Analysis Techniques for Adaptive Online Learning

H. Brendan McMahan
Google, Inc.
mcmahan@google.com

Abstract

We survey tools for the analysis of Follow-The-Regularized-Leader (FTRL) and Dual Averaging algorithms when the regularizer (prox-function) is chosen adaptively based on the data. Adaptivity can be used to prove regret bounds that hold on every round (rather than a specific final round $T$), and also allows for data-dependent regret bounds as in AdaGrad-style algorithms. We present results from a large number of prior works in a unified manner, using a modular analysis that isolates the key arguments in easily re-usable lemmas. Our results include the first fully-general analysis of the FTRL-Proximal algorithm (a close relative of mirror descent), supporting arbitrary norms and non-smooth regularizers.

1 Introduction

We consider the problem of online convex optimization over a series of rounds $t \in \{1, 2, \ldots \}$. On each round the algorithm selects a predictor $x_t \in \mathbb{R}^n$, and then an adversary selects a convex loss function $f_t$, and the algorithm suffers loss $f_t(x_t)$. The goal is to minimize

$$\text{Regret}(x^*) \equiv \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(x^*),$$

the difference between the algorithm’s loss and the loss of a fixed predictor $x^*$ chosen with full knowledge of the sequence of $f_t$. When a particular set of comparators $X$ is fixed in advance, one is often interested in $\text{Regret}(X) \equiv \sup_{x^* \in X} \text{Regret}(x^*)$; since $X$ is often a norm ball, it is often preferable to simply bound $\text{Regret}(x^*)$ by a function of $\|x^*\|$. Online algorithms with good regret bounds can be used for a wide variety of prediction and learning tasks (Cesa-Bianchi and Lugosi, 2006, Shalev-Shwartz, 2012). Our goal is to provide sufficient exposition for a reader with only limited familiarity with online convex optimization; more seasoned readers may wish to skip to the highlights.

We consider the family of Follow-The-Regularized-Leader (FTRL, or FoReL) algorithms (Shalev-Shwartz, 2007, Shalev-Shwartz and Singer, 2007, Rakhlin, 2008, McMahan and Streeter, 2010, McMahan, 2011a). Hazan (2010) and Shalev-Shwartz (2012) provide a comprehensive survey of analysis techniques for non-adaptive members of this algorithm family, where the regularizer is typically fixed for all rounds and chosen with knowledge of $T$. In this survey, we allow the regularizer to change adaptively over the course of an unknown-horizon game.

\footnote{Theorems 1 and 2 are the main results proved; they are stated in enough generality to cover many known results for general and strongly convex functions. The proofs follow immediately by combining Lemmas 6 and 8; all necessary proofs take only about a page.}
Given a sequence $r_0, r_1, r_2, \ldots$ of incremental regularizers, we consider the algorithm that plays $x_1 \in \arg \min_x r_0(x)$, and thereafter plays
\begin{equation}
x_{t+1} = \arg \min_x f_{1:t}(x) + r_{0:t}(x),
\end{equation}
where we use the compressed summation notation $f_{1:t}(x) = \sum_{s=1}^t f_s(x)$ (we also use this notation for sums of scalars or vectors). The algorithms we consider are adaptive in that each $r_t$ can be chosen based on $f_1, f_2, \ldots, f_t$. For convenience, we define functions $h_t$ where $h_0(x) = r_0(x)$ and $h_t(x) = f_t(x) + r_t(x)$ for $t \geq 1$, so $x_{t+1} = \arg \min_x h_{0:t}(x)$. Generally we will assume the $f_t$ are convex, and the $r_t$ are chosen so that $r_{0:t}$ (or $h_{0:t}$) is strongly convex for all $t$ (e.g., $r_{0:t}(x) = \frac{1}{2\eta^2} \|x\|^2$; see Section 3 for a review of important definitions and results from convex analysis). The name FTRL comes from a style of analysis that considers the regret of the Be-The-Leader algorithm (playing $x_t = \arg \min_x f_{1:t}(x)$) and then of Follow-The-Leader (playing $x_t = \arg \min_x f_{1:t-1}(x)$).

In addition to providing regret bounds for all rounds $t$, this framework is also particularly suitable for analyzing algorithms that adapt their regularization or norms based on the observed data, for example those of McMahan and Streeter (2010) and Duchi et al. (2011). \footnote{This work first appeared simultaneously with McMahan and Streeter (2010) as Duchi et al. (2010).}

**Linearization** In practice, it may be computationally infeasible to solve the optimization problem of Eq. (1). A key point is that we can derive a wide variety of first-order algorithms by linearizing the $f_t$, and running the algorithm on these linear functions. Algorithm 1 gives the general algorithmic scheme. For convex differentiable $f_t$, let $x_t$ be defined as above, and let $g_t = \nabla f_t(x_t)$. Then, convexity implies for any comparator $x^*$, $f_t(x_t) - f_t(x^*) \leq g_t \cdot (x_t - x^*)$. If we let $\hat{f}_t(x) = g_t \cdot x$, then for any algorithm the regret against the functions $\hat{f}_t$ upper bounds the regret against the original $f_t$. Note we can construct the functions $\hat{f}_t$ on the fly (after observing $x_t$ and $f_t$) and then present them to the algorithm, resulting in a much easier to compute update Eq. (2). For example, if we take $r_0(x) = \frac{1}{2\eta^2} \|x\|^2$ then we can solve Eq. (1) in closed form, yielding $x_{t+1} = -\eta g_{t+1}$ (that is, this FTRL algorithm is exactly constant step size online gradient descent). However, whenever possible we will state our results in terms of general $f_t$, since one can always simply take $\hat{f}_t = f_t$ when appropriate.

Further note that we could in general run the algorithm on any $f_t$ that satisfy $\hat{f}_t(x_t) = f_t(x_t) - f_t(x^*)$ for all $x^*$ and have the regret bound achieved for the $\hat{f}$ also apply to the original $f$. This is generally accomplished by constructing a lower bound that is tight

---

**Algorithm 1** General Scheme for Linearized FTRL

**Parameters:** Initial regularization function $r_0$, scheme for choosing $r_t$.  
$z \leftarrow 0 \in \mathbb{R}^n$ // Maintains $g_{1:t}$  
$x_1 \leftarrow \arg \min_x z \cdot x + r_0(x)$  
for $t = 1, 2, \ldots, T$ do  
- Play $x_t$, observe loss function $f_t$, incur loss $f_t(x_t)$  
- Compute sub-gradient $g_t \in \partial f_t(x_t)$  
- Choose $r_t$, possibly using $x_t$ and $g_t$  
- $z \leftarrow z + g_t$  
- $x_{t+1} \leftarrow \arg \min_x z \cdot x + r_{1:t}(x)$  
end for
at \( x_t \), that is \( \hat{f}_t(x) \leq f_t(x) \) for all \( x \) and \( \hat{f}_t(x_t) = f_t(x_t) \). A linear lower bound is always possible for convex functions, but for example if the \( f_t \) are all strongly convex, better bounds are possible by taking \( \hat{f}_t \) to be an appropriate quadratic lower bound.

