t^k$. //Approximates $\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)$ via m^{th} order Taylor expansion
  Compute $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \widetilde{H}_t \nabla\Phi_{t+1}(\boldsymbol{w}_t)$. //Computable in $O(md^2)$ (Lines 7-12 of Alg. 3)
  if $\left| \|\boldsymbol{w}_{t+1}\|^2 - \|\boldsymbol{u}_t\|^2 \right| \leq c \cdot (1 - \|\boldsymbol{u}_t\|^2)$ **then**
    Set $\boldsymbol{u}_{t+1} = \boldsymbol{u}_t$. //No landmark update; $\boldsymbol{u}_t$ can be used for the next Taylor expansion
  else
    Set $\boldsymbol{u}_{t+1} = \boldsymbol{w}_{t+1}$. //Updated landmark; this happens at most $\widetilde{O}(\sqrt{T})$ times (Lemma 4)
  end if
end for

3.1. Efficient Computation of Newton Iterates

Algorithm 1 generates iterates $(\boldsymbol{w}_t)$ that satisfy the approximate equality in (2) without evaluating the inverse Hessian $\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)$ exactly at every round t. In particular, Algorithm 1 computes the inverse Hessian $\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)$ exactly at most $\widetilde{O}(\sqrt{T})$ times after T rounds, and approximates it using a Taylor expansion the rest of the time, resulting in a low amortized computational complexity. What makes this possible are certain self-concordance properties of the log-barrier Ψ in the definition of Φ_t. These properties imply that as long as the iterates $(\boldsymbol{w}_t)$ are stable enough, it is possible to efficiently approximate $\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)\nabla\Phi_{t+1}(\boldsymbol{w}_t)$ for most rounds $t \geq 1$ using Taylor expansions

around a small number of landmark iterates $\boldsymbol{w}_{\tau_1}, \dots, \boldsymbol{w}_{\tau_N}$, where $1 \le \tau_1 \le \cdots \le \tau_N \le T$. At round $t \ge 1$, $\boldsymbol{u}_t$ in Algorithm 1 represents the current landmark and $\boldsymbol{u}_t \neq \boldsymbol{u}_{t-1}$ only if $\boldsymbol{w}_t$ is "far enough" from the most-recent landmark $\boldsymbol{u}_{t-1}$, in which case $\boldsymbol{w}_t$ is set as the current landmark, i.e. $\boldsymbol{u}_t = \boldsymbol{w}_t$ (see Lines 9-12 of Alg. 1). The next lemma, whose proof is in App. C.2, shows that the total number of unique landmarks used by Alg. 1 is small relative to the number of rounds T. The proof of the lemma relies on the stability of the Newton iterates conferred by the non-linear terms in (Φ_t).

Lemma 4 (Stability) *Let $\beta, c \in (0,1)$, $B > 0$, and $\eta \ge 1$. Further, let $(\boldsymbol{u}_t)$ be as in Algorithm 1 with parameters (B, η, β, c) and suppose that Assumption 1 holds with $\mathfrak{B} \le B$. Then, it holds that*

$$\sum_{t=1}^{T-1} \mathbb{I}\{\boldsymbol{u}_{t+1} \neq \boldsymbol{u}_t\} \le 8\sqrt{\frac{2T\ln(d + B^2T/d)}{c^2\eta\beta}}.$$

On the rounds where the landmarks are updated (i.e. on the rounds t where $\boldsymbol{u}_t \neq \boldsymbol{u}_{t-1}$) Algorithm 1 computes the inverse Hessian $\nabla^{-2}\Phi_{t+1}(\boldsymbol{u}_t)$ exactly, which can be done in $O(d^\omega)$ (Cormen et al., 2009). Next, we will show that on any other round (i.e. a round where $\boldsymbol{u}_t = \boldsymbol{u}_{t-1}$), Algorithm 1 efficiently approximates $\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)$ and computes $\boldsymbol{w}_{t+1}$ in $\widetilde{O}(d^2)$, implying (thanks to Lemma 4) a total computational complexity of at most $\widetilde{O}(d^2T + d^\omega\sqrt{T})$ for Algorithm 1.

Efficient approximation of the Hessian. On round $t \ge 1$, Algorithm 1 computes the iterate $\boldsymbol{w}_{t+1}$ via a Taylor expansion of order $m = O(\ln T)$ using the current landmark $\boldsymbol{u}_t$ as follows:

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \sum_{k=1}^{m+1} \gamma_t^{k-1} H_t^k \nabla\Phi_{t+1}(\boldsymbol{w}_t), \quad \text{where} \quad \gamma_t \coloneqq \frac{2d\eta}{1 - \|\boldsymbol{u}_t\|^2} - \frac{2d\eta}{1 - \|\boldsymbol{w}_t\|^2}, \tag{4}$$

and $H_t^{-1} = \nabla^2\Phi_{t+1}(\boldsymbol{w}_t) + \gamma_t I$. Now, with $\Theta\colon \gamma \mapsto (\nabla^2\Phi_{t+1}(\boldsymbol{w}_t) + \gamma I)^{-1}$ and the definitions of γ_t and H_t, the sum $\sum_{k=1}^{m+1} \gamma_t^{k-1} H_t^k$ is simply the mth-order Taylor expansion of $\Theta(0) = \nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)$ at the point γ_t. This expansion provides an accurate approximation of $\Theta(0) = \nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)$, when $\left|\frac{1-\|\boldsymbol{u}_t\|^2}{1-\|\boldsymbol{w}_t\|^2} - 1\right| < 1$ (i.e. when γ_t is small enough), which is an invariant of Algorithm 1 when $c < 1$—see Line 9 of Algorithm 1. We formalize this in the next lemma.

Lemma 5 *Let $\beta, c \in (0,1)$, $\eta \ge 1$, and $B > 0$. Further, let γ_t be as in (4) and (H_t) be as in Algorithm 1 with parameters (B, η, β, c). Then, for any $m \ge 1$, we have*

$$\left\| \nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t) - \sum_{k=1}^{m+1} \gamma_t^{k-1} H_t^k \right\| \le \frac{c^m}{2\eta d \cdot (1-c)}.$$

One implication of this result is that the order m of the expansion need only be of order $O(\ln T)$ (the exact choice of m is specified in Alg. 3) to obtain a poly$(\frac{1}{T})$-accurate approximation of the Hessian (which is all we need for our target regret). Thus, given the matrix H_t, the vector $\boldsymbol{w}_{t+1}$ in (4) (i.e. the output of Algorithm 1) can be computed in $O(md^2) = \widetilde{O}(d^2)$ using m matrix-vector multiplications.

It remains to consider the computational cost of H_t itself. First, note that H_t we can be written as $H_t = \left(A_t^{-1} + \frac{4\eta d \boldsymbol{w}_t\boldsymbol{w}_t^\top}{(1-\|\boldsymbol{w}_t\|^2)^2}\right)^{-1}$, where $A_t = \left(\frac{2\eta d I}{1-\|\boldsymbol{u}_t\|^2} + (d + \eta B^2)I + \beta\sum_{s=1}^t \boldsymbol{g}_s\boldsymbol{g}_s^\top\right)^{-1}$. On the rounds where $\boldsymbol{u}_t \neq \boldsymbol{u}_{t-1}$, Algorithm 1 computes H_t using a full matrix inverse, costing $O(d^\omega)$. On the other hand, on a round t where $\boldsymbol{u}_t = \boldsymbol{u}_{t-1}$, we have $A_t = (A_{t-1}^{-1} + \beta\boldsymbol{g}_t\boldsymbol{g}_t^\top)^{-1}$. Thus, A_t can be updated in $O(d^2)$ using A_{t-1} (which is maintain by Algorithm 1) and a rank-one update of the inverse. Further, since $H_t = \left(A_t^{-1} + \frac{4\eta d \boldsymbol{w}_t\boldsymbol{w}_t^\top}{(1-\|\boldsymbol{w}_t\|^2)^2}\right)^{-1}$, H_t can also be computed in $O(d^2)$ using A_t and another

rank-one update of the inverse. Since the number of rounds where $\boldsymbol{u}_t \neq \boldsymbol{u}_{t-1}$ is bounded by $\widetilde{O}(\sqrt{T})$ (Lemma 4), the total cost of computing (H_t) and $(\boldsymbol{w}_t)$ is at most $\widetilde{O}(d^2T + d^\omega\sqrt{T})$.

Having established that $(\boldsymbol{w}_t)$ approximate Newton iterates well (thanks to Lemma 5 and Line 8 of Algorithm 1), we are now in a good position to bound the regret of Algorithm 1.

3.2. Regret Guarantee

First, we note that when Assumption 1 holds with $\mathfrak{B} \le B$, for some $B > 0$, and the losses (ℓ_t) are α-exp-concave, then for $\beta \le \frac{1}{8B} \wedge \frac{\alpha}{2}$, we have

$$\ell_t(\boldsymbol{w}_t) - \ell_t(\boldsymbol{w}) \le \langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \beta\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2/2, \quad \text{for all } \boldsymbol{w} \in \mathcal{B}(1). \tag{5}$$

This result follows from (Hazan et al., 2007, Lemma 3). Thus, to bound the regret of Algorithm 1, it suffices to bound the sum $\sum_{t=1}^T(\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \beta\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2/2)$. To get a better handle on this sum, we will use the FTRL iterates $(\boldsymbol{x}_t)$ in (3); in particular, we will add and subtract terms of the form $\langle \boldsymbol{x}_t, \boldsymbol{g}_t\rangle$ and use Hölder's inequality to obtain the following bound (see details in Appendix A):

$$\begin{aligned}
&\sum_{t=1}^T \left(\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \frac{\beta}{2}\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2\right) \\
&\le \sum_{t=1}^T \left(\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \frac{\beta}{2}\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2\right) + (1 + 2\beta B)\sum_{t=1}^T \|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)}\|\boldsymbol{g}_t\|_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)},
\end{aligned} \tag{6}$$

for all $\boldsymbol{w} \in \mathcal{B}(1)$. The first sum on the RHS of (6) is the regret of FTRL with respect to the surrogate losses $(\boldsymbol{w} \mapsto \langle \boldsymbol{w}, \boldsymbol{g}_t\rangle + \beta\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle/2)$; these can be thought of as quadratic approximations of the actual losses (ℓ_t). The remaining term in (6) can be bounded by $\sum_{t\in[T]} \frac{1+2\beta B}{\sqrt{\eta}}\|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)}$ using that the local norms $(\|\boldsymbol{g}_t\|_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)})$ of the gradients are bounded by $1/\sqrt{\eta}$ thanks to our choice of regularizer Φ_t (we bound these local norms in Lemma 19 in the appendix). Thus, in light of (6) and (5), to bound the regret of Algorithm 1 it suffices to:

I. Bound the sum $\sum_{t\in[T]} \|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)}$.

II. Bound the regret of FTRL with respect to the surrogate losses $(\boldsymbol{w} \mapsto \langle \boldsymbol{w}, \boldsymbol{g}_t\rangle + \beta\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle/2)$.

Point I: Bounding the sum of deviations. The sum in Point I measures the deviation of the outputs of Algorithm 1 from the FTRL iterates in the norms induces by the Hessians of the potentials (Φ_t). In Lemma 20 of App. C.1, we bound the sum in Point I from above by $\frac{16\sqrt{d}}{\beta\sqrt{\eta}}\ln(d + \frac{B^2T}{d})$ using that:

(i) $(\boldsymbol{w}_t)$ are approximate Newton iterates; this follows from (4) and Lemma 5 in the prequel.

(ii) The Newton iterates are close to the FTRL iterates $(\boldsymbol{x}_t)$ for an appropriate choice of parameters (B, η, β). Our proof of the latter fact (see proof of Lemma 20) is similar to one by Mhammedi and Rakhlin (2022) who used damped Newton iterates to approximate FTRL iterates.

We will also use facts (i) and (ii) in the proof of our main theorem in this section (Theorem 7) to show that the iterates $(\boldsymbol{w}_t)$ of Algorithm 1 are always within $\mathcal{B}(1)$.

Point II: Bounding the FTRL surrogate regret. It remains to bound the regret of FTRL with respect to the surrogate losses, which we do next (see proof in Appendix C.4):

Lemma 6 (Surrogate regret of FTRL) *Let $\beta \in (0,1)$, $B > 0$, and $\eta \geq 1$. If Assumption 1 holds with $\mathfrak{B} \leq B$, then the iterates $(\boldsymbol{x}_t)$ in (3) satisfy, for all $\boldsymbol{w} \in \operatorname{int} \mathcal{B}(1)$,*

$$\sum_{t=1}^{T} \left(\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle - \frac{\beta}{2} \langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle^2 \right) \leq \Psi(\boldsymbol{w}) + \frac{d + \eta B^2}{2} \|\boldsymbol{w}\|^2 + \left(\frac{2d}{\beta} + \frac{32\sqrt{d}}{3\eta^2} \right) \ln(d + B^2 T/d).$$

We can now bound the (surrogate) regret of Alg. 1 using the bound on the sum of deviations in Point I, the surrogate regret bound of FTRL, and (6) (see proof in Appendix C.5):

Theorem 7 (Surrogate regret of Alg. 1) *Let $\beta \in (0, 1/8)$, $c \in (0,1)$, $B > 0$, and $\eta \geq 11$. Further, let $(\boldsymbol{w}_t)$ be the iterates of Algorithm 1 with parameters (B, η, β, c) and suppose that Assumption 1 holds with $\mathfrak{B} \leq B$. Then, we have $\boldsymbol{w}_t \in \operatorname{int} \mathcal{B}(1)$, for all $t \geq 1$, and for all $\boldsymbol{w} \in \operatorname{int} \mathcal{B}(1)$,*

$$\sum_{t=1}^{T} \left(\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle - \frac{\beta}{2} \langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle^2 \right) \leq \Psi(\boldsymbol{w}) + \frac{d + \eta B^2}{2} \|\boldsymbol{w}\|^2 + \left(\frac{3d}{\beta} + B d^{1/2} \right) \ln(d + B^2 T/d). \tag{7}$$

The total computational complexity of the instance of Algorithm 1 under consideration is bounded by $O\left(m d^2 T + c^{-1} d^{\omega} \sqrt{\beta^{-1} T \ln(d + B^2 T/d)} \right)$, where $m = O(\ln T)$ is as in Algorithm 3.

From surrogate regret to actual regret. For any $\beta \leq \frac{1}{8B} \wedge \frac{\alpha}{2}$, Eq. (5) and Theorem 7 together immediately imply a $O(d \ln(d+T))$ bound on the actual regret $\sum_{t=1}^{T} (\ell_t(\boldsymbol{w}_t) - \ell_t(\boldsymbol{w}))$, for all $\boldsymbol{w} \in \mathcal{B}(1 - \frac{1}{T})$; shrinking the unit ball by $(1 - \frac{1}{T})$ ensures that $\Psi(\boldsymbol{w})$ in (7) is at most $O(\ln T)$. This guarantee can easily be extended to the whole unit ball $\mathcal{B}(1)$ for B-Lipschitz losses (ℓ_t) using that $\ell_t((1-\frac{1}{T})\boldsymbol{w}) \leq \ell_t(\boldsymbol{w}) + \frac{1}{T}(\ell_t(0) - \ell_t(\boldsymbol{w})) \leq \ell_t(\boldsymbol{w}) + \frac{B}{T}$, for all $\boldsymbol{w} \in \mathcal{B}(1)$, by convexity of the losses (ℓ_t). Finally, we note that we state a bound on the surrogate regret in Theorem 7 instead of the actual regret because it will be convenient in the sequel when we generalize beyond the Euclidean ball.

