Online Convex Optimization with Continuous Switching Constraint

Anonymous Author(s)
Affiliation
Address
email

Abstract

In many sequential decision making applications, the change of decision would bring an additional cost, such as the wear-and-tear cost associated with changing server status. To control the switching cost, we introduce the problem of online convex optimization with continuous switching constraint, where the goal is to achieve a small regret given a budget on the *overall* switching cost. We first investigate the hardness of the problem, and provide a lower bound of order $\Omega(\sqrt{T})$ when the switching cost budget $S = \Omega(\sqrt{T})$, and $\Omega(\min\{TS, T\})$ when $S = O(\sqrt{T})$, where T is the time horizon. The essential idea is to carefully design an adaptive adversary, who can adjust the loss function according to the cumulative switching cost of the player incurred so far based on the orthogonal technique. We then develop a simple gradient-based algorithm which enjoys the minimax optimal regret bound. Finally, we show that, for strongly convex functions, the regret bound can be improved to $O(\log T)$ for $S = \Omega(\log T)$, and $O(\min\{T/\exp(S) + S, T\})$ for $S = O(\log T)$.

1 Introduction

Online convex optimization (OCO) is a fundamental framework for studying sequential decision making problems (Shalev-Shwartz, 2011). Its protocol can be seen as a game between a player and an adversary: In each round t, firstly, the player selects an action $\mathbf{w}_t$ from a convex set $\mathcal{D} \subseteq \mathbb{R}^d$. After submitting the answer, a loss function $f_t : \mathcal{D} \mapsto \mathbb{R}$ is revealed, and the player suffers a loss $f_t(\mathbf{w}_t)$. The goal is to minimize the regret:

$$R = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w} \in \mathcal{D}} \sum_{t=1}^{T} f_t(\mathbf{w}), \tag{1}$$

which is the difference between the cumulative loss of the player and that of the best action in hindsight.

Over the past decades, the problem of OCO has been extensively studied, yielding various algorithms and theoretical guarantees (Hazan, 2016; Orabona, 2019). However, most of the existing approaches allow the player to switch her action *freely* during the learning process. As a result, these methods become unsuitable for many real-life scenarios, such as the online shortest paths problem (Koolen et al., 2010), and portfolio management (Dekel et al., 2014; Vittori et al., 2020), where the switching of actions brings extra cost, and the budget for the overall switching cost is strictly constrained. To address this problem, recent advances in OCO introduced the switching-constrained problem (Altschuler & Talwar, 2018; Chen et al., 2020), where a hard constraint is imposed to the

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

number of the player's action shifts, i.e.,

$$\sum_{t=2}^{T} \{\mathbf{w}_t \neq \mathbf{w}_{t-1}\} \leq K, \tag{2}$$

and the goal is to minimize regret under a fixed budget K. For this problem, Chen et al. (2020) have shown that, given any K, we could precisely control the the overall switching cost in (2), while achieving a minimax regret bound of order $\Theta(T/\sqrt{K})$.

One limitation of (2) is that it treats different amounts of changes between $\mathbf{w}_{t-1}$ and $\mathbf{w}_t$ equally, since the *binary* function is used as the penalty for action shifts. However, as observed by many practical applications, e.g., thermal management (Zanini et al., 2010), video streaming (Joseph & de Veciana, 2012) and multi-timescale control (Goel et al., 2017), the price paid for large and small action changes are not the same. Specifically, for these scenarios, the switching cost between two consecutive rounds is typically characterized by a ℓ_2-norm function, i.e., $\|\mathbf{w}_t - \mathbf{w}_{t-1}\|$. Motivated by this observation, in this paper, we introduce a novel OCO setting, named OCO with continuous switching constraint (OCO-CSC), where the player needs to choose actions under a hard constraint on the *overall ℓ_2-norm switching cost*, i.e.,

$$\sum_{t=2}^{T} \|\mathbf{w}_t - \mathbf{w}_{t-1}\| \leq S, \tag{3}$$

where S is a budget given by the environment. The main advantage of OCO-CSC is that, equipped with (3), we could have a more delicate control on the overall switching cost compared to the binary constraint in (2).

For the proposed problem, we firstly observe that, an $O(T/\sqrt{S})$ regret bound can be achieved by using the method proposed for the switching-constrained OCO (Chen et al., 2020) under a proper configuration of K. However, this bound is not tight, since there is a large gap from the lower bound established in this paper. Specifically, we provide a lower bound of order $\Omega(\sqrt{T})$ when $S = \Omega(\sqrt{T})$, and $\Omega(\min\{\frac{T}{S}, T\})$ when $S = O(\sqrt{T})$. Our basic framework for constructing the lower bound follows the classical linear game (Abernethy et al., 2008), while we adopt a novel continuous-constraint-related dynamic policy for the adversary, which allows it to adaptively change the loss function according to the player's cumulative switching costs. Furthermore, we prove that the classical online gradient descent (OGD) with an appropriately chosen step size is able to obtain the matching upper bound. These results demonstrate that there is a *phase transition* phenomenon between large and small switching budget regimes, which is in sharp contrast to the switching-constrained setting, where the minimax bound always decreases with $\Theta(1/\sqrt{K})$. Finally, we propose a variant of OGD for λ-strongly convex functions, which can achieve an $O(\log T)$ regret bound when $S = \Omega(\log T)$, and an $O(T/\exp(S) + S)$ regret bound when $S = O(\log T)$.

