Discretization gives Uniformity: Surveying a Universal Algorithm for Online Convex Optimization

Miles Pophal MAP10046@NYU.EDU
251 Mercer St, NY, NY

Alexandre Kaiser AMK1004@NYU.EDU
251 Mercer St, NY, NY

Editor: None

Abstract

We motivate and detail the development of a universal algorithm for online convex optimization. Specifically, we develop the theory to explain how one could develop such an algorithm from first principles and prior work, and follow up with full descriptions and derivations of regret bounds for the meta algorithm.

Keywords: List of keywords

1. Introduction

Online convex optimization (OCO) has become an essential problem in machine learning due to its overall application across the field, from optimizing traffic to solving gradient descent problems. The field has produced a diverse array of algorithms to leverage various degrees of prior information about a given learning scenario, thus the challenge for practitioners is how to select the algorithm that makes the most use of their particular problem.

1.1. Bound Knowledge

In all generality, online convex optimization has been solved by many algorithms (Shalev-Shwartz et al. (2007), Tieleman and Hinton (2012), Zeiler (2012)) for an optimal regret per round of $\mathcal{O}(\sqrt{T})$. However, the literature has focused on stricter convex geometries, most notably strong-convexity and exponential-concavity, in an effort to achieve even faster vanishing regret $\mathcal{O}(\log T)$. In the case of strongly-convex and exponentially-concave loss functions, most regret-minimizing algorithms (Hazan et al. (2007),Duchi et al. (2011),Shalev-Shwartz et al. (2007)) require prior knowledge about the specific modulus of the convex geometry. The need for these additional assumptions on the geometry can often limit its use in practice as most loss functions are not necessarily known.

1.2. Adapt-ML-Prod

In the domain of online prediction with expert advice, a learner iteratively makes decisions based on the guidance of multiple experts, adjusting strategies across several rounds to minimize cumulative loss. This setting usually involves assigning weights to expert opinions, where each weight sum reflects the learner's strategy in balancing between differing expert losses. The primary goal is to control cumulative regret relative to the best-performing expert, with standard regret bounds typically at $\sqrt{K \log(T)}$. Recent advancements have focused on refining these bounds by incorporating loss variance, leading to second-order bounds that promise more nuanced regret minimization under certain conditions. Notably, the development of the Adapt-ML-Prod algorithm by Gaillard et al. (2014) introduced a way to account for the existing regret of each expert to minimize the added regret of the algorithm to a near constant $\mathcal{O}(\log \log T)$ term.

2. Motivation

In this section we seek to develop an algorithm by motivating the design choices and explaining some simplifications. Firstly, being a meta-algorithm means we have to make a selection to known algorithms to run and give us a prediction, i.e., functioning as experts in the usual online sense. This forces us to estimate the moduli of strong convexity and $\exp-$concavity (see Subsection 2.1) for each expert we initialize for strongly convex functions (resp. $\exp-$concave functions). To estimate these hyperparameters, we see that discretizing yields desired properties like uniformity in experts, which can be generalized to the idea that "discretization gives uniformity" (in some sense) which we detail in Subsection 2.2. Finally, the meta-algorithm uses second-order bounds which linearized losses motivate with their relation to strong convexity and $\exp-$concavity as explained in Subsection 2.3.

2.1. Estimating Hyperparameters

In theory, the modulus of a strongly convex function f, i.e., the largest α for f to be $\alpha-$strongly convex may take any value in $\mathbb{R}_+$ which cannot be estimated well. If we were to initialize experts for each real (resp. rational) number even in a bounded interval, we would have an uncountable number (resp. countably infinite). However, if we could find a bounded interval to represent the potential space of moduli, then our solution naturally simplifies. Known algorithms like projected stochastic gradient descent (PSGD, see (Mohri, p. 9) for an example) enjoys a regret bound with constant C,

$$R_T \leq \frac{C}{\alpha}(1 + \log T),$$

for f_t $\alpha-$strongly convex. If it happens $\alpha \sim 1/T$, then the regret bound would become

$$R_T \lesssim T(1 + \log T),$$

where $\lesssim$ denotes being less or equal in order of growth. This is impractical because general bounds are usually $\Omega(\log T)$ anyway (Mohri, p. 29) when they come from mirror descent. Hence, with increasing average regret, these bounds become uninformative and we can effectively exclude $\alpha < 1/T$ from our considerations.