Remark 8 *We note that the final regret bound we will get will have an additive $O(B^2)$ term that is not present in the bound ONS; the latter is of order $O(d(\beta^{-1} + B)\ln T)$ (Hazan et al., 2007). This term stems from the regularization term $\eta B^2 \|\boldsymbol{x}\|^2/2$ in our definition of Φ_t, which our current analysis requires.*[2]

Computational complexity. Finally, we note that from Theorem 7, the average per-iteration computational cost of Algorithm 1 is bounded by $\widetilde{O}(d^2 + d^{\omega}/\sqrt{T})$. We will later leverage this fact to show that Algorithm 1 can be used in the stochastic exp-concave setting to find an ε-optimal point in $\widetilde{O}(d^3/\varepsilon)$ time complexity (this remains true even if we take $\omega = 3$), which improves over the previous best $\widetilde{O}(d^4/\varepsilon)$ (Koren, 2013). In the next section, we generalize the result of Theorem 7 beyond the Euclidean ball for both the online and stochastic settings.

2. We thank anonymous reviewers for pointing out that the additive $O(B^2)$ term is not present in the ONS bound. While it may be possible to removed it with a more refined analysis, we realized that this may involve some significant changes to the analysis, and so we leave this exercise for future work.

4. Online/Stochastic Exp-Concave Optimization Over General Sets

In this section, we extend the results of Section 3 to a general convex set $\mathcal{C}$. One approach to achieving this would be to simply swap the log-barrier for the Euclidean ball in the definition of Φ_t in (1) with a self-concordant barrier for the set $\mathcal{C}$. This would come with two main challenges. First, the barrier of a general convex set $\mathcal{C}$ that is, say, not a ball or a polytope, is typically hard to compute. This means that our algorithm from the previous section, which requires the gradients and Hessians of the barrier of the set of interest, is not a candidate for efficient exp-concave optimization over general convex sets. The second challenge is in the ability to approximate the inverse Hessian of an arbitrary barrier via a Taylor expansion. The fact that we were able to do this for the log-barrier of the Euclidean ball has to do with the special structure of the Hessian in this case. In particular, the Hessian of Ψ (the log-barrier of the Euclidean ball) at a point $\boldsymbol{w}$ depends only on the norm $\|\boldsymbol{w}\|$ of $\boldsymbol{w}$ and the outer product $\boldsymbol{w}\boldsymbol{w}^\top$. If we ignore the outer product part, which is easy to deal with when it comes to computing the inverse Hessian thanks to the Sherman–Morrison formula, we are left with only a dependence in the norm $\|\boldsymbol{w}\|$. Therefore, a Taylor expansion in 1d is sufficient to approximate the inverse of the Hessian of Ψ at $\boldsymbol{w}$. This is exactly what we do in Algorithm 1. For the barrier of a general convex set, one would require a multivariate Taylor expansion to approximate the inverse of its Hessian, which can not always be done efficiently.

Given the challenges faced when changing the barrier regularizer to extend Algorithm 1 to general convex sets, we instead reduce the OXO problem over general convex sets to one over the Euclidean ball. In the rest of this section, we present our new efficient OXO algorithm; Algorithm 2 with subroutine $\mathcal{A}$ set as Algorithm 1. Algorithm 2, which is taken from (Cutkosky, 2020; Mhammedi, 2022), reduces OCO over any convex set $\mathcal{C}$ to OCO over a Euclidean ball. In fact, Algorithm 2 (an algorithm over $\mathcal{C}$) essentially inherits the regret guarantee of its subroutine $\mathcal{A}$, which is a subroutine over a Euclidean ball that contains $\mathcal{C}$. Thus, the problem becomes one of designing a subroutine $\mathcal{A}$ with a good regret guarantee for exp-concave losses, which we have already tackled in Section 3. Next, we describe and analyze Algorithm 2 in the general OXO setting before specializing the results to the stochastic exp-concave setting in Section 4.2.

4.1. Efficient Online Exp-Concave Optimization via Reduction to the Ball

Before stating the regret guarantee of our algorithm in the online setting, we first formalize the assumptions we make starting with the Lipschitzness and exp-concavity of the losses.

Assumption 2 *For $\alpha > 0$, the functions $(f_t : \mathcal{C} \to \mathbb{R})$ in Algorithm 2 are α-exp-concave and $\sup\{\|\boldsymbol{\zeta}\| : \boldsymbol{\zeta} \in \partial f_t(\boldsymbol{w})\} \leq 1$, for any $\boldsymbol{w} \in \mathcal{C}$ (i.e. (f_t) are 1-Lipschitz). Furthermore, $\mathcal{C} \subseteq \mathcal{B}(1)$.*

We note that assuming that (f_t) are 1-Lipschitz and $\mathcal{C} \subseteq \mathcal{B}(1)$ comes with no loss of generality as one can always re-scale the losses and the set $\mathcal{C}$ to satisfy this condition, provided the Lipschitz constant is known. When the Lipschitz constant is unknown, it is possible to adapt to it using known techniques such as those in (Mhammedi et al., 2019; Cutkosky, 2019; Mhammedi and Koolen, 2020). For some of our results in this section, we will assume that the set $\mathcal{C}$ is centrally symmetric.

Assumption 3 *The set $\mathcal{C}$ is centrally-symmetric, i.e. $\mathcal{C} = -\mathcal{C}$, and $\mathcal{B}(1/\sqrt{d}) \subseteq \mathcal{C} \subseteq \mathcal{B}(1)$.*

Here again, there is no loss of generality in assuming that $\mathcal{B}(1/\sqrt{d}) \subseteq \mathcal{C} \subseteq \mathcal{B}(1)$ when the set is centrally-symmetric since, in this case, it is always possible to apply a certain affine transformation (that puts the set into the isotropic position) to satisfy this condition (see e.g. (Lovász and Vempala,

Algorithm 2 A Reduction to Online Exp-Concave Optimization Over the Euclidean ball.

Require: **I)** An OCO algorithm $\mathcal{A}$ over $\mathcal{B}(1) \supseteq \mathcal{C}$; **II)** A convex function $\rho: \mathbb{R}^d \to \mathbb{R}_{\geq 0} \cup \{+\infty\}$.

Initialize $\mathcal{A}$ and set $\boldsymbol{w}_1 \in \mathcal{B}(1)$ to $\mathcal{A}$'s first output.
for $t = 1, 2, \ldots$ **do**
  Define $S(\boldsymbol{w}) = \inf_{\boldsymbol{x} \in \mathcal{C}} \rho(\boldsymbol{w} - \boldsymbol{x})$. `//Easy to compute for the intended choices of` ρ
  Set $\boldsymbol{\nu}_t \in \partial S(\boldsymbol{w}_t)$ and $\gamma_t = \langle \boldsymbol{\nu}_t, \boldsymbol{w}_t \rangle$.
  Play $\boldsymbol{u}_t = \mathbb{I}_{\{\gamma_t \geq 1\}} \boldsymbol{w}_t / \gamma_t + \mathbb{I}_{\{\gamma_t < 1\}} \boldsymbol{w}_t$. `//`$\boldsymbol{u}_t$ `represents the "projection" of` $\boldsymbol{w}_t$ `onto` $\mathcal{C}$.
  Observe subgradient $\boldsymbol{\zeta}_t \in \partial f_t(\boldsymbol{u}_t)$.
  Set $\boldsymbol{g}_t = \boldsymbol{\zeta}_t - \mathbb{I}_{\langle \boldsymbol{\zeta}_t, \boldsymbol{w}_t \rangle < 0} \langle \boldsymbol{\zeta}_t, \boldsymbol{u}_t \rangle \boldsymbol{\nu}_t$
  Set $\mathcal{A}$'s tth loss function to $\ell_t : \boldsymbol{w} \mapsto \langle \boldsymbol{g}_t, \boldsymbol{w} \rangle$.
  Set $\boldsymbol{w}_{t+1} \in \mathcal{B}(1)$ to $\mathcal{A}$'s $(t+1)$th output given the history $((\boldsymbol{w}_i, \ell_i)_{i \leq t})$.
end for

2006, Section 5)). With our approach, we are able to leverage the fact that a set is centrally-symmetric for more efficient OXO (see Remark 11 below), which is why we treat this case separately in what follows.

The next lemma essentially states that the instantaneous regret of Algorithm 2 can be bounded by that of its subroutine $\mathcal{A}$, and bounds the norm of the subgradients that the latter receives.

Lemma 9 *Suppose that Assumption 2 holds and let* $\beta \leq \frac{1}{8} \wedge \frac{\alpha}{2}$. *Then, the following holds:*

(a) If $\rho(\cdot)$ *in Algorithm 2 is set to* $\|\cdot\|$, *then* $\|\boldsymbol{g}_t\| \leq 1$.

(b) If $\rho(\cdot)$ *in Algorithm 2 is set to* $\gamma_{\mathcal{C}}(\cdot)$ *and* $\mathcal{C}$ *satisfies Assumption 3, then* $\|\boldsymbol{g}_t\| \leq 1 + \sqrt{d}$.

Furthermore, the iterates $(\boldsymbol{u}_t)$ *of the instance of Algorithm 2 in either* (a) *or* (b) *satisfy*

$$\forall \boldsymbol{w} \in \mathcal{C}, \quad f_t(\boldsymbol{u}_t) - f_t(\boldsymbol{w}) \leq \langle \boldsymbol{\zeta}_t, \boldsymbol{u}_t - \boldsymbol{w} \rangle - \frac{\beta}{2} \langle \boldsymbol{\zeta}_t, \boldsymbol{u}_t - \boldsymbol{w} \rangle^2 \leq \langle \boldsymbol{g}_t, \boldsymbol{w}_t - \boldsymbol{w} \rangle - \frac{\beta}{2} \langle \boldsymbol{g}_t, \boldsymbol{w}_t - \boldsymbol{w} \rangle^2. \quad (8)$$

Proof **[Case (a)]** When $\rho(\cdot) \equiv \|\cdot\|$, Alg. 2 matches (Cutkosky, 2020, Alg. 1), and so by (Cutkosky, 2020, Thm. 2), we have that I) $\|\boldsymbol{g}_t\| \leq \|\boldsymbol{\zeta}_t\| \leq 1$ (last inequality follows by Assump. 2), and II)

$$\forall \boldsymbol{w} \in \mathcal{C}, \quad \langle \boldsymbol{\zeta}_t, \boldsymbol{u}_t - \boldsymbol{w} \rangle \leq \langle \boldsymbol{g}_t, \boldsymbol{w}_t - \boldsymbol{w} \rangle. \quad (9)$$

This, together with the fact that the function $x \mapsto x - \beta x^2/2$ is non-decreasing over $[0, 1/\beta]$ and $\langle \boldsymbol{g}_t, \boldsymbol{w}_t - \boldsymbol{w} \rangle \leq \|\boldsymbol{g}_t\| \|\boldsymbol{w}_t - \boldsymbol{w}\| \leq 2 \leq 1/\beta$ implies the second inequality in (8).

[Case (b)] Now, when $\rho(\cdot) \equiv \gamma_{\mathcal{C}}(\cdot)$ ($\gamma_{\mathcal{C}}(\cdot)$ is the gauge function of the set $\mathcal{C}$—see §2), then Algorithm 2 matches (Mhammedi, 2022, Alg. 1), and so by (Mhammedi, 2022, Lemma 7), we have $\|\boldsymbol{g}_t\| \leq (1+\kappa)\|\boldsymbol{\zeta}_t\| \leq 1+\kappa$, where $\kappa = R/r$ and r, R are such that $\mathcal{B}(r) \subseteq \mathcal{C} \subseteq \mathcal{B}(R)$. By Assumption 3, we have $\kappa = \sqrt{d}$ and so $\|\boldsymbol{g}_t\| \leq 1 + \sqrt{d}$. On the other hand, since $\rho(\cdot) \equiv \gamma_{\mathcal{C}}(\cdot)$, the function S in Algorithm 2 satisfies $\langle \boldsymbol{\nu}, \boldsymbol{w} \rangle = \gamma_{\mathcal{C}}(\boldsymbol{w})$, for all $\boldsymbol{\nu} \in \partial S(\boldsymbol{w})$ (see e.g. (Mhammedi, 2022, Lemma 6)). This means that $\gamma_t = \langle \boldsymbol{\nu}_t, \boldsymbol{w}_t \rangle = \gamma_{\mathcal{C}}(\boldsymbol{w}_t)$, and so $\boldsymbol{u}_t = \mathbb{I}_{\{\gamma_{\mathcal{C}}(\boldsymbol{w}_t) \geq 1\}} \boldsymbol{w}_t / \gamma_{\mathcal{C}}(\boldsymbol{w}_t) + \mathbb{I}_{\{\gamma_{\mathcal{C}}(\boldsymbol{w}_t) < 1\}} \boldsymbol{w}_t$. Using this, and the triangle inequality, we get

$$|\langle \boldsymbol{g}_t, \boldsymbol{w}_t \rangle| \leq |\langle \boldsymbol{\zeta}_t, \boldsymbol{w}_t \rangle| + |\langle \boldsymbol{\zeta}_t, \boldsymbol{u}_t \rangle \langle \boldsymbol{\nu}_t, \boldsymbol{w}_t \rangle| = |\langle \boldsymbol{\zeta}_t, \boldsymbol{w}_t \rangle| + |\langle \boldsymbol{\zeta}_t, \boldsymbol{w}_t \rangle \langle \boldsymbol{\nu}_t, \boldsymbol{u}_t \rangle| \overset{(*)}{\leq} 2|\langle \boldsymbol{\zeta}_t, \boldsymbol{w}_t \rangle| \leq 2 \leq \frac{1}{2\beta},$$

where in $(*)$ we used that I) $\boldsymbol{u}_t \in \mathcal{C}$ and $\boldsymbol{\nu}_t \in \partial S(\boldsymbol{w}_t) \subseteq \mathcal{C}^\circ$ (see (Mhammedi, 2022, Lemma 6) for the set inclusion); and II) that $|\langle \boldsymbol{\nu}, \boldsymbol{u}\rangle| \le 1$, for all $\boldsymbol{\nu} \in \mathcal{C}^\circ$ and $\boldsymbol{u} \in \mathcal{C}$, which follows by definition of the polar set $\mathcal{C}^\circ$ (see §2) and the fact that $\mathcal{C}$ is centrally-symmetric. By a similar argument, we also have $|\langle \boldsymbol{g}_t, \boldsymbol{w}\rangle| \le 1/(2\beta)$, for all $\boldsymbol{w} \in \mathcal{C}$, and so

$$|\langle \boldsymbol{g}_t, \boldsymbol{w}_t - \boldsymbol{w}\rangle| \le 1/\beta, \quad \forall \boldsymbol{w} \in \mathcal{C}. \tag{10}$$

On the other hand, by (Mhammedi, 2022, Lemma 7), we also have that $\langle \boldsymbol{\zeta}_t, \boldsymbol{u}_t - \boldsymbol{w}\rangle \le \langle \boldsymbol{g}_t, \boldsymbol{w}_t - \boldsymbol{w}\rangle$, for all $\boldsymbol{w} \in \mathcal{C}$. Using this, Eq. (10), and that the function $x \mapsto x - \beta x^2/2$ is non-decreasing over $[0, 1/\beta]$, implies the second inequality in (8).