**Prox-Functions, Regularization and the FTRL-Proximal Algorithm** We refer to the functions \( r_{0,t} \) as regularization functions, with \( r_t \) the incremental increase in regularization on round \( t \) (generally we will assume \( r_t(x) \geq 0 \)). This is regularization in the sense of Follow-The-Regularized-Leader, and these \( r_t \) terms should be viewed as part of the algorithm itself. In Dual Averaging, \( r_{0,t} \) is called the *prox-function* \(^{[Nesterov, 2009]} \), and is generally assumed to be minimized at a fixed constant point, without loss of generality the origin. We use the terms Regularized Dual Averaging (RDA) or just DA to refer specifically to this case, though more typically DA also implies linearizing the \( f_t \) (we will say more about linearization in Section 1.2).

We will refer to the algorithm as *FTRL-Proximal* when each incremental regularization function \( r_t \) is globally minimized by \( x_t \), and call such \( r_t \) incremental proximal regularizers. When we make neither a proximal nor origin-centered assumption on the \( r_t \), we refer to general FTRL algorithms. Using proximal regularizers will prove useful in the analysis, as in addition to \( x_t = \arg\min_x f_{1:t-1}(x) + r_{0:t-1}(x) \) by Eq. (1), this choice of \( r_t \) ensures

\[
x_t = \arg\min_x f_{1:t-1}(x) + r_{0:t}(x)
\]

as well. This means we can view both \( x_t \) and \( x_{t+1} \) as being defined in terms of minimization with respect to the regularizer \( r_{0:t}(x) \), which simplifies the analysis and to some extent strengthens the results.

The actual convex optimization problem we are solving may itself contain regularization terms, this time in the usual machine learning sense of the word. For example, we might have \( f_t(x) = \ell_t(x) + \lambda_1 \| x \|_1 \), where the \( \lambda_1 \| x \|_1 \) is an \( L_1 \) regularization term as in the LASSO method, and \( \ell_t \) measures the loss on the \( t \)th training example. The algorithms here handle this seamlessly; we note only that it is generally preferable to only apply the linearization to the part of the objective where it is necessary computationally; in this case, for example, we would take \( f_t(x) = g_t \cdot x + \lambda_1 \| x \| \), where \( g_t \in \partial \ell_t(x_t) \). Note that whether we view the \( \lambda_1 \| x \|_1 \) terms as part of the \( f_t \) or the \( r_t \) does not matter to the algorithm, and only changes the interpretation of the regret bounds.

FTRL-Proximal algorithms are close relatives of online gradient descent and online mirror descent; however, it is exactly this ability to encode the full \( L_1 \) penalty (or some other non-smooth regularizer) in the update Eq. (1) that gives them a significant advantage \(^{[McMahan, 2011a]} \).

### 1.1 Analysis Techniques and Main Results

We break the analysis of adaptive FTRL algorithms into three main components, which helps to modularize the arguments. In Section 2 we provide two inductive lemmas that express the Regret on round \( t \) as a regularization term on the comparator \( x^* \), namely \( r_{0:T}(x^*) \), plus a sum of per-round terms. Generally, this reduces the problem of bounding Regret to that of bounding these per-round terms. The first of these results is often referred to as the *FTRL Lemma*; the second was called the *Strong FTRL Lemma* in \(^{[McMahan, 2011b]} \), but for linear functions it is also closely connected to the primal-dual analysis of online algorithms. In Section 3 we review some standard results from convex analysis, and prove lemmas that
making bounding the per-round terms straightforward. The final regret bounds are then
proved in Section 4 as straightforward corollaries of these results. Before stating the main
theorems, we introduce some additional notation.

**Notation and Definitions** We consider extended convex functions \( \psi : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\} \),
where the domain of \( \psi \) is the set \( \{x : \psi(x) < \infty\} \). We write \( \partial \psi(x) \) for the subdifferential
of \( f \) at \( x \). A subgradient \( g \in \partial \psi(x) \) satisfies \( \psi(y) \geq \psi(x) + g \cdot (y - x) \) for all \( y \).
A function \( \psi : \mathcal{X} \to \mathbb{R} \cup \{\infty\} \) is \( \sigma \)-strongly convex w.r.t. a norm \( \| \cdot \| \) on \( \mathcal{X} \) if for all \( x, y \in \mathcal{X} \) and any
\( g \in \partial \psi(x) \), we have

\[
\psi(y) \geq \psi(x) + g \cdot (y - x) + \frac{\sigma}{2} \|y - x\|^2.
\]

The convex conjugate (or Fenchel conjugate) of an arbitrary function \( \psi : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\} \) is
\[
\psi^*(g) \equiv \sup_x g \cdot x - \psi(x).
\]

For a norm \( \| \cdot \| \), the dual norm is given by

\[
\|x\|_* \equiv \sup_{y : \|y\| \leq 1} x \cdot y.
\]

We denote the indicator function on a convex set \( \mathcal{X} \) by

\[
\Psi_{\mathcal{X}}(x) = \begin{cases}
0 & \text{if } x \in \mathcal{X} \\
\infty & \text{otherwise}
\end{cases}.
\]

We can summarize our basic assumptions as follows:

**Setting 1.** We consider the algorithm that plays according to Eq. (1) based on \( r_t \) that
satisfy \( r_t(x) \geq 0 \) for \( t \in \{0, 1, \ldots, T\} \), against a sequence of convex cost functions \( f_t : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\} \).

