Combining Online Learning Guarantees

Ashok Cutkosky
Google
ashok@cutkosky.com

Abstract

We show how to take any two parameter-free online learning algorithms with different regret guarantees and obtain a single algorithm whose regret is the minimum of the two base algorithms. Our method is embarrassingly simple: just add the iterates. This trick can generate efficient algorithms that adapt to many norms simultaneously, as well as providing diagonal-style algorithms that still maintain dimension-free guarantees. We then proceed to show how a variant on this idea yields a black-box procedure for generating optimistic online learning algorithms. This yields the first optimistic regret guarantees in the unconstrained setting and generically increases adaptivity. Further, our optimistic algorithms are guaranteed to do no worse than their non-optimistic counterparts regardless of the quality of the optimistic estimates provided to the algorithm.

1 Online Learning

We consider the classic online learning problem with linear losses [1, 2, 3], sometimes called online linear optimization. Online learning is a game in which for each of \( T \) rounds, the learning algorithm outputs some vector \( w_t \) in some convex domain \( W \), and then the environment reveals a vector \( g_t \) and the algorithm suffers loss \( \langle g_t, w_t \rangle \). The objective is to minimize the regret, which is the total loss relative to some benchmark point \( u \):

\[
R_T(u) := \sum_{t=1}^{T} \langle g_t, w_t - u \rangle
\]

Although this formulation appears to only apply to a simple linear environment, algorithms that guarantee low regret can actually be automatically applied to general stochastic convex optimization problems found throughout machine learning [4], and so many of the popular optimization algorithms in use today (e.g. [5, 6]) are in fact online linear optimization algorithms.

Our first goal is to provide a “meta-algorithm” that combines online learning algorithms in a black-box manner to obtain an algorithm that achieves the best properties of the individual algorithms. Our technique applies to any algorithm that guarantees \( R_T(0) \) is bounded by a constant, notably including the “parameter-free” algorithms that obtain regret bounds of the form \( R_T(u) = \tilde{O}(\|u\| \sqrt{T}) \) without knowledge of \( \|u\| \). There are already a number of such algorithms which guarantee regret bounds adapting to different characteristics of the sequence \( g_t \) or comparison point \( u \) [7, 8, 9, 10, 11]. Our meta-algorithm frees the user from having to choose which algorithm is best for the task at hand.

Next, we develop a variation of this algorithm-combining technique that yields optimistic regret guarantees. In optimistic online learning, the algorithm is provided with a “hint” \( h_t \) that is some estimate of \( g_t \) before deciding on the prediction \( w_t \). The goal is to use \( h_t \) in such a way that the regret is very small when \( h_t \) is a good estimate of \( g_t \) [12, 13, 14, 15]. A classic optimistic regret bound when \( W \) has diameter \( D = \sup_{x, y \in W} \|x - y\| \) is:

\[
R_T(u) \leq O \left( D \sqrt{\sum_{t=1}^{T} \|g_t - h_t\|^2} \right)
\]

Our approach is a reduction that takes an algorithm obtaining regret \( R_T(u) \leq B(u) \sqrt{\sum_{t=1}^{T} \|g_t\|^2} \) for some arbitrary...
function $B$ and returns an algorithm obtaining regret

$$R_T(u) \leq O \left[ B(u) \min \left( \sqrt{\sum_{t=1}^{T} \|g_t\|^2}, \sqrt{\sum_{t=1}^{T} \|g_t - h_t\|^2} \right) \right]$$

This improves on prior results in several ways. First, our algorithm is a generic reduction, and so can be applied to make optimistic versions of any new algorithms that may yet be invented. Second, it allows us to construct the first parameter-free optimistic algorithm (e.g. unbounded $W$, $\dot{B}(u) = \tilde{O}(\|u\|)$). Third, when $W$ is unconstrained, we can improve our results to replace $\sum_{t=1}^{T} \|g_t - h_t\|^2$ with $\max(\sum_{t=1}^{T} \|g_t - h_t\|^2 - \|h_t\|^2, 1)$. Finally, our optimistic algorithm is “safe” in the sense that even if the hints $h_t$ are very bad we still do no worse than the original algorithm.

This paper is organized as follows. First, we introduce our technique for combining online learning guarantees (Section 2), and provide an efficient algorithm that adapts to many norms at the same time as a simple example of the technique in action. Next, we apply this technique to generate optimistic algorithms in unconstrained domains (Section 3), and see a generic improvement in adaptivity over prior optimistic algorithms while also maintaining good performance in the face of poor-quality $h_t$. We then proceed to adapt our optimistic algorithm to constrained domains (Section 4), matching prior bounds while again being robust to bad $h_t$. Finally, we demonstrate how to take advantage of multiple sequences of hints (Section 5), obtaining an optimistic guarantee that matches the performance on the best sequence of hints in hindsight. We conclude with a simple trick showing how to compete with the best fixed hint (Section 6).

### 1.1 Definitions and Notation

Throughout this paper we assume $W$ is a convex subset of a real Hilbert space. Given a norm $\| \cdot \|$, we write $\| \cdot \|_*$ to indicate the dual norm $\| g \|_* = \sup_{\| x \| \leq 1} \langle g, x \rangle$. We always use $\| \cdot \|$ to indicate the Hilbert space norm unless otherwise stated, so that $\| \cdot \| = \| \cdot \|_*$ by the standard identification of a Hilbert space with its dual. Given a convex function $f$, we write $x \in \partial f(y)$ to indicate that $x$ is a subgradient of $f$ at $y$. We interchangeably refer to $g_t$ as losses and gradients. We will often assume the $g_t$ are bounded $\|g_t\| \leq 1$ for all $t$, which will be stated explicitly in the hypotheses of the relevant results. As usual, $\epsilon$ indicates the base of the natural logarithm.

### 2 Combining Parameter-Free Algorithms

In this section we provide our technique for combining incomparable regret guarantees. Our technique is most effective on algorithms that ensure $R_T(0) \leq \epsilon$ for some (usually user-specified) $\epsilon$. There has been much recent work on this style of algorithm [16, 17, 8, 18, 10], yielding so-called parameter-free algorithms that achieve optimal or near-optimal regret guarantees up to log factors. These works provide various improvements in adaptivity to the norm of $u$ or the gradients $g_t$. However, there is no one uniformly-dominant adaptive guarantee. As a simple example, under the assumption $\|g_t\|_* \leq 1$ for all $t$, recently [10] provided algorithms that obtain

$$R_T(u) \leq \tilde{O} \left[ \epsilon + \frac{\|u\|}{\sqrt{T}} \max \left( \log \left( \frac{\|u\| T}{\epsilon} \right), \sqrt{\sum_{t=1}^{T} \|g_t\|^2 \log \left( \frac{\|u\| T}{\epsilon} \right)} \right) \right]$$

(1)

for any norm fixed $\| \cdot \|$ such that $\| \cdot \|^2$ is $\lambda$-strongly convex with respect to the norm $\| \cdot \|$.