Finally, for both cases (a) and (b), the first inequality in (8) follows from Assumption 2 (i.e. the exp-concavity of the losses) and the range assumption on β (see (Hazan et al., 2007, Lemma 3)). ■

Using Lemma 9, we now state the main guarantee of Algorithm 2 when its subroutine $\mathcal{A}$ is set to Algorithm 1, which outputs approximate Newton iterates over the unit Euclidean ball. Before stating this guarantee, we recall that $\mathcal{T}_{\mathsf{sep}}(\mathcal{C})$ [resp. $\mathcal{T}_{\mathsf{proj}}(\mathcal{C})$] denotes the computational complexity of a Separation Oracle [resp. Euclidean Projection Oracle] for the set $\mathcal{C}$.

Theorem 10 *Suppose that Assumption 2 holds and the subroutine $\mathcal{A}$ in Algorithm 2 is set to Alg. 1 with parameters $B > 0$, $\eta \ge 11$, $c = 1/4$, and $\beta \le \frac{1}{8} \wedge \frac{\alpha}{2}$. Then, the following holds:*

(a) When $\rho(\cdot)$ in Alg. 2 is set to $\|\cdot\|$ and $B = 1$, the regret of Alg. 2 after T rounds is bounded by $O(d(\alpha^{-1} + G)\ln(dT))$, and the comp. complexity is bounded by $\widetilde{O}((\mathcal{T}_{\mathsf{proj}}(\mathcal{C}) + d^2)T + d^\omega T^{\frac{1}{2}})$.

(b) When $\rho(\cdot)$ is set to $\gamma_{\mathcal{C}}(\cdot)$; $\mathcal{C}$ satisfies Assumption 3 (i.e. $\mathcal{C}$ is centrally-symmetric); and $B = 1 + \sqrt{d}$, the regret of Alg. 2 after T rounds is bounded by $O(d(\alpha^{-1} + G)\ln(dT))$, and the total computational complexity is bounded by $\widetilde{O}((\mathcal{T}_{\mathsf{sep}}(\mathcal{C}) + d^2)T + d^\omega T^{\frac{1}{2}})$.

Proof We first analyze the regret then consider the computational complexity. Fix $\boldsymbol{w} \in \mathcal{C}$ and let $\widetilde{\boldsymbol{w}} := (1 - 1/T)\boldsymbol{w}$. We bound the regret of Alg. 2 as

$$\begin{aligned}
\sum_{t=1}^T (f_t(\boldsymbol{u}_t) - f_t(\boldsymbol{w})) &= \sum_{t=1}^T (f_t(\boldsymbol{u}_t) - f_t(\widetilde{\boldsymbol{w}})) + \sum_{t=1}^T (f_t(\widetilde{\boldsymbol{w}}) - f_t(\boldsymbol{w})), \\
&\le \sum_{t=1}^T (f_t(\boldsymbol{u}_t) - f_t(\widetilde{\boldsymbol{w}})) + 1, \quad ((f_t) \text{ are 1-Lipschitz and } \mathcal{C} \subseteq \mathcal{B}(1)) \\
&\le \sum_{t=1}^T \left(\langle \boldsymbol{w}_t - \widetilde{\boldsymbol{w}}, \boldsymbol{g}_t\rangle - \frac{\beta}{2}\langle \boldsymbol{w}_t - \widetilde{\boldsymbol{w}}, \boldsymbol{g}_t\rangle^2 \right) + 1. \quad \text{(by Lemma 9)}
\end{aligned} \tag{11}$$

The first sum on the RHS of (11) is the surrogate regret (see §3.2) of Alg. 1—the subroutine $\mathcal{A}$ of Alg. 2—against comparator $\widetilde{\boldsymbol{w}}$. To bound this surrogate regret, we will use Theorem 7. But first, we need to verify that Assumption 1, under which Theorem 7 holds, is satisfied with $\mathfrak{B} = B$ for the sequence $(\boldsymbol{g}_t)$. Thanks to Assumption 2 [resp. 3] and Lemma 9, Assumption 1 is satisfied for the sequence $(\boldsymbol{g}_t)$ with $\mathfrak{B} = 1$ [resp. $\mathfrak{B} = 1 + \sqrt{d}$] when $\rho(\cdot) \equiv \|\cdot\|$ [resp. $\rho(\cdot) \equiv \gamma_{\mathcal{C}}(\cdot)$]. Thus, by Theorem 7, Eq. (11), and the facts that $1 - \|\widetilde{\boldsymbol{w}}\|^2 \ge 1 - (1 - \frac{1}{T})^2 = \frac{2}{T} - \frac{1}{T^2} \ge \frac{1}{T}$ and $B\sqrt{d} \le 2d$ (for both cases (a) and (b)), we get that in both cases of the theorem's statement:

$$\sum_{t=1}^T (f_t(\boldsymbol{u}_t) - f_t(\boldsymbol{w})) \le \eta d \ln T + \frac{d + \eta d}{2}\|\widetilde{\boldsymbol{w}}\|^2 + \frac{5d\ln(d+T)}{\beta}. \tag{12}$$

Using that $\|\widetilde{\boldsymbol{w}}\| \le 1$ in (12) implies the desired regret bound.

The computational complexity of Algorithm 2 is bounded by that of subroutine $\mathcal{A}$, which by Lemma 4 is less than $\widetilde{O}(d^2T + d^\omega\sqrt{T})$, plus T times the computational complexity $\mathcal{T}_{\mathsf{grad}}(S)$ of evaluating a subgradient of S (this is required in Line 4 of Alg. 2). In case (a), S is differentiable everywhere except at the origin and $\nabla S(\boldsymbol{w}) = (\boldsymbol{w} - \Pi_{\mathcal{C}}(\boldsymbol{w}))/\|\boldsymbol{w} - \Pi_{\mathcal{C}}(\boldsymbol{w})\|$, for $\boldsymbol{w} \in \mathbb{R}^d \setminus \{\mathbf{0}\}$, where $\Pi_{\mathcal{C}}(\boldsymbol{w})$ denotes the Euclidean projection of $\boldsymbol{w}$ onto $\mathcal{C}$. Thus, $\mathcal{T}_{\mathsf{grad}}(S) \le O(\mathcal{T}_{\mathsf{proj}}(\mathcal{C}))$. In case (b), i.e. when $\rho(\cdot) \equiv \gamma_{\mathcal{C}}(\cdot)$, we know that $\mathcal{T}_{\mathsf{grad}}(S)$ can be bounded by $\widetilde{O}(\mathcal{T}_{\mathsf{sep}}(\mathcal{C}))$ since S and its subgradients can be evaluated using a Separation Oracle and a binary search (see e.g. (Mhammedi, 2022, §B.2) and (Lee et al., 2018)). ■

Remark 11 (Computational Complexity) *The function S is convex for the choices of ρ in Theorem 10, and its subgradients (which are required in Alg. 2) can be computed using either a Euclidean projection Oracle in case $\rho(\cdot) \equiv \|\cdot\|$, or a Separation Oracle with a binary search routine in case $\rho(\cdot) \equiv \gamma_{\mathcal{C}}(\cdot)$ (see e.g. (Mhammedi, 2022)). For many sets of interest $\mathcal{C}$, a Separation Oracle can be implemented in complexity $\mathcal{T}_{\mathsf{sep}}(\mathcal{C}) \le \widetilde{O}(d^2)$. A particularly relevant case is when a Membership Oracle for $\mathcal{C}$ can be implemented in $O(d)$ time, then $\mathcal{T}_{\mathsf{sep}}(\mathcal{C}) \le \widetilde{O}(d^2)$ (Lee et al., 2018). In many cases, we also have $\mathcal{T}_{\mathsf{proj}}(\mathcal{C}) \le \widetilde{O}(d^2)$ and, crucially, $\mathcal{T}_{\mathsf{proj}}(\mathcal{C})$ can be much smaller than the cost of a Mahalanobis projection, which is require by ONS. Finally, since $\mathcal{T}_{\mathsf{sep}}(\mathcal{C}) \le \mathcal{T}_{\mathsf{proj}}(\mathcal{C})$ in general, our approach is able to leverage that a set is centrally-symmetric for more efficient OXO.*

4.2. Application to Stochastic Exp-Concave Optimization

We now instantiate our results in the stochastic setting where the sequence of losses (f_t) are of the form $f_t(\cdot) = f(\cdot, \xi_t)$, where (ξ_t) are i.i.d. such that $f(\cdot) = \mathbb{E}[f(\cdot, \xi_t)]$ and f is an exp-concave function. Specifically, we will make the following standard assumption in line with Koren (2013).

Assumption 4 *The functions (f_t) in Alg. 2 are such that $f_t(\cdot) = f(\cdot, \xi_t)$ and $\xi_1, \xi_2, \dots$ are i.i.d. random variables in some set Ξ and for all $\xi \in \Xi$, $\boldsymbol{w} \to f(\boldsymbol{w}, \xi)$ is α-exp-concave, for $\alpha > 0$, and $\sup\{\|\boldsymbol{\zeta}\| : \boldsymbol{\zeta} \in \partial^{(1,0)} f(\boldsymbol{w}, \xi)\} \le 1$. Furthermore, $\mathcal{C} \subseteq \mathcal{B}(1)$.*

The assumption that $\sup\{\|\boldsymbol{\zeta}\| : \boldsymbol{\zeta} \in \partial^{(1,0)} f(\boldsymbol{w}, \xi)\} \le 1$ comes with no loss of generality as we can always re-scale the losses. We note that Assumption 4 implies Assumption 2, and so the results of Theorem 10 apply. Using online-to-batch-conversion, the results of Theorem 10 translate into excess-risk bounds in the stochastic setting (the proof of the next theorem, which uses Theorem 10 and online-to-batch-conversion, is somewhat standard and we postpone it to Appendix C.6).

Theorem 12 *Let $\varepsilon \le 1/d$. Suppose that Assumption 4 holds and the subroutine $\mathcal{A}$ in Algorithm 2 is set to Alg. 1 with parameters $B > 0, \eta \ge 11$, $c = 1/4$, and $\beta = \frac{1}{8} \wedge \frac{\alpha}{2}$. Further, suppose that either*

(a) $\rho(\cdot)$ in Alg 2 is set to $\|\cdot\|$ and $B = 1$, or

(b) $\rho(\cdot)$ is set to $\gamma_{\mathcal{C}}(\cdot)$, $\mathcal{C}$ satisfies Assumption 3 (i.e. $\mathcal{C}$ is centrally-symmetric), and $B = 1 + \sqrt{d}$.

Then, for $T = \frac{d \ln(d/\varepsilon)}{\alpha\varepsilon}$ and $\bar{\boldsymbol{u}}_T := \frac{1}{T}\sum_{t=1}^T \boldsymbol{u}_t$, where $(\boldsymbol{u}_t)$ are the iterates of Alg. 2, we have

$$\mathbb{E}\left[f(\bar{\boldsymbol{u}}_T) - \inf_{\boldsymbol{u} \in \mathcal{C}} f(\boldsymbol{u})\right] \le O(\varepsilon), \quad \text{where} \quad f(\cdot) := \mathbb{E}[f(\cdot, \xi)].$$

The comp. costs in cases (a) and (b) are, respectively, $\widetilde{O}(\frac{d}{\varepsilon}(d^2 + \mathcal{T}_{\mathsf{proj}}(\mathcal{C})))$ and $\widetilde{O}(\frac{d}{\varepsilon}(d^2 + \mathcal{T}_{\mathsf{sep}}(\mathcal{C})))$.

In the regime where $\varepsilon > \frac{1}{d}$ (i.e. the regime not covered by Theorem 12), one can simply use projected OGD to find an ε-optimal solution with total computational complexity at most $O(\frac{1}{\varepsilon^2}(d+\mathcal{T}_{\mathsf{proj}}(\mathcal{C}))) \leq O(\frac{d}{\varepsilon}(d+\mathcal{T}_{\mathsf{proj}}(\mathcal{C})))$ (where the last inequality follows by the fact that $\varepsilon > \frac{1}{d}$), which is better than the computational complexity in Theorem 12 for general convex sets. (Though, we note that for small enough ε, the complexity of OGD becomes worse than that of our algorithm and ONS.)

We also note that the instance of Algorithm 2 in Theorem 12 can be used as a black box within an existing meta-algorithm due to (Mehta, 2017, Algorithm 1) to achieve an excess risk guarantee with high probability (instead of in expectation). The computational complexity of the meta algorithm will only be worse than that of the instance of Alg. 2 in Theorem 12 by log factors in T and d.

Implications for the open problem by Koren (2013). The observation made in the previous paragraph and the results of Theorem 12 directly answer one of the questions posed by Koren (2013). There, Koren (2013) asks about the existence of an algorithm for stochastic exp-concave optimization over the Euclidean ball that can find an ε-optimal point using fewer than $\widetilde{O}(d^4/\varepsilon)$ arithmetic operations. For the case of a Euclidean ball, we have $\mathcal{T}_{\mathsf{proj}}(\mathcal{C}) \leq O(d)$, and so the instance of Algorithm 2 in Theorem 12 [resp. OGD] finds an ε-optimal point in less than $\widetilde{O}(d^3/\varepsilon)$ time when $\varepsilon \leq 1/d$ [resp. $\varepsilon > 1/d$]. This remains true for general [resp. centrally-symmetric] convex sets as long as $\mathcal{T}_{\mathsf{proj}}(\mathcal{C}) \leq \widetilde{O}(d^2)$ [resp. $\mathcal{T}_{\mathsf{sep}}(\mathcal{C}) \leq \widetilde{O}(d^2)$], and even if we take the matrix multiplication constant to be $\omega = 3$. We conjecture that $O(d^3/\varepsilon)$ is the best one can do in general (see Appendix E).

Acknowledgment

ZM acknowledges support from the ONR through awards N00014-20-1-2336 and N00014-20-1-2394. We thank Adam Block and Ayush Sekhari for their helpful comments on the presentation.

References

Jacob D Abernethy, Elad Hazan, and Alexander Rakhlin. Interior-point methods for full-information and bandit online learning. *IEEE Transactions on Information Theory*, 58(7):4164–4175, 2012.

Naman Agarwal, Satyen Kale, and Julian Zimmert. Efficient methods for online multiclass logistic regression. In *International Conference on Algorithmic Learning Theory*, pages 3–33. PMLR, 2022.

Katy S Azoury and Manfred K Warmuth. Relative loss bounds for on-line density estimation with the exponential family of distributions. *Machine Learning*, 43(3):211–246, 2001.

Peter L Bartlett, Wouter M Koolen, Alan Malek, Eiji Takimoto, and Manfred K Warmuth. Minimax fixed-design linear regression. In *Conference on Learning Theory*, pages 226–239. PMLR, 2015.

Richard H Byrd, Samantha L Hansen, Jorge Nocedal, and Yoram Singer. A stochastic quasi-newton method for large-scale optimization. *SIAM Journal on Optimization*, 26(2):1008–1031, 2016.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Stein Clifford. *Introduction to Algorithms*, volume 3rd edition. MIT Press, Cambridge, MA, 2009.

Thomas M Cover. Universal portfolios. *Mathematical Finance*, 1(1):1–29, 1991.

Ashok Cutkosky. Artificial constraints and hints for unbounded online learning. In *Conference on Learning Theory*, pages 874–894. PMLR, 2019.

Ashok Cutkosky. Parameter-free, dynamic, and strongly-adaptive online learning. In *International Conference on Machine Learning*, pages 2250–2259. PMLR, 2020.