On the other end, if a given function f is $\alpha > 1$ strongly convex, then it it also $1-$strongly convex because the latter quadratic still fits between the function and its tangent plane. Alternatively, by definition of strong-convexity,

$$f(w) + \langle \delta f(w), w' - w \rangle + \frac{1}{2}\|w - w'\|^2 < f(w) + \langle \delta f(w), w' - w \rangle + \frac{\alpha}{2}\|w - w'\|^2 \leq f(w')$$

as f is $\alpha-$strongly convex. Hence, we don't consider $\alpha > 1$ for strong convexity either and our new interval to consider is $\alpha \in [1/T, 1]$ to cover all useful cases.

2.2. Discretization and Uniformity

Now that we have the bounded interval $[0, T]$ to work with, we need to get a better estimation on the interior. Since we seek to have this be uniform in experts, we want a condition which gives us uniformity. A general technique for getting some kind of uniform result over large sets is to discretize them because by separating the set into finite components and a small error term, the overall nature can be controlled as finite sets naturally give uniform results (e.g., the maximum of a real function over all points always exists). For example, to get a uniform bound over all $\rho \in (0, r]$ for SVM (see theorem 5.9 in Mohri et al. (2018)), the breakdown looks like

$$(0, r] = (r/2, r] \sqcup \cdots \sqcup (r/2^n, r/2^{n-1}] \sqcup \text{ length } \varepsilon$$

if $n \sim \log_2(2\varepsilon/r)$, and the $\sqcup$ denotes the disjoint union. Another example is the doubling trick, where we get uniformity over the randomness of the horizon T (a random variable in full generality) by discretizing the positive reals with $I_k = [2^k, 2^{k+1}]$. It signals two conditions that discretization needs to meet to yield uniformity

1. Each of the finite parts is easily controllable,
2. The remainder can be made small using other methods.

Condition (1) in the doubling trick is setting the learning rate to the worst-case η_k for each I_k, allowing a regret bound to be uniform from $[0, 2^n]$, where $T \geq n$ is the only problem. For doubling trick we can get an intuition that the remainder term is controllable (condition 2) because of the exponential growth. If we try and consider $\phi(x) = x^\alpha$ (needs to be monotonic for doubling trick to apply) and get uniformity by assuming $T > \phi(n)$, we can use $I_k = [\phi(k), \phi(k+1)]$ and get

$$\sum_{k=1}^{n} \sqrt{\phi(k)\frac{\log N}{2}} \asymp n^{\alpha/2+1}\left(\frac{\log N}{2}\right) \lesssim T^{1/\alpha+1/2}\left(\frac{\log N}{2}\right).$$

This bound is uninformative for $\alpha \leq 2$ because it achieves constant (or worse) average regret, and only in the limit $\alpha \to +\infty$, do we achieve the expected $\sqrt{T}$ bound for regret. Exponentials achieve this bound by default, so it makes sense to split our interval

$$[1/T, 1] \xrightarrow[\text{approx. by}]{} \left\{ \frac{1}{T}, \frac{2}{T}, \ldots, \frac{2^k}{T} \right\},$$

where $k = \lceil \log_2 T \rceil$ to ensure the closest approximation of the interval. Later this will be defined as $\mathcal{P}_{str}, \mathcal{P}_{exp}$ in the algorithm, but now it is important to see each one has size $\mathcal{O}(\log_2 T)$.