2 Related Work

In this section, we briefly review related work on online convex optimization.

2.1 Classical OCO

The framework of OCO is established by the seminal work of Zinkevich (2003). For general convex functions, Zinkevich (2003) shows that online gradient descent (OGD) with step size on the order of $O(1/\sqrt{t})$ enjoys an $O(\sqrt{T})$ regret bound. For λ-strongly convex functions, Hazan et al. (2007) prove that OGD with step size of order $O(1/[\lambda t])$ achieves an $O(\log T)$ regret bound. Both bounds have been proved to be minimax optimal (Abernethy et al., 2008). For exponentially concave functions, the state-of-the-art algorithm is online Newton step (ONS) , (Hazan et al., 2007), which enjoys an $O(d \log T)$ regret bound, where d is the dimensionality.

2.2 Switching-constrained OCO

One related line of research is the switching-constrained setting, where the player is only allowed to change her action no more than K times. This setting has been studied in various online learning

scenarios, such as prediction with expert advice (Altschuler & Talwar, 2018) and bandits problems (Simchi-Levi & Xu, 2019; Dong et al., 2020; Ruan et al., 2020). In this paper, we focus on online convex optimization. Jaghargh et al. (2019) firstly consider this problem, and develop a novel online algorithm based on the Poison Process, which can achieve an expected regret of order $O(T^{3/2}/\mathrm{E}[K])$ for any given expected switching budget $\mathrm{E}[K]$. Therefore, the regret will become sublinear for $\mathrm{E}[K] = o(\sqrt{T})$. Later, Chen et al. (2020) propose a variant of the classical OGD based on the mini-batch approach, which enjoys an $O(T/\sqrt{K})$ regret bound for any given budget K. They also prove that this result is minimax optimal by establishing a matching $\Omega(T/\sqrt{K})$ lower bound. We note that, when the action set is bounded (i.e., $\max_{\mathbf{w}_1,\mathbf{w}_2\in\mathcal{D}} \|\mathbf{w}_1 - \mathbf{w}_2\| \leq D$), since

$$\sum_{t=2}^{T} \|\mathbf{w}_t - \mathbf{w}_{t-1}\| \leq DK,$$

we could set $K = \lfloor S/D \rfloor$ to satisfy (3) and immediately obtain an $O(T/\sqrt{S})$ regret for OCO-CSC, but there is still a large gap from the lower bound we provide in this paper. Very recently, a concurrent work (Sherman & Koren, 2021) considered switching-constrained OCO under oblivious adversary setting, and derived $\Theta(T/S)$ bounds. However, we note that: (i) lower bound for the switching-constraint OCO does not translate to a lower bound for our continuous switching cost setting; and (ii) their upper bound for oblivious only holds in expectation, and has an undesirable $\sqrt{d}$ factor.

2.3 OCO with Ramp Constraints

Another related setting is OCO with ramp constraints, which is studied by Badiei et al. (2015). In this setting, at each round, the player must choose an action satisfying the following inequality:

$$|w_{t,i} - w_{t-1,i}| \leq X_i, \tag{4}$$

where $w_{t,i}$ denotes the i-th dimension of $\mathbf{w}_t$, and X_i is a constant factor. The constraint in (4) limits the player's action switching in a *per-round* and *per-dimension* level. This is very different from the constraint we proposed in (3), which mainly focus on the *long-term* and *overall* switching cost. Moreover, we note that, Badiei et al. (2015) assume the player could get access to a sequence of *future* loss functions before choosing $\mathbf{w}_t$, while in this paper we follow the classical OCO framework in which the player can only make use of the historical data.

2.4 OCO with Long-term Constraints

Our proposed problem is also related to OCO with long-term constraints (Mahdavi et al., 2012; Jenatton et al., 2016; Yu et al., 2017), where the action set is written as m convex constraints, i.e.,

$$\mathcal{D} = \{\mathbf{w} \in \mathbb{R}^d : g_i(\mathbf{w}) \leq 0, i \in [m]\}, \tag{5}$$

and we only require these constraints to be satisfied in the long term, i.e., $\sum_{t=1}^{T} g_i(\mathbf{w}_t) \leq 0, i \in [m]$. The goal is to minimize regret while keeping $\sum_{t=1}^{T} g_i(\mathbf{w}_t)$ small. We note that, in this setting, the action set is *expressed* by the constraint, which is in contrast to OCO-CSC, where the constraint and the decision set $\mathcal{D}$ are independent. Moreover, the constraint in OCO-CSC is time-variant and decided by the historical decisions, while the constraint in (5) is static (or stochastic, considered by Yu et al., 2017). Recently, several work start to investigate OCO with long-term and time-variant constraints, but this task is proved to be impossible in general (Mannor et al., 2009). Therefore, existing studies have to consider more restricted settings, such as weaker definitions of regret (Neely & Yu, 2017; Liakopoulos et al., 2019; Yi et al., 2020; Valls et al., 2020).