Further, by running a single 1-dimensional copy of this algorithm in each coordinate of a $d$-dimensional problem and rescaling $\epsilon$ to $\epsilon/d$, we can obtain the regret:

$$R_T(u) \leq \tilde{O} \left[ \epsilon + \sum_{t=1}^{T} |u_t| \max \left( \log \left( \frac{|d|u_t| T}{\epsilon} \right), \sqrt{\sum_{t=1}^{T} |g_{t,i}|^2 \log \left( \frac{|d|u_t| T}{\epsilon} \right)} \right) \right]$$

(2)

These regret guarantees are optimal (up to log factors) and also incomparable a priori. Depending on the gradients $g_t$ and the benchmark $u$, it may be best to use the per-coordinate algorithm or it may be best to use some particular norm. Thus the “ultimate adaptive algorithm” would be able to achieve the best of all these bounds in hindsight. One
Theorem 1. Suppose $W$ is a Hilbert space. Let $A$ and $B$ be two online linear optimization algorithms that guarantee regret $R_T^A(u)$ and $R_T^B(u)$ respectively. Let $w_t^A$ and $w_t^B$ be their respective predictions on the loss sequence $g_1, \ldots, g_T$. Let $w_t = w_t^A + w_t^B$. Then we have

$$R_T(u) = \sum_{t=1}^T \langle g_t, w_t - u \rangle \leq \inf_{x+y=u} R_T^A(x) + R_T^B(y)$$

In particular, if $R_T^A(0) \leq \epsilon$ and $R_T^B(0) \leq \epsilon$, we have

$$R_T(u) \leq \epsilon + \min(R_T^A(u), R_T^B(u))$$

Proof. The proof is one line:

$$\sum_{t=1}^T \langle g_t, w_t - u \rangle = \sum_{t=1}^T \langle g_t, w_t^A - x \rangle + \langle g_t, w_t^B - y \rangle \leq R_T^A(x) + R_T^B(y)$$

With this strategy it is clear that we can combine any $k$ algorithms and obtain only an additive penalty of $(k-1)\epsilon$ over the best of their regret bounds. Since parameter-free algorithms with guarantees like (1) and (2) depend on $\log(1/\epsilon)$, we can replace $\epsilon$ with $\epsilon/k$ to increase the regret by a factor of $\log(k)$ in exchange for guaranteeing only $\epsilon$ regret at 0. Finally, we note that our assumption that $W$ is an entire Hilbert space can usually be removed using the unconstrained-to-constrained reduction of [10].

We can gain some more insight into why a result such as Theorem 1 should be expected to exist by appealing to the equivalence between regret bounds and concentration inequalities outlined by [21]. Roughly speaking, this result says that a regret bound of $R_T$ implies that sums of mean-zero random variables concentrate about their mean with a radius of roughly $R_T$ - and vice versa. Therefore one should be able to convert a concentration bound into an online learning algorithm, although the conversion may be very computationally taxing. There is already an extremely popular technique for combining concentration inequalities - the union bound - so there should be a corresponding way to combine regret bounds. In this way we can view Theorem 1 as providing an extremely efficient online learning analog to the union bound.

One immediate application of Theorem 1 is to combine an algorithm that obtains the bound (2) with one that obtains the bound (1) where $\| \cdot \|_2$. This yields an algorithm that simultaneously enjoys a “dimension-free” bound with respect to $\| \cdot \|_2$ while also reaping the benefits of per-coordinate updates when the gradients $g_i$ or comparison point $u$ are sparse.

A second application is to adapt to many norms simultaneously. For example, in the following Theorem we construct an algorithm that adapts to any $p$-norm for $p \in [1, 2]$. The strategy is simple: first, we show that by selecting a discrete grid of $\log(d)$ different $p_i$, we can ensure $\|x\|_p$ is within a constant of $\|x\|_{p_i}$ for some $i$ for any $p \in [1, 2]$ (Lemma 8). Then we observe that $\| \cdot \|_p$ is $p-1$-strongly-convex with respect to itself, so that combining the guarantee (1) with our algorithm-combining strategy immediately yields the desired results (Theorem 2).

Theorem 2. Suppose $g_t \in \mathbb{R}^d$ satisfies $\|g_t\|_2 \leq 1$ for all $t$. Then there exists an online algorithm that runs in time $O(d \log(d))$ per update that obtains regret

$$R_T(u) \leq \tilde{O}(\epsilon \log(d) + \inf_{p \in [1, 2]} \|u\|_p \sqrt{\frac{1}{\epsilon}} \sum_{t=1}^T \|g_t\|_q^2)$$

where for any $p, q$ is such that $\frac{1}{p} + \frac{1}{q} = 1$. 


Now we turn our best-of-all-worlds strategy into an optimistic online learning algorithm. Specifically, we will provide an optimistic regret guarantee. Importantly, this extra analysis allows us to dispense with the requirement that we use two approaches using the technique of Theorem 1, and then add some more detailed analysis to derive the optimistic regret guarantee. for all $t$

Let $A$ be an algorithm that guarantees regret $\epsilon$.

Proof. Consider $q_0 = 2$ and $\frac{1}{q_i} = \frac{1}{q_{i-1}} - \frac{1}{\log(d)}$ for all $i \leq \log(d)/2$ and $p_i$ given by $\frac{1}{p_i} + \frac{1}{q_i} = 1$. Note that there are $O(\log(d))$ different indices $i$. Then by Lemma 3 for any $p \in [1, 2]$, there is some $p_i \geq p$ such that $\|u\|_{p_i} \leq \|u\|_p$ and $\|g\|_{q_i} \leq \epsilon \|g\|_{q_i}$ for all $g$. Recall that $\|\cdot\|_{p_i}$ is $p_i - 1$-strongly convex with respect to $\|\cdot\|_{p_i}$. Then considering running one algorithm for each $p_i$ that guarantees regret

$$R_T(u) \leq \epsilon + \tilde{O}\left(\frac{\|u\|_{p_i}}{\sqrt{p_i}} \sum_{t=1}^{T} \|g_t\|_{q_i}^{p_i}\right)$$

where $\frac{1}{p_i} + \frac{1}{q_i} = 1$. Note that this is possible because $\|g_t\|_{q_i} \leq \|g_t\|_2 \leq 1$ for all $t$. Then by combining all $O(\log(d))$ of these algorithms using Theorem 1, we obtain the stated result.

Similar bounds have been shown in previous work: [20] achieved a similar bound using an expert algorithm to combine the base algorithms. However, the expert algorithm dominates the runtime and leads to both $O(T)$ time per update, and also to loss of adaptivity to the sum of the squared norms of the gradients. Also, [10] provides an algorithm that adapts to any sequence of norms simultaneously, but the algorithm requires $O(d^2)$ time per update and incurs an extra $\sqrt{d}$ factor in the regret bound. In contrast, the algorithm presented above is simple, adaptive, and efficient.