2.3. Linearized Losses

Looking at equivalent definitions of strong convexity and $\exp-$concavity (these are for bounded gradients and domains), we can say

$$\lambda - \text{strongly convex } f \iff f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\lambda}{2}\|y - x\|^2, \tag{1}$$

$$\alpha - \exp \text{concave } f \iff f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\beta}{2}\langle \nabla f(x), y - x \rangle^2, \tag{2}$$

where $\beta = \frac{1}{2}\min\left\{\frac{1}{4GD}, \alpha\right\}$ from lemma 3.5 in Zhang et al. (2021). Both of these involve second-order bounds which we know show up in the subgradient expansion term in almost every online convex optimization regret proof (say in Mohri). Based on Subsection 1.2, we would like to get a time-dependent loss function to compare our guess with the expert's which can make use of the second-order terms which appear in strongly convex or exponentially concave functions. A candidate loss function could be

$$\ell_t(x) = \langle \nabla f_t(x_t), x - x_t \rangle$$

which is clearly comparable for $\exp-$concavity, and by Cauchy-Schwarz is reasonable for comparison with strongly convex functions. This is called the linearized loss and the initial intuition can be thought that if f_t has a subgradient, then with the only algorithm we know we should expect $x = x_t - \eta \nabla f_t(x_t)$ for some small $\eta > 0$, meaning

$$\ell_t(x) = \langle \nabla f_t(x_t), -\eta \nabla f_t(x_t) \rangle = -\eta \|f_t(x_t)\| < 0,$$

and forward gradient steps increase the loss. This can't cover all directions but in the half plane $\langle y, \nabla f_t(x_t) \rangle < 0$, $x = x_t + y$ does have negative loss and otherwise incurs nonnegative loss. The linearized loss is useful because if we decompose the regret of the meta algorithm by adding the

term with u_t-the guess of any expert

$$R_T = \underbrace{\sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(u_t)}_{\text{meta part}} - \underbrace{\sum_{t=1}^{T} f_t(u_t) - \min_{x\in\mathcal{X}} \sum_{t=1}^{T} f_t(x)}_{\text{expert regret}}, \tag{3}$$

$$\leq \sum_{t=1}^{T} \langle \nabla f_t(x_t), x_t - u_t\rangle - \frac{\lambda}{2}\|x_t - u_t\|^2 + \left(\sum_{t=1}^{T} f_t(u_t) - \min_{x\in\mathcal{X}} \sum_{t=1}^{T} f_t(x)\right)$$

$$= \sum_{t=1}^{T} (\ell_t(x_t) - \ell_t(u_t)) - \frac{\lambda}{2}\|x_t - u_t\|^2 + \left(\sum_{t=1}^{T} f_t(u_t) - \min_{x\in\mathcal{X}} \sum_{t=1}^{T} f_t(x)\right),$$

using the definition of strong convexity on f_t to arrive at the inequality. Similarly, for $\alpha - \exp -$concavity, one has

$$R_T \leq \sum_{t=1}^{T} (\ell_t(x_t) - \ell_t(u_t)) - \frac{\alpha}{2}|\ell_t(x_t) - \ell_t(u_t)|^2 + \left(\sum_{t=1}^{T} f_t(u_t) - \min_{x\in\mathcal{X}} \sum_{t=1}^{T} f_t(x)\right),$$

showing why having second-order bounds would be nice in both cases. The issue now is these losses are not adapted to $[0,1]$ valued, but if we assume $\|\nabla f_t\|_2 \leq G, \|x\|_2 \leq D$ holding over $\mathcal{X}$, then by Cauchy-Schwarz we know the loss of the $i-th$ expert, ℓ_t^i is bounded like

$$|\langle \nabla f_t(x_t), x_t^i - x_t\rangle| \leq GD \implies \ell_t^i + GD \in [0, 2GD] \implies$$

$$\ell_t^i := \frac{\langle \nabla f_t(x_t), x_t^i - x_t\rangle + GD}{2GD} = \frac{\langle \nabla f_t(x_t), x_t^i - x_t\rangle}{2GD} + \frac{1}{2} \tag{4}$$

can be taken as the definition of the linearized loss for expert i on the $t-$th round.