2.5 Smoothed OCO

The problem of smoothed OCO is originally proposed in the dynamic right-sizing for power-proportional data centers (Lin et al., 2012b), and has received great research interests during the past decade (Lin et al., 2012a; Bansal et al., 2015; Antoniadis & Schewior, 2017; Chen et al., 2018; Goel et al., 2019). In smoothed OCO, at each round, the learner will incur a *hitting* cost $f_t(\cdot)$ as well as a *switching* cost $\|\mathbf{w}_t - \mathbf{w}_{t-1}\|$, and the goal is to minimize dynamic regret (Zinkevich, 2003)

or competitive ratio (Borodin & El-Yaniv, 2005) with respect to $f_t(\mathbf{w}_t) + \|\mathbf{w}_t - \mathbf{w}_{t-1}\|$. This setting is significantly different from OCO-CSC, where the goal is to minimize regret with respect to $f_t(\cdot)$, and the overall switching cost is limited by a given budget. Additionally, we note that, similar to Badiei et al. (2015), studies for the smoothed OCO typically assume the player could see $f_t(\cdot)$ or sometimes a window of future loss functions (Chen et al., 2015, 2016; Li et al., 2018) before choosing $\mathbf{w}_t$. By contrast, in OCO-CSC the player can not obtain these additional information.

3 Main Results

In this section, we present the algorithms and theoretical guarantees for OCO-CSC. Before proceeding to the details, following previous work, we introduce some standard definitions (Boyd & Vandenberghe, 2004) and assumptions (Abernethy et al., 2008).

Defination 1 *A function $f : \mathcal{D} \mapsto \mathbb{R}$ is convex if* $\forall \mathbf{w}_1, \mathbf{w}_2 \in \mathcal{D}$,

$$f(\mathbf{w}_1) \geq f(\mathbf{w}_2) + \nabla f(\mathbf{w}_2)^\top (\mathbf{w}_1 - \mathbf{w}_2). \tag{6}$$

Defination 2 *A function $f : \mathcal{D} \mapsto \mathbb{R}$ is λ-strongly convex if* $\forall \mathbf{w}_1, \mathbf{w}_2 \in \mathcal{D}$,

$$f(\mathbf{w}_1) \geq f(\mathbf{w}_2) + \nabla f(\mathbf{w}_2)^\top (\mathbf{w}_1 - \mathbf{w}_2) + \frac{\lambda}{2}\|\mathbf{w}_1 - \mathbf{w}_2\|^2. \tag{7}$$

Assumption 1 *$\mathcal{D}$ is a d-dimensional ball of radius $\frac{D}{2}$, i.e., $\mathcal{D} = \{\mathbf{w} | \mathbf{w} \in \mathbb{R}^d, \|\mathbf{w}\| \leq \frac{D}{2}\}$.*

Assumption 2 *The gradients of all the online functions are bounded by G, i.e.,*

$$\max_{\mathbf{w} \in \mathcal{D}} \|\nabla f_t(\mathbf{w})\| \leq G, \ \forall t \in [T]. \tag{8}$$

3.1 Lower Bound for Convex Functions

Algorithm 1 Adversary's Policy

```
τ = 1, l_τ = 1
Observe the player's action w_1
Choose m_1 such that m_1^⊤ w_1 = 0
for t = 2 to T do
    Observe the player's action w_t
    if Σ_{j=l_τ+1}^{t} ‖w_j − w_{j−1}‖ ≤ c/S then
        Choose m_t = m_{t−1}
    else
        Choose m_t such that m_t^⊤ w_t ≥ 0 and m_t^⊤ (Σ_{j=1}^{t−1} m_j) ≥ 0.
        Set τ = τ + 1, l_τ = t
    end if
end for
```

We first describe the adversary's policy for obtaining the lower bound. Following previous work Abernethy et al. (2008), our proposed policy is based on the *linear game*, i.e., in each round, the adversary chooses from a set of bounded linear functions:

$$F = \{f(\cdot) : \mathcal{D} \mapsto \mathbb{R} | f(\mathbf{w}) = \mathbf{m}^\top \mathbf{w}, \|\mathbf{m}\| = G\},$$

which is a subset of convex functions satisfying Assumptions 1 and 2. For this setting, the regret can be written as

$$\sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w} \in \mathcal{D}} \sum_{t=1}^{T} f_t(\mathbf{w}) = \sum_{t=1}^{T} \mathbf{m}_t^\top \mathbf{w}_t - \min_{\mathbf{w} \in \mathcal{D}} \left(\sum_{t=1}^{T} \mathbf{m}_t \right)^\top \mathbf{w} = \sum_{t=1}^{T} \mathbf{m}_t^\top \mathbf{w}_t + \frac{D}{2} \left\| \sum_{t=1}^{T} \mathbf{m}_t \right\|, \tag{9}$$

where the third equality is because the minimum is only obtained when

$$\mathbf{w} = -D\frac{\sum_{t=1}^{T}\mathbf{m}_t}{2\|\sum_{t=1}^{T}\mathbf{m}_t\|}.$$

According to (9), to get a tight lower bound for R, we have to make both $\sum_{t=1}^{T}\mathbf{m}_t^\top\mathbf{w}_t$ and $\|\sum_{t=1}^{T}\mathbf{m}_t\|$ as large as possible. One classical way to achieve this goal is through the *orthogonal technique* (Abernethy et al., 2008; Chen et al., 2020), that is, in round t, the adversary chooses $\mathbf{m}_t$ such that $\mathbf{m}_t^\top\mathbf{w}_t \geq 0$ and $\mathbf{m}_t^\top(\sum_{i=1}^{t-1}\mathbf{m}_i) \geq 0$. Note that such a $\mathbf{m}_t$ can always be found for $d \geq 2$. For this technique, it can be easily shown that $\sum_{t=1}^{T}\mathbf{m}_t^\top\mathbf{w}_t \geq 0$, while $\|\sum_{t=1}^{T}\mathbf{m}_t\| \geq G\sqrt{T}$, which implies an $\Omega(DG\sqrt{T})$ lower bound.