This best-of-all-worlds technique has powerful applications beyond simply combining existing regret guarantees. In particular, it enables us to combine algorithms that do not guarantee sublinear regret with algorithms that do have reasonable worst-case regret guarantees. This enables us to generate algorithms that perform well all the time (because they do no worse than the algorithm with a worst-case guarantee), but may sometimes perform much better because sometimes the algorithm without a sublinear regret guarantee may “get lucky” and perform extremely well. In the following sections, we elaborate on this idea to develop optimistic online algorithms.

### 3 Optimism

Now we turn our best-of-all-worlds strategy into an optimistic online learning algorithm. Specifically, we will provide a black-box reduction that converts any algorithm $A$ that obtains regret

$$R_T(u) \leq B(u) \sum_{t=1}^{T} \|g_t\|^2$$

into an optimistic algorithm obtaining regret

$$R_T(u) \leq B(u) \sum_{t=1}^{T} \|g_t - h_t\|^2$$

We first tackle the problem in the case that $W$ is an entire Hilbert space (no constraints), and then move to a constrained setting in Section 4.

Our strategy uses a 1-dimensional parameter-free algorithm to take advantage of the hint $h_t$. Intuitively, if $h_t = g_t$ for all $t$, then playing $w_t = -y h_t$ for some sufficiently large positive constant $y$ will yield small regret. We can learn this constant $y$ on-the-fly by using a 1-dimensional online algorithm. Alternatively, if the hints are bad, then simply running $A$ will yield reasonably low regret (although perhaps not as low as in the former case). We combine these two approaches using the technique of Theorem 1 and then add some more detailed analysis to derive the optimistic regret guarantee. Importantly, this extra analysis allows us to dispense with the requirement that $A$ guarantees regret $\epsilon$ at the origin, so that we can make optimistic versions of essentially any adaptive online learning algorithm.

**Theorem 3.** Let $W$ be a Hilbert space. Suppose $A$ guarantees regret

$$R_T^A(u) \leq A_T(u) + B_T(u) \sum_{t=1}^{T} \|g_t\|^2$$

then

$$R_T(u) \leq \tilde{O}(\|w_T\|_2^2)$$
Next, we recall that [10] provides a 1-D algorithm that satisfies the conditions of Theorem 3 for on gradients $g_t$ and suppose $B$ guarantees regret

$$R_T^B(u) \leq \epsilon + |u|C \log(1 + |u|T^c/\epsilon) + |u|D \sqrt{\sum_{t=1}^{T} z_t^2 \log(1 + |u|T^c/\epsilon)}$$

on gradients $z_t$ with $|z_t| \leq 1$, where $A_T$ and $B_T$ are arbitrary non-negative functions and $C$, $c$ and $D$ and $\epsilon$ are arbitrary nonnegative constants. Finally, suppose $\|h_t\| \leq 1$ and $\|g_t\| \leq 1$ for all $t$. Then Algorithm 1 guarantees regret

$$R_T(u) \leq B_T(u) \left[ (2C + D^2) \log(e + B_T(u)T^c/\epsilon) + \sum_{t=1}^{T} \|h_t - g_t\|^2 - \|h_t\|^2 \right]^{1/2} + DB_T(u) \sqrt{\log(e + B_T(u)T^c/\epsilon)} + A_T(u) + \epsilon$$

where $[X]_1$ denotes $\max(X, 1)$. Further, Algorithm 1 simultaneously guarantees regret

$$R_T(u) \leq \epsilon + R_T^A(u)$$

Let us unpack this Theorem. If we remove all logarithmic factors, then the Theorem states that

$$R_T(u) \leq \tilde{O} \left( B_T(u) \left[ \sum_{t=1}^{T} \|h_t - g_t\|^2 - \|h_t\|^2 \right]^{1/2} + A_T(u) + \epsilon \right)$$

Next, we recall that [10] provides a 1-D algorithm that satisfies the conditions of Theorem 3 for $B$ as well as an algorithm that satisfies the conditions for $A$ with $B_T(u) = O(\|u\| \sqrt{\log(\|u\| T^c/\epsilon)})$ and $A_T(u) = O(\|u\| \log(\|u\| T^c/\epsilon)) + \epsilon$ [1]. Thus using these algorithms we obtain:

$$R_T(u) \leq \tilde{O} \left( \epsilon + \|u\| \left[ \sum_{t=1}^{T} \|h_t - g_t\|^2 - \|h_t\|^2 \right]^{1/2} \right)$$

This is already somewhat better (modulo log factors) than the standard optimistic guarantee by virtue of the $-\|h_t\|^2$ terms. Further, this algorithm is unconstrained, and to our knowledge is the first unconstrained algorithm to achieve this optimistic guarantee. Even more, the second part of the Theorem shows that we never do worse than the base algorithm $A$ regardless of the values of $h_t$. This greatly robustifies optimistic online algorithms, as it allows the use of arbitrary hint sequences that may have absolutely no relationship with $g_t$ without harming the regret guarantees.

Now we provide the proof of Theorem 3. As sketched above, the main idea is that we are using Theorem 1 to combine the regret of $A$ and $B$. By careful analysis of the regret of these two algorithms we can interpolate between the optimal scenario for $B$ (i.e. when the $g_t = h_t$ for all $t$), and the more general adversarial scenario.

---

Algorithm 1 Optimistic Reduction

**Input:** Online learning algorithm $A$ with domain $W$ and $B$ with domain $\mathbb{R}$.

for $t = 1$ to $T$ do
   Get $x_t$ from $A$ and $y_t$ from $B$.
   Get hint $h_t$.
   Play $w_t = x_t - y_t h_t$, receive loss $g_t$.
   Send $g_t$ to $A$ as the $t$th loss.
   Send $-\langle g_t, h_t \rangle$ to $B$ as the $t$th loss.
end for
Proof. We write the regret

\[ R_T(u) = \sum_{t=1}^{T} (g_t, w_t - u) \]

\[ = \sum_{t=1}^{T} (g_t, x_t - u) - (g_t, h_t) y_t \]

\[ \leq R_T^A(u) + R_T^B(y) - \sum_{t=1}^{T} (g_t, h_t) y \]

Now we can actually immediately see the second part of the Theorem: just set \( y = 0 \) and observe that \( R_T^B(0) \leq \epsilon \).