We can now introduce the theorems which will be our main focus. First, we consider a
bound for FTRL-Proximal:

**Theorem 1W. Weak FTRL-Proximal Bound** Consider Setting 1 and further suppose
the \( r_t \) are chosen such that \( r_0 \) is \( 1 \)-strongly-convex w.r.t. some norm \( \| \cdot \|_t \) for \( x \in \text{dom } r_0 \),
and further the \( r_t \) are proximal, that is \( x_t \) is a global minimizer of \( r_t \). Then, choosing any
\( g_t \in \partial f_t(x_t) \) on each round, for any \( x^* \in \mathbb{R}^n \),

\[
\text{Regret}(x^*) \leq r_0(x^*) + \sum_{t=1}^{T} \|g_t\|_t^2.
\]

The proof of this theorem relies on the standard FTRL Lemma (Lemma 5 in Section 2).

**Theorem 1W** is the best possible using this lemma, but using the Strong FTRL Lemma
(Lemma 6), we can improve this result by a constant factor:

**Theorem 1. Strong FTRL-Proximal Bound** Under the same conditions as Theorem 1W we can improve the bound to

\[
\text{Regret}(x^*) \leq r_0(x^*) + \frac{1}{T} \sum_{t=1}^{T} \|g_t\|_t^2.
\]
Finally, we have a general bound for any FTRL algorithm (including RDA):

**Theorem 2. General FTRL Bound** Consider Setting [1] and suppose the $r_t$ are chosen such that $h_{0:t} + f_{t+1}$ is 1-strongly-convex w.r.t. some norm $\| \cdot \|_{(t)}$ for $x \in \text{dom} r_{0:t}$. Then,

$$\text{Regret}(x^*) \leq r_{0:T-1}(x^*) + \frac{1}{2} \sum_{t=1}^{T} \|g_t\|_{(t-1),*}^2.$$  

We state these bounds in terms of strong convexity conditions on $h_{0:t}$ in order to also cover the case where the $f_t$ are themselves strongly convex. In fact, if each $f_t$ is strongly convex, then we can choose $r_t(x) = 0$ for all $x$, and Theorems 1 and 2 produce identical bounds (and algorithms). On the other hand, when the $f_t$ are not strongly convex (e.g., linear), a sufficient condition for all these theorems is choosing the $r_t$ such that $r_{0:t}$ is 1-strongly-convex w.r.t. $\| \cdot \|_{(t)}$. It is worth emphasizing here the “off-by-one” difference between Theorems 1 and 2 in this case: we can choose $r_t$ based on $g_t$, and when using proximal regularizers, this lets us influence the norm we use to measure $g_t$ in the final bound (namely the $\|g_t\|_{(t-1),*}$ term); this is not possible using Theorem 2 since we have $\|g_t\|_{(t-1),*}^2$. This makes constructing AdaGrad-style adaptive learning rate algorithms for FTRL-Proximal easier (McMahan and Streeter, 2010), whereas with Dual Averaging algorithms one must start with slightly more regularization. We will see this in more detail in the next section.

When it is not known a priori whether the loss functions $f_t$ are strongly convex, the $r_t$ can be chosen adaptively to add only as much strong convexity as needed, following Bartlett et al. (2007).

Theorem 2 leads immediately to a bound for dual averaging algorithms (Nesterov, 2009), including the Regularized Dual Averaging (RDA) algorithm of Xiao (2009), and its AdaGrad variant (Duchi et al., 2011) (in fact, this statement is equivalent to Duchi et al. (2011, Prop. 2) when we assume the $f_t$ are not strongly convex). As in these cases, Theorem 2 is usually applied to regularizers where 0 is a global minimizer of each $r_t$, so 0 is a global minimizer of $r_{0:T}$ as well. The theorem does not require this; however, such a condition is usually necessary to bound $r_{0,T-1}(x^*)$ and hence Regret$(x^*)$ in terms of $\|x^*\|$.

Less general versions of these theorems often assume that each $r_{0:t}$ is say $\alpha_t$-strongly-convex with respect to a fixed norm $\| \cdot \|$. Our results include this as a special case, see Lemma 3 in Section 3 as well as discussion in the next section.

These theorems can also be used to analyze non-adaptive algorithms. If we choose $r_0(x)$ to be a fixed non-adaptive regularizer (perhaps chosen with knowledge of $T$) that is 1-strongly convex w.r.t. $\| \cdot \|$, and all $r_t(x) = 0$ for $t \geq 1$, then we have $\|x\|_{(t),*} = \|x\|_*$ for all $t$, and so both Theorems provide the identical statement

$$\text{Regret}(x^*) \leq r_0(x^*) + \frac{1}{2} \sum_{t=1}^{T} \|g_t\|_*^2.$$  

Theorem 1 can also be applied in this way, but it again loses a factor of $\frac{1}{2}$; this gives, e.g., Shalev-Shwartz (2012, Theorem 2.11).

### 1.2 Application to Specific Algorithms

Before proving these theorems, we discuss some simple applications to a variety of algorithms. We will use the following lemma, which collects some straightforward facts for the
sequence of incremental regularizers \( r_t \). These claims are immediate consequences of the relevant definitions.

**Lemma 3.** Consider a sequence of \( r_t \) as in Setting [1]. Then, since \( r_t(x) \geq 0 \), we have \( r_{0:t}(x) \leq r_{0:t-1}(x) \), and so \( r_{0:t}(x) \leq r_{0:t-1}(x) \). If each \( r_t \) is \( \sigma_t \)-strongly convex w.r.t. a norm \( \| \cdot \| \) for \( \sigma_t \geq 0 \), then \( r_{0:t} \) is \( \sigma_{0:t} \)-strongly convex w.r.t. \( \| \cdot \| \), or equivalently, is \( 1 \)-strongly-convex w.r.t. \( \| x \|_{(t)} = \sqrt{\sigma_{0:t}} \| x \| \).