3. The Universal Algorithm

Now that we have a linearized loss to compare the meta-algorithms guess with each experts guess and a finite set of possible moduli for strong convexity/exponential concavity all that is left is to come up with a strategy for making a guess based on experts. We can let the set of experts be denoted $\mathcal{E}$ and for each expert i, assign it a weight $0 \leq p_t^i \leq 1$ such that they sum to 1 and let

$$x_t = \sum_{i=1}^{|\mathcal{E}|} p_t^i x_t^i,$$

and similarly

$$\ell_t = \sum_{i=1}^{|\mathcal{E}|} p_t^i \ell_t^i = \sum_{i=1}^{|\mathcal{E}|} p_t^i \left(\frac{\langle \nabla f_t(x_t), x_t^i - x_t\rangle + GD}{2GD}\right) = \left\langle \nabla f_t(x_t), \sum_{i=1}^{|\mathcal{E}|} p_t^i x_t^i - \sum_{i=1}^{|\mathcal{E}|} p_t^i x_t \right\rangle + \frac{1}{2}$$

$$= \langle \nabla f_t(x_t), x_t - x_t\rangle + \frac{1}{2} = \frac{1}{2}, \tag{5}$$

using the definition of ℓ_t^i from 4, linearity of $\langle \nabla f_t(x_t), \cdot \rangle$, the definition of x_t, and $\sum_i p_t^i = 1$.

3.1. Algorithmic Details

The important part of the meta algorithm is the ability to keep track of the best expert as it runs. We use Algorithm 2 (Adapt-ML-prod seen in 1.2) from Gaillard et al. (2014) because of the control on second-order terms it features. Specifically, the learning rate and weight updates are as follows

$$
\begin{aligned}
p_t^i &= \frac{\eta_{t-1}^i w_{t-1}^i}{\sum_{i=1}^{|\mathcal{E}|} \eta_{t-1}^i w_{t-1}^i} \\
\eta_{t-1}^i &= \min\left\{\frac{1}{2}, \sqrt{\frac{\log|\mathcal{E}|}{1+\sum_{s=1}^{t-1}(\ell_s - \ell_s^i)^2}}\right\}, \qquad t \geq 1 \\
w_{t-1}^i &= \left[w_{t-2}^i + w_{t-2}^i \eta_{t-2}^i (\ell_{t-1} - \ell_{t-1}^i)\right]^{\frac{\eta_{t-1}^i}{\eta_{t-2}^i}}, \qquad w_0^i = \frac{1}{|\mathcal{E}|}.
\end{aligned}
\tag{6}
$$

In Gaillard et al. (2014), the authors don't seem to give clear intuition to the choices for these learning rates and we treat their optimality as a black box. For the full algorithm, see Algorithm 1 below.

Algorithm 1: The Universal Algorithm

Data: $\mathcal{A}_{str}, \mathcal{A}_{exp}, \mathcal{A}_{con}$ algorithm sets, $\mathcal{P}_{str}, \mathcal{P}_{exp}$ parameter sets
$\mathcal{E} \leftarrow \emptyset$;
for $(A, \lambda) \in \mathcal{A}_{str} \times \mathcal{P}_{str}$ **do**
  Create expert $E(A, \lambda)$;
  Append $\mathcal{E} \leftarrow \mathcal{E} \cup E(A, \lambda)$;
end
for $(A, \alpha) \in \mathcal{A}_{exp} \times \mathcal{P}_{exp}$ **do**
  Create expert $E(A, \alpha)$;
  Append $\mathcal{E} \leftarrow \mathcal{E} \cup E(A, \alpha)$;
end
for $A \in \mathcal{A}_{con}$ **do**
  Create expert $E(A)$;
  Append $\mathcal{E} \leftarrow \mathcal{E} \cup E(A)$;
end
for $t = \{1, \ldots, T\}$ **do**
  for $E^i \in \mathcal{E}$ **do**
    Calculate p_t^i using (6) for E^i ;
    Receive x_t^i from E^i ;
  end
  Calculate $x_t = \sum_{i=1}^{|\mathcal{E}|} p_t^i x_t^i$;
  Observe loss $f_t(\cdot)$; `#full information case`
  for $E \in \mathcal{E}$ **do**
    Send loss data $f_t(\cdot)$ to expert E
  end
end

The algorithm is simple because the weight updates and their corresponding optimality among experts is doing the real work. In fact, the importance of the adapt-ml-prod bounds being second-order cannot be overstated as we will see from the regret bounds in 3.2.