Ashok Cutkosky and Francesco Orabona. Black-box reductions for parameter-free online learning in banach spaces. In *Conference On Learning Theory*, pages 1493–1529. PMLR, 2018.

Dean P Foster. Prediction in the worst case. *The Annals of Statistics*, pages 1084–1090, 1991.

Dylan J Foster, Satyen Kale, Haipeng Luo, Mehryar Mohri, and Karthik Sridharan. Logistic regression: The importance of being improper. In *Conference On Learning Theory*, pages 167–208. PMLR, 2018.

Pierre Gaillard, Sébastien Gerchinovitz, Malo Huard, and Gilles Stoltz. Uniform regret bounds over rd for the sequential linear regression problem with the square loss. In *Algorithmic Learning Theory*, pages 404–432. PMLR, 2019.

Philip E Gill and Walter Murray. Quasi-newton methods for unconstrained optimization. *IMA Journal of Applied Mathematics*, 9(1):91–108, 1972.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Fundamentals of convex analysis*. Springer Science & Business Media, 2004.

S Indrapriyadarsini, Shahrzad Mahboubi, Hiroshi Ninomiya, and Hideki Asai. A stochastic quasi-newton method with nesterov's accelerated gradient. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 743–760. Springer, 2019.

Rémi Jézéquel, Dmitrii M Ostrovskii, and Pierre Gaillard. Efficient and near-optimal online portfolio selection. *arXiv preprint arXiv:2209.13932*, 2022.

Tomer Koren. Open problem: Fast stochastic exp-concave optimization. In *Conference on Learning Theory*, pages 1073–1075. PMLR, 2013.

Yin Tat Lee, Aaron Sidford, and Santosh S Vempala. Efficient convex optimization with membership oracles. In *Conference On Learning Theory*, pages 1292–1294. PMLR, 2018.

László Lovász and Santosh Vempala. Simulated annealing in convex bodies and an o*(n4) volume algorithm. *Journal of Computer and System Sciences*, 72(2):392–417, 2006.

Haipeng Luo, Alekh Agarwal, Nicolo Cesa-Bianchi, and John Langford. Efficient second order online learning by sketching. *Advances in Neural Information Processing Systems*, 29, 2016.

Haipeng Luo, Chen-Yu Wei, and Kai Zheng. Efficient online portfolio with logarithmic regret. *Advances in neural information processing systems*, 31, 2018.

Mehrdad Mahdavi, Lijun Zhang, and Rong Jin. Lower and upper bounds on the generalization of stochastic exponentially concave optimization. In *Conference on Learning Theory*, pages 1305–1320. PMLR, 2015.

Jack J. Mayo, Hédi Hadiji, and Tim van Erven. Scale-free unconstrained online learning for curved losses. In *Conference on Learning Theory, 2-5 July 2022, London, UK*, volume 178, pages 4464–4497. PMLR, 2022.

Nishant Mehta. Fast rates with high probability in exp-concave statistical learning. In *Artificial Intelligence and Statistics*, pages 1085–1093. PMLR, 2017.

Zakaria Mhammedi. Efficient projection-free online convex optimization with membership oracle. In *Conference on Learning Theory*, pages 5314–5390. PMLR, 2022.

Zakaria Mhammedi and Wouter M. Koolen. Lipschitz and comparator-norm adaptivity in online learning. In *Proceedings of Thirty Third Conference on Learning Theory*, volume 125, pages 2858–2887. PMLR, 2020.

Zakaria Mhammedi and Alexander Rakhlin. Damped online newton step for portfolio selection. In *Conference on Learning Theory, 2-5 July 2022, London, UK*, volume 178 of *Proceedings of Machine Learning Research*, pages 5561–5595. PMLR, 2022.

Zakaria Mhammedi, Wouter M Koolen, and Tim van Erven. Lipschitz adaptivity with multiple learning rates in online learning. In *Conference on Learning Theory*, pages 2490–2511. PMLR, 2019.

Aryan Mokhtari and Alejandro Ribeiro. Stochastic quasi-newton methods. *Proceedings of the IEEE*, 108(11):1906–1922, 2020.

Arkadi S Nemirovski and Michael J Todd. Interior-point methods for optimization. *Acta Numerica*, 17:191–234, 2008.

Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning via sequential complexities. *J. Mach. Learn. Res.*, 16(1):155–186, 2015.

Philippe Rigollet and Jan-Christian Hutter. High dimensional statistics. *Lecture Notes*, 2019. URL https://math.mit.edu/~rigollet/PDFs/RigNotes17.pdf.

Nicol N Schraudolph, Jin Yu, and Simon Günter. A stochastic quasi-newton method for online convex optimization. In *Artificial intelligence and statistics*, pages 436–443. PMLR, 2007.

P. M. Vaidya. An algorithm for linear programming which requires o(((m+n)n2+(m+n)1.5n)l) arithmetic operations. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, page 29–38, New York, NY, USA, 1987. Association for Computing Machinery. ISBN 0897912217.

Tim van Erven, Dirk van der Hoeven, Wojciech Kotlowski, and Wouter M. Koolen. Open problem: Fast and optimal online portfolio selection. In *Conference on Learning Theory*, pages 3864–3869. PMLR, 2020.

Volodya Vovk. Competitive on-line linear regression. *Advances in Neural Information Processing Systems*, 10, 1997.

Volodya Vovk. Competitive on-line statistics. *International Statistical Review*, 69(2):213–248, 2001.

Contents

1 Introduction **1**

2 Preliminaries **4**

3 Efficient Online Exp-Concave Optimization Over a Ball **4**
3.1 Efficient Computation of Newton Iterates 5
3.2 Regret Guarantee 7

4 Online/Stochastic Exp-Concave Optimization Over General Sets **9**
4.1 Efficient Online Exp-Concave Optimization via Reduction to the Ball 9
4.2 Application to Stochastic Exp-Concave Optimization 12

Appendices **18**

A Omitted Pseudocode and Regret Decomposition Details **18**
A.1 Pseudocode of Algorithm 1 18
A.2 Details on the Surrogate Regret Decomposition 19

B Background on Self-Concordant Functions **19**

C Proofs of Section 3 **20**
C.1 Helper Lemmas 20
C.2 Proof of Lemma 4 21
C.3 Proof of Lemma 5 23
C.4 Proof of Lemma 6 24
C.5 Proof of Theorem 7 25
C.6 Proof of Theorem 12 26

D Proofs of Helper Lemmas **26**
D.1 Proof of Lemma 18 26
D.2 Proof of Lemma 19 26
D.3 Proof of Lemma 20 27

E Special Case of Linear Regression **31**

Appendices

Appendix A. Omitted Pseudocode and Regret Decomposition Details

A.1. Pseudocode of Algorithm 1

Algorithm 3 Pseudocode for OQNS (Algorithm 1).

Require: Parameters $B, \eta, \beta, c > 0$, and $m = \left\lceil -\log_c \left(\frac{12(4+32/\eta^2)^2(2\eta d + B^2\eta + (B+2\beta B^2)T)^2 T}{(1-c)} \right) \right\rceil$.

Set $\boldsymbol{u}_1 = \boldsymbol{w}_1 = \mathbf{0}$, $V_0 = 0$, $A_0 = I/(2\eta d + d + \eta B^2)$, and $\mathbf{S}_0 = \mathbf{G}_0 = \mathbf{0}$.
for $t = 1, \dots, T$ **do**
  Play $\boldsymbol{w}_t$ and observe $\boldsymbol{g}_t \in \partial \ell_t(\boldsymbol{w}_t)$.
  Set $\mathbf{G}_t = \mathbf{G}_{t-1} + \boldsymbol{g}_t$, $\mathbf{S}_t = \mathbf{S}_{t-1} + \boldsymbol{g}_t \boldsymbol{g}_t^\top \boldsymbol{w}_t$, and $V_t = V_{t-1} + \boldsymbol{g}_t \boldsymbol{g}_t^\top$.
  Set $\nabla_t = \frac{2\eta d \boldsymbol{w}_t}{1-\|\boldsymbol{w}_t\|^2} + (d + \eta B^2)\boldsymbol{w}_t + \beta V_t \boldsymbol{w}_t - \beta \mathbf{S}_t + \mathbf{G}_t$. // $\nabla_t = \nabla \Phi_{t+1}(\boldsymbol{w}_t)$
  Set $A_t = A_{t-1} - \frac{\beta A_{t-1} \boldsymbol{g}_t \boldsymbol{g}_t^\top A_{t-1}}{2 + \beta \boldsymbol{g}_t^\top A_{t-1} \boldsymbol{g}_t}$ and $H_t = A_t - \frac{4\eta d A_t \boldsymbol{w}_t \boldsymbol{w}_t^\top A_t}{(1-\|\boldsymbol{w}_t\|^2)^2 + 4d\eta \boldsymbol{w}_t^\top A_t \boldsymbol{w}_t}$.
  Set $\boldsymbol{\Delta}_t = \widetilde{\boldsymbol{\Delta}}_t = H_t \nabla_t$.
  for $k = 1, \dots, m$ **do**
    Update $\widetilde{\boldsymbol{\Delta}}_t \leftarrow \left(\frac{2\eta d}{1-\|\boldsymbol{u}_t\|^2} - \frac{2\eta d}{1-\|\boldsymbol{w}_t\|^2} \right) H_t \widetilde{\boldsymbol{\Delta}}_t$.
    Update $\boldsymbol{\Delta}_t \leftarrow \boldsymbol{\Delta}_t + \widetilde{\boldsymbol{\Delta}}_t$.
  end for
  Set $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \boldsymbol{\Delta}_t$. // $\boldsymbol{w}_{t+1} \approx \boldsymbol{w}_t - \nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)\nabla \Phi_{t+1}(\boldsymbol{w}_t)$.
  if $|\|\boldsymbol{w}_{t+1}\|^2 - \|\boldsymbol{u}_t\|^2| \le c \cdot (1 - \|\boldsymbol{u}_t\|^2)$ **then**
    Set $\boldsymbol{u}_{t+1} = \boldsymbol{u}_t$.
  else
    Set $\boldsymbol{u}_{t+1} = \boldsymbol{w}_{t+1}$.
    Set $A_t = \left(\frac{2\eta d I}{1-\|\boldsymbol{w}_{t+1}\|^2} + dI + \eta B^2 I + \beta V_t \right)^{-1}$. // $A_t = \left(\nabla^2 \Phi_{t+1}(\boldsymbol{w}_{t+1}) - \frac{4\eta d \boldsymbol{w}_{t+1} \boldsymbol{w}_{t+1}^\top}{(1-\|\boldsymbol{w}_{t+1}\|^2)^2} \right)^{-1}$
  end if
end for

A.2. Details on the Surrogate Regret Decomposition

In this subsection, we provide more details on the regret decomposition in (6) under Assumption 1 with $\mathfrak{B} \le B$. For all $\boldsymbol{w} \in \mathcal{B}(1)$, we have

$$\begin{aligned}
&\sum_{t=1}^{T}\left(\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \frac{\beta}{2}\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2\right)\\
&= \sum_{t=1}^{T}\left(\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \frac{\beta}{2}(\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle + \langle \boldsymbol{w}_t - \boldsymbol{x}_t, \boldsymbol{g}_t\rangle)^2\right) + \sum_{t=1}^{T}\langle \boldsymbol{w}_t - \boldsymbol{x}_t, \boldsymbol{g}_t\rangle,\\
&= \sum_{t=1}^{T}\left(\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \frac{\beta}{2}\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2\right) + \sum_{t=1}^{T}(1 - \beta\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle)\langle \boldsymbol{w}_t - \boldsymbol{x}_t, \boldsymbol{g}_t\rangle - \frac{\beta}{2}\sum_{t=1}^{T}\langle \boldsymbol{w}_t - \boldsymbol{x}_t, \boldsymbol{g}_t\rangle^2,\\
&\le \sum_{t=1}^{T}\left(\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \frac{\beta}{2}\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2\right) + \sum_{t=1}^{T}(1 - \beta\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle)\langle \boldsymbol{w}_t - \boldsymbol{x}_t, \boldsymbol{g}_t\rangle,\\
&\le \sum_{t=1}^{T}\left(\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle - \frac{\beta}{2}\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2\right) + (1 + 2\beta B)\sum_{t=1}^{T}\|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)}\|\boldsymbol{g}_t\|_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)}, \qquad (13)
\end{aligned}$$

where the last inequality follows by Hölder's inequality (to bound $|\langle \boldsymbol{w}_t - \boldsymbol{x}_t, \boldsymbol{g}_t\rangle|$), and that $|\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle| \le 2B$ (this follows by the fact that Assumption 1 holds with $\mathfrak{B} \le B$ and that $\boldsymbol{x}_t, \boldsymbol{w} \in \mathcal{B}(1)$).

Appendix B. Background on Self-Concordant Functions

In this section, we define self-concordant functions and present some of their properties that we make heavy use of in the proofs of our results. We start by the definition of a self-concordant function. For the rest of this section, we let $\mathcal{K}$ be a convex compact set with non-empty interior $\operatorname{int}\mathcal{K}$. For a twice [resp. thrice] differentiable function, we let $\nabla^2 f(\boldsymbol{u})$ [resp. $\nabla^3 f(\boldsymbol{u})$] be the Hessian [resp. third derivative tensor] of f at $\boldsymbol{u}$.

Definition 13 *A convex function $f\colon \operatorname{int}\mathcal{K} \to \mathbb{R}$ is called* self-concordant *with constant $M_f \ge 0$, if f is C^3 and satisfies* **I)** *$f(\boldsymbol{x}_k) \to +\infty$ for $\boldsymbol{x}_k \to \boldsymbol{x} \in \partial\mathcal{K}$; and* **II)**

$$\forall \boldsymbol{x} \in \operatorname{int}\mathcal{K}, \forall \boldsymbol{u} \in \mathbb{R}^d, \quad |\nabla^3 f(\boldsymbol{x})[\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{u}]| \le 2M_f\|\boldsymbol{u}\|^3_{\nabla^2 f(\boldsymbol{x})}.$$

Note that by definition, if f is self-concordant with constant $M_f \ge 0$ it is also self-concordant with any constant $M \ge M_f$. For a self-concordant function f and $\boldsymbol{x} \in \operatorname{dom} f$, the quantity $\lambda(\boldsymbol{x}, f) \coloneqq \|\nabla f(\boldsymbol{x})\|_{\nabla^{-2} f(\boldsymbol{x})}$, known as the *Newton decrement*, will be instrumental in our proofs. The following two lemmas contain properties of the Newton decrement and Hessians of self-concordant functions, which we will use repeatedly throughout (see e.g. Nemirovski and Todd (2008); Nesterov et al. (2018)).