**Dual Averaging** If we choose \( r_t(x) = \frac{\eta}{\sqrt{T}} \| x \|_2^2 \), then \( r_{0:t} \) is \( 1 \)-strongly-convex w.r.t. the norm \( \| x \|_{(t)} = \sqrt{\sigma_{0:t}} \| x \|_2 \), which has dual norm \( \| x \|_{(t),*} = \frac{1}{\sqrt{\sigma_{0:t}}} \| x \|_2 \). It will be convenient to let \( \eta_t = \frac{R}{\sqrt{2G} \sqrt{t+1}} \), and as the notation suggests, \( \eta_t \) is exactly analogous to a learning rate as in gradient descent. Note that any non-increasing learning rate schedule can be expressed in this manner by choosing: \( \sigma_t = 1/\eta_t - 1/\eta_{t-1} \). With this definition, plugging into Theorem [2] then gives

\[
\text{Regret} \leq \frac{1}{2} \sum_{t=1}^{T} \eta_{t-1} \| g_t \|_2^2 + \frac{1}{2\eta_{T-1}} \| x^* \|_2^2.
\]

Suppose we know \( \| g_t \|_2 \leq G \), and we guess a bound \( \| x^* \|_2 \leq R \) in advance. Then, with the choice \( \eta_t = \frac{R}{\sqrt{2G} \sqrt{t+1}} \), using the inequality \( \sum_{t=1}^{T} \frac{1}{\sqrt{t}} \leq 2\sqrt{T} \), we arrive at

\[
\text{Regret}(x^*) \leq \frac{\sqrt{2}}{2} \left( R + \frac{\| x^* \|_2^2}{R} \right) G \sqrt{T}.
\]

When in fact \( \| x^* \| \leq R \), we have \( \text{Regret} \leq \sqrt{2RG} \sqrt{T} \), but the bound is valid (and meaningful) for arbitrary \( x^* \in \mathbb{R}^n \).

DA can also be restricted to play from a closed bounded feasible set \( \mathcal{X} \) by including \( \Psi_\mathcal{X} \) in \( r_0 \); this does not change the bound of Eq. [4], but it means it only applies to \( x^* \in \mathcal{X} \). Additional non-smooth regularization can also be applied by adding the appropriate terms to \( r_0 \) (or any of the \( r_t \); for example, we can add an \( L_1 \) and \( L_2 \) penalty by adding the terms \( \lambda_1 \| x \|_1 + \lambda_2 \| x \|_2^2 \). When in addition the \( f_t \) are linearized, this produces the RDA algorithm of [Xiao, 2011].

**FTRL-Proximal** Suppose \( \mathcal{X} \subseteq \{ x \mid \| x \|_2 \leq R \} \), and we choose \( r_0(x) = \Psi_\mathcal{X}(x) \) and for \( t > 1, r_t(x) = \frac{\eta}{\sqrt{T}} \| x-x_t \|_2^2 \). Then \( r_{0:t} \) is \( 1 \)-strongly-convex w.r.t. the norm \( \| x \|_{(t)} = \sqrt{\sigma_{0:t}} \| x \|_2 \), which has dual norm \( \| x \|_{(t),*} = \frac{1}{\sqrt{\sigma_{0:t}}} \| x \|_2 \). Note \( r_{0:t}(x^*) \leq \frac{\sqrt{2}}{2} (2R)^2 \) for any \( x^* \in \mathcal{X} \), since each \( x_t \in \mathcal{X} \). Thus, applying Theorem [4], we have

\[
\forall x^* \in \mathcal{X}, \quad \text{Regret}(x^*) \leq \frac{1}{2} \sum_{t=1}^{T} \eta_t \| g_t \|_2^2 + \frac{1}{2\eta_{T}} (2R)^2,
\]

where we let \( \eta_t = \frac{1}{\sigma_{t-1}} \). Choosing \( \eta_t = \frac{\sqrt{2}R}{G \sqrt{T}} \) and assuming \( \| x^* \| \leq \mathbb{R}^2 \), we have

\[
\text{Regret}(x^*) \leq 2\sqrt{2RG} \sqrt{T}.
\]
FTRL-Proximal with Diagonal Matrix Learning Rates  For simplicity, first consider the 1-dimensional problem. Let $r_0 = \Psi_X$ with $X = [-R, R]$, and fix a learning-rate schedule for FTRL-Proximal where

$$\eta_t = \frac{\sqrt{2R}}{\sqrt{\sum_{s=1}^{t} g_s^2}}$$

for use in Eq. (5). This gives

$$\text{Regret}(x^*) \leq 2\sqrt{2R} \left( \sum_{s=1}^{t} g_s^2 \right), \quad (7)$$

where we have used Lemma 4 (stated at the end of this section), which generalizes the standard inequality $\sum_{t=1}^{T} 1/\sqrt{t} \leq 2\sqrt{T}$. This gives us a fully adaptive version of Eq. (6): not only do we not need to know $T$ in advance, we also do not need to know a bound on the norms of the gradients $G$. Rather, the bound is fully adaptive and we see, for example, that the bound only depends on rounds where the gradient is nonzero (as one would expect).

We do, however, require that $R$ is chosen in advance; for algorithms that avoid this, see Streeter and McMahan (2012), Orabona (2013), McMahan and Abernethy (2013).

To arrive at a diagonal AdaGrad-style algorithm for $n$-dimensions we need only apply the above technique on a per-coordinate basis. Note Streeter and McMahan (2010) takes this approach directly; the more general analysis here allows us to handle arbitrary feasible sets and $L_1$ or other non-smooth regularization. Define $r_t(x) = \frac{1}{2} \| Q_{1:t} (x - x_t) \|_2^2$ for $Q_{t} \succeq 0$, so $r_{0:t}$ is 1-strongly-convex w.r.t. the norm $\| x \|_{(t)} = \| (Q_{1:t})^{-\frac{1}{2}} x \|_2$, which has dual norm $\| x \|_{(t)^*} = \| (Q_{1:t})^{-\frac{1}{2}} x \|_2$. We then define diagonal $Q_t$ so that $i$th diagonal entry of $Q_{1:t}$ is $\sqrt{\sum_{s=1}^{t} g_{s,i}^2}$, and let $r_0(x) = \Psi_X(x)$ for a closed and bounded convex $X$. Then, plugging into Theorem 1 recovers McMahan and Streeter (2010, Theorem 2), and we can improve by a constant factor using Theorem 1. Essentially, this bound amounts to summing Eq. (7) across all $n$ dimensions; a careful analysis shows this bound is at least as good (and often better) than that of Eq. (6).

Full matrix learning rates can be derived using a matrix generalization of Lemma 4, e.g., Duchi et al. (2011, Lemma 10); however, since this requires $O(n^2)$ space and potentially $O(n^2)$ time per round, in practice these algorithms are often less useful than the diagonal varieties.

It is perhaps not immediately clear that this algorithm is easy and efficient to implement. In fact, however, taking the linear approximation to $f_t$, one can see $h_{1:t}(x) = g_{1:t} \cdot x + r_{1:t}(x)$ is itself just a quadratic which can represented using two length-$n$ vectors, one to maintain the linear terms ($g_{1:t}$ plus some adjustment terms) and one to maintain $\sum_{s=1}^{t} g_{s,i}^2$, from which the diagonal entries of $Q_{1:t}$ can be constructed. For full pseudo-code which also incorporates $L_1$ and $L_2$ regularization, see McMahan et al. (2013).