The above policy does not take the constraint on the player's action shifts into account. In the following, we show that, by designing a more adaptive adversary which automatically adjusts its action based on the player's historical switching costs, we can obtain a tighter lower bound when S is small. The details is summarized in Algorithm 1. Specifically, in the first round, after observing the player's action $\mathbf{w}_1$, the adversary just simply chooses $f_1(\mathbf{w}) = \mathbf{m}_1^\top\mathbf{w}$ such that $\mathbf{m}_1^\top\mathbf{w}_1 = 0$ (Step 3). For round $t \geq 2$, the adversary divides the time horizon into several epochs. Let the number of epochs be N. For each round t in epoch $\tau \in [N]$, after obtaining $\mathbf{w}_t$, the adversary checks if the cumulative switching cost of the player inside epoch τ exceeds a threshold (Step 6). To be more specific, the adversary will check if

$$\sum_{j=l_\tau+1}^{t}\|\mathbf{w}_j - \mathbf{w}_{j-1}\| \leq \frac{c}{S},$$

where l_τ is the start point of epoch τ, $\ell_1 = 1$, and $c > 0$ is a constant factor. If the inequality holds, then the adversary will keep the action unchanged (Step 7); otherwise, the adversary will find a new $\mathbf{m}_t$ based on the orthogonal technique, i.e., find $\mathbf{m}_t$ such that $\mathbf{m}_t^\top\mathbf{w}_t \geq 0$ and $\mathbf{m}_t^\top M_{t-1} \geq 0$, where $M_{t-1} = \sum_{j=1}^{t-1}\mathbf{m}_j$, and then start a new epoch (Steps 9-10).

The essential idea behind the above policy is that the adversary adaptively divides T iterations into N epochs, such that for each epoch $\tau \in [N]$, the cumulative switching cost inside of τ is upper bounded by

$$\sum_{j=l_\tau+1}^{l_{\tau+1}-1}\|\mathbf{w}_j - \mathbf{w}_{j-1}\| \leq \frac{c}{S},$$

and for each epoch $\tau \in [N-1]$,

$$\sum_{j=l_\tau+1}^{l_{\tau+1}}\|\mathbf{w}_j - \mathbf{w}_{j-1}\| > \frac{c}{S}.$$

The above two inequalities help us obtain novel lower bounds for the two terms at the R.H.S. of (9) respectively which depend on S. Specifically, we prove the following two lemmas.

Lemma 1 *We have*

$$\sum_{t=1}^{T}\mathbf{m}_t^\top\mathbf{w}_t \geq -\frac{cGT}{S}.$$

Lemma 2 *We have*

$$\left\|\sum_{t=1}^{T}\mathbf{m}_t\right\| \geq G\frac{T\sqrt{c}}{\sqrt{S^2+c}}.$$

By appropriately tuning the parameter c, we finally prove the following lower bound.

Theorem 1 *For any online algorithm, under any given switching cost budget S, Algorithm 1 can generate a series of loss functions $f_1(\cdot), \ldots, f_T(\cdot)$ satisfying Assumptions 1 and 2, such that*

$$R \geq \begin{cases} 0.5DG\sqrt{T}, & S \in [D\sqrt{T}, DT] \\ 0.05DG\frac{DT}{S}, & S \in [D, D\sqrt{T}) \\ 0.05DGT, & S \in [0, D). \end{cases}$$

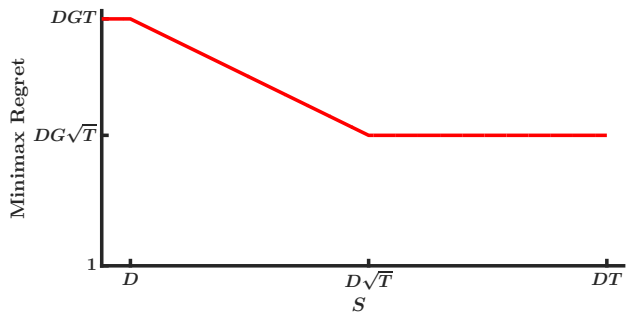

Figure 1: Minimax regret of OCO-CSC. Axies are plotted in log-log scale.

Remark 1 The above theorem implies that, when $S \leq D$, the lower bound for OCO-CSC is linear with respect to T; When $S \in [D, D\sqrt{T})$, it's possible to achieve sublinear results, and the lower bound decreases with T/S; for sufficiently large S, i.e., when $S = \Omega(D\sqrt{T})$, the lower bound is $\Omega(DG\sqrt{T})$, which matches the lower bound for the general OCO problem (Abernethy et al., 2008). Note that in this case the lower bound will not further improve as S increases, which is very different from the switching-constrained setting, where the lower bound is $\Omega(T/\sqrt{K})$, which means that increasing the budget K is always beneficial. Finally, we note that, since

$$\frac{1}{D}\sum_{t=2}^{T}\|\mathbf{w}_t - \mathbf{w}_{t-1}\| \leq \sum_{t=2}^{T}\{\mathbf{w}_t \neq \mathbf{w}_{t-1}\},$$

lower bound for the binary switching cost setting (Altschuler & Talwar, 2018; Chen et al., 2020) does not translate to lower bound for our continuous switching cost setting.