Lemma 14 *Let $f\colon \operatorname{int}\mathcal{K} \to \mathbb{R}$ be a self-concordant function with constant $M_f \ge 1$. Further, let $\boldsymbol{x} \in \operatorname{int}\mathcal{K}$ and $\boldsymbol{x}_f \in \operatorname{argmin}_{\boldsymbol{x}\in\mathcal{K}} f(\boldsymbol{x})$. Then,* **I)** *whenever $\lambda(\boldsymbol{x}, f) < 1/M_f$, we have*

$$\|\boldsymbol{x} - \boldsymbol{x}_f\|_{\nabla^2 f(\boldsymbol{x}_f)} \vee \|\boldsymbol{x} - \boldsymbol{x}_f\|_{\nabla^2 f(\boldsymbol{x})} \le \lambda(\boldsymbol{x}, f)/(1 - M_f\lambda(\boldsymbol{x}, f));$$

and **II)** *for any $M \ge M_f$, the* Newton step *$\boldsymbol{x}_+ \coloneqq \boldsymbol{x} - \nabla^{-2} f(\boldsymbol{x})\nabla f(\boldsymbol{x})$ satisfies $\boldsymbol{x}_+ \in \operatorname{int}\mathcal{K}$ and $\lambda(\boldsymbol{x}_+, f) \le M\lambda(\boldsymbol{x}, f)^2/(1 - M\lambda(\boldsymbol{x}, f))^2$.*

Lemma 15 *Let $f\colon \operatorname{int}\mathcal{K} \to \mathbb{R}$ be a self-concordant function with constant M_f and $\boldsymbol{x} \in \operatorname{int}\mathcal{K}$. Then, for any $\boldsymbol{y}$ such that $r \coloneqq \|\boldsymbol{y}-\boldsymbol{x}\|_{\nabla^2 f(\boldsymbol{x})} < 1/M_f$, we have*

$$(1-M_f r)^2 \nabla^2 f(\boldsymbol{y}) \preceq \nabla^2 f(\boldsymbol{x}) \preceq (1-M_f r)^{-2} \nabla^2 f(\boldsymbol{x}).$$

The following result from (Nesterov et al., 2018, Theorem 5.1.5) will be useful to show that the iterates of our algorithms are always within the feasible set.

Lemma 16 *Let $f\colon \operatorname{int}\mathcal{K} \to \mathbb{R}$ be a self-concordant function with constant $M_f \geq 1$ and $\boldsymbol{x} \in \operatorname{int}\mathcal{K}$. Then, $\mathcal{E}_{\boldsymbol{x}} \coloneqq \{\boldsymbol{w} \in \mathbb{R}^d \colon \|\boldsymbol{w}-\boldsymbol{x}\|_{\nabla^2 f(\boldsymbol{x})} < 1/M_f\} \subseteq \operatorname{int}\mathcal{K}$. Furthermore, for all $\boldsymbol{w} \in \mathcal{E}_{\boldsymbol{x}}$, we have*

$$\|\boldsymbol{w}-\boldsymbol{x}\|_{\nabla^2 f(\boldsymbol{w})} \leq \frac{\|\boldsymbol{w}-\boldsymbol{x}\|_{\nabla^2 f(\boldsymbol{x})}}{1-M_f \|\boldsymbol{w}-\boldsymbol{x}\|_{\nabla^2 f(\boldsymbol{x})}}.$$

Finally, we will also make use of the following result due to Mhammedi and Rakhlin (2022):

Lemma 17 *Let $f\colon \operatorname{int}\mathcal{K} \to \mathbb{R}$ be a self-concordant function with constant $M_f > 0$. Then, for any $\boldsymbol{x}, \boldsymbol{y} \in \operatorname{int}\mathcal{K}$ such that $r \coloneqq \|\boldsymbol{x}-\boldsymbol{y}\|_{\nabla^2 f(\boldsymbol{x})} < 1/M_f$, we have*

$$\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\|^2_{\nabla^{-2} f(\boldsymbol{x})} \leq \frac{1}{(1-M_f r)^2} \|\boldsymbol{y}-\boldsymbol{x}\|^2_{\nabla^2 f(\boldsymbol{x})}.$$

We now have all the tools we require for the analysis of our OXO algorithms.

Appendix C. Proofs of Section 3

In this section, we prove the statements in Section 3. For this, we need a set of helper lemmas that we state in the next subsection.

C.1. Helper Lemmas

The proofs of this section are in Appendix D. First, we establish that the functions (Φ_t) are self-concordant.

Lemma 18 *The function Φ_t in (1) is a self-concordant function with constant $M_{\Phi_t} \leq 1/\sqrt{d\eta}$.*

The next lemma gives a bound on the local gradients norms, which will be useful throughout (the proof is similar to ones in (Luo et al., 2018; Mhammedi and Rakhlin, 2022)):

Lemma 19 *Let $\beta \in (0,1)$, $B > 0$, and $\eta \geq 1$. Further, let $(\boldsymbol{g}_t)$ be such that $\|\boldsymbol{g}_t\| \leq B$, for all $t \geq 1$. Then, for any sequence $(\boldsymbol{y}_t) \subset \operatorname{int}\mathcal{B}(1)$, the potential functions (Φ_t) in (1) satisfy*

$$\forall t \in [T], \quad \|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{y}_t)} \leq 1/\eta \quad \text{and} \quad \sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{y}_t)} \leq \frac{d \ln(d + TB^2/d)}{\beta}.$$

The main technical heavy lifting in the paper is done in the proof of the next lemma. Some of the steps in the proof of the lemma that involve bounding the distance between $\boldsymbol{w}_t$ and $\boldsymbol{x}_t$ are similar to those found in (Abernethy et al., 2012, Proof of Lemma 4.1) and (Mhammedi and Rakhlin, 2022, Proof of Lemma 8).

Lemma 20 (Master Lemma) *Let $\beta, c \in (0,1)$, $B > 0$, and $\eta \geq 1$. Further, let $(\boldsymbol{w}_t)$ be the iterates of Algorithm 1 with parameters (B, η, β, c) and suppose that Assumption 1 holds with $\mathfrak{B} \leq B$. Then, we have* I) *$(\boldsymbol{w}_t) \subset \operatorname{int} \mathcal{B}(1)$; and* II)

$$\forall t \geq 1, \quad \frac{\sqrt{\eta d}}{4}\left(\|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2 \Phi_t(\boldsymbol{w}_t)} - \frac{1}{T}\right) \leq \frac{\sqrt{\eta d}}{2}\left(\lambda(\boldsymbol{w}_t, \Phi_t) - \frac{1}{T}\right) \leq \lambda(\boldsymbol{w}_{t-1}, \Phi_t)^2 \leq \frac{4}{\eta}.$$

Further, we have $\sum_{t=1}^T \|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2 \Phi_t(\boldsymbol{w}_t)} \leq 1 + \frac{16\sqrt{d}}{3\beta\sqrt{\eta}} \ln(d + \frac{B^2 T}{d})$ and

$$\sum_{t=1}^T \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|^2_{\nabla^2 \Psi(\boldsymbol{w}_t)} + \sum_{t=1}^T \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|^2_{\nabla^2 \Psi(\boldsymbol{w}_{t-1})} \leq \frac{8d \ln(d + B^2 T/d)}{\beta}. \tag{14}$$

C.2. Proof of Lemma 4

For the proof of Lemma 4, we need the following elementary result.

Lemma 21 *Let $\Psi(\boldsymbol{x}) \coloneqq -\eta d \ln(1 - \|\boldsymbol{x}\|^2)$. For any $\boldsymbol{u}, \boldsymbol{w} \in \mathcal{B}(1)$, we have*

$$\frac{1}{\eta d}\|\boldsymbol{w} - \boldsymbol{u}\|^2_{\nabla^2 \Psi(\boldsymbol{w})} \geq \frac{(\|\boldsymbol{w}\|^2 - \|\boldsymbol{u}\|^2)^2}{(1 - \|\boldsymbol{w}\|^2)^2}.$$

Proof Fix $\boldsymbol{u}, \boldsymbol{w} \in \mathcal{B}(1)$. We have

$$\begin{aligned}
\frac{1}{2\eta d}\|\boldsymbol{w} - \boldsymbol{u}\|^2_{\nabla^2 \Psi(\boldsymbol{w})} &= (\boldsymbol{w} - \boldsymbol{u})^\top \left(\frac{I}{1 - \|\boldsymbol{w}\|^2} + \frac{2\boldsymbol{w}\boldsymbol{w}^\top}{(1 - \|\boldsymbol{w}\|^2)^2}\right)(\boldsymbol{w} - \boldsymbol{u}), \\
&= \frac{\|\boldsymbol{w}\|^2 + \|\boldsymbol{u}\|^2 - 2\boldsymbol{w}^\top \boldsymbol{u} - 2\|\boldsymbol{w}\|^2 \boldsymbol{w}^\top \boldsymbol{u} + \|\boldsymbol{w}\|^4 + 2(\boldsymbol{w}^\top \boldsymbol{u})^2 - \|\boldsymbol{w}\|^2 \|\boldsymbol{u}\|^2}{(1 - \|\boldsymbol{w}\|^2)^2}, \\
&= \frac{2(\|\boldsymbol{w}\|^2 - \boldsymbol{w}^\top \boldsymbol{u})(1 - \boldsymbol{w}^\top \boldsymbol{u})}{(1 - \|\boldsymbol{w}\|^2)^2} + \frac{\|\boldsymbol{u}\|^2 - \|\boldsymbol{w}\|^2}{1 - \|\boldsymbol{w}\|^2}, \\
&= \frac{2(\|\boldsymbol{w}\|^2 - \boldsymbol{w}^\top \boldsymbol{u})(1 - \|\boldsymbol{w}\|^2)}{(1 - \|\boldsymbol{w}\|^2)^2} + \frac{2(\|\boldsymbol{w}\|^2 - \boldsymbol{w}^\top \boldsymbol{u})^2}{(1 - \|\boldsymbol{w}\|^2)^2} + \frac{\|\boldsymbol{u}\|^2 - \|\boldsymbol{w}\|^2}{1 - \|\boldsymbol{w}\|^2}.
\end{aligned}$$

Now using that $-\boldsymbol{w}\boldsymbol{u} = 2^{-1}(\|\boldsymbol{w} - \boldsymbol{u}\|^2 - \|\boldsymbol{w}\|^2 - \|\boldsymbol{u}\|^2)$, we get that

$$\begin{aligned}
\frac{1}{2\eta d}\|\boldsymbol{w} - \boldsymbol{u}\|^2_{\nabla^2 \Psi(\boldsymbol{w})} &= \frac{\|\boldsymbol{w} - \boldsymbol{u}\|^2 + \|\boldsymbol{w}\|^2 - \|\boldsymbol{u}\|^2}{1 - \|\boldsymbol{w}\|^2} + \frac{2(\|\boldsymbol{w}\|^2 - \boldsymbol{w}^\top \boldsymbol{u})^2}{(1 - \|\boldsymbol{w}\|^2)^2} + \frac{\|\boldsymbol{u}\|^2 - \|\boldsymbol{w}\|^2}{1 - \|\boldsymbol{w}\|^2}, \\
&= \frac{\|\boldsymbol{w} - \boldsymbol{u}\|^2}{1 - \|\boldsymbol{w}\|^2} + \frac{(\|\boldsymbol{w} - \boldsymbol{u}\|^2 + \|\boldsymbol{w}\|^2 - \|\boldsymbol{u}\|^2)^2}{2(1 - \|\boldsymbol{w}\|^2)^2}.
\end{aligned} \tag{15}$$

Now consider the function $f : X \to \frac{X}{1 - \|\boldsymbol{w}\|^2} + \frac{(X + \|\boldsymbol{w}\|^2 - \|\boldsymbol{u}\|^2)^2}{2(1 - \|\boldsymbol{w}\|^2)^2}$. Note that $\operatorname{sgn}(f'(X)) = \operatorname{sgn}(X - \|\boldsymbol{u}\|^2 + 1)$. Thus, since $\|\boldsymbol{u}\|^2 \leq 1$, the function f is non-decreasing over $\mathbb{R}_{\geq 0}$, and so $f(\|\boldsymbol{w} - \boldsymbol{u}\|^2) \geq f(0)$. Using this with (15), we get

$$\frac{1}{\eta d}\|\boldsymbol{w} - \boldsymbol{u}\|^2_{\nabla^2 \Psi(\boldsymbol{w})} \geq \frac{(\|\boldsymbol{w}\|^2 - \|\boldsymbol{u}\|^2)^2}{(1 - \|\boldsymbol{w}\|^2)^2}.$$

■

Proof of Lemma 4 Let $i_1, \dots, i_n$ be the rounds t where $\boldsymbol{u}_t \neq \boldsymbol{u}_{t-1}$, and note that by Line 9 of Algorithm 1, we have

$$|\|\boldsymbol{u}_{i_{k+1}}\|^2 - \|\boldsymbol{u}_{i_k}\|^2| > c \cdot (1 - \|\boldsymbol{u}_{i_k}\|^2), \quad \forall k \in [n-1]. \tag{16}$$

Further, let

$$\alpha_t \coloneqq \frac{\|\boldsymbol{w}_{t+1}\|^2 - \|\boldsymbol{w}_t\|^2}{1 - \|\boldsymbol{w}_{t+1}\|^2}, \quad \text{and} \quad \mu_t \coloneqq \frac{\|\boldsymbol{w}_t\|^2 - \|\boldsymbol{w}_{t+1}\|^2}{1 - \|\boldsymbol{w}_t\|^2}.$$

Fix $k \in [n-1]$. Suppose that $(\sum_{t=i_k}^{i_{k+1}-1} \alpha_t) \vee (\sum_{t=i_k}^{i_{k+1}-1} \mu_t) \leq 1/2$ and let $m_k \coloneqq i_{k+1} - i_k$. In this case, by (16) we have that

$$\begin{aligned}
\ln(1+c) &\leq \left(\ln \frac{1 - \|\boldsymbol{u}_{i_k}\|^2}{1 - \|\boldsymbol{u}_{i_{k+1}}\|^2}\right) \vee \left(\ln \frac{1 - \|\boldsymbol{u}_{i_{k+1}}\|^2}{1 - \|\boldsymbol{u}_{i_k}\|^2}\right), \\
&\leq \left(\ln \prod_{t=i_k}^{i_{k+1}-1} (1+\alpha_t)\right) \vee \left(\ln \prod_{t=i_k}^{i_{k+1}-1} (1+\mu_t)\right), \\
&= \left(\sum_{t=i_k}^{i_{k+1}-1} \ln(1+\alpha_t)\right) \vee \left(\sum_{t=i_k}^{i_{k+1}-1} \ln(1+\mu_t)\right), \\
&\leq \ln\left(1 + \frac{1}{m_k} \sum_{t=i_k}^{i_{k+1}-1} \alpha_t\right)^{m_k} \vee \ln\left(1 + \frac{1}{m_k} \sum_{t=i_k}^{i_{k+1}-1} \mu_t\right)^{m_k}, \quad \text{(Jensen)} \\
&\leq \ln\left(1 + 2\sum_{t=i_k}^{i_{k+1}-1} \alpha_t\right) \vee \ln\left(1 + 2\sum_{t=i_k}^{i_{k+1}-1} \mu_t\right),
\end{aligned}$$

where the last inequality follows by the facts that $(\sum_{t=i_k}^{i_{k+1}-1} \alpha_t) \vee (\sum_{t=i_k}^{i_{k+1}-1} \mu_t) \leq 1/2$ and $(1+x)^r \leq 1 + \frac{rx}{1-(r-1)x}$, for all $x \in (-1, \frac{1}{r-1}]$ and $r \geq 1$. Now, using that $\ln(1+x) \leq x$ for $x \geq 0$ and $\ln(1+x) \geq x/2$, for $x \in (0,1)$, we get that

$$\begin{aligned}
\frac{c}{2} \leq \ln(1+c) &\leq \left(2\sum_{t=i_k}^{i_{k+1}-1} \alpha_t\right) \vee \left(2\sum_{t=i_k}^{i_{k+1}-1} \mu_t\right), &(17)\\
&\leq 2\left(\sqrt{m_k \sum_{t=i_k}^{i_{k+1}-1} \alpha_t^2}\right) \vee \left(\sqrt{m_k \sum_{t=i_k}^{i_{k+1}-1} \mu_t^2}\right), \quad \text{(Jensen)} \\
&\leq 2\sqrt{m_k \sum_{t=i_k}^{i_{k+1}-1} \alpha_t^2 + m_k \sum_{t=i_k}^{i_{k+1}-1} \mu_t^2}. &(18)
\end{aligned}$$

