Understand Dynamic Regret with Switching Cost for Online Decision Making

YAWEI ZHAO, School of Computer, National University of Defense Technology, China
QIAN ZHAO, College of Mathematics and System Science, Xinjiang University, China
XINGXING ZHANG, Institute of Information Science, and Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing Jiaotong University, China
EN ZHU and XINWANG LIU, School of Computer, National University of Defense Technology, China
JIANPING YIN, School of Computer, Dongguan University of Technology, China

As a metric to measure the performance of an online method, dynamic regret with switching cost has drawn much attention for online decision making problems. Although the sublinear regret has been provided in much previous research, we still have little knowledge about the relation between the *dynamic regret* and the *switching cost*. In the article, we investigate the relation for two classic online settings: Online Algorithms (OA) and Online Convex Optimization (OCO). We provide a new theoretical analysis framework that shows an interesting observation; that is, the relation between the switching cost and the dynamic regret is different for settings of OA and OCO. Specifically, the switching cost has significant impact on the dynamic regret in the setting of OA. But it does not have an impact on the dynamic regret in the setting of OCO. Furthermore, we provide a lower bound of regret for the setting of OCO, which is same with the lower bound in the case of no switching cost. It shows that the switching cost does not change the difficulty of online decision making problems in the setting of OCO.

CCS Concepts: • **Computing methodologies** → **Machine learning algorithms**; • **Mathematics of computing** → *Convex optimization*;

Additional Key Words and Phrases: Online decision making, dynamic regret, switching cost, online algorithms, online convex optimization, online mirror descent

28

ACM Reference format:
Yawei Zhao, Qian Zhao, Xingxing Zhang, En Zhu, Xinwang Liu, and Jianping Yin. 2020. Understand Dynamic Regret with Switching Cost for Online Decision Making. *ACM Trans. Intell. Syst. Technol.* 11, 3, Article 28 (April 2020), 21 pages.
https://doi.org/10.1145/3375788

This work was supported by the National Key R & D Program of China 2018YFB1003203 and the National Natural Science Foundation of China (Grants No. 61672528, No. 61773392, and No. 61671463).
Authors' addresses: Y. Zhao, E. Zhu (corresponding author), and X. Liu, School of Computer, National University of Defense Technology, 109 Deya Road, Changsha, Hunan 410073, China; email: zhaoyawei@nudt.edu.cn, enzhu@nudt.edy.cn, xinwangliu@nudt.edu.cn; Q. Zhao, College of Mathematics and System Science, Xinjiang University, Urumqi, Xinjiang, 830001 China; email: mcsqian.zhao@gmail.com; X. Zhang, Institute of Information Science and Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing Jiaotong University, Beijing, 100044 China; email: zhangxing@bjtu.edu.cn; J. Yin, School of Computer, Dongguan University of Technology, Dongguan, Guangdong, 523808, China; email: jpyin@dgut.edu.cn.

2157-6904/2020/04-ART28 $15.00
https://doi.org/10.1145/3375788

1 INTRODUCTION

Online Algorithms (OA)[1] [14, 15, 37] and Online Convex Optimization (OCO) [9, 23, 38] are two important settings of online decision making. Methods in both OA and OCO settings are designed to make a decision at every round and then use the decision as a response to the environment. Their major differences are outlined as follows.

- For every round, methods in the setting of OA are able to know a loss function first and then make a decision as the response to the environment.
- However, for every round, methods in the setting of OCO have to make a decision before knowing the loss function. Thus, the environment may be adversarial to decisions of those methods.

Both of them have a large number of practical scenarios. For example, both the k-server problem [4, 26] and the Metrical Task Systems (MTS) problem [1, 4, 10] are usually studied in the setting of OA. Other problems include online learning [29, 39, 42, 43], online recommendation [41], online classification [6, 18], online portfolio selection [28], and model predictive control [36], which are usually studied in the setting of OCO.

Most recent research has begun to investigate performance of online methods in both OA and OCO settings by using *dynamic regret with switching cost* [15, 30]. It measures the difference between the cost yielded by real-time decisions and the cost yielded by the optimal decisions. Comparing with the classic static regret [9], it has two major differences.