AdaGrad-RDA  Similar ideas can be applied RDA (where we center each $r_t$ at the origin), but again one must use some care due to the “off-by-one” difference in the bounds. For example, for the diagonal algorithm, it is necessary to choose per-coordinate learning rates

$$\eta_t \approx \frac{1}{\sqrt{G^2 + \sum_{s=1}^{t} g_s^2}}$$

for use in Eq. (6). This gives
where \( |g_t| \leq G \). Thus, we arrive at an algorithm that is almost (but not quite) fully adaptive in the gradients, since a modest dependence on the initial guess \( G \) of the maximum per-coordinate gradient remains in the bound. This offset appears, for example, as the \( \delta I \) terms added to the learning rate matrix \( H_t \) in Figure 1 of [Duchi et al. (2011)].

**Strongly Convex Functions** Suppose each loss function \( f_t \) is 1-strongly-convex w.r.t. a norm \( \| \cdot \| \), and let \( r_t(x) = 0 \) for all \( t \) (that is, we play the Follow-The-Leader (FTL) algorithm). Define \( \|x\|_{(t)} = \sqrt{T} \|x\| \), and observe \( h_{0,t}(x) \) is 1-strongly-convex w.r.t. \( \| \cdot \|_{(t)} \). Then, applying either Theorem 1 or 2, we have

\[
\text{Regret}(x^*) \leq \frac{1}{2} \sum_{t=1}^{T} \|g_t\|_{(t),*}^2 = \frac{1}{2} \sum_{t=1}^{T} \frac{1}{t} \|g_t\|^2 \leq \frac{G^2}{2} (1 + \log T),
\]

where we have used the standard inequality \( \sum_{t=1}^{T} \frac{1}{t} < 1 + \log T \) and assumed \( \|g_t\| \leq G \). This recovers, e.g., Kakade and Shalev-Shwartz (2008, Cor. 1) for the exact FTL algorithm. On the other hand, for a 1-strongly-convex \( f_t \) with \( g_t \in \partial f_t(x_t) \) we have by definition

\[
f_t(x) \geq f_t(x_t) + g_t(x - x_t) + \frac{1}{2} \|x - x_t\|^2.
\]

Thus, we can define a \( \hat{f}_t \) equal to the right-hand-side of the above inequality, so \( \hat{f}_t(x) \leq f_t(x) \) and \( \hat{f}_t(x_t) = f_t(x_t) \). Running FTL on these functions produces an identical regret bound; this gives rise to the online gradient descent algorithm for strongly convex functions given by [Hazan et al. (2007)].

**Lemma 4.** For any non-negative real numbers \( a_1, a_2, \ldots, a_n \),

\[
\sum_{i=1}^{n} \frac{a_i}{\sqrt{\sum_{j=1}^{n} a_j}} \leq 2 \sqrt{\sum_{i=1}^{n} a_i}.
\]

For a proof see [Auer et al. (2002)] or [Streeter and McMahan (2010, Lemma 1)].

## 2 Inductive Lemmas

In this section we consider two lemmas that let us analyze arbitrary FTRL-style algorithms. The first is quite well known:

**Lemma 5 (Standard FTRL Lemma).** Let \( f_t \) be a sequence of arbitrary (e.g., non-convex) loss functions, and let \( r_t \) be arbitrary non-negative regularization functions, such that \( x_{t+1} = \arg\min_x h_{0,t}(x) \) is well defined. Then, the algorithm that plays these \( x_t \) satisfies

\[
\text{Regret}(x^*) \leq r_{0:T}(x^*) + \sum_{t=1}^{T} f_t(x_t) - f_t(x_{t+1}).
\]

For example, see [Kalai and Vempala (2003), Hazan (2008), Hazan (2010, Lemma 1), and Shalev-Shwartz (2012, Lemma 2.3)]. The proof of this lemma (e.g., see [McMahan and Streeter (2010, Lemma 3)]) relies on showing that if one could run the Be-The-Leader algorithm by playing \( x_t = \arg\min_x f_t(x) \) (which requires peeking ahead at \( f_t \) to choose \( x_t \)), then the
player’s regret is bounded above by zero. However, as we see by comparing Theorems 1 and 1; this analysis loses a factor of 1/2 on one of the terms. The key is that being the leader is actually strictly better than playing the post-hoc optimal point. The following result captures this fact, and hence allows for tighter bounds:

**Lemma 6 (Strong FTRL Lemma).** Under the same conditions as Lemma 5, we can tighten the bound to

\[
\text{Regret}(x^*) \leq r_{0:T}(x^*) + \sum_{t=1}^{T} h_{0:t}(x_t) - h_{0:t}(x_{t+1}) - r_t(x_t).
\]  

(8)

We immediately have the following corollary, which relates the above statement to the primal-dual style of analysis:

**Corollary 7.** Consider the same conditions as Lemma 6, but further suppose the loss functions are linear, \( f_1(x) = g_{t} \cdot x_t \). Then,

\[
h_{0:t}(x_t) - h_{0:t}(x_{t+1}) - r_t(x_t) = r_{0:t}^*(-g_{1:t}) - r_{0:t-1}^*(-g_{1:t-1}) + g_t \cdot x_t,
\]

(9)

which implies

\[
\text{Regret}(x^*) \leq r_{0:T}(x^*) + \sum_{t=1}^{T} r_{0:t}^*(-g_{1:t}) - r_{0:t-1}^*(-g_{1:t-1}) + g_t \cdot x_t.
\]

Lemma 6 was introduced in McMahan (2011b). Corollary 7 can easily be proved directly using the Fenchel-Young inequality, and has a longer history in the literature. Our statement directly matches the first claim of Orabona (2013, Lemma 1), and see also Kakade et al. (2012, Corollary 4).

Note that Lemma 6 is strictly stronger than Corollary 7: it applies to non-convex \( f_t \) and \( r_t \). Further, even for convex \( f_t \), it can be more useful: for example, we can easily analyze strongly-convex \( f_t \) with all \( r_t(x) = 0 \) using the first statement; however, a direct application of the second statement becomes vacuous as \( r_{0:t}^*(g) = \infty \) whenever \( g \neq 0 \) (this issue can be surmounted with some technical care, see Orabona et al. [2013], Section 4 in particular).