Remark 2 We summarize our main ideas for proving Theorem 1 as follows. For the proposed adversary, we firstly show that (Lemma 1, Eq. (17)), inside of each batch, there exists a negative error term caused by the possible *non-orthogonality* between the player's decisions and the adversary's choice. To maximize this error term and further the lower bound, the adversary should change its choice as many times as possible (thus require a *small* threshold); On the other hand, Lemma 2 (Eq. (21)) indicates that, the lower bound is tighter when the number of batches is small (thus require a *large* threshold). Based on the two lemmas, in the proof of Theorem 1, we show that the final lower bound is a function of the threshold. Very luckily, we find that the *optimal* lower bound can be derived by choosing a threshold that maximizes the function.

3.2 Upper Bounds

In this section, we provide the algorithm for obtaining the upper bound. Before introducing our method, we note that, as mentioned in Section 2.2, the mini-batch OGD algorithm proposed by Chen et al. (2020) enjoys an $O(T/\sqrt{S})$ regret bound for OCO-CSC, which is suboptimal based on the lower bound we constructed at the last section. In the following, we show that, perhaps a bit surprisingly, the classical online gradient descent with an appropriately chosen step size is sufficient for obtaining the matching upper bound. Specifically, in round t, we update $\mathbf{w}_t$ by

$$\mathbf{w}_{t+1} = \Pi_{\mathcal{D}}\left[\mathbf{w}_t - \eta\nabla f_t(\mathbf{w}_t)\right], \tag{10}$$

where $\Pi_{\mathcal{D}}[\mathbf{p}]$ denotes projecting $\mathbf{p}$ into $\mathcal{D}$, i.e.,

$$\Pi_{\mathcal{D}}[\mathbf{p}] = \operatorname*{argmin}_{\mathbf{w}\in\mathcal{D}}(\mathbf{w}-\mathbf{p})^\top(\mathbf{w}-\mathbf{p}).$$

For this algorithm, we prove the following theoretical guarantee.

Theorem 2 *Suppose Assumptions 1 and 2 hold, and all loss functions are convex. Then, under any given switching cost budget S, OGD with step size*

$$\eta = \begin{cases} \frac{D}{G\sqrt{T}}, & S \in [D\sqrt{T}, DT] \\ \frac{S}{GT}, & S \in [0, D\sqrt{T}) \end{cases} \tag{11}$$

satisfies (3)*, and achieves the following regret:*

$$R \leq \begin{cases} DG\sqrt{T}, & S \in [D\sqrt{T}, DT] \\ DG\frac{DT}{S}, & S \in [D, D\sqrt{T}) \\ DGT, & S \in [0, D). \end{cases}$$

Remark 3 Theorems 1 and 2 show that our proposed algorithm enjoys an $O(GD\sqrt{T})$ minimax regret bound for $S = \Omega(D\sqrt{T})$, and $O(DG\min\{DT/S, T\})$ regret bound for $S = O(D\sqrt{T})$. We illustrate the relationship between the minimax regret and S in Figure 1.

Although the analysis above implies that the theoretical guarantee for OCO-CSC is unimproveable in general, in the following, we show that tighter bounds is still achievable when the loss functions are strongly convex. Specifically, when there are no constraints on the switching cost, the state-of-the-art algorithm is OGD with a time variant step size $\eta_t = 1/[\lambda t]$, which enjoys an $O(\log T)$ regret bound. For the OCO-CSC setting, in order to control the overall switching cost, we propose to add a tuning parameter at the denominator of the step size. To be more specific, in round t, we update $\mathbf{w}_t$ by

$$\mathbf{w}_{t+1} = \Pi_{\mathcal{D}}\left[\mathbf{w}_t - \eta_t \nabla f_t(\mathbf{w}_t)\right], \tag{12}$$

where $\eta_t = \frac{1}{\lambda(t+c)}$, and $c > 0$ is a constant factor. By configuring c properly, we can obtain the following regret bound.

Theorem 3 *Suppose Assumptions 1 and 2 hold, and all loss functions are λ-strongly convex. Then, under any given switching cost budget S, the algorithm in* (12) *with*

$$c = \begin{cases} 0, & S \in [\frac{2G}{\lambda}\log(T+1), DT] \\ \frac{T}{\exp(\frac{\lambda}{2G}S)-1} - 1, & S \in [0, \frac{2G}{\lambda}\log(T+1)) \end{cases} \tag{13}$$

satisfies (3)*, and achieves $R \leq \lambda D^2 + \frac{2G^2}{\lambda}\log(T+1)$. for $S \in [\frac{2G}{\lambda}\log(T+1), DT]$, and*

$$R \leq \min\left\{\frac{\lambda T D^2}{\exp(\frac{\lambda}{2G}S) - 1} + GS, DGT\right\}$$

for $S \in [0, \frac{2G}{\lambda}\log(T+1))$.

Remark 4 Theorem 3 implies that, when $S \geq \frac{2G}{\lambda}\log(T+1)$, the proposed algorithm enjoys an $O(\log T)$ optimal regret bound; for $S \leq \frac{2G}{\lambda}\log(T+1)$, the proposed algorithm achieves an $O(T/\exp(S) + S)$ regret bound. To obtain a sublinear regret bound, consider $S = \frac{2G}{\lambda}\log(T^\alpha + 1)$. In this case, we have

$$R \leq \lambda D^2 T^{1-\alpha} + \frac{2G^2}{\lambda}\log(T^\alpha + 1),$$

which is sublinar for $\alpha \in (0, 1]$.