- First, it allows optimal decisions to change within a threshold over time, which is necessary in the dynamic environment.[2]
- Second, the cost yielded by a decision consists of two parts: the *operating cost* and the *switching cost*, while the classic static regret only contains the operating cost.

The switching cost measures the difference between two successive decisions, which is needed in many practical scenarios such as service management in electric power network [35] and dynamic resource management in data centers [31, 33, 40]. However, we still have little knowledge about the relation between the dynamic regret and the switching cost. In the article, we are motivated by the following fundamental questions:

- *Does the switching cost impact the dynamic regret of an online method?*
- *Does the problem of online decision making become more difficult due to the switching cost?*

To answer those challenging questions, we investigate online mirror descent in settings of OA and OCO and provide a new theoretical analysis framework. According to our analysis, we find an interesting observation, that is, *the switching cost does impact on the dynamic regret in the setting of OA. But it has no impact on the dynamic regret in the setting of OCO.* Specifically, when the switching cost is measured by $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^\sigma$ with $1 \leq \sigma \leq 2$, the dynamic regret for an OA method is $\mathcal{O}(T^{\frac{1}{\sigma+1}} D^{\frac{\sigma}{\sigma+1}})$, where T is the maximal number of rounds and D is the given budget of dynamics. But the dynamic regret for an OCO method is $\mathcal{O}(\sqrt{TD} + \sqrt{T})$, which is same with the case of no switching cost [20, 21, 50, 51]. Furthermore, we provide a lower bound of dynamic regret, namely $\Omega(\sqrt{TD} + \sqrt{T})$ for the OCO setting. Since the lower bound is still same with the case of no switching cost [50], it implies that *the switching cost does not change the difficulty of the online decision making problem for the OCO setting*. Comparing with previous results, our new analysis is

[1] Some literature denote OA by "smoothed online convex optimization."

[2] Generally, the dynamic environment means the distribution of the data stream may change over time.

more general than previous results. We define a new dynamic regret with a generalized switching cost and provide new regret bounds. It is novel to analyze and provide the tight regret bound in the dynamic environment, since previous analysis cannot work directly for the generalized dynamic regert. In a nutshell, our main contributions are summarized as follows:

- We propose a new general formulation of the dynamic regret with switching cost and then develop a new analysis framework based on it.
- We provide $O(T^{\frac{1}{\sigma+1}}D^{\frac{\sigma}{\sigma+1}})$ regret with $1 \le \sigma \le 2$ for the setting of OA and $O(\sqrt{TD} + \sqrt{T})$ regret for the setting of OCO by using the online mirror descent.
- We provide a lower bound $\Omega(\sqrt{TD} + \sqrt{T})$ regret for the setting of OCO, which matches with the upper bound.

The article is organized as follows. Section 2 reviews related literature. Section 3 presents the preliminaries. Section 4 presents our new formulation of the dynamic regret with switching cost. Section 5 presents a new analysis framework and main results. Section 6 presents extensive empirical studies. Section 7 conludes the article and presents future work.

2 RELATED WORK

In the section, we review related literatures briefly.

2.1 Competitive Ratio and Regret

Although the competitive ratio is usually used to analyze OA methods, and the regret is used to analyze OCO methods, recent research aims to develop unified frameworks to analyze the performance of an online method in both settings [1–3, 8, 11–13]. Reference [8] provides an analysis framework that is able to achieve sublinear regret for OA methods and constant competitive ratio for OCO methods. References [1, 11, 12] use a general OCO method, namely online mirror descent in the OA setting, and improve the existing competitive ratio analysis for k-server and MTS problems. Different from them, we extend the existing regret analysis framework to handle a general switching cost and focus on investigating the relation between regret and switching cost. Reference [3] provides a lower bound for the OCO problem in the competitive ratio analysis framework, but we provide the lower bound in the regret analysis framework. References [2, 13] study the regret with switching cost in the OA setting, but the relation between them is not studied. Comparing with References [2, 13], we extend their analysis and present a more generalized bound of dynamic regret (see Theorem 1).