With this out of the way, we continue to unpack our regret inequality:

\[ R_T(u) \leq A_T(u) + B_T(u) \sqrt{\sum_{t=1}^{T} \|g_t\|^2 + yD \sum_{t=1}^{T} \langle g_t, h_t \rangle^2 \log(1 + |y|T^c/\epsilon)} \]

\[ + \epsilon + yC \log(1 + |y|T^c/\epsilon) - \sum_{t=1}^{T} (g_t, h_t) y \]

\[ \leq A_T(u) + (B_T(u) + yD \sqrt{\log(1 + |y|T^c/\epsilon)}) \sqrt{\sum_{t=1}^{T} \|g_t\|^2 - \sum_{t=1}^{T} \langle g_t, h_t \rangle y} \]

\[ + \epsilon + yC \log(1 + |y|T^c/\epsilon) \]

Where in the second line we used \( \|h_t\| \leq 1 \).

Now consider the identity \(-2\langle g_t, h_t \rangle = \|g_t - h_t\|^2 - \|g_t\|^2 - \|h_t\|^2\). Applying this yields

\[ R_T(u) \leq yC \log(1 + |y|T^c/\epsilon) + (B_T(u) + yD \sqrt{\log(1 + |y|T^c/\epsilon)}) \sqrt{\sum_{t=1}^{T} \|g_t\|^2} \]

\[ - \frac{y}{2} \sum_{t=1}^{T} \|g_t\|^2 + \frac{y}{2} \sum_{t=1}^{T} \|h_t - g_t\|^2 - \|h_t\|^2 + A_T(u) + \epsilon \]

\[ \leq yC \log(1 + |y|T^c/\epsilon) + \frac{y}{2} \sum_{t=1}^{T} \|h_t - g_t\|^2 - \|h_t\|^2 \]

\[ + \sup_{X \geq 0} \left[ (B_T(u) + yD \sqrt{\log(1 + |y|T^c/\epsilon)}) \sqrt{X} - \frac{y}{2} X \right] + A_T(u) + \epsilon \]

\[ \leq yC \log(1 + |y|T^c/\epsilon) + \frac{y}{2} \sum_{t=1}^{T} \|h_t - g_t\|^2 - \|h_t\|^2 \]

\[ + \frac{(B_T(u) + yD \sqrt{\log(1 + |y|T^c/\epsilon)})^2}{2y} + A_T(u) + \epsilon \]
where in the last line we have used the assumption $y \geq 0$. Now we optimize $y$:

$$R_T(u) \leq \inf_{y \geq 0} \left[ yC \log(1 + yT^c/\epsilon) + \frac{y}{2} \sum_{t=1}^{T} \| h_t - g_t \|^2 - \| h_t \|^2 \right. $$

$$+ \left. \left( B_T(u) + yD \sqrt{\log(1 + yT^c/\epsilon)} \right)^2 \right] + A_T(u) + \epsilon$$

$$\leq \inf_{y \geq 0} \left[ \frac{y}{2} \left( 2C + D^2 \right) \log(1 + yT^c/\epsilon) + \sum_{t=1}^{T} \| h_t - g_t \|^2 - \| h_t \|^2 \right] + \frac{B_T(u)^2}{2y} + DB_T(u) \sqrt{\log(e + yT^c/\epsilon)} + A_T(u) + \epsilon$$

This infimum is computed in Lemma 9 yielding:

$$R_T(u) \leq B_T(u) \sqrt{\left[ (2C + D^2) \log(e + B_T(u)T^c/\epsilon) + \sum_{t=1}^{T} \| h_t - g_t \|^2 - \| h_t \|^2 \right]} + DB_T(u) \sqrt{\log(e + B_T(u)T^c/\epsilon)} + A_T(u) + \epsilon$$

as desired.

\[ \square \]

4 Constrained Optimism

The reduction Algorithm 1 requires an unconstrained domain in order to form the updates $x_t - y_t h_t$. To move to the constrained setting, we use the unconstrained-to-constrained reduction from [10]. When used out-of-the-box, this reduction converts an unconstrained algorithm whose regret as a function of the gradients $g_t$ is $R_T(u, g_t, \ldots, g_T)$ into a constrained algorithm that obtains regret $2R_T(u, \tilde{g}_t, \ldots, \tilde{g}_T)$ where $\tilde{g}_t$ is a “surrogate gradient” with $\| \tilde{g}_t \| \leq \| g_t \|$. Unfortunately, this is not quite good enough to maintain optimism as $\tilde{g}_t$ may be less similar to $h_t$ than $g_t$ was. However, by inspecting the internals of the reduction, we can remedy this issue.

Specifically, the reduction of [10] replaces the iterates $w_t$ of the unconstrained online learning algorithm with $\Pi(w_t)$ and the gradients $g_t$ with $\tilde{g}_t = \frac{h_t}{2} + \frac{\|h_t\|}{2} \nabla S(w_t)$, where $\Pi(w) = \arg\min_{w' \in W} \| w - w' \|$ and $S(w) = \| w - \Pi(x) \|$ = $\inf_{w' \in W} \| x - w' \|$ (in Hilbert spaces, $\arg\min_{w' \in W} \| w - w' \|$ is always a singleton). We therefore apply the same transformation to the hint $h_t$, replacing it with $\tilde{h}_t = \frac{h_t}{2} + \frac{\|h_t\|}{2} \nabla S(h_t)$. This strategy is actually somewhat more subtle than it appears because $w_t$ is a function of $\tilde{h}_t$: $w_t = x_t - y_t \tilde{h}_t$, and so the setting $\tilde{h}_t = \frac{h_t}{2} + \frac{\|h_t\|}{2} \nabla S(h_t)$ actually represents an equation that must be solved for the value of $\tilde{h}_t$. Fortunately, it turns out that this equation is not too difficult to solve, with the help of the following Lemma:

Lemma 4. Let $W$ be a convex domain in a Hilbert space $H$. Let $x \in H$, $y \in \mathbb{R}$, $h \in H$. Let $z \in \partial S(x - yh/2)$. Suppose $\frac{\|y\|^2}{2} \leq S(x - yh/2)$. Then $z \in \partial S(x - yh)$ for $\tilde{h} = \frac{h}{2} + \frac{\|h\|^2}{2}$. If instead $\frac{\|y\|^2}{2} < S(x - yh/2)$, then $az \in \partial S(x - yh)$ for $\tilde{h} = \frac{h}{2} + \frac{\|h\|^2}{2} - \frac{2S(x - yh)}{y|y|}$. where $a = 2S(x - yh)\frac{y}{y|h|}$.

Intuitively, this Lemma tells us that most of the time if we set $\tilde{h}_t = \frac{h_t}{2} + \frac{\|h_t\|^2}{2}$ for $z = \nabla S(x_t - y_t h_t/2)$, we will have $\tilde{h}_t = \frac{h_t}{2} + \frac{\|h_t\|^2}{2} \nabla S(w_t)$, where $w_t = x_t - y_t \tilde{h}_t$. This suggests the reduction given by Algorithm 2 for constrained optimism.