4 Theoretical Analysis

In this section, we present the proofs for the main conclusions.

4.1 Proof of Lemma 1

For any epoch $\tau \in [N]$ of length 1, we have

$$\mathbf{m}_{l_\tau}^\top \mathbf{w}_{l_\tau} \geq 0. \tag{14}$$

For any epoch $\tau \in [N]$ whose length is greater than 1, we have $\forall t \in [l_\tau, l_{\tau+1} - 1]$,

$$\|\mathbf{w}_t - \mathbf{w}_{l_\tau}\| \leq \sum_{j=l_\tau+1}^{t} \|\mathbf{w}_j - \mathbf{w}_{j-1}\| \leq \frac{c}{S}, \tag{15}$$

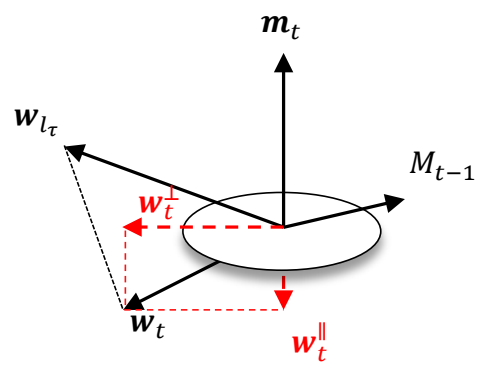

Figure 2: Decomposition of $\mathbf{w}_t$ when $d = 3$.

where the first inequality is based on the triangle inequality, and the second inequality is guaranteed by Step 6 of the adversary's policy. Next, we decompose $\mathbf{w}_t$ into two terms: $\mathbf{w}_t = \mathbf{w}_t^{\parallel} + \mathbf{w}_t^{\perp}$, where $\mathbf{w}_t^{\parallel}$ is the component parallel to $\mathbf{m}_t$, and $\mathbf{w}_t^{\perp}$ the component parallel to the $(d-1)$-normal-hyperplane of $\mathbf{m}_t$. We illustrate the decomposition for $d = 3$ in Figure 2. Based on the decomposition, we have

$$\mathbf{m}_t^\top \mathbf{w}_t = \mathbf{m}_t^\top (\mathbf{w}_t^{\perp} + \mathbf{w}_t^{\parallel}) = \mathbf{m}_t^\top \mathbf{w}_t^{\parallel} = -G\|\mathbf{w}_t^{\parallel}\| \geq -G\|\mathbf{w}_t - \mathbf{w}_{l_\tau}\| \geq -\frac{cG}{S}, \tag{16}$$

where the first inequality is based on the triangle inequality, and the fact that $\mathbf{w}_{l_\tau}$ is always above the $(d-1)$-normal-hyperplane. The second inequality is derived from (15). Combining (14) and (16), we know that for $t \in [T]$, $\mathbf{m}_t^\top \mathbf{w}_t \geq -\frac{cG}{S}$, and thus

$$\sum_{t=1}^{T} \mathbf{m}_t^\top \mathbf{w}_t \geq -\frac{cGT}{S}. \tag{17}$$

4.2 Proof of Lemma 2

Based on Step 6 of the adversary's policy, we know that, for epoch $\tau \in [N-1]$,

$$\sum_{j=l_\tau+1}^{l_{\tau+1}} \|\mathbf{w}_j - \mathbf{w}_{j-1}\| > \frac{c}{S}, \tag{18}$$

since otherwise the adversary will not start a new epoch at $\ell_{\tau+1}$ (note that the cumulative switching cost at the last epoch does not have this lower bound). Thus, the total switching cost in the first $[N-1]$ epochs is lower bounded by

$$\sum_{\tau=1}^{N-1} \sum_{j=l_\tau+1}^{l_{\tau+1}} \|\mathbf{w}_j - \mathbf{w}_{j-1}\| > (N-1)\frac{c}{S}.$$

On the other hand, since the overall budget is S, we know

$$\sum_{\tau=1}^{N-1} \sum_{j=l_\tau+1}^{l_{\tau+1}} \|\mathbf{w}_j - \mathbf{w}_{j-1}\| \leq S.$$

Thus

$$N \leq \frac{S^2}{c} + 1. \tag{19}$$

Let L_τ be the length of epoch $\tau \in [N]$. Based on the Step 9, we know that for each epoch τ, $\mathbf{m}_{l_\tau} M_{l_\tau - 1} \geq 0$. Thus, we have

$$\left\|\sum_{t=1}^{T} \mathbf{m}_t\right\|^2 = \left\|L_N \mathbf{m}_{l_N} + \sum_{\tau=1}^{N-1} L_\tau \mathbf{m}_{l_\tau}\right\|^2 \geq \|L_N \mathbf{m}_{l_N}\|^2 + \left\|\sum_{i=1}^{N-1} L_\tau \mathbf{m}_{l_\tau}\right\|^2 \geq G^2 \sum_{\tau=1}^{N} L_i^2. \tag{20}$$

Thus

$$\left\|\sum_{t=1}^{T}\mathbf{m}_t\right\| \geq G\sqrt{\sum_{\tau=1}^{N}L_\tau^2} \geq G\frac{T}{\sqrt{N}} \geq G\frac{T\sqrt{c}}{\sqrt{S^2+c}}, \tag{21}$$

where the first inequality is derived from (20), the second inequality is based on Cauchy-Schwarz inequality, and the final inequality is derived from (19).