So far, we have assumed that $(\sum_{t=i_k}^{i_{k+1}-1} \alpha_t) \vee (\sum_{t=i_k}^{i_{k+1}-1} \mu_t) \leq 1/2$. If this does not hold, then we have $(\sum_{t=i_k}^{i_{k+1}-1} \alpha_t) \vee (\sum_{t=i_k}^{i_{k+1}-1} \mu_t) \geq 1/2$. This implies (17) from which (18) follows. Now, (18) implies

$$\sum_{t=i_k}^{i_{k+1}-1} \alpha_t^2 + \sum_{t=i_k}^{i_{k+1}-1} \mu_t^2 \geq \frac{c^2}{16 m_k}.$$

Thus, by summing over $k = 1, \dots, n-1$, and using Lemma 21 and Lemma 20 (in particular (14)), we get

$$\frac{8\ln(d + B^2T/d)}{\eta\beta} \geq \sum_{t=1}^{T}(\alpha_t^2 + \mu_t^2) \geq \sum_{k=1}^{n}\frac{c^2}{16m_k} = \sum_{k=1}^{n}\frac{c^2}{16(i_{k+1} - i_k)} \geq \frac{c^2n^2}{16T},$$

where the last inequality follows by the fact that $x \mapsto 1/x$ is convex and Jensen's inequality. By rearranging, we get that

$$n \leq 8\sqrt{\frac{2T\ln(d + B^2T/d)}{c^2\eta\beta}}.$$

■

C.3. Proof of Lemma 5

Proof Fix $m \geq 1$ and let $\alpha_t \coloneqq \frac{\|\boldsymbol{w}_t\|^2 - \|\boldsymbol{u}_t\|^2}{1 - \|\boldsymbol{w}_t\|^2}$. We have

$$\alpha_t = \frac{1 - \|\boldsymbol{u}_t\|^2}{1 - \|\boldsymbol{w}_t\|^2} - 1 = -\frac{1 - \|\boldsymbol{u}_t\|^2}{2\eta d}\gamma_t,$$

where we recall that $\gamma_t = \frac{2d\eta}{1 - \|\boldsymbol{u}_t\|^2} - \frac{2d\eta}{1 - \|\boldsymbol{w}_t\|^2}$. Note that H_t in Alg. 1 satisfies

$$\begin{aligned}
H_t^{-1} &= \frac{2\eta dI}{1 - \|\boldsymbol{u}_t\|^2} + \frac{4\eta d\boldsymbol{w}_t\boldsymbol{w}_t^\top}{(1 - \|\boldsymbol{w}_t\|^2)^2} + (d + \eta B^2)I + \beta V_t, \qquad (19)\\
&= \nabla^2\Phi_{t+1}(\boldsymbol{w}_t) - \frac{2\eta dI}{1 - \|\boldsymbol{w}_t\|^2} + \frac{2\eta dI}{1 - \|\boldsymbol{u}_t\|^2},\\
&= \nabla^2\Phi_{t+1}(\boldsymbol{w}_t) - \frac{2\eta dI}{1 - \|\boldsymbol{u}_t\|^2}\left(\frac{1 - \|\boldsymbol{u}_t\|^2}{1 - \|\boldsymbol{w}_t\|^2} - 1\right),\\
&= \nabla^2\Phi_{t+1}(\boldsymbol{w}_t) - \frac{2\eta d\alpha_t I}{1 - \|\boldsymbol{u}_t\|^2}.
\end{aligned}$$

Therefore, if we let $U_t \coloneqq (1 - \|\boldsymbol{u}_t\|^2)H_t^{-1}/(2\eta d)$, we have

$$\begin{aligned}
\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t) &= \left(\frac{2\eta d\alpha_t I}{1 - \|\boldsymbol{u}_t\|^2} + H_t^{-1}\right)^{-1},\\
&= \frac{1 - \|\boldsymbol{u}_t\|^2}{2\eta d}\left(\alpha_t I + \frac{1 - \|\boldsymbol{u}_t\|^2}{2\eta d}H_t^{-1}\right)^{-1},\\
&= \frac{1 - \|\boldsymbol{u}_t\|^2}{2\eta d}(\alpha_t I + U_t)^{-1},\\
&= \frac{1 - \|\boldsymbol{u}_t\|^2}{2\eta d}U_t^{-1}(I + \alpha_t U_t^{-1})^{-1}. \qquad (20)
\end{aligned}$$

Now, by (19), we have $U_t \succeq I$ and so $\|U_t^{-1}\| \leq 1$. Using this and that $|\alpha_t| \leq c < 1$ (this is an invariant of Algorithm 1—see Line 9 of Alg. 1), we have

$$(1+\alpha_t U_t^{-1})^{-1} = \sum_{k=0}^{\infty} (-\alpha_t)^k U_t^{-k}, \quad \text{and} \quad \left\| (1+\alpha_t U_t)^{-1} - \sum_{k=0}^{m} (-\alpha_t)^k U_t^{-k} \right\| \leq \frac{c^m}{1-c}.$$

Therefore, by (20) and the fact that $\|U_t^{-1}\| \leq 1$ we have

$$\left\| \nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t) - \frac{1-\|\boldsymbol{u}_t\|^2}{2d\eta} \sum_{k=1}^{m+1} (-\alpha_t)^{k-1} U_t^{-k} \right\| \leq \frac{(1-\|\boldsymbol{u}_t\|^2)\cdot c^m}{2\eta d \cdot (1-c)}.$$

Now, the fact that $\frac{1-\|\boldsymbol{u}_t\|^2}{2d\eta} \sum_{k=1}^{m+1} (-\alpha_t)^{k-1} U_t^{-k} = \sum_{k=1}^{m+1} \gamma_t^{k-1} H_t^k$ completes the proof. ■

C.4. Proof of Lemma 6

Proof Fix $\boldsymbol{w} \in \mathcal{B}(1)$. Let $\phi_t(\boldsymbol{x}) \coloneqq \boldsymbol{x}^\top \boldsymbol{g}_t + \beta\langle \boldsymbol{g}_t, \boldsymbol{x} - \boldsymbol{w}_t\rangle^2/2$ and $\phi_0(\boldsymbol{x}) \coloneqq \Psi(\boldsymbol{x}) + (d+\eta B^2)\|\boldsymbol{x}\|^2/2$, and note that $\Phi_t(\boldsymbol{x}) = \sum_{s=0}^{t-1} \phi_s(\boldsymbol{x})$ and $\boldsymbol{x}_t \in \operatorname{argmin}_{\boldsymbol{x}\in\mathcal{B}(1)} \sum_{s=0}^{t-1} \phi_s(\boldsymbol{x})$. By (Cesa-Bianchi and Lugosi, 2006, Lemma 3.1), we have

$$\sum_{t=0}^{T} \phi_t(\boldsymbol{x}_{t+1}) \leq \sum_{t=0}^{T} \phi_t(\boldsymbol{w}),$$

which implies that

$$\sum_{t=1}^{T} \langle \boldsymbol{x}_{t+1} - \boldsymbol{w}, \boldsymbol{g}_t\rangle \leq \Psi(\boldsymbol{w}) + \frac{d+\eta B^2}{2}\|\boldsymbol{w}\|^2 + \frac{\beta}{2}\sum_{t=1}^{T} \langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t\rangle^2. \tag{21}$$

Now, if suffices to bound the sum $\sum_{t=1}^{T} \langle \boldsymbol{x}_t - \boldsymbol{x}_{t+1}, \boldsymbol{g}_t\rangle$. By Taylor's theorem, the exists $\boldsymbol{y}_t$ in the segment $[\boldsymbol{x}_t, \boldsymbol{x}_{t+1}]$ such that

$$\begin{aligned}
\Phi_{t+1}(\boldsymbol{x}_t) - \Phi_{t+1}(\boldsymbol{x}_{t+1}) &\geq \nabla\Phi_{t+1}(\boldsymbol{x}_{t+1})^\top(\boldsymbol{x}_t - \boldsymbol{x}_{t+1}) + \frac{1}{2}\|\boldsymbol{x}_t - \boldsymbol{x}_{t+1}\|^2_{\nabla^2\Phi_{t+1}(\boldsymbol{y}_t)}, \\
&\geq \frac{1}{2}\|\boldsymbol{x}_t - \boldsymbol{x}_{t+1}\|^2_{\nabla^2\Phi_{t+1}(\boldsymbol{y}_t)},
\end{aligned} \tag{22}$$

where the last inequality uses the fact that $\boldsymbol{x}_{t+1} \in \operatorname{argmin}_{\boldsymbol{x}\in\mathcal{B}(1)} \Phi_{t+1}(\boldsymbol{x})$ is in the interior of $\mathcal{B}(1)$ by self-concordance of Φ_{t+1}. On the other hand, using the convexity of Φ_{t+1} and the fact that $\nabla\Phi_{t+1}(\boldsymbol{x}_t) = \nabla\phi_{t+1}(\boldsymbol{x}_t) + \nabla\Phi_t(\boldsymbol{x}_t) = \nabla\phi_{t+1}(\boldsymbol{x}_t)$ (by optimality of $\boldsymbol{x}_t$), we get that

$$\begin{aligned}
\Phi_{t+1}(\boldsymbol{x}_t) - \Phi_{t+1}(\boldsymbol{x}_{t+1}) &\leq \langle \boldsymbol{x}_t - \boldsymbol{x}_{t+1}, \nabla\phi_{t+1}(\boldsymbol{x}_t)\rangle, \\
&= \langle \boldsymbol{x}_t - \boldsymbol{x}_{t+1}, \boldsymbol{g}_t\rangle(1+\beta\langle \boldsymbol{g}_t, \boldsymbol{x}_t - \boldsymbol{w}_t\rangle), \\
&\leq \|\boldsymbol{x}_t - \boldsymbol{x}_{t+1}\|_{\nabla^2\Phi_{t+1}(\boldsymbol{y}_t)} \cdot \|\boldsymbol{g}_t\|_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{y}_t)} \cdot (1+\beta\langle \boldsymbol{g}_t, \boldsymbol{x}_t - \boldsymbol{w}_t\rangle).
\end{aligned}$$

Combining this and (22), we get

$$\|\boldsymbol{x}_t - \boldsymbol{x}_{t+1}\|_{\nabla^2\Phi_{t+1}(\boldsymbol{y}_t)} \leq 2\,\|\boldsymbol{g}_t\|_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{y}_t)}\,(1+\beta\langle \boldsymbol{g}_t, \boldsymbol{x}_t - \boldsymbol{w}_t\rangle).$$

Using this and Hölder's inequality leads to

$$\langle \boldsymbol{g}_t, \boldsymbol{x}_t - \boldsymbol{x}_{t+1}\rangle \leq \|\boldsymbol{g}_t\|_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{y}_t)}\,\|\boldsymbol{x}_t - \boldsymbol{x}_{t+1}\|_{\nabla^2\Phi_{t+1}(\boldsymbol{y}_t)} \leq 2\,\|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{y}_t)}\,(1+\beta\langle \boldsymbol{g}_t, \boldsymbol{x}_t - \boldsymbol{w}_t\rangle).$$

Thus, by summing this inequality for $t = 1, \dots, T$, we get that

$$\begin{aligned}
\sum_{t=1}^{T} \langle \boldsymbol{g}_t, \boldsymbol{x}_t - \boldsymbol{x}_{t+1} \rangle &\le 2 \sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{y}_t)} + 2\beta \sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{y}_t)} \langle \boldsymbol{g}_t, \boldsymbol{x}_t - \boldsymbol{w}_t \rangle, \\
&\le 2 \sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{y}_t)} + 2\beta \sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{y}_t)} \|\boldsymbol{g}_t\|_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)} \|\boldsymbol{x}_t - \boldsymbol{w}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)}, \\
&\le \frac{2d \ln(d + B^2T/d)}{\beta} + \frac{2\beta}{\eta^{3/2}} \sum_{t=1}^{T} \|\boldsymbol{x}_t - \boldsymbol{w}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)}, \text{ (Lem. 19 and } \nabla^2\Phi_{t+1} \succeq \nabla^2\Phi_t) \\
&\le \left(\frac{2d}{\beta} + \frac{32d^{1/2}}{3\eta^2} \right) \ln(d + B^2T/d),
\end{aligned}$$

where the last inequality follows from the bound on $\sum_{t=1}^{T} \|\boldsymbol{x}_t - \boldsymbol{w}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)}$ from Lemma 20. Combining this with (21), we get the desired bound. ■

C.5. Proof of Theorem 7

Proof First, the fact that $(\boldsymbol{w}_t) \subset \operatorname{int} \mathcal{B}(1)$ follows from Lemma 20. Now, by the surrogate regret decomposition in (13) and the fact that $\|\boldsymbol{g}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)} \le 1/\sqrt{\eta}$ (Lemma 19), we have, for all $\boldsymbol{w} \in \operatorname{int} \mathcal{B}(1)$,

$$\begin{aligned}
&\sum_{t=1}^{T} \left(\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle - \frac{\beta}{2} \langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle^2 \right) \\
&\le \sum_{t=1}^{T} \left(\langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle - \frac{\beta}{2} \langle \boldsymbol{x}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle^2 \right) + (1 + 2\beta B) \sum_{t=1}^{T} \|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)} / \sqrt{\eta}. \qquad (23)
\end{aligned}$$

The first sum on the RHS (23) represents the surrogate regret of FTRL, and the second sum measures the deviation of the iterates of Alg. 3 from the FTRL iterates $(\boldsymbol{x}_t)$. Plugging the bound on the surrogate regret of FTRL [resp. $\sum_{t=1}^{T} \|\boldsymbol{w}_t - \boldsymbol{p}_t\|_{\nabla^2\Phi_t(\boldsymbol{w}_t)}$] from Lemma 6 [resp. Lemma 20] in (23), we get that, for all $\boldsymbol{w} \in \operatorname{int} \mathcal{B}(1)$,

$$\begin{aligned}
&\sum_{t=1}^{T} (\langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle - \beta \langle \boldsymbol{w}_t - \boldsymbol{w}, \boldsymbol{g}_t \rangle^2 / 2) \\
&\le \Psi(\boldsymbol{w}) + \frac{d + \eta B^2}{2} \|\boldsymbol{w}\|^2 + \left(\frac{2d}{\beta} + \frac{32\sqrt{d}}{3\eta^2} + \frac{16\sqrt{d}}{3\beta\eta} + \frac{32B\sqrt{d}}{3\eta} \right) \ln(d + B^2T/d), \\
&\le \Psi(\boldsymbol{w}) + \frac{d + \eta B^2}{2} \|\boldsymbol{w}\|^2 + \left(\frac{3d}{\beta} + Bd^{\frac{1}{2}} \right) \ln(d + B^2T/d), \qquad (24)
\end{aligned}$$

where (24) follows by the fact that $\frac{32\sqrt{d}}{3\eta^2} + \frac{16\sqrt{d}}{3\beta\eta} \le \frac{d}{\beta}$ and $32/(3\eta) \le 1$ (since $\beta \in (0, 1/8)$, $\eta \ge 11$, and $d \ge 1$).