Theorem 5. Under the same assumptions as Theorem 3 with the exception that $W$ is now a convex domain in a Hilbert space rather than necessarily the entire space. Then Algorithm 2 guarantees regret

$$R_T(u) \leq 2B_T(u) \left[ (2C + D^2) \log(e + B_T(u)T^c/\epsilon) + \sum_{t=1}^{T} \| h_t - g_t \|^2 \right]$$

$$+ 2 DB_T(u) \sqrt{\log(e + B_T(u)T^c/\epsilon)} + 2A_T(u) + 2\epsilon$$
Many Hints At Once

The classical optimistic online learning setup considers a single hint $h_t$ provided at each round. However, one could also imagine a scenario in which multiple hints $h_{t,i}, \ldots, h_{t,i}$ are provided in each round. Our reduction allows us to handle this case seamlessly by combining the best-of-all-words analysis of Theorem 1 with Algorithm 1. In a nutshell, we consider updates of the form $x_t - \sum_{i=1}^{k} y_{t,i} h_{t,i}$ and use $k$ independent 1-dimensional optimizers to optimize each $y_{t,i}$. This roughly corresponds to applying Theorem 1 to the problem of choosing the best hints, and so we suffer only an additive penalty of $\epsilon k$ to compete with the best hint sequence. The resulting pseudocode is in Algorithm 3, which we analyze in Theorem 6.

This reduction obtains the guarantee

**Theorem 6.** Under the same assumptions as Theorem 1, Algorithm 1 guarantees regret

\[
R_T(u) \leq B_T(u) \left[ (2C + D^2) \log(e + B_T(u)T^c/\epsilon) + \sum_{t=1}^{T} \|h_{t,i} - g_t\|^2 - \|h_{t,i}\|^2 \right]^{1/2} + DB_T(u) \sqrt{\log(e + B_T(u)T^c/\epsilon)} + A_T(u) + k\epsilon
\]
Thus we can use an adaptive gradient descent algorithm with domain $W$ and $B$ with domain $\mathbb{R}$.

By utilizing Theorem 6, we can obtain both bounds at the same time. We suggest a simple scheme that generates hints that perform as well as this latter bound, which somewhat streamlines the analysis of [12].

The above technique can actually be improved in the unconstrained setting. Since our unconstrained optimistic technique is quite simple: we use an online learning algorithm to choose $h_t$ in response to losses $\ell_t$. Conveniently, $\ell_t(h)$ is strongly-convex and so if we use the Follow-The-Leader algorithm to pick $h_t$ (which corresponds to using the running-average $h_t = \frac{1}{T} \sum_{i=1}^{T} g_i$), we obtain [3]:

$$\sum_{t=1}^{T} \ell_t(h_t) - \ell_t(\overline{h}) \leq O(\log(T))$$

Plugging this into the regret bound of Theorem 3, we have regret

$$R_T(u) \leq \tilde{O}\left(e + \frac{\|u\|}{\sqrt{T}} \sum_{t=1}^{T} \|g_t - \|^2\right) \tag{3}$$

Up to log factors, this represents a generic improvement in adaptivity over the standard regret bound that depends on $\sum_{t=1}^{T} \|g_t\|^2$, and generalizes the regret bound of [12] to unconstrained domains. Further, we remark in Appendix A that the existence of this algorithm actually provides a simple proof of an empirical Bernstein bound in Hilbert spaces.

The above technique can actually be improved in the unconstrained setting. Since our unconstrained optimistic guarantee depends on $\sum_{t=1}^{T} \|g_t - h_t\| - \|h_t\|^2 = \sum_{t=1}^{T} \langle g_t, g_t - 2h_t \rangle$ rather than $\sum_{t=1}^{T} \|g_t - h_t\|^2$, we can set $\ell_t(h) = \langle g_t, g_t - 2h \rangle$ to obtain an even tighter bound. Notice now that $\ell_t$ is no longer strongly-convex, but it is still convex. Thus we can use an adaptive gradient descent algorithm with domain $\{\|h\| \leq 1\}$ (e.g. Adagrad) [5, 22] to obtain:

$$\sum_{t=1}^{T} \ell_t(h_t) - \ell_t(\overline{h}) \leq O\left(\sqrt{\frac{\sum_{t=1}^{T} \|g_t\|^2}{T}}\right)$$

### Algorithm 3 Optimism with Many Hints

**Input:** Online learning algorithm $A$ with domain $W$ and $B$ with domain $\mathbb{R}$.

Initialize $k$ copies of $B, B_1, \ldots, B_k$.

for $t = 1$ to $T$

- Get $x_t$ from $A$ and $y_{t,i}$ from $B_i$ for $i \in [1,k]$.
- Get hints $h_{t,1}, \ldots, h_{t,k}$.
- Play $w_t = x_t - \sum_{i=1}^{k} y_{t,i}, h_{t,i}$, receive loss $g_t$.
- Send $g_t$ to $A$ as the $t$th loss.
- Send $-\langle g_{t,i}, h_t \rangle$ to $B_i$ as the $t$th loss for all $i$.

end for

for all $i$, where $[X]_1$ denotes $\max (X, 1)$. Further, Algorithm 7 simultaneously guarantees regret

$$R_T(u) \leq k \epsilon + R_T^A(u)$$

### 6 Best Fixed Hint

So far we have discussed how to use hints $h_t$ effectively, but given no consideration to where the hints come from. In many cases, there is no external oracle providing hints and so they must be constructed from other information. One popular choice is $h_t = g_{t-1}$. This yields bounds that depend on $\sum_{t=1}^{T} \|g_t - g_{t-1}\|^2$ and so obtain low regret when the gradients are “slowly varying”. Another approach suggested by [12] yields regret bounds that depend on $\sum_{t=1}^{T} \|g_t - \overline{g}\|^2$ where $\overline{g} = \frac{1}{T} \sum_{t=1}^{T} g_t$ - which is an optimistic regret bound using the best fixed hint. In this section, we suggest a simple scheme that generates hints that perform as well as this latter bound, which somewhat streamlines the analysis of [12].