2.2 Dynamic Regret and Switching Cost

Regret is widely used as a metric to measure the performance of OCO methods. When the environment is static, e.g., the distribution of data stream does not change over time, online mirror descent yields $O(\sqrt{T})$ regret for convex functions and $O(\log T)$ regret for strongly convex functions [9, 23, 38]. When the distribution of data stream changes over time, online mirror descent yields $O(\sqrt{TD} + \sqrt{T})$ regret for convex functions [20], where D is the given budget of dynamics. Additionally, Reference [51] first investigates online gradient descent in the dynamic environment and obtains $O(\sqrt{TD} + \sqrt{T})$ regret (by setting $\eta \propto \sqrt{\frac{D}{T}}$) for convex f_t. Note that the dynamic regret used in Reference [51] does not contain swtiching cost. References [21, 22] use similar but more general definitions of dynamic regret and still achieves $O(\sqrt{TD} + \sqrt{T})$ regret. Furthermore, Reference [50] presents that the lower bound of the dynamic regret is $\Omega(\sqrt{TD} + \sqrt{T})$. Other previous research investigates the regret under different definitions of dynamics such as parameter variation [19, 34, 44, 47], functional variation [7, 25, 46], gradient variation [17], and the mixed regularity

Table 1. Summary of Differences between OA and OCO

Algo.	Make decision first?	Observe f_t first?	Metric	Has SC?
OA	no	yes	competitive ratio	yes
OCO	yes	no	regret	no

SC represents "switching cost."

[16, 24]. Note that the dynamic regret in those previous studies does not contain switching cost, which is significantly different from our work. Our new analysis shows that this bound is achieved and optimal when there is switching cost in the regret (see Theorems 2 and 3). The proposed analysis framework thus shows how the switching cost impacts the dynamic regret for settings of OA and OCO, which leads to new insights to understand online decision making problems.

3 PRELIMINARIES

In the section, we present the preliminaries of online algorithms and online convex optimization and highlight their differences. Then, we present the dynamic regret with switching cost, which is used to measure the performance of both OA methods and OCO methods.

3.1 Online Algorithms and Online Convex Optimization

Comparing with the setting of OCO [9, 23, 38], OA has the following major difference.

- OA assumes that the loss function, e.g., f_t, is known before making the decision at every round. But OCO assumes that the loss function, e.g., f_t, is given after making the decision at every round.
- The performance of an OA method is measured by using the *competitive ratio* [15], which is defined by

$$\frac{\left[\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|\right)\right]}{\left[\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t^*) + \|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\|\right)\right]}.$$

  Here, $\{\mathbf{x}_t^*\}_{t=1}^T$ is denoted by

$$\{\mathbf{x}_t^*\}_{t=1}^T = \operatorname*{argmin}_{\{\mathbf{z}_t\}_{t=1}^T \in \tilde{\mathcal{L}}_D^T} \sum_{t=1}^{T}\left(f_t(\mathbf{z}_t) + \|\mathbf{z}_t - \mathbf{z}_{t-1}\|\right),$$

  where $\tilde{\mathcal{L}}_D^T := \{\{\mathbf{z}_t\}_{t=1}^T : \sum_{t=1}^T \|\mathbf{z}_t - \mathbf{z}_{t-1}\| \le D\}$. D is the given budget of dynamics. It is the best offline strategy, which is yielded by knowing all the requests beforehand [15]. Note that $\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\|$ is the switching cost yielded by A at the tth round. But OCO is usually measured by the *regret*, which is defined by

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\{\mathbf{z}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^{T} f_t(\mathbf{z}_t),$$

  where $\mathcal{L}_D^T := \{\{\mathbf{z}_t\}_{t=1}^T : \sum_{t=1}^{T-1} \|\mathbf{z}_{t+1} - \mathbf{z}_t\| \le D\}$. D is also the given budget of dynamics. Note that the regret in classic OCO algorithm does not contain the switching cost.

To make it clear, we use Table 1 to highlight their differences.