4.3 Proof of Theorem 1

When $S \geq D\sqrt{T}$, the lower bound can be directly obtained by using the minimax linear game provided by (Abernethy et al., 2008). When $S \in [D, D\sqrt{T})$, by the definition of regret, we have

$$R = \underbrace{\sum_{t=1}^{T}\mathbf{m}_t^\top\mathbf{w}_t}_{a_1} \underbrace{- \min_{\mathbf{w}\in\mathcal{D}}\left(\sum_{t=1}^{T}\mathbf{m}_t\right)^\top\mathbf{w}}_{a_2}.$$

Based on Lemmas 1 and 2, we get

$$R = a_1 + a_2 \geq G\frac{0.5DT\sqrt{c}}{\sqrt{S^2+c}} - \frac{cGT}{S}. \tag{22}$$

Let $c = c'D^2$, and we have

$$R \geq G\frac{0.5DT\sqrt{c'D^2}}{\sqrt{S^2+c'D^2}} - \frac{c'D^2GT}{S} = GD^2T\left(\frac{0.5\sqrt{c'}}{\sqrt{S^2+c'D^2}} - \frac{c'}{S}\right) \geq GD^2\frac{T}{S}\left(\frac{0.5\sqrt{c'}}{\sqrt{1+c'}} - c'\right).$$

where the second inequality is due to $D \leq S$. Note that the R.H.S. of the above inequality is a function of c'. To maximize the lower bound, we should solve the following convex problem:

$$\operatorname*{argmax}_{x>0}\frac{0.5\sqrt{x}}{\sqrt{1+x}} - x,$$

which is equivalent to finding the solution of the following equation:

$$16x^4 + 32x^3 + 49x^2 + 15x - 1 = 0.$$

it can be easily shown that the optimal solution $x_* \approx 0.056$. Thus, by setting $c' = 0.056$, we get

$$R \geq 0.05GD^2\frac{T}{S}. \tag{23}$$

For $S \in (0, D]$, based on (22) and setting $c = 0.056S^2$, we have

$$R \geq GT\left(\frac{0.5D\sqrt{0.056S^2}}{\sqrt{S^2+0.056S^2}} - \frac{0.056S^2}{S}\right) \geq GDT\left(\frac{0.028}{\sqrt{1.056}} - 0.056\right) \geq 0.05GDT.$$

where the second inequality is because $S \leq D$.

5 Conclusion and Future Work

In this paper, we propose a variant of the classical OCO problem, named OCO with continuous switching constraint, where the player suffers a ℓ_2-norm switching cost for each action shift, and the overall switching cost is constrained by a given budget S. We first propose an adaptive mini-batch policy for the adversary, based on which we prove that the lower bound for this problem is $\Omega(\sqrt{T})$ when $S = \Omega(\sqrt{T})$, and $\Omega(\min\{\frac{T}{S}, T\})$ when $S = O(\sqrt{T})$. Next, we demonstrate that OGD with a proper configuration of the step size achieves the minimax optimal regret bound. Finally, for λ-strongly convex functions, we develop a variant of OGD, which has a tunable parameter at the denominator, and we show that it enjoys an $O(\log T)$ regret bound when $S = \Omega(\log T)$, and an $O(T/\exp(S) + S)$ regret bound when $S = O(\log T)$.

In the future, we would like to investigate how to extend our setting to other online learning scenarios, such as bandit convex optimization (Flaxman et al., 2005), and OCO in changing environments (Hazan & Seshadhri, 2009). Moreover, it is also an interesting question to study the switching constraint problem under other distance metrics such as the Bregman divergence.

References

Abernethy, J., Bartlett, P. L., Rakhlin, A., and Tewari, A. Optimal stragies and minimax lower bounds for online convex games. In *Proceedings of the 21st Annual Conference on Learning Theory*, pp. 415–423, 2008.

Altschuler, J. and Talwar, K. Online learning over a finite action set with limited switching. In *Proceedings of the 31st Annual Conference on Learning Theory*, pp. 1569–1573, 2018.

Antoniadis, A. and Schewior, K. A tight lower bound for online convex optimization with switching costs. In *International Workshop on Approximation and Online Algorithms*, pp. 164–175, 2017.

Badiei, M., Li, N., and Wierman, A. Online convex optimization with ramp constraints. In *2015 54th IEEE Conference on Decision and Control*, pp. 6730–6736, 2015.

Bansal, N., Gupta, A., Krishnaswamy, R., Pruhs, K., Schewior, K., and Stein, C. A 2-competitive algorithm for online convex optimization with switching costs. In *Algorithms and Techniques for Approximation, Randomization, and Combinatorial Optimization*, 2015.

Borodin, A. and El-Yaniv, R. *Online computation and competitive analysis*. cambridge university press, 2005.

Boyd, S. and Vandenberghe, L. *Convex Optimization*. Cambridge University Press, 2004.

Chen, L., Yu, Q., Lawrence, H., and Karbasi, A. Minimax regret of switching-constrained online convex optimization: No phase transition. In *Advances in Neural Information Processing Systems 33*, 2020.

Chen, N., Agarwal, A., Wierman, A., Barman, S., and Andrew, L. L. Online convex optimization using predictions. In *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pp. 191–204, 2015.

Chen, N., Comden, J., Liu, Z., Gandhi, A., and Wierman, A. Using predictions in online optimization: Looking forward with an eye on the past. *ACM SIGMETRICS Performance Evaluation Review*, 44(1):193–206, 2016.