3.2. Bounds and Proofs

Theorem 1 *Let f_t all be $\lambda-$strongly convex with $\lambda \in [1/T, 1]$ and assume $\|\nabla f_t\|_2 \leq G$, $\|x\|_2 \leq D$. Then for $\hat{\lambda} \leq \lambda \leq 2\hat{\lambda}$ with $\hat{\lambda} \in \mathcal{P}_{str}$ has the regret of Algorithm 1 is bounded by*

$$\begin{aligned} R_T &\leq \min_{A \in \mathcal{A}_{str}} R_T(A, \hat{\lambda}) + 2\Gamma G D \left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) + \frac{\Gamma^2 G^2}{2\lambda \log|\mathcal{E}|}, \\ &= \min_{A \in \mathcal{A}_{str}} R_T(A, \hat{\lambda}) + \mathcal{O}\left(\frac{\log\log T}{\lambda}\right). \end{aligned}$$

Here Γ *comes from Corollary 4 in Gaillard et al. (2014) as*

$$\Gamma = 3\log|\mathcal{E}| + \log\left(1 + \frac{|\mathcal{E}|}{2e}(1 + \log(T+1))\right) = \mathcal{O}(\log\log T)$$

is $\mathcal{O}(\log\log T)$ *because* $|\mathcal{E}| = \mathcal{O}(\log T)$ *by construction.*

Proof We start by computing the regret from the linearized losses and the bound from Corollary 4 in Gaillard et al. (2014), i.e,

$$\sum_{t=1}^{T}(\ell_t - \ell_t^i) \leq \frac{\Gamma}{\sqrt{\log|\mathcal{E}|}}\sqrt{1 + \sum_{t=1}^{T}(\ell_t - \ell_t^i)^2} + 2\Gamma \tag{7}$$

By the definition of ℓ_t^i, ℓ_t in (4),(5) we can note

$$\ell_t - \ell_t^i = \frac{1}{2} - \left(\frac{\langle\nabla f_t(x_t), x_t^i - x_t\rangle}{2GD} + \frac{1}{2}\right) = \frac{\langle\nabla f_t(x_t), x_t - x_t^i\rangle}{2GD}$$

where the order x_t, x_t^i is reversed to account for the $-$sign. Multiplying both sides of the inequality with $2GD$ (and bringing inside the square root) allows us to write the bound (7) as

$$\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle \leq \frac{\Gamma}{\sqrt{\log|\mathcal{E}|}}\sqrt{4G^2D^2 + \sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle^2} + 4\Gamma GD, \tag{8}$$

and using the fact $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ (for $a, b \geq 0$, and it can be shown easily by squaring both sides) we get

$$\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle \leq 2\Gamma GD\left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) + \frac{\Gamma}{\sqrt{\log|\mathcal{E}|}}\sqrt{\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle^2}.$$

In order to remove the square root we can treat this term as a multiplication i.e.,

$$\frac{\Gamma}{\sqrt{\log|\mathcal{E}|}}\sqrt{\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle^2} = \sqrt{\frac{\Gamma^2G^2}{\lambda\log|\mathcal{E}|}\cdot\frac{\lambda}{G^2}\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle^2},$$

the term G^2/λ gets multiplied and divided inside so we can make eventual use of strong convexity. This is natural because the only term strong convexity will help us with involves $\|x_t - x_t^i\|^2$, so we need to apply a λ there sometime. Now we can split these terms using arithmetic mean-geometric mean (AM-GM) inequality as

$$\sqrt{\frac{\Gamma^2G^2}{\lambda\log|\mathcal{E}|}\cdot\frac{\lambda}{G^2}\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle^2} \leq \frac{\Gamma^2G^2}{2\lambda\log|\mathcal{E}|} + \frac{\lambda}{2G^2}\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle^2,$$

and moreover, an application of Cauchy-Schwarz on the latter term to say

$$\frac{\lambda}{2G^2}\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle^2 \leq \frac{\lambda}{2G^2}\sum_{t=1}^{T}\|\nabla f_t(x_t)\|^2\|x_t - x_t^i\|^2.$$