3.2 Dynamic Regret with Switching Cost

Although the analysis framework of OA and OCO is different, the *dynamic regret with switching cost* is a popular metric to measure the performance of both OA and OCO [15, 30]. Formally, for an algorithm A, its dynamic regret with switching cost $\widetilde{\mathcal{R}}_D^A$ is defined by

$$\widetilde{\mathcal{R}}_D^A := \sum_{t=1}^{T} f_t(\mathbf{x}_t) + \sum_{t=1}^{T-1} \|\mathbf{x}_{t+1} - \mathbf{x}_t\| - \min_{\{\mathbf{z}_t\}_{t=1}^T \in \mathcal{L}_D^T} \left(\sum_{t=1}^{T} f_t(\mathbf{z}_t) + \sum_{t=1}^{T-1} \|\mathbf{z}_{t+1} - \mathbf{z}_t\| \right), \tag{1}$$

where $\mathcal{L}_D^T := \{\{\mathbf{z}_t\}_{t=1}^T : \sum_{t=1}^{T-1} \|\mathbf{z}_{t+1} - \mathbf{z}_t\| \le D\}$. Here, $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|$ represents the switching cost at the tth round. D is the given budget of dynamics in the dynamic environment. When $D = 0$, all optimal decisions are same. With the increase of D, the optimal decisions are allowed to change to follow the dynamics in the environment. It is necessary when the distribution of data stream changes over time.

3.3 Notations and Assumptions

We use the following notations in the article.

- The bold lowercase letters, e.g., $\mathbf{x}$, represent vectors. The normal letters, e.g., μ, represent a scalar number.
- $\|\cdot\|$ represents a general norm of a vector.
- $\mathcal{X}^T$ represents Cartesian product, namely, $\underbrace{\mathcal{X} \times \mathcal{X} \times \cdots \times \mathcal{X}}_{T \text{ times}}$. $\mathcal{F}^T$ has the similar meaning.
- Bregman divergence $B_\Phi(\mathbf{x}, \mathbf{y})$ is defined by $B_\Phi(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) - \Phi(\mathbf{y}) - \langle \nabla\Phi(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle$.
- $\mathcal{A}$ represents a set of all possible online methods, and $A \in \mathcal{A}$ represents some a specific online method.
- $\lesssim$ represents 'less than equal up to a constant factor'.
- $\mathbb{E}$ represents the mathematical expectation operator.

Our assumptions are presented as follows. They are widely used in previous literature [9, 15, 23, 30, 38].

Assumption 1. *The following basic assumptions are used throughout the article.*

- *For any $t \in [T]$, we assume that f_t is convex and has L-Lipschitz gradient.*
- *The function Φ is μ-strongly convex, that is, for any $\mathbf{x} \in \mathcal{X}$ and $\mathbf{y} \in \mathcal{X}$, $B_\Phi(\mathbf{x}, \mathbf{y}) \ge \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2$.*
- *For any $\mathbf{x} \in \mathcal{X}$ and $\mathbf{y} \in \mathcal{X}$, there exists a positive constant R such that*
$$\max\left\{B_\Phi(\mathbf{x}, \mathbf{y}), \|\mathbf{x} - \mathbf{y}\|^2\right\} \le R^2.$$
- *For any $\mathbf{x} \in \mathcal{X}$, there exists a positive constant G such that*
$$\max\left\{\|\nabla f_t(\mathbf{x})\|^2, \|\nabla\Phi(\mathbf{x})\|^2\right\} \le G^2$$

4 DYNAMIC REGRET WITH GENERALIZED SWITCHING COST

In the section, we propose a new formulation of dynamic regret, which contains a generalized switching cost. Then, we highlight the novelty of this formulation and present the online mirror decent method for setting of OA and OCO.

4.1 Formulation

For an algorithm $A \in \mathcal{A}$, it yields a cost at the end of every round, which consists of two parts: *operating cost* and *switching cost*. At the tth round, the *operating cost* is incurred by $f_t(\mathbf{x}_t)$, and the *switching cost* is incurred by $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^\sigma$ with $1 \le \sigma \le 2$. The optimal decisions are denoted by $\{\mathbf{y}_t^*\}_{t=1}^T$, which is denoted by

$$\{\mathbf{y}_t^*\}_{t=1}^T = \operatorname*{argmin}_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^{T} f_t(\mathbf{y}_t) + \sum_{t=1}^{T-1} \|\mathbf{y}_{t+1} - \mathbf{y}_t\|^\sigma .$$

Here, $\mathcal{L}_D^T$ is denoted by

$$\mathcal{L}_D^T = \left\{ \{\mathbf{y}_t\}_{t=1}^T : \sum_{t=1}^{T-1} \|\mathbf{y}_{t+1} - \mathbf{y}_t\| \le D \right\}.$$

D is a given budget of dynamics, which measures how much the optimal decision, i.e., $\mathbf{y}_t^*$ can change over t. With the increase of D, those optimal decisions can change over time to follow the dynamics in the environment effectively.