We now look at the computational complexity of Algorithm 1. The most computationally expansive step in Algorithm 1 is in Line 4, which involves a full matrix inverse when $\boldsymbol{u}_t \ne \boldsymbol{u}_{t-1}$ (see also Line 17 of Algorithm 3; the pseudo-code of Alg. 1). However, by Lemma 4, the inverse need only be computed at most $O(c^{-1}\sqrt{\beta^{-1}T \ln(d + B^2T/d)})$ times after T rounds. The next most

computationally expansive step in Alg. 1 is in Line 8, which involves computing the output $\boldsymbol{w}_{t+1}$. The output $\boldsymbol{w}_{t+1}$ in Line 8 can be computed in $O(md^2)$ using m matrix-vector multiplications (see Lines 7-12 of Alg. 3). Thus, the claim on the total computational complexity of the algorithm follows by the fact that $m \leq O\left(1 + \log_c \frac{d+\eta+\beta+B+T}{1-c}\right)$ (the exact choice of m can be found in Algorithm 3). ■

C.6. Proof of Theorem 12

Proof The result follows by Thm. 10 and standard online-to-batch conversion. If we let $\mathrm{Reg}_T(\cdot)$ be the regret of Alg. 2 in response to the i.i.d. loss functions (f_t) and $\boldsymbol{u}_* \in \operatorname{argmin}_{\boldsymbol{u}\in\mathcal{C}} f(\boldsymbol{u})$, where $f(\cdot) \coloneqq \mathbb{E}[f_t(\cdot)]$, then the average iterate $\bar{\boldsymbol{u}}_T$ of Alg. 2 after T rounds satisfies

$$\mathbb{E}[f(\bar{\boldsymbol{u}}_T)] - \inf_{\boldsymbol{u}\in\mathcal{C}} f(\boldsymbol{u}) \overset{(*)}{\leq} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[f_t(\boldsymbol{u}_t) - f_t(\boldsymbol{u}_*)] = \frac{\mathbb{E}[\mathrm{Reg}_T(\boldsymbol{u}_*)]}{T},$$

where $(*)$ follows by Jensen's inequality and the fact that $\boldsymbol{u}_t$ is independent of f_t. Plugging the regret bounds of Algorithm 2 in the settings of Theorem 10 implies that

$$\mathbb{E}[f(\bar{\boldsymbol{u}}_T)] - \inf_{\boldsymbol{u}\in\mathcal{C}} f(\boldsymbol{u}) \leq O(d(\alpha^{-1} + G)\ln(dT)/T).$$

By choosing $T = \frac{d}{\alpha\varepsilon}\ln\frac{d}{\varepsilon}$, we get that $\mathbb{E}[f(\bar{\boldsymbol{u}}_T)] - \inf_{\boldsymbol{u}\in\mathcal{C}} f(\boldsymbol{u}) \leq O(\varepsilon)$. The bounds on the computational complexity follow directly from those in Theorem 10 by plugging-in the choice $T = \frac{d}{\alpha\varepsilon}\ln\frac{d}{\varepsilon}$ and using the fact that $\varepsilon \leq 1/d$. This conclusion remains true even if we take $\omega = 3$. ■

Appendix D. Proofs of Helper Lemmas

D.1. Proof of Lemma 18

Proof First, we note that $\boldsymbol{x} \mapsto \Psi(\boldsymbol{x})/(\eta d) = -\ln(1 - \|\boldsymbol{x}\|^2)$ is self-concordant with constant 1 (see e.g. (Nesterov et al., 2018, Exampled 5.1.1)). Thus, Ψ is a self-concordant function with constant $1/\sqrt{\eta d}$; this follows by the fact that if a function f is self-concordant with constant M_f, then αf, for $\alpha > 0$, it is self-concordant with constant $1/\sqrt{\alpha}$ (see e.g. (Nesterov et al., 2018, Corollary 5.1.3)). On the other hand, since $\Phi_t(\boldsymbol{x})$ is equal to $\Psi(\boldsymbol{x})$ plus a quadratic in $\boldsymbol{x}$, then Φ_t is self-concordant with the same constant as Ψ (see e.g. (Nesterov et al., 2018, Corollary 5.1.2)). ■

D.2. Proof of Lemma 19

Proof First note that $\eta\boldsymbol{g}_t\boldsymbol{g}_t^\top \leq \eta B^2 I \leq \nabla^2\Phi_t(\boldsymbol{y}_t) - I$. This together with the fact that $\eta \geq \beta$ implies that

$$\|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{y}_t)} \leq \|\boldsymbol{g}_t\|^2_{(I+\eta\boldsymbol{g}_t\boldsymbol{g}_t^\top)^{-1}} = \boldsymbol{g}_t^\top(I + \eta\boldsymbol{g}_t\boldsymbol{g}_t^\top)^{-1}\boldsymbol{g}_t \leq 1/\eta,$$

$$\text{and} \qquad \|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{y}_t)} = \boldsymbol{g}_t^\top\left(\nabla^2\Psi(\boldsymbol{y}_t) + dI + \eta B^2 I + \beta V_{t-1}\right)^{-1}\boldsymbol{g}_t \leq \beta^{-1}\boldsymbol{g}_t^\top Q_t^{-1}\boldsymbol{g}_t. \tag{25}$$

where $Q_t \coloneqq dI + \sum_{s=1}^t \boldsymbol{g}_s\boldsymbol{g}_s^\top$. Thus, by (25) and (Hazan et al., 2007, Lemma 11), we have

$$\|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{y}_t)} \leq \frac{1}{\beta}\sum_{t=1}^{T}\boldsymbol{g}_t^\top Q_t^{-1}\boldsymbol{g}_t \leq \frac{1}{\beta}\ln\frac{|Q_T|}{|Q_0|} \leq \frac{d\ln(d + TB^2/d)}{\beta},$$

where the second inequality uses the fact $\ln|Q_0| = d\ln d$ and by Jensen's inequality $\ln|Q_T| \le d\ln\frac{\operatorname{tr} Q_T}{d} \le d\ln\left(d + \sum_{t=1}^T \|\boldsymbol{g}_t\|_2^2/d\right) \le d\ln(d + B^2T/d)$. This completes the proof. ■

D.3. Proof of Lemma 20

For the proof of Lemma 20, we need two additional lemmas that we now state and prove:

Lemma 22 *Let $\eta, t \ge 1$. If $\boldsymbol{w}_1, \dots, \boldsymbol{w}_{t-1} \in \mathcal{B}(1)$, then the FTRL iterate $\boldsymbol{x}_t$ in (3) satisfies:*

$$\frac{2\eta d}{1 - \|\boldsymbol{x}_t\|^2} \le 2(2\eta d + B^2\eta + (B + 2\beta B^2)(t-1)).$$

Proof Since $\Psi(\boldsymbol{x})$ is a self-concordant barrier, we have $\boldsymbol{x}_t \in \operatorname{int}\mathcal{B}(1)$. Thus, by the first-order optimality condition involving $\boldsymbol{x}_t$, we have

$$\frac{2\eta d\boldsymbol{x}_t}{1 - \|\boldsymbol{x}_t\|^2} + (d + B^2\eta)\boldsymbol{x}_t + \beta\sum_{s=1}^{t-1}\boldsymbol{g}_s\boldsymbol{g}_s^\top(\boldsymbol{x}_t - \boldsymbol{w}_s) + \mathbf{G}_{t-1} = \mathbf{0}.$$

This implies that

$$\frac{2\eta d\|\boldsymbol{x}_t\|}{1 - \|\boldsymbol{x}_t\|^2} \le d + B^2\eta + (B + 2\beta B^2)(t-1). \tag{26}$$

If $\|\boldsymbol{x}_t\| \le 1/2$, then we are done since in this case $1/(1 - \|\boldsymbol{x}_t\|^2) \le 4/3 \le 2$. Otherwise, (26) directly implies that

$$\frac{2\eta d}{1 - \|\boldsymbol{x}_t\|^2} \le 2(2\eta d + B^2\eta + (B + 2\beta B^2)(t-1)).$$

■

Lemma 23 *Let $\eta, t \ge 1$, $C > 0$, and $\boldsymbol{x}_t \in \operatorname{argmin}_{\boldsymbol{x}\in\mathcal{B}(1)}\Phi_t(\boldsymbol{x})$. If $\boldsymbol{w}_1, \dots, \boldsymbol{w}_{t-1} \in \mathcal{B}(1)$, then for any $\boldsymbol{u} \in \operatorname{int}\mathcal{B}(1)$ such that $\|\boldsymbol{u} - \boldsymbol{x}_t\|^2_{\nabla^2\Psi(\boldsymbol{u})} \le \eta C^2$, we have*

$$\|\nabla\Phi_t(\boldsymbol{u})\| \le (4 + 2C)(2\eta d + B^2\eta + (B + 2\beta B^2)(t-1)),$$
$$\text{and} \qquad \nabla^2\Phi_t(\boldsymbol{u}) \le 7(1 + C)^2(2\eta d + B^2\eta + (B + 2\beta B^2)(t-1))^2 I.$$

Proof Fix $\boldsymbol{u} \in \operatorname{int}\mathcal{B}(1)$ such that $\|\boldsymbol{u} - \boldsymbol{x}_t\|^2_{\nabla^2\Psi(\boldsymbol{u})} \le \eta C^2$. By Lemma 21, we have

$$C^2 \ge \left(\frac{\|\boldsymbol{u}\|^2 - \|\boldsymbol{x}_t\|^2}{1 - \|\boldsymbol{u}\|^2}\right)^2 = \left(\frac{1 - \|\boldsymbol{x}_t\|^2}{1 - \|\boldsymbol{u}\|^2} - 1\right)^2.$$

This implies that

$$\frac{2\eta d}{1 - \|\boldsymbol{u}\|^2} \le \frac{2\eta d\cdot(1 + C)}{1 - \|\boldsymbol{x}_t\|^2} \le 2(1 + C)(2\eta d + B^2\eta + (B + 2\beta B^2)(t-1)), \tag{27}$$

where the last inequality follows by Lemma 22. Therefore, by the triangle inequality, we have

$$\begin{aligned}\|\nabla\Phi_t(\boldsymbol{u})\| &\le \left\|\frac{2\eta d\boldsymbol{u}}{1-\|\boldsymbol{u}\|^2}\right\| + \left\|(d+B^2\eta)\boldsymbol{u} + \beta\sum_{s=1}^{t-1}\boldsymbol{g}_s\boldsymbol{g}_s^\top(\boldsymbol{u}-\boldsymbol{w}_s) + \mathbf{G}_{t-1}\right\|,\\ &\le \frac{2\eta d}{1-\|\boldsymbol{u}\|^2} + (d+B^2\eta) + 2\beta B^2 + B(t-1),\\ &\le (4+2C)(2\eta d + B^2\eta + (B+2\beta B^2)(t-1)).\end{aligned}$$

On the other hand, we have

$$\begin{aligned}\nabla^2\Phi_t(\boldsymbol{u}) &= \frac{2\eta d I}{1-\|\boldsymbol{u}\|^2} + \frac{4\eta d\boldsymbol{u}\boldsymbol{u}^\top}{(1-\|\boldsymbol{u}\|^2)^2} + (d+\eta B^2)I + \beta V_t,\\ &\le 7(1+C)^2(2\eta d + B^2\eta + (B+2\beta B^2)(t-1))^2 I.\end{aligned}$$

where the last inequality follows from (27). ■

Proof of Lemma 20 Define

$$\widetilde{\boldsymbol{w}}_{t+1} \coloneqq \boldsymbol{w}_t - \nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)\nabla\Phi_{t+1}(\boldsymbol{w}_t), \quad \text{and} \quad \widetilde{\nabla}_t \coloneqq \sum_{k=1}^{m+1}\left(\frac{2\eta}{1-\|\boldsymbol{u}_t\|^2} - \frac{2\eta}{1-\|\boldsymbol{w}_t\|^2}\right)^{k-1} H_t^k\nabla_t,$$

and note that $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \widetilde{\nabla}_t$. By induction, we will show that for all $s \ge 1$,

$$\boldsymbol{w}_s \in \operatorname{int}\mathcal{B}(1) \;\&\; \frac{\sqrt{\eta d}}{4}(\|\boldsymbol{w}_s - \boldsymbol{x}_s\|_{\nabla^2\Phi_s(\boldsymbol{w}_s)} - \epsilon) \le \frac{\sqrt{\eta d}}{2}(\lambda(\boldsymbol{w}_s,\Phi_s) - \epsilon) \le \lambda(\boldsymbol{w}_{s-1},\Phi_s)^2 \le \frac{4}{\eta}, \tag{28}$$

where $\epsilon = 1/T$ and $\boldsymbol{w}_0 = \mathbf{0}$ by convention. The base case follows trivially since $\nabla\Phi_1(\boldsymbol{w}_0) = \nabla\Phi_1(\boldsymbol{w}_1) = \mathbf{0}$ and $\boldsymbol{w}_1 = \boldsymbol{x}_1$. Suppose that (28) holds for $s = t$. We will show that it holds for $s = t+1$. By the expression of Φ_{t+1} in (1), we have $\nabla\Phi_{t+1}(\boldsymbol{w}_t) = \boldsymbol{g}_t + \nabla\Phi_t(\boldsymbol{w}_t)$, and so by the fact that $(a+b)^2 \le 2a^2 + 2b^2$, we get

$$\begin{aligned}\lambda(\boldsymbol{w}_t,\Phi_{t+1})^2 &= \|\nabla\Phi_{t+1}(\boldsymbol{w}_t)\|^2_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)},\\ &\le 2\|\nabla\Phi_t(\boldsymbol{w}_t)\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)} + 2\|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)}, \quad (\nabla^2\Phi_{t+1}(\cdot) \succeq \nabla^2\Phi_t(\cdot))\\ &= 2\lambda(\boldsymbol{w}_t,\Phi_t)^2 + 2\|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)}, &(29)\\ &\le 2(8^2/\eta^3 + 16\epsilon/\eta^{3/2} + \epsilon^2) + 2/\eta, &(30)\\ &\le 4/\eta, &(31)\end{aligned}$$

where in (30) we used the induction hypothesis in (28) for $s = t$ and the bound on $\|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)}$ from Lemma 19; and (31) uses the range assumptions on η and that $\epsilon = 1/T$. Since $\widetilde{\boldsymbol{w}}_{t+1}$ is the standard Newton step, Lemma 14 and the fact that $\lambda(\boldsymbol{w}_t,\Phi_{t+1}) \le (1-1/\sqrt{2})\sqrt{\eta d}$ (which follows from (31) and the range assumption on η), we have

$$\lambda(\widetilde{\boldsymbol{w}}_{t+1},\Phi_{t+1}) \le \frac{2}{\sqrt{\eta d}}\lambda(\boldsymbol{w}_t,\Phi_{t+1})^2. \tag{32}$$

Furthermore, since $\boldsymbol{x}_{t+1}$ is the minimizer of Φ_{t+1} and $\lambda(\widetilde{\boldsymbol{w}}_{t+1}, \Phi_{t+1}) \le \sqrt{\eta d}/2$, we have $\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{x}_{t+1}\|_{\nabla^2\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})} \le 2\lambda(\widetilde{\boldsymbol{w}}_{t+1}, \Phi_{t+1})$ (by Lemma 14 again). Combining this with (32) and (31) implies that $\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{x}_{t+1}\|^2_{\nabla^2\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})} \le \eta C^2$ with $C = 16/\eta^2$. Thus, Lemma 23 implies that

$$\nabla^2\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1}) \le 7(1 + 32/\eta^2)^2(2\eta d + B^2\eta + (B + 2\beta B^2)t)^2 I. \tag{33}$$