Finally, using the assumption $\|\nabla f_t(x_t)\|^2 \leq G^2$, we note

$$\frac{\lambda}{2G^2}\sum_{t=1}^{T}\|\nabla f_t(x_t)\|^2\|x_t - x_t^i\|^2 \leq \sum_{t=1}^{T}\frac{\lambda}{2}\|x_t - x_t^i\|^2,$$

and to summarize,

$$\sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle \leq 2\Gamma GD\left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) + \frac{\Gamma^2G^2}{2\lambda\log|\mathcal{E}|} + \sum_{t=1}^{T}\frac{\lambda}{2}\|x_t - x_t^i\|^2.$$

Now we can use strong convexity and bringing the last term to the LHS to argue

$$\begin{aligned}\sum_{t=1}^{T} f_t(x_t) - f_t(x_t^i) &\leq \sum_{t=1}^{T}\left(\langle\nabla f_t(x_t), x_t - x_t^i\rangle - \sum_{t=1}^{T}\frac{\lambda}{2}\|x_t - x_t^i\|^2\right)\\ &\leq 2\Gamma GD\left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) + \frac{\Gamma^2G^2}{2\lambda\log|\mathcal{E}|},\end{aligned}$$

the desired bound. The critical term lies in the second order control and how it relates to strong convexity, otherwise this would not be possible. The other portion of regret, i.e., the second term from (3) can be bounded as

$$\sum_{t=1}^{T} f_t(u_t) - \min_{x\in\mathcal{X}}\sum_{t=1}^{T} f_t(x) \leq R_T(A, \hat{\lambda})$$

as $\hat{\lambda} \leq \lambda$ and any expert algorithm $E(A, \hat{\lambda})$ makes the correct assumption on the strong convexity allowing it to enjoy the correct bounds. Since the algorithm was independent of this construction, we can take the minimum over all algorithms $A \in \mathcal{A}_{str}$ to get our final result

$$R_T \leq \min_{A\in\mathcal{A}_{str}} R_T(A, \hat{\lambda}) + 2\Gamma GD\left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) + \frac{\Gamma^2G^2}{2\lambda\log|\mathcal{E}|}.$$

■

Now we turn to the case of f_t being $\alpha -$ exp-concave.

Theorem 2 *Let f_t all be $\alpha - \exp -$concave with $\alpha \in [1/T, 1]$ and assume $\|\nabla f_t\|_2 \leq G, \|x\|_2 \leq D$. Then for $\hat{\alpha} \leq \alpha \leq \hat{\alpha}$ with $\hat{\alpha} \in \mathcal{P}_{exp}$ has the regret of Algorithm 1 is bounded by*

$$\begin{aligned} R_T &\leq \min_{A \in \mathcal{A}_{exp}} R_T(A, \hat{\alpha}) + 2\Gamma GD\left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) + \frac{\Gamma^2}{2\beta \log|\mathcal{E}|}, \\ &= \min_{A \in \mathcal{A}_{exp}} R_T(A, \hat{\alpha}) + \mathcal{O}\left(\frac{\log\log T}{\alpha}\right). \end{aligned}$$

Here, $\beta = \frac{1}{2}\min\left\{\frac{1}{4GD}, \alpha\right\}$ as in (2).

Proof We can approach this similarly to Theorem 1. From (2), we can take inspiration and modify the step from Theorem 1 but multiply and divide by $\beta \neq 0$ to get

$$\frac{\Gamma}{\sqrt{\log|\mathcal{E}|}}\sqrt{\sum_{t=1}^{T}\langle \nabla f_t(x_t), x_t - x_t^i\rangle^2} = \sqrt{\frac{\Gamma^2}{\beta\log|\mathcal{E}|}\cdot\beta\sum_{t=1}^{T}\langle \nabla f_t(x_t), x_t - x_t^i\rangle^2}.$$