Chen, N., Goel, G., and Wierman, A. Smoothed online convex optimization in high dimensions via online balanced descent. In *Proceedings of the 31st Annual Conference on Learning Theory*, pp. 1574–1594, 2018.

Dekel, O., Ding, J., Koren, T., and Peres, Y. Bandits with switching costs: T 2/3 regret. In *Proceedings of the 46th annual ACM symposium on Theory of computing*, pp. 459–467, 2014.

Dong, K., Li, Y., Zhang, Q., and Zhou, Y. Multinomial logit bandit with low switching cost. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 2607–2615, 2020.

Flaxman, A. D., Kalai, A. T., and McMahan, H. B. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the 16th annual ACM-SIAM symposium on Discrete algorithms*, pp. 385–394, 2005.

Goel, G., Chen, N., and Wierman, A. Thinking fast and slow: Optimization decomposition across timescales. In *2017 IEEE 56th Annual Conference on Decision and Control*, pp. 1291–1298, 2017.

Goel, G., Lin, Y., Sun, H., and Wierman, A. Beyond online balanced descent: An optimal algorithm for smoothed online optimization. In *Advances in Neural Information Processing Systems 32*, pp. 1875–1885, 2019.

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

Hazan, E. and Seshadhri, C. Efficient learning algorithms for changing environments. In *Proceedings of the 26th International Conference on Machine Learning*, pp. 393–400, 2009.

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

Jaghargh, M. R. K., Krause, A., Lattanzi, S., and Vassilvtiskii, S. Consistent online optimization: Convex and submodular. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2241–2250, 2019.

Jenatton, R., Huang, J., and Archambeau, C. Adaptive algorithms for online convex optimization with long-term constraints. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 402–411, 2016.

Joseph, V. and de Veciana, G. Jointly optimizing multi-user rate adaptation for video transport over wireless systems: Mean-fairness-variability tradeoffs. In *Proceedings of the 31st Annual IEEE International Conference on Computer Communications*, pp. 567–575, 2012.

Koolen, W. M., Warmuth, M. K., Kivinen, J., et al. Hedging structured concepts. In *Proceedings of the 23rd Annual Conference on Learning Theory*, pp. 93–105, 2010.

Li, Y., Qu, G., and Li, N. Using predictions in online optimization with switching costs: A fast algorithm and a fundamental limit. In *2018 Annual American Control Conference*, pp. 3008–3013, 2018.

Liakopoulos, N., Destounis, A., Paschos, G., Spyropoulos, T., and Mertikopoulos, P. Cautious regret minimization: Online optimization with long-term budget constraints. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 3944–3952, 2019.

Lin, M., Liu, Z., Wierman, A., and Andrew, L. L. Online algorithms for geographical load balancing. In *International Green Computing Conference*, pp. 1–10, 2012a.

Lin, M., Wierman, A., Andrew, L. L., and Thereska, E. Dynamic right-sizing for power-proportional data centers. *IEEE/ACM Transactions on Networking*, 21(5):1378–1391, 2012b.

Mahdavi, M., Jin, R., and Yang, T. Trading regret for efficiency: online convex optimization with long term constraints. *The Journal of Machine Learning Research*, 13(1):2503–2528, 2012.

Mannor, S., Tsitsiklis, J. N., and Yu, J. Y. Online learning with sample path constraints. *Journal of Machine Learning Research*, 10(3), 2009.

Neely, M. J. and Yu, H. Online convex optimization with time-varying constraints. *arXiv preprint arXiv:1702.04783*, 2017.

Orabona, F. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

Ruan, Y., Yang, J., and Zhou, Y. Linear bandits with limited adaptivity and learning distributional optimal design. *arXiv preprint arXiv:2007.01980*, 2020.

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

Sherman, U. and Koren, T. Lazy oco: Online convex optimization on a switching budget. *Proceedings of the 34th Annual Conference on Learning Theory*, 2021.

Simchi-Levi, D. and Xu, Y. Phase transitions and cyclic phenomena in bandits with switching constraints. In *Advances in Neural Information Processing Systems 32*, pp. 7523–7532, 2019.

Valls, V., Iosifidis, G., Leith, D., and Tassiulas, L. Online convex optimization with perturbed constraints: Optimal rates against stronger benchmarks. In *International Conference on Artificial Intelligence and Statistics*, pp. 2885–2895, 2020.

Vittori, E., de Luca, M. B., Trovò, F., and Restelli, M. Dealing with transaction costs in portfolio optimization: Online gradient descent with momentum. In *ACM International Conference on AI in Finance*, pp. 1–8, 2020.

Yi, X., Li, X., Xie, L., and Johansson, K. H. Distributed online convex optimization with time-varying coupled inequality constraints. *IEEE Transactions on Signal Processing*, 68:731–746, 2020.

Yu, H., Neely, M. J., and Wei, X. Online convex optimization with stochastic constraints. In *Advances in Neural Information Processing Systems 30*, pp. 1427–1437, 2017.

Zanini, F., Atienza, D., De Micheli, G., and Boyd, S. P. Online convex optimization-based algorithm for thermal management of mpsocs. In *Proceedings of the 20th symposium on Great lakes symposium on VLSI*, pp. 203–208, 2010.

Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning*, pp. 928–936, 2003.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See future work.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] There are no foreseen ethical or societal consequences for the research presented herein.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]