On the other hand, since $\nabla_t = \nabla\Phi_{t+1}(\boldsymbol{w}_t)$ we have,

$$\begin{aligned}
\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_{t+1}\| &= \left\|\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t)\nabla\Phi_{t+1}(\boldsymbol{w}_t) - \sum_{k=1}^{m+1}\left(\frac{2\eta}{1-\|\boldsymbol{u}_t\|^2} - \frac{2\eta}{1-\|\boldsymbol{w}_t\|^2}\right)^{k-1} H_t^k \nabla_t\right\|, \\
&= \left\|\left(\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t) - \sum_{k=1}^{m+1}\left(\frac{2\eta}{1-\|\boldsymbol{u}_t\|^2} - \frac{2\eta}{1-\|\boldsymbol{w}_t\|^2}\right)^{k-1} H_t^k\right)\nabla\Phi_{t+1}(\boldsymbol{w}_t)\right\|, \\
&\le \left\|\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_t) - \sum_{k=1}^{m+1}\left(\frac{2\eta}{1-\|\boldsymbol{u}_t\|^2} - \frac{2\eta}{1-\|\boldsymbol{w}_t\|^2}\right)^{k-1} H_t^k\right\| \cdot \|\nabla\Phi_{t+1}(\boldsymbol{w}_t)\|, \\
&\le \frac{c^m \cdot (4 + 32/\eta^2)(2\eta d + B^2\eta + (B + 2\beta B^2)t)}{2\eta d \cdot (1-c)},
\end{aligned} \tag{34}$$

where the last inequality follows by Lemma 23 (which holds with $C = 16/\eta^2$ by (28)) and Lemma 5. Combining (34) with (33) implies that

$$\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_{t+1}\|_{\nabla^2\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})} \le \epsilon' \coloneqq \frac{3c^m \cdot (4 + 32/\eta^2)^2(2\eta d + B^2\eta + (B + 2\beta B^2)t)^2}{2\eta d \cdot (1-c)}. \tag{35}$$

We now show that this implies that $\boldsymbol{w}_{t+1} \in \mathcal{B}(1)$. First, since $\boldsymbol{w}_t \in \mathcal{B}(1)$ and

$$\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_t\|_{\nabla^2\Phi_{t+1}(\boldsymbol{w}_t)} = \lambda(\boldsymbol{w}_t, \Phi_{t+1}) \overset{(*)}{\le} 2/\sqrt{\eta} < \sqrt{\eta d}/2, \tag{36}$$

where $(*)$ follows by (31), we have that $\widetilde{\boldsymbol{w}}_{t+1} \in \mathcal{B}(1)$ by Lemma 16. Now, by our choice of m in Algorithm 1, we have $\epsilon' < \sqrt{\eta d}/4$, and so (35) implies that $\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_{t+1}\|_{\nabla^2\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})} < \sqrt{\eta d}/4$. Therefore, $\boldsymbol{w}_{t+1} \in \mathcal{B}(1)$ by Lemma 16, since $\widetilde{\boldsymbol{w}}_{t+1} \in \mathcal{B}(1)$.

We now bound the Newton decrement $\lambda(\boldsymbol{w}_{t+1}, \Phi_{t+1})$. First, by Lemma 16 and the fact that $\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_{t+1}\|_{\nabla^2\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})} < \sqrt{\eta d}/4$, we have

$$\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_{t+1}\|_{\nabla^2\Phi_{t+1}(\boldsymbol{w}_{t+1})} \le 2\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_{t+1}\|_{\nabla^2\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})} \le 2\epsilon' < \sqrt{\eta d}/2. \tag{37}$$

Using this, we get

$$\begin{aligned}
\lambda(\boldsymbol{w}_{t+1}, \Phi_{t+1}) &= \|\nabla\Phi_t(\boldsymbol{w}_{t+1})\|_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_{t+1})} \\
&\le \|\nabla\Phi_t(\widetilde{\boldsymbol{w}}_{t+1})\|_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_{t+1})} + \|\nabla\Phi_{t+1}(\boldsymbol{w}_{t+1}) - \nabla\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})\|_{\nabla^{-2}\Phi_{t+1}(\boldsymbol{w}_{t+1})}, \\
&\le (1 - \epsilon'/\sqrt{\eta d})^{-1}\|\nabla\Phi_t(\widetilde{\boldsymbol{w}}_{t+1})\|_{\nabla^{-2}\Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})} + 2\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_{t+1}\|_{\nabla^2\Phi_{t+1}(\boldsymbol{w}_{t+1})}, \qquad (38) \\
&\le \lambda(\widetilde{\boldsymbol{w}}_{t+1}, \Phi_{t+1}) + 2\epsilon' \cdot \lambda(\widetilde{\boldsymbol{w}}_{t+1}, \Phi_{t+1})/\sqrt{\eta d} + 4\epsilon', \\
&\le \lambda(\widetilde{\boldsymbol{w}}_{t+1}, \Phi_{t+1}) + \epsilon, \qquad (39)
\end{aligned}$$

where (38) uses Lemmas 15 and 17, and the last inequality follows by (32), (31) and the fact that $8\epsilon' \le \epsilon = 1/T$ by the choice of m in Algorithm 1. Now, since $\boldsymbol{x}_{t+1}$ is the minimizer of

Φ_{t+1} and $\lambda(\boldsymbol{w}_{t+1}, \Phi_{t+1}) \leq \sqrt{\eta d}/2$ (by (39), (32), and (31)), we have $\|\boldsymbol{w}_{t+1} - \boldsymbol{x}_{t+1}\|_{\nabla^2 \Phi_{t+1}(\boldsymbol{w}_t)} \leq 2\lambda(\boldsymbol{w}_{t+1}, \Phi_{t+1})$ (by Lemma 14). Combining this with (39), (32), and (31), implies (28) for $s = t+1$, which concludes the induction.

We now use (28) together with (29) to bound the sums

$$S_1 \coloneqq \sum_{t=1}^{T} \|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2 \Phi_t(\boldsymbol{w}_t)}, \; S_2 \coloneqq \sum_{t=1}^{T} \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|^2_{\nabla^2 \Psi(\boldsymbol{w}_t)}, \; \& \; S_3 \coloneqq \sum_{t=1}^{T} \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|^2_{\nabla^2 \Psi(\boldsymbol{w}_{t-1})}.$$

To this end, we will first bound the sum $\sum_{t=1}^{T} \lambda(\boldsymbol{w}_t, \Phi_t)^i$, for $i = 1, 2$. Using that $\lambda(\boldsymbol{w}_{t+1}, \Phi_{t+1}) \leq 2(\eta d)^{-1/2} \lambda(\boldsymbol{w}_t, \Phi_{t+1})^2 + \epsilon$ (by (28)) and (29), we get

$$\lambda(\boldsymbol{w}_{t+1}, \Phi_{t+1}) \leq 4(\eta d)^{-1/2} \lambda(\boldsymbol{w}_t, \Phi_t)^2 + 4(\eta d)^{-1/2} \|\boldsymbol{g}_t\|^2_{\nabla^{-2} \Phi_t(\boldsymbol{w}_t)} + \epsilon. \tag{40}$$

Summing (40), for $t = 1, \ldots, T$, rearranging, and using that $\lambda(\boldsymbol{w}_{T+1}, \Phi_{T+1}) \geq 0$, we get

$$\sum_{t=2}^{T} \left(\lambda(\boldsymbol{w}_t, \Phi_t) - \frac{4}{\sqrt{\eta d}} \lambda(\boldsymbol{w}_t, \Phi_t)^2 \right) \leq \frac{4}{\sqrt{\eta d}} \lambda(\boldsymbol{w}_1, \Phi_1)^2 + \frac{4}{\sqrt{\eta d}} \sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2_{\nabla^{-2} \Phi_t(\boldsymbol{w}_t)} + T\epsilon.$$

Using (28) and the range assumption on η, we have $0 \leq \frac{4}{\sqrt{\eta d}} \lambda(\boldsymbol{w}_t, \Phi_t) \leq 32/\eta^2 + 4\epsilon/\sqrt{\eta} \leq 1/4$. Therefore, we have

$$\begin{aligned} \frac{3}{4} \sum_{t=1}^{T} \lambda(\boldsymbol{w}_t, \Phi_t) &\leq \lambda(\boldsymbol{w}_1, \Phi_1) + \frac{4}{\sqrt{\eta d}} \sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2_{\nabla^{-2} \Phi_t(\boldsymbol{w}_t)}, \\ &\leq \frac{1}{16} + \frac{4}{\sqrt{\eta d}} \sum_{t=1}^{T} \|\boldsymbol{g}_t\|^2_{\nabla^{-2} \Phi_t(\boldsymbol{w}_t)} \leq \frac{1}{16} + \frac{4d \ln(d + B^2 T/d)}{\beta \sqrt{\eta d}}, \end{aligned} \tag{41}$$

where the last inequality follows by Lemma 19 and the range assumption on η. Now, using the fact that $\boldsymbol{x}_t$ is the minimizer of Φ_t, we have $\|\boldsymbol{w}_t - \boldsymbol{x}_t\|_{\nabla^2 \Phi_t(\boldsymbol{w}_t)} \leq 2\lambda(\boldsymbol{w}_t, \Phi_t)$, which implies the desired bound the sum S_1. We now bound S_2 and S_3. By Lemma 15 and (37), we have

$$\begin{aligned} \|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\|_{\nabla^2 \Psi(\boldsymbol{w}_{t+1})} &\leq 2\|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\|_{\nabla^2 \Psi(\widetilde{\boldsymbol{w}}_{t+1})}, \\ &\leq 2\|\boldsymbol{w}_{t+1} - \widetilde{\boldsymbol{w}}_{t+1}\|_{\nabla^2 \Psi(\widetilde{\boldsymbol{w}}_{t+1})} + 2\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_t\|_{\nabla^2 \Psi(\widetilde{\boldsymbol{w}}_{t+1})}, \\ &\leq 2\epsilon' + \|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_t\|_{\nabla^2 \Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1})} \quad \text{(by (37) and } \nabla^2 \Phi_{t+1}(\widetilde{\boldsymbol{w}}_{t+1}) \geq \nabla^2 \Psi(\widetilde{\boldsymbol{w}}_{t+1})), \\ &\leq \epsilon + 2\|\widetilde{\boldsymbol{w}}_{t+1} - \boldsymbol{w}_t\|_{\nabla^2 \Phi_{t+1}(\boldsymbol{w}_t)}, \quad \text{(by (36) and Lemma 16)} \\ &= \epsilon + \sqrt{2}\lambda(\boldsymbol{w}_t, \Phi_{t+1}). \quad \text{(by (36))} \end{aligned} \tag{42}$$

On the other hand, since $\epsilon + \sqrt{2}\lambda(\boldsymbol{w}_t, \Phi_{t+1}) \leq \sqrt{\eta d}/8$ (by (28)), Lemma 16 and (42) imply that

$$\|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\|_{\nabla^2 \Psi(\boldsymbol{w}_t)} \leq \sqrt{2} \|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\|_{\nabla^2 \Psi(\boldsymbol{w}_{t+1})}.$$

Now, to get the desired results, it suffices to bound the sum $S_2 \coloneqq \sum_{t=1}^T \|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\|^2_{\nabla^2\Psi(\boldsymbol{w}_t)}$. Using (29), (42), and the fact that $\lambda(\boldsymbol{w}_t, \Phi_{t+1}) \le 1$ (by (28)), we have

$$\begin{aligned}
\sum_{t=1}^T \|\boldsymbol{w}_{t+1} - \boldsymbol{w}_t\|^2_{\nabla^2\Psi(\boldsymbol{w}_{t+1})} &\le T\epsilon^2 + \sum_{t=1}^T 2^{3/2}\epsilon\lambda(\boldsymbol{w}_t, \Phi_{t+1}) + \sum_{t=1}^T 2\lambda(\boldsymbol{w}_t, \Phi_{t+1})^2, \\
&\le 5T\epsilon^2 + \sum_{t=1}^T \lambda(\boldsymbol{w}_t, \Phi_t)^2/2 + 4\sum_{t=1}^T \|\boldsymbol{g}_t\|^2_{\nabla^{-2}\Phi_t(\boldsymbol{w}_t)}, \\
&\le 5T\epsilon^2 + \sum_{t=1}^T \lambda(\boldsymbol{w}_t, \Phi_t)^2 + \frac{4d\ln(d + B^2T/d)}{\beta}, \quad \text{(by Lemma 19)} \\
&\le 5T\epsilon^2 + \sum_{t=1}^T \lambda(\boldsymbol{w}_t, \Phi_t)/2 + \frac{4d\ln(d + B^2T/d)}{\beta}, \quad (\lambda(\boldsymbol{w}_t, \Phi_t) \le 1 \text{ by (28)}), \\
&\le 5T\epsilon^2 + \frac{6d\ln(d + B^2T/d)}{\beta},
\end{aligned}$$

where the last inequality follows by (41). ■

Appendix E. Special Case of Linear Regression

Without additional assumptions on the data-generating distribution, we conjecture that it is not possible to find an ε-optimal point in Stochastic Exp-Concave Optimization using fewer than $O(d^3/\varepsilon)$ arithmetic operations if one insists on a computational complexity that scales with $1/\varepsilon$ (instead of $1/\varepsilon^2$, for example). One observation that lead us to this conjecture is that even in the simple special case of Linear Regression (LR) with the square loss, it is not clear if one can find an ε-optimal point using fewer than $O(d^3/\varepsilon)$ arithmetic operations.

In the LR setting with the square loss, one can assume that the covariates $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots \in \mathbb{R}^d$ are i.i.d., and $y_t = \boldsymbol{w}_\star^\top \boldsymbol{x}_t + \varepsilon_t$, $t \ge 1$, for some fixed $\boldsymbol{w}_\star \in \mathbb{R}^d$ (to be learned/approximated) and some i.i.d. noise variables $\varepsilon_1, \varepsilon_2, \ldots$. In this case, a natural estimator for $\boldsymbol{w}_\star$ is the Empirical Risk Minimizer (ERM) $\widehat{\boldsymbol{w}} \in \operatorname{argmin}_{\boldsymbol{w}\in\mathbb{R}^d} \sum_{t=1}^T (\boldsymbol{w}^\top \boldsymbol{x}_t - y_t)^2$, which admits the closed form expression

$$\widehat{\boldsymbol{w}} = (X^\top X)^\dagger X^\top \boldsymbol{y},$$

where X [resp. $\boldsymbol{y}$] is the matrix [resp. vector] whose tth row is $\boldsymbol{x}_t^\top$ [resp. y_t], and $\dagger$ denotes the pseudo-inverse. To ensure that $\widehat{\boldsymbol{w}}$ is an ε-optimal point, in the sense that $\mathbb{E}[(\widehat{\boldsymbol{w}}^\top \boldsymbol{x} - \boldsymbol{w}_\star^\top \boldsymbol{x})^2] \le \varepsilon$, standard generalization arguments say that T needs be at least $\Omega(d/\varepsilon)$, in general (see e.g. (Rigollet and Hutter, 2019, Corollary 4.13)). For such a T, X is a matrix in $\mathbb{R}^{d/\varepsilon \times d}$, and so evaluating even $X^\top X$ in the expression of $\widehat{\boldsymbol{w}}$ would require d^3/ε arithmetic operations. A similar number of arithmetic operations would, in general, be needed to project $\widehat{\boldsymbol{w}}$ onto a feasible set in case of constrained Linear Regression.