The technique is quite simple: we use an online learning algorithm to choose $h_t$. Define $\ell_t(h) = \|g_t - h\|^2$, then we have

$$\sum_{t=1}^{T} \|g_t - h_t\|^2 \leq \sum_{t=1}^{T} \ell_t(h_t) - \ell_t(\overline{h}) + \sum_{t=1}^{T} \|g_t - \overline{g}\|^2$$

for any arbitrary $\overline{h}$. Further $\sum_{t=1}^{T} \ell_t(h_t) - \ell_t(\overline{h})$ is simply the regret of an online algorithm that plays $h_t$ in response to losses $\ell_t$. Conveniently, $\ell_t(h)$ is strongly-convex and so if we use the Follow-The-Leader algorithm to pick $h_t$ (which corresponds to using the running-average $h_t = \frac{1}{T} \sum_{i=1}^{T} g_i$), we obtain [3]:

$$\sum_{t=1}^{T} \ell_t(h_t) - \ell_t(\overline{h}) \leq O(\log(T))$$

Plugging this into the regret bound of Theorem 3, we have regret

$$R_T(u) \leq \tilde{O}\left(e + \frac{\|u\|}{\sqrt{T}} \sum_{t=1}^{T} \|g_t - \|^2\right) \tag{3}$$

Up to log factors, this represents a generic improvement in adaptivity over the standard regret bound that depends on $\sum_{t=1}^{T} \|g_t\|^2$, and generalizes the regret bound of [12] to unconstrained domains. Further, we remark in Appendix A that the existence of this algorithm actually provides a simple proof of an empirical Bernstein bound in Hilbert spaces.
In this case the optimal value of $h$ is

$$h = \frac{\sum_{t=1}^{T} g_t}{\sum_{t=1}^{T} g_t},$$

so that we have

$$\sum_{t=1}^{T} \|g_t - h_t\|^2 - \|h_t\|^2 \leq \sum_{t=1}^{T} \|g_t - \overline{h}\|^2 - \|\overline{h}\|^2 + \epsilon_t(h_t) - \epsilon_t(\overline{h})$$

$$\leq \sum_{t=1}^{T} \|g_t\|^2 - 2 \sum_{t=1}^{T} g_t + O\left(\sqrt{\sum_{t=1}^{T} \|g_t\|^2}\right)$$

If we then apply the optimistic bound of Theorem 3 we have

$$R_T(u) \leq \tilde{O}\left(\epsilon + \|u\| \sqrt{\sum_{t=1}^{T} \|g_t\|^2 - 2 \sum_{t=1}^{T} g_t} + \sqrt{\sum_{t=1}^{T} \|g_t\|^2}\right)$$

7 Conclusion

We introduced the simple strategy of adding iterates as a method for obtaining best-of-all-worlds style regret guarantees in parameter-free online learning. Further, a variation on this technique yields optimistic regret bounds. Our optimistic algorithm is a generic reduction that converts any adaptive online learning algorithm into an optimistic algorithm. This extends optimism to unconstrained domains, allows algorithms to use many sequences of hints, and does not degrade performance when the hints are poor. Finally, we provide a simple technique that competes with the best fixed hint, which can be used to provide a simple proof of an empirical Bernstein bound. Intuitively, we achieved optimism by combining an algorithm that had an excellent best-case guarantee but a poor worst-case guarantee with a “safety-net” algorithm that had reasonable worst-case guarantees. It is our hope that similar synergies with other algorithms will yield further increases in adaptivity.

References

[1] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In Proceedings of the 20th International Conference on Machine Learning (ICML-03), pages 928–936, 2003.

[2] Shai Shalev-Shwartz. Online learning and online convex optimization. Foundations and Trends in Machine Learning, 4(2):107–194, 2011.

[3] H. Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. arXiv preprint arXiv:1403.3465, 2014.

[4] Nicolo Cesa-Bianchi, Alex Conconi, and Claudio Gentile. On the generalization ability of on-line learning algorithms. Information Theory, IEEE Transactions on, 50(9):2050–2057, 2004.

[5] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. In Conference on Learning Theory (COLT), 2010.

[6] Stephane Ross, Paul Mineiro, and John Langford. Normalized online learning. In Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence (UAI), 2013.

[7] Francesco Orabona. Simultaneous model selection and optimization through parameter-free stochastic learning. In Advances in Neural Information Processing Systems, pages 1116–1124, 2014.

[8] Dylan J Foster, Alexander Rakhlin, and Karthik Sridharan. Adaptive online learning. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems 28, pages 3375–3383. Curran Associates, Inc., 2015.

[9] Francesco Orabona and Dávid Pál. Coin betting and parameter-free online learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 577–585. Curran Associates, Inc., 2016.
A Concentration Inequality

In this section we convert the “best fixed hint” bound in Section 6 into a concentration inequality in Hilbert spaces following the approach of [21], who describe an elegant general equivalence between online learning algorithms and concentration inequalities. Although we suspect the constants in our bound can be significantly improved by a more involved direct analysis, we think the simplicity of this argument is interesting in of itself. First we describe the general procedure to turn regret bounds into concentration inequalities. We run an online algorithm with gradients $g_t = X_t - \mathbb{E}[X_t]$ where $X_1, \ldots, X_t$ are i.i.d. random variables such that $\|X_t - \mathbb{E}[X_t]\| \leq 1$ with probability 1. Suppose our algorithm guarantees $R_T(0) \leq \epsilon$ for some $\epsilon$. Then if we set $u = -\epsilon \frac{\sum_{t=1}^{T} g_t}{\sum_{t=1}^{T} g_t}$ we have:

$$
\epsilon - \sum_{t=1}^{T} (g_t, w_t) = c \left\| \sum_{t=1}^{T} g_t \right\| - R_T(u) + \epsilon = \mathbb{E} \left[ c \left\| \sum_{t=1}^{T} g_t \right\| - R_T(u) + \epsilon \right]
$$
Further, since $\epsilon - \sum_{t=1}^{T} \langle g_t, w_t \rangle = \epsilon - R_T(0)$, we have $\epsilon - \sum_{t=1}^{T} \langle g_t, w_t \rangle \geq 0$ so that Markov’s inequality we can say that with probability at least $1 - \delta$,

$$c \left\| \sum_{t=1}^{T} g_t \right\| - R_T(u) + \epsilon \leq \frac{\epsilon}{\delta}$$

This, in tandem with appropriate algebra, provides a concentration inequality of roughly $\| \sum_{t=1}^{T} X_t - \mathbb{E}[\sum_{t=1}^{T} X_t] \| \leq R_T \left( - \frac{\sum_{t=1}^{T} g_t}{\| \sum_{t=1}^{T} g_t \|} \right)$. In particular, since $R_T(u)$ depends on $g_t - \mathbb{E}$, we have $g_t - \mathbb{E} = X_t - \mathbb{X}$ where $\mathbb{X} = \frac{1}{T} \sum_{t=1}^{T} X_t$, and so we recover the empirical Bernstein inequality [23], generalized to Hilbert spaces:

**Theorem 7.** Suppose $X_1, \ldots, X_T$ are i.i.d. random variables in a Hilbert space such that $\| X_t - \mathbb{E}[X_t] \| \leq 1$ with probability 1. Then with probability at least $1 - \delta$,

$$\left\| \sum_{t=1}^{T} X_t - \mathbb{E} \left[ \sum_{t=1}^{T} X_t \right] \right\| \leq \tilde{O} \left( 1 + \sqrt{\sum_{t=1}^{T} \left\| X_t - \mathbb{E}[X_t] \right\|^2} \right)$$