Next, using the same AM-GM inequality arrive at

$$\sqrt{\frac{\Gamma^2}{\beta\log|\mathcal{E}|}\cdot\beta\sum_{t=1}^{T}\langle \nabla f_t(x_t), x_t - x_t^i\rangle^2} \leq \frac{\Gamma^2}{2\beta\log|\mathcal{E}|} + \sum_{t=1}^{T}\frac{\beta}{2}\langle \nabla f_t(x_t), x_t - x_t^i\rangle^2,$$

allowing us to combine all the terms from the meta-regret (first component of (3)) as

$$\begin{aligned} \sum_{t=1}^{T} f_t(x_t) - f_t(x_t^i) &\leq \sum_{t=1}^{T}\left(\langle \nabla f_t(x_t), x_t - x_t^i\rangle - \sum_{t=1}^{T}\frac{\beta}{2}\langle \nabla f_t(x_t), x_t - x_t^i\rangle^2\right) \\ &\leq 2\Gamma GD\left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) + \frac{\Gamma^2}{2\beta\lambda\log|\mathcal{E}|}, \end{aligned}$$

our desired bound. The second regret split in (3) is handled identically because we did not use the function structure except in $\mathcal{A}_{str}, \mathcal{P}_{str}$ which now become $\mathcal{A}_{exp}, \mathcal{P}_{exp}$. Hence,

$$R_T \leq \min_{A \in \mathcal{A}_{exp}} R_T(A, \hat{\alpha}) + 2\Gamma GD\left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) + \frac{\Gamma^2}{2\log|\mathcal{E}|},$$

but we need to argue why this is $\mathcal{O}\left(\frac{\log\log T}{\alpha}\right)$ because now there is a β dependence. Firstly, $\log|\mathcal{E}|, \Gamma \in \mathcal{O}(\log\log T)$ via construction and Corollary 4 in Gaillard et al. (2014). Hence,

$$\begin{aligned} 2\Gamma GD\left(2 + \frac{1}{\sqrt{\log|\mathcal{E}|}}\right) &= \mathcal{O}\left(\log\log T\right), \\ \frac{\Gamma^2}{2\beta\lambda\log|\mathcal{E}|} &= \mathcal{O}\left(\frac{(\log\log T)^2}{\beta\log\log T}\right) = \mathcal{O}\left(\frac{\log\log T}{\beta}\right), \end{aligned}$$

making the dominant term the second one. Note as β appears in the denominator and we are looking for asymptotic behavior, the only way we can have an increasing bound is if $\beta \to 0$, where we can say $\beta = \alpha/2$ eventually. Hence,

$$\mathcal{O}\left(\frac{\log\log T}{\beta}\right) = \mathcal{O}\left(\frac{\log\log T}{\alpha}\right)$$

because the factor of 2 means nothing for big-O notation. ■

Finally, we need to consider the case where the f_t are only convex. We can't expect as good performance here just by nature of a weaker assumption, and this is formalized in Theorem 3 below.

Theorem 3 *Let f_t all be convex and assume $\|\nabla f_t\|_2 \leq G, \|x\|_2 \leq D$. Then the regret of Algorithm 1 is bounded by*

$$\begin{aligned} R_T &\leq \min_{A\in\mathcal{A}_{con}} R_T(A) + 4\Gamma GD + \frac{\Gamma D}{\sqrt{\log|\mathcal{E}|}}\sqrt{4G^2 + \sum_{t=1}^{T}\|\nabla f_t(x_t)\|^2} \\ &= \min_{A\in\mathcal{A}_{con}} R_T(A) + \mathcal{O}\left(\sqrt{T\log\log T}\right). \end{aligned}$$

Proof If we use the same expansion in (8), we have

$$\begin{aligned} \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(x_t^i) &\leq \sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle \\ &\leq \frac{\Gamma}{\sqrt{\log|\mathcal{E}|}}\sqrt{4G^2D^2 + \sum_{t=1}^{T}\langle\nabla f_t(x_t), x_t - x_t^i\rangle^2} + 4\Gamma GD, \\ \text{Cauchy-Schwarz}\} \qquad &\leq \frac{\Gamma}{\sqrt{\log|\mathcal{E}|}}\sqrt{4G^2D^2 + \sum_{t=1}^{T}\|f_t(x_t)\|^2\|x_t - x_t^i\|^2} + 4\Gamma GD. \end{aligned}$$