**Proof of Lemma 6.** First, we bound a quantity that is essentially our regret if we had played the FTL algorithm against the functions \( h_1, \ldots, h_T \) (for convenience, we include a \(-h_0(x^*)\) term as well).

\[
\sum_{t=1}^{T} h_t(x_t) - h_{0:T}(x^*) = \sum_{t=1}^{T} (h_{0:t}(x_t) - h_{0:t-1}(x_t)) - h_{0:T}(x^*)
\]

\[
\leq \sum_{t=1}^{T} (h_{0:t}(x_t) - h_{0:t-1}(x_t)) - h_{0:T}(x_{T+1}) \quad \text{since } x_{T+1} \text{ minimizes } h_{0:T}
\]

\[
= \sum_{t=1}^{T} (h_{0:t}(x_t) - h_{0:t}(x_{t+1})),
\]

where the last line follows by simply re-indexing the \(-h_{0:t}\) terms and dropping the non-positive term \(-h_0(x_1) = -r_0(x_1) \leq 0\). Expanding the definition of \( h \) on the left-hand-side of the above inequality gives

\[
\sum_{t=1}^{T} (f_t(x_t) + r_t(x_t)) - f_{1:T}(x^*) - r_{0:T}(x^*) \leq \sum_{t=1}^{T} (h_{0:t}(x_t) - h_{0:t}(x_{t+1})).
\]
Re-arranging the inequality proves the lemma.

Remark: we could get equality if we included the non-positive term $h_{1:T}(x_{T+1}) - h_{1:T}(x^*)$ on the RHS, since we can assume $r_0(x_1) = 0$ without loss of generality. Further, if one is actually interested in the performance of the FTL algorithm against the $h_t$ (e.g., if all the $r_t$ are uniformly zero), then choosing $x^* = x_{T+1}$ is natural.

**Proof of Corollary 7.** Using the definition of the Fenchel conjugate and of $x_{t+1}$, we have

$$r_0^*(-g_{1:t}) = \max_x -g_{1:t} \cdot x - r_0 (x) = -(\min_x g_{1:t} \cdot x + r_0 (x)) = -h_0 (x_{t+1}).$$  \hspace{1cm} (10)

Now, observe that

$$h_0 (x_{t+1}) = g_{1:t} \cdot x_{t+1} + r_0 (x_{t+1}) = h_{0:t-1} (x_{t+1}) + g_t \cdot x_{t+1} = -r_0^* (-g_{1:t-1}) + g_t \cdot x_{t+1},$$

where the last line uses Eq. (10). Combining this with $-h_0 (x_{t+1}) = r_0^* (-g_{1:t})$ proves Eq. (9). \hfill $\square$

### 3 Tools from Convex Analysis

Here we highlight a few key tools from convex analysis that will be applied to bounding the per-round terms that appear in the preceding lemmas. For more details, see Shalev-Shwartz (2012), Rockafellar (1997), Shalev-Shwartz (2007), and many of the other papers cited here. The following lemma is a powerful tool for bounding the per-round terms of both Lemma 5 and 6. We defer the proofs of the results in this section to Appendix A.

**Lemma 8.** Let $\phi_1 : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ be a convex function such that $x_1 = \arg \min_x \phi_1 (x)$ exists. Let $\psi$ be a convex function, let $\phi_2 (x) = \phi_1 (x) + \psi (x)$, and suppose $\phi_2$ is strongly convex w.r.t. norm $\| \cdot \|$. Let $b \in \partial \psi (x_1)$ and let $x_2 = \arg \min_x \phi_2 (x)$. Then,

$$\| x_1 - x_2 \| \leq \| b \|_*,$$

and for any $x'$,

$$\phi_2 (x_1) - \phi_2 (x') \leq \frac{1}{2} \| b \|_*^2.$$

When $\phi_1$ and $\psi$ are quadratics (one possibly linear) and the norm is the corresponding $L_2$ norm, both statements in the above lemma hold with equality.

The concept of strong smoothness plays a key role in the proof of the above lemma. A function $\psi$ is $\sigma$-strongly-smooth with respect to a norm $\| \cdot \|$ if it is differentiable and for all $x, y$ we have

$$\psi (y) \leq \psi (x) + \nabla \psi (x) \cdot (y - x) + \frac{\sigma}{2} \| y - x \|^2. \hspace{1cm} (11)$$

There is a fundamental duality between strongly convex and strongly smooth functions:

**Lemma 9.** Let $\psi$ be closed and convex. Then $\psi$ is $\sigma$-strongly convex with respect to the norm $\| \cdot \|$ if and only if $\psi^*$ is $\frac{1}{\sigma}$-strongly smooth with respect to the dual norm $\| \cdot \|_*$. \hfill $\square$
For the strongly convexity implies strongly smooth direction see Shalev-Shwartz (2007, Lemma 15), and for the other direction see Kakade et al. (2012, Theorem 3). The following Lemma captures two useful properties of strongly smooth functions:

**Lemma 10.** Let $\psi$ be $1$-strongly convex w.r.t $\|\cdot\|$, so $\psi^*$ is $1$-strongly smooth with respect to $\|\cdot\|$. Then,

$$\|\nabla \psi^*(z) - \nabla \psi^*(z')\| \leq \|z - z'\|_* \tag{12}$$

and

$$\arg \min_x g \cdot x + \psi(x) = \nabla \psi^*(-g). \tag{13}$$

This lemma implies that when $f_t(x) = g_t \cdot x$ and $r_{0:t}$ is strongly convex, then the update of the algorithm Eq. (1) can be written as $x_{t+1} = \nabla r_{0:t}^*(-g_t)$; this notation is commonly used, especially in the context of mirror descent algorithms.

## 4 Regret Bound Proofs

### 4.1 Analysis of FTRL-Proximal using the Standard FTRL Lemma

In this section, we prove Theorem 1 using strong smoothness via Lemma 8. For general convex $f_t$, an alternative proof that uses strong convexity directly can also be done, closely following Shalev-Shwartz (2012, Sec. 2.5.2).