**Proof.** Recall that our proof strategy is to run an online learning algorithm on the gradient sequence $g_t = X_t - \mathbb{E}[X_t]$, which yields

$$R_T(u) = \sum_{t=1}^{T} \langle g_t, w_t \rangle + c \left\| \sum_{t=1}^{T} g_t \right\|$$

$$\epsilon = \mathbb{E} \left[ c \left\| \sum_{t=1}^{T} g_t \right\| - R_T(u) + \epsilon \right]$$

Suppose the online learning algorithm is an algorithm that obtains an optimistic guarantee with a best fixed-hint:

$$R_T(u) \leq O \left( \epsilon + \| u \| \sqrt{\sum_{t=1}^{T} \| g_t - \mathbb{E} \|^{2} \log(\| u \| T / \epsilon) + \| u \| \log(\| u \| T / \epsilon)} \right)$$

where $\mathbb{E} = \frac{1}{T} \sum_{t=1}^{T} g_t$. Then $R_T(0) \leq \epsilon$ with probability 1, so that $\epsilon - \sum_{t=1}^{T} \langle g_t, w_t \rangle \geq 0$ with probability 1. Therefore by Markov’s inequality, with probability at least $1 - \delta$,

$$c \left\| \sum_{t=1}^{T} g_t \right\| - R_T(u) + \epsilon \leq \frac{\epsilon}{\delta}$$

$$\left\| \sum_{t=1}^{T} g_t \right\| \leq \frac{R_T(u)}{c} + \epsilon (1 + \delta)$$

$$\left\| \sum_{t=1}^{T} X_t - \mathbb{E} \left[ \sum_{t=1}^{T} X_t \right] \right\| \leq \inf_{c} \frac{R_T(u)}{c} - \epsilon c + \frac{\epsilon}{\delta}$$

Now setting $\epsilon = \delta$ and $c = 1$ we have with probability at least $1 - \delta$:

$$\left\| \sum_{t=1}^{T} X_t - \mathbb{E} \left[ \sum_{t=1}^{T} X_t \right] \right\| \leq R_T(u) - \epsilon + 1$$

Plugging in our optimistic regret bound we have with probability at least $1 - \delta$:

$$\left\| \sum_{t=1}^{T} X_t - \mathbb{E} \left[ \sum_{t=1}^{T} X_t \right] \right\| \leq O \left( 1 + \| u \| \sqrt{\sum_{t=1}^{T} \| g_t - \mathbb{E} \|^{2} \log(\| u \| T / \delta) + \log(\| u \| T / \delta)} \right)$$

$$\leq O \left( 1 + \sum_{t=1}^{T} \left\| X_t - \mathbb{X} \right\|^{2} \log(\| u \| T / \delta) + \log(\| u \| T / \delta) \right)$$

where we have observed that $\| u \| = c$ and $X_t - \frac{1}{T} \sum_{t=1}^{T} X_t = g_t - \mathbb{E}$.
B  Technical Lemmas

In this section we prove the Lemmas used in the main text. First, the following Lemma shows that we can discretize the space of \( p \)-norms:

**Lemma 8.** Let \( q_0 = 2 \) and \( \frac{1}{q_i} = \frac{1}{\log(d)} = \frac{1}{q_{i-1}} \) for all \( i \in \{1, \ldots, \lfloor \log(d)/2 \rfloor \} \). Let \( p_i \) be defined by \( \frac{1}{p_i} + \frac{1}{q_i} = 1 \). Then for any \( p \in [1, 2] \), there exists \( i \) such that \( p_i \geq p \), \( \|x\|_p \leq \|x\|_p \), and \( \|x\|_{q_i} \leq \|x\|_q \) for all \( x \), where \( \frac{1}{q_i} + \frac{1}{q_i} = 1 \).

**Proof.** First, we claim that \( \|x\|_q \leq d^{1/q'-1/q} \|x\|_{q'} \) for any \( q' \leq q \). To see this, observe that without loss of generality we may set \( \|x\|_q = 1 \), and attempt to maximize \( \|x\|_{q'} \) subject to the constraint \( \|x\|_q = 1 \). Then by application of Lagrange multipliers, we have

\[
q' x_i^{q'-1} = \lambda q x_i^{q'-1}
\]

for all \( i \). From this we see that any non-zero \( x_i \)s must all be equal to each other. Let \( n \) be the number of non-zero \( x_i \)s, and let \( z \) be their common value. Then we wish to maximize \( n z^q \) subject to \( n z^q = 1 \). This yields \( n z^q = n^{1-q'/q} \). This clearly grows with \( n \), which can be at most \( d \). Thus we see \( \|x\|_{q'} \leq d^{1-q'/q} \), which implies \( \|x\|_{q'} \leq d^{1/q'-1/q} \) as desired.

Now we can move on to prove the Lemma. Let \( i \) be the largest value such that \( q_i \leq q \). Then by the recursive definition of \( q_i \), we must have \( \frac{1}{q_i} - \frac{1}{q} \leq \frac{1}{\log(d)} \) so that \( \|x\|_{q_i} \leq d^{1/\log(d)} \|x\|_q = e \|x\|_q \). Further, since \( q_i \leq q, p_i \geq p \) so that \( \|x\|_{p_i} \leq \|x\|_p \). \( \square \)

The following Lemma is used to optimize \( y \) in the proof of Theorem 5.

**Lemma 9.** Suppose \( A, B, C, D, E \) are non-negative constants. Then

\[
\inf_{y \geq 0} \left[ y(A + B \log(e + Cy)) + \frac{D^2}{y} + E \sqrt{\log(e + Cy)} \right] 
\leq 2D \sqrt{[A + B \log(e + CD)]} + E \sqrt{\log(e + CD)}
\]

where \( [X]_1 = \max(X, 1) \).

**Proof.** We just guess a value for \( y \):

\[
y = \frac{D}{\sqrt{[A + B \log(e + CD)]}}
\]

Then the result follows from the fact that \( \log \) is an increasing function and \( y \leq D \). \( \square \)

This final Lemma is allows us to compute the modified hint values \( \tilde{h}_t \) needed to convert our unconstrained optimistic algorithm into a constrained algorithm.

**Lemma 10.** Let \( W \) be a convex domain in a Hilbert space. Let \( x \notin W \). Then for any \( t \in [0, S(x)] \), we have \( \delta \in S(x - t\delta) \) for all \( \delta \in \partial S(x) \). Further, we have \( a\delta \in \partial S(x - S(x)\delta) \) for all \( a \in [0, 1] \).