Now, we use the crude bound $\|x_t - x_t^i\|^2 \leq D^2$ to get

$$\sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(x_t^i) \leq \frac{\Gamma D}{\sqrt{\log|\mathcal{E}|}}\sqrt{4G^2 + \sum_{t=1}^{T}\|f_t(x_t)\|^2} + 4\Gamma GD, \tag{9}$$

which when combined with the same expert regret term in (3) yields

$$R_T \leq \min_{A\in\mathcal{A}_{con}} R_T(A) + 4\Gamma GD + \frac{\Gamma D}{\sqrt{\log|\mathcal{E}|}}\sqrt{4G^2 + \sum_{t=1}^{T}\|\nabla f_t(x_t)\|^2},$$

and the asymptotics come from

$$4\Gamma GD = \mathcal{O}(\log\log T)$$

$$\frac{\Gamma D}{\sqrt{\log|\mathcal{E}|}}\sqrt{4G^2+\sum_{t=1}^{T}\|\nabla f_t(x_t)\|^2} \leq \underbrace{\frac{\Gamma D}{\sqrt{\log|\mathcal{E}|}}}_{\mathcal{O}(\sqrt{\log\log T})}\underbrace{\sqrt{4G^2+G^2T}}_{\mathcal{O}(\sqrt{T})},$$

and hence, we are done. ■

4. Discussion

4.1. Novelty

The Universal Strategy for online Convex optimization (USC) is among the first algorithms that minimize regret across strongly-convex, exp-concave and general convex functions. Its near-assumption-free requirements for close-to-optimal performance makes it a leading candidate for general use in the field. Nevertheless, the biggest strength of the USC algorithm is undoubtedly that it works for any black-box OCO solver in each of the three geometries. This allows it to be implemented alongside any existing and future methods, promoting more research into those areas.

4.2. Limitations

As alluded to above, although the USC algorithm no longer requires an assumption on the modulus of the geometry, there are other assumptions that need to be made. Firstly, the algorithm is designed for a fixed time horizon T in order to adequately span the required expert spaces set by $\mathcal{P}_{str}$ and $\mathcal{P}_{exp}$. Due to these reliances at initialization, the fixed horizon cannot be avoided using traditional methods such as the doubling trick.

Another potential limitation is the crude bound employed in the proof of Theorem 3, specifically (9). By bounding each of the differences $\|x_t - x_t^i\|$ by the entire diameter of the set, any information on the experts "learning" is cast aside. Specifically, these terms may help counter the norm gradient terms also present, if those were to be large.

In addition to the fixed time horizon, the use of Adapt-ML-Prod for the meta-algorithm requires an assumption that bounds the domain and gradients of the problem. However, it is worth noting that the authors conclude that these assumptions could potentially be circumvented by implementing different meta-algorithms, as long as the meta-algorithm has the second-order bounds.

5. Conclusion

The authors of the USC algorithm have introduced a compeling novel algorithm for OCO, which leverages most of the field's existing methods to solve for a best-of-all-worlds solution. Although

it is constrained by a set of assumptions, most notably by the need for a preset time horizon, its contributions to the field are significant and could lead to a whole new set of expert-agnostic OCO algorithms.

References

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

Pierre Gaillard, Gilles Stoltz, and Tim Van Erven. A second-order bound with excess losses, 2014.

Elad Hazan, Ammit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization, 2007.

M Mohri. Advanced machine learning slides: Online convex optimization. URL https://cims.nyu.edu/~mohri/amls/aml_oco.pdf.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. The MIT Press, 2 edition, 2018.

Shai Shalev-Shwartz, Yoram Singer, and Nathan Srebro. Pegasos: primal estimated sub-gradient solver for svm, 2007.

Tijmen Tieleman and Geoffrey Hinton. "divide the gradient by a running average of its recent magnitude.", 2012.

Matthew D. Zeiler. Adadelta: An adaptive learning rate method, 2012.

Lijun Zhang, Guanghui Wang, Jinfeng Yi, and Tianbao Yang. A simple yet universal strategy for online convex optimization, 2021.