**Proof of Theorem 1**

Applying Lemma 5, it is sufficient consider a fixed $t$ and upper bound $f_t(x_t) - f_t(x_{t+1})$. For this fixed $t$, define a helper function $\phi_t(x) = f_{1:t-1}(x) + r_{0:t}(x)$. Observe $x_t = \arg \min_x \phi_t(x)$ since $x_t$ is a minimizer of $r_t(x)$, and by definition of the update $x_t$ is a minimizer of $f_{1:t-1}(x) + r_{0:t-1}(x)$. Let $\phi_t(x) = \phi_t(x) + f_t(x) = h_{0:t}(x)$, so $\phi_t$ is $1$-strongly convex with respect to $\|\cdot\|_{(t)}$ by assumption. Then, we have

$$f_t(x_t) - f_t(x_{t+1}) \leq g_t(x_t - x_{t+1})$$

Convexity of $f_t$ and $g_t \in \partial f_t(x_t)$

$$\leq \|g_t\|_{(t),*} \|x_t - x_{t+1}\|_{(t)}$$

Dual norms

$$\leq \|g_t\|_{(t),*} \|g_t\|_{(t),*} = \|g_t\|_{(t),*}^2$$

Lemma 8

Interestingly, it appears difficult to achieve a tight (up to constant factors) analysis of non-proximal (e.g., RDA) FTRL algorithms using Theorem 8. The Strong FTRL Lemma, however, will allow us to accomplish this.

### 4.2 Analysis using the Strong FTRL Lemma

In this section, we prove Theorem 1 and Theorem 2 using Lemma 6. Stating these two analyses in a common framework makes clear exactly where the “off-by-one” problem arises for RDA, and how assuming proximal $r_t$ resolves this issue. The key tool is Lemma 8 though for completeness we also provide an analysis of Theorem 2 from Eq. (9) directly using strong smoothness.
Proximal Regularizers (Proof of Theorem 1) Take $\phi_1(x) = f_{1:t-1}(x) + r_{0:t}(x) = h_{0:t}(x) - f_t(x)$. Since the $r_t$ are proximal (so $x_t$ is a global minimizer of $r_t$) we have $x_t = \arg\min_x \phi_1(x)$, and $x_{t+1} = \arg\min_x \phi_1(x) + f_t(x)$. Thus,

\[
h_{0:t}(x_t) - h_{0:t}(x_{t+1}) - r_t(x_t) \leq h_{0:t}(x_t) - h_{0:t}(x_{t+1}) = \phi_1(x_t) + f_t(x_t) - \phi_1(x_{t+1}) - f_t(x_{t+1}) \leq \frac{1}{2} \|g_t\|_{\ell(t)}^2,
\]

where the last line follows by applying Lemma 3 to $\phi_1$ and $\phi_2(x) = \phi_1(x) + f_t(x) = h_{0:t}(x)$. Plugging back into Eq. (8) completes the proof.

Non-proximal Regularizers (Proof Theorem 2) For Lemma 2 take $\phi_1(x) = h_{0:t-1}(x)$ and $\phi_2(x) = h_{0:t-1}(x) + f_t(x)$, so $x_t = \arg\min_x \phi_2(x)$, and by assumption $\phi_2$ is 1-strongly-convex w.r.t. $\|\cdot\|_{(t-1)}$. Then, applying Lemma 3 to $\phi_2$, we have

\[
h_{0:t}(x_t) - h_{0:t}(x_{t+1}) - r_t(x_t) = \phi_2(x_t) + r_t(x_t) - \phi_2(x_{t+1}) - r_t(x_{t+1}) - r_t(x_t) \leq \frac{1}{2} \|g_t\|_{\ell(t)}^2,
\]

where we have used the assumption that $r_t(x) \geq 0$ to drop the $-r_t(x_{t+1})$ term. We can now plug this bound into Theorem 1, Eq. (8). However, we need to make one additional observation: the choice of $r_T$ does not impact $\|\cdot\|_{\ell(T)}$, and only increases $r_{0:T}(x^*)$. Further, $r_T$ does not influence any of the points $x_1, \ldots, x_T$ played by the algorithm. Thus, for analysis purposes, we can take $r_T(x) = 0$ without loss of generality, and hence replace $r_{0:T}$ with $r_{0:T-1}$ in the final bound.

Remark: The final argument in this proof is another manifestation of the “off-by-one” difference between FTRL-Proximal and RDA. The FTRL-Proximal bound essentially depends on $r_1, \ldots, r_T$ (we can essentially take $r_0(x) = 0$), whereas RDA depends on $r_0, \ldots, r_{T-1}$.

Non-proximal Regularizers via Potential functions We give an alternative proof of Thm 2 for linear functions, $f_t(x) = g_t \cdot x$, using Eq. (9). Recall in this case $x_t = \nabla r^*_{1:t-1}(-g_{1:t-1})$, and by Lemma 3 $r^*_{1:t-1}$ is 1-strongly-smooth with respect to $\|\cdot\|_{(t-1)}$, and so

\[
r^*_{1:t-1}(-g_{1:t}) \leq r^*_{1:t-1}(-g_{1:t-1}) - x_t \cdot g_t + \frac{1}{2} \|g_t\|_{\ell(t-1)}^2,
\]

and we can bound the per-round terms in Eq. (9) by

\[
r^*_{1:t}(-g_{1:t}) - r^*_{1:t-1}(-g_{1:t-1}) + x_t \cdot g_t \leq r^*_{1:t}(-g_{1:t}) - r^*_{1:t-1}(-g_{1:t}) + \frac{1}{2} \|g_t\|_{\ell(t-1)}^2 \leq \frac{1}{2} \|g_t\|_{\ell(t-1)}^2,
\]

where we have used the fact that $r^*_{1:t-1}(-g_{1:t}) \geq r^*_{1:t}(-g_{1:t})$ (Lemma 3).
References

Peter Auer, Nicolò Cesa-Bianchi, and Claudio Gentile. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 2002.

Peter L. Bartlett, Elad Hazan, and Alexander Rakhlin. Adaptive online gradient descent. In *NIPS*, 2007.

Nicolò Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. In *COLT*, 2010.

John C. Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

Elad Hazan. Extracting certainty from uncertainty: Regret bounded by variation in costs. In *COLT*, 2008.

Elad Hazan. The convex optimization approach to regret minimization, 2010.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Mach. Learn.*, 69:169–192, December 2007. ISSN 0885-6125. doi: 10.1007/s10994-007-5016-8.

S. Kakade and Shalev-Shwartz. Mind the duality gap: Logarithmic regret algorithms for online optimization. In *NIPS*, 2008.