**Proof.** First, observe that from Proposition 1 and Theorem 4 from [10] we have \( S \) is 1-Lipschitz and \( \partial S(x) = \left\{ \frac{x - \Pi(x)}{\|x - \Pi(x)\|} \right\} \). Therefore \( \delta = \frac{x - \Pi(x)}{\|x - \Pi(x)\|} \) and so \( S(x - t) \leq S(x - t\delta) \leq \|x - t\delta - \Pi(x)\| = S(x) - t \), where the first inequality is from Lipschitzness and the last equality from definition of \( \delta \). Therefore \( \Pi(x) = \Pi(x - t\delta) \), and so the first part of the Lemma follows from Theorem 4 of [10]. For the second part, we observe that \( 0 \in \partial S(x - S(x)\delta) \) because \( x - S(x)\delta \notin W \). Therefore \( a\delta + (1 - a)0 = a\delta \) in \( \partial S(x - \delta) \), proving the second part of the Lemma. \( \square \)

**Lemma 4** is now an immediate corollary of Lemma 10.

C  Proof of Theorem 5

We restate the Theorem below for reference:

**Theorem 5.** Under the same assumptions as Theorem 5 with the exception that \( W \) is now a convex domain in a Hilbert space rather than necessarily the entire space. Then Algorithm 5 guarantees regret

\[
R_T(u) \leq 2B_T(u) \sqrt{\left( (2C + D^2) \log(e + B_T(u)T^c/e) + \sum_{t=1}^T \|h_t - g_t\|^2 \right) + 2DB_T(u) \sqrt{\log(e + B_T(u)T^c/e) + 2A_T(u) + 2e}}
\]

13
where \( |X| \) denotes \( \max(X, 1) \). Further, Algorithm 7 simultaneously guarantees regret

\[
R_T(u) \leq 2 \epsilon + 2A_T(u) + 2B_T(u) \sum_{t=1}^{T} \|g_t\|^2
\]

**Proof.** Define \( \ell_t(w) = \frac{1}{2} (\langle g_t, w \rangle + \|g\|S(w)) \). Then by Lemma 4, \( z_t \in \partial S(\tilde{w}_t) \), so that \( \tilde{g}_t = g_t + \frac{z_t}{\|z_t\|} \in \partial \ell_t(\tilde{w}_t) \). Now we apply the definition of \( \ell_t \), \( \tilde{w}_t \) and Cauchy-Schwarz just as in [10] to obtain \( \frac{1}{2} (\langle g_t, \tilde{w}_t - u \rangle) \leq \ell_t(\tilde{w}_t) - \ell_t(u) \leq \langle \tilde{g}_t, \tilde{w}_t - u \rangle \). Thus we may analyze the regret of the \( \tilde{w}_t \)'s with respect to the \( \tilde{g}_t \)'s. This is encouraging, because the \( \tilde{w}_t \)'s are constructed using Algorithm 1 on hints \( h_t \) and gradients \( \tilde{g}_t \).

Now we continue as in the proof of Theorem 5. We write the regret

\[
\frac{1}{2} R_T(u) \leq \sum_{t=1}^{T} (\tilde{g}_t, \tilde{w}_t - u) = \sum_{t=1}^{T} (\tilde{g}_t, x_t - u) - (\tilde{g}_t, h_t) y_t \leq R_A^T(u) + R_B^T(y) - \sum_{t=1}^{T} (\tilde{g}_t, h_t) y
\]

Now again we can actually immediately see the second part of the Theorem by setting \( y = 0 \) and observing \( \|\tilde{g}_t\| \leq \|g_t\| \) for all \( t \).

For the first part of the Theorem, by exactly the same argument as the proof of Theorem 3 we have

\[
\frac{1}{2} R_T(u) \leq B_T(u) \left[ (2C + D^2) \log(e + B_T(u) T^c / \epsilon) + \sum_{t=1}^{T} \|h_t - \tilde{g}_t\|^2 - \|h_t\|^2 \right]^{1/2} + DB_T(u) \sqrt{\log(e + B_T(u) T^c / \epsilon)} + A_T(u) + \epsilon
\]

Finally, observe that

\[
\|h_t - \tilde{g}_t\| \leq \frac{\|h_t - g_t\|}{2} + \frac{\|h_t - \|g_t\|\|}{2} \leq \frac{\|h_t - g_t\|}{2} + \frac{\|h_t - \|g_t\|\|}{2}
\]

where in the second line we observed that \( \|z_t\| \leq 1 \) (because \( S \) is 1-Lipschitz), and in the last line we observe that \( |||a|| - ||b||| \leq ||a - b|| \) for all \( a, b \) by triangle inequality. Putting this together we have

\[
\frac{1}{2} R_T(u) \leq B_T(u) \left[ (2C + D^2) \log(e + B_T(u) T^c / \epsilon) + \sum_{t=1}^{T} \|h_t - g_t\|^2 \right]^{1/2} + DB_T(u) \sqrt{\log(e + B_T(u) T^c / \epsilon)} + A_T(u) + \epsilon
\]

as desired. \( \square \)

**D Proof of Theorem 6**

We restate the Theorem below for reference:

**Theorem 6.** Under the same assumptions as Theorem 3 Algorithm 7 guarantees regret

\[
R_T(u) \leq B_T(u) \left[ (2C + D^2) \log(e + B_T(u) T^c / \epsilon) + \sum_{t=1}^{T} \|h_{t,i} - g_{t,i}\|^2 - \|h_{t,i}\|^2 \right]^{1/2} + DB_T(u) \sqrt{\log(e + B_T(u) T^c / \epsilon)} + A_T(u) + k \epsilon
\]

for all \( i \), where \( |X| \) denotes \( \max(X, 1) \). Further, Algorithm 7 simultaneously guarantees regret

\[
R_F(u) \leq k \epsilon + R_F^\#(u)
\]
Proof. Just as in the proof of Theorem 3, we write the regret

\[ R_T(u) = \sum_{t=1}^{T} \langle g_t, w_t - u \rangle \]

\[ = \sum_{t=1}^{T} \langle g_t, x_t - u \rangle - \sum_{i=1}^{k} \langle g_t, h_{t,i} \rangle y_{t,i} \]

\[ \leq R_A^A(u) + \sum_{i=1}^{k} R_B^{B_i}(y_i) - \sum_{i=1}^{k} \sum_{t=1}^{T} \langle g_t, h_{t,i} \rangle y_{t,i} \]

Now again we can actually immediately see the second part of the Theorem by setting \( y_i = 0 \) for all \( i \) and observing that

\[ \sum_{i=1}^{k} R_B^{B_i}(0) \leq \epsilon k. \]

Further, choose any particular index \( i \). Then set \( y_j = 0 \) for all \( j \neq i \) and we have

\[ R_T(u) \leq R_A^A(u) + (k - 1)\epsilon + R_B^{B_i}(y_i) - \sum_{t=1}^{T} \langle g_t, h_{t,i} \rangle y_{t,i} \]

Now the rest of the proof is identical to that of Theorem 3. \qed