Sham M. Kakade, Shai Shalev-Shwartz, and Ambuj Tewari. Regularization techniques for learning with matrices. *Journal of Machine Learning Research*, 2012.

Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and Systems Sciences*, 71(3), 2005. ISSN 0022-0000.

H. Brendan McMahan. Follow-the-regularized-leader and mirror descent: Equivalence theorems and L1 regularization. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011a.

H. Brendan McMahan. A unified analysis of regularized dual averaging and composite mirror descent with implicit updates. http://arxiv.org/abs/1009.3240, 2011b.

H. Brendan McMahan and Jacob Abernethy. Minimax optimal algorithms for unconstrained linear optimization. In *NIPS*, 2013.

H. Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. In *COLT*, 2010.

H. Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, Sharat Chikkerur, Dan Liu, Martin Wattenberg, Arnar Hrafnkelsson, Tom Boulos, and Jeremy Kubica. Ad click prediction: a view from the trenches. In *KDD*, 2013.
Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Math. Program.*, 120 (1), April 2009.

Francesco Orabona. Dimension-free exponentiated gradient. In *NIPS*, 2013.

Francesco Orabona, Koby Crammer, and Nicolò Cesa-Bianchi. A generalized online mirror descent with applications to classification and regression. *CoRR*, abs/1304.2994, 2013.

Alexander Rakhlin. Lecture notes on online learning, 2008.

Ralph T. Rockafellar. *Convex Analysis (Princeton Landmarks in Mathematics and Physics)*. Princeton University Press, 1997.

Shai Shalev-Shwartz. *Online Learning: Theory, Algorithms, and Applications*. PhD thesis, The Hebrew University of Jerusalem, 2007.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 2012.

Shai Shalev-Shwartz and Yoram Singer. A primal-dual perspective of online learning algorithms. *Machine Learning*, 2007.

Matthew Streeter and H. Brendan McMahan. No-regret algorithms for unconstrained online convex optimization. In *NIPS*, 2012.

Matthew J. Streeter and H. Brendan McMahan. Less regret via online conditioning. 2010.

Lin Xiao. Dual averaging method for regularized stochastic learning and online optimization. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *NIPS*, 2009.
A Proofs For Section 3

**Proof of Lemma 1.** Applying Eq. (11) twice, we have
\[
\psi^*(u) \leq \psi^*(w) + \nabla \psi^*(w) (u - w) + \frac{1}{2} \| u - w \|^2
\]
\[
\psi^*(w) \leq \psi^*(u) + \nabla \psi^*(u) (w - u) + \frac{1}{2} \| u - w \|^2
\]
Adding these inequalities gives the following chain of equivalent inequalities:
\[
\nabla \psi^*(u) (u - w) + \nabla \psi^*(w) (w - u) \leq \| u - w \|^2
\]
\[
\nabla (\psi^*(u) - \psi^*(w)) (u - w) \leq \| u - w \|^2
\]
\[
\| \nabla (\psi^*(u) - \psi^*(w)) \| \cdot \| (u - w) \| \leq \| u - w \|^2
\]
\[
\| \nabla (\psi^*(u) - \psi^*(w)) \| \leq \| u - w \|.
\]

In order to prove Lemma 8 we first prove a somewhat easier result:

**Lemma 11.** Let \( \phi_1 : \mathbb{R}^n \to \mathbb{R} \) be strongly convex w.r.t. \( \| \cdot \| \), and let \( x_1 = \arg \min_x \phi_1(x) \), and define \( \phi_2(x) = \phi_1(x) + b \cdot x \) for \( b \in \mathbb{R}^n \). Letting \( x_2 = \arg \min_x \phi_2(x) \), we have
\[
\phi_2(x_1) - \phi_2(x_2) \leq \frac{1}{2} \| b \|^2, \quad \text{and} \quad \| x_1 - x_2 \| \leq \| b \|.
\]

**Proof.** We have
\[
-\phi_1^*(0) = -\max_x 0 \cdot x - \phi_1(x) = \min_x \phi_1(x) = \phi_1(x_1).
\]
and similarly,
\[
-\phi_1^*(-b) = -\max_x -b \cdot x - \phi_1(x) = \min_x b \cdot x + \phi_1(x) = b \cdot x_2 + \phi_1(x_2).
\]
Since \( x_1 = \nabla \phi_1^*(0) \), Eq. (11) gives
\[
\phi_1^*(-b) \leq \phi_1^*(0) + x_1 \cdot (-b - 0) + \frac{1}{2} \| b \|^2.
\]
Combining these facts, we have
\[
\phi_1(x_1) + b \cdot x_1 - \phi_1(x_2) - b \cdot x_2 = -\phi_1^*(0) + b \cdot x_1 + \phi_1^*(-b)
\]
\[
\leq -\phi_1^*(0) + b \cdot x_1 + \phi_1^*(0) + x_1 \cdot (-b) + \frac{1}{2} \| b \|^2
\]
\[
= \frac{1}{2} \| b \|^2.
\]
For the second part, observe \( \nabla \phi_1^*(0) = x_1 \), and \( \nabla \phi_1^*(-b) = x_2 \) and so \( \| x_1 - x_2 \| \leq \| b \| \), using both parts of Lemma 10. \( \square \)

**Proof of Lemma 5.** We are given that \( \phi_2(x) = \phi_1(x) + \psi(x) \) is 1-strongly convex w.r.t. \( \| \cdot \| \). The key trick is to construct an alternative \( \phi_1^* \) that is also 1-strongly convex with respect to this same norm, but has \( x_1 \) as a minimizer. Fortunately, this is easily possible: define \( \phi_1^*(x) = \phi_1(x) + \psi(x) - b \cdot x \), and note \( \phi_1 \) is 1-strongly convex w.r.t. \( \| \cdot \| \) since it differs from \( \phi_2 \) only by a linear function. Since \( b \in \partial \psi(x_1) \) it follows that 0 is in \( \partial (\psi(x) - b \cdot x) \) at \( x = x_1 \), and so \( x_1 = \arg \min_x \phi_1^*(x) \). Note \( \phi_2(x) = \phi_1^*(x) + b \cdot x \). Applying Lemma 11 to \( \phi_1^* \) and \( \phi_2 \) completes the proof, noting for any \( x' \) we have \( \phi_2(x_1) - \phi_2(x') \leq \phi_2(x_1) - \phi_2(x_2) \). \( \square \)