Denote an optimal method A^*, which yields the optimal sequence of decisions $\{\mathbf{y}_t^*\}_{t=1}^T$. Its total cost is denoted by

$$\mathrm{cost}(A^*) = \sum_{t=1}^{T} f_t(\mathbf{y}_t^*) + \sum_{t=1}^{T-1} \|\mathbf{y}_{t+1}^* - \mathbf{y}_t^*\|^\sigma .$$

Similarly, the total cost of an algorithm $A \in \mathcal{A}$ is denoted by

$$\mathrm{cost}(A) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) + \sum_{t=1}^{T-1} \|\mathbf{x}_{t+1} - \mathbf{x}_t\|^\sigma .$$

Definition 1. For any algorithm $A \in \mathcal{A}$, its dynamic regret $\mathcal{R}_D^A$ with switching cost is defined by

$$\mathcal{R}_D^A := \mathrm{cost}(A) - \mathrm{cost}(A^*). \tag{2}$$

Our new formulation of the dynamic regret $\mathcal{R}_D^A$ makes a balance between the operating cost and the switching cost, which is different from the previous definition of the dynamic regret in [20, 21, 51].

Note that the freedom of σ with $1 \le \sigma \le 2$ allows our new dynamic regret $\mathcal{R}_D^A$ to measure the performance of online methods for a large number of problems. Some problems such as dynamic control of data centers [32], stock portfolio management [27], require to be sensitive to the small change between successive decisions, and the switching cost in these problems is usually bounded by $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|$. But many problems such as dynamic placement of cloud service [49] need to bound the large change between successive decisions effectively, and the switching cost in these problems is usually bounded by $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2$.

4.2 Novelty of the New Formulation

Our new formulation of the dynamic regret is more general than previous formulations [15, 30], which are presented as follows.

- **Support more general switching cost.** Reference [15] defines the dynamic regret with switching cost by Equation (1). It is a special case of our new formulation (2) by setting $\sigma = 1$. The sequence of optimal decisions $\{\mathbf{y}_t^*\}_{t=1}^T$ is dominated by $\{f_t\}_{t=1}^T$ and D and does not change over $\{\mathbf{x}_t\}_{t=1}^T$. $\mathcal{R}_D^A$ is thus impacted by $\{\mathbf{x}_t\}_{t=1}^T$ for the given $\{f_t\}_{t=1}^T$ and D. Generally, $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|$ is more sensitive to measure the slight change between $\mathbf{x}_{t+1}$ and $\mathbf{x}_t$ than $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2$. But, for some problems such as the dynamic placement of cloud service [49], the switching cost at the tth round is usually measured by $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2$, instead of $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|$. The previous formulation in Reference [15] is not suitable to bound the

ALGORITHM 1: MD-OA: Online Mirror Descent for OA.

Require: The learning rate γ, and the number of rounds T.

```
for t = 1, 2, ..., T do
    Observe the loss function f_t.                                    ▷ Observe f_t first.
    Query a gradient ĝ_t ∈ ∇f_t(x_{t-1}).
    x_t = argmin_{x∈X} ⟨ĝ_t, x − x_{t-1}⟩ + (1/γ) B_Φ(x, x_{t-1}).    ▷ Play a decision after knowing f_t.
return x_T
```

switching cost for those problems. Benefiting from $1 \leq \sigma \leq 2$, (2) supports more general switching cost than previous work.

- **Support more general convex f_t.** Reference [30] defines the the dynamic regret with switching cost by

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) + \sum_{t=1}^{T-1} \|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 - \min_{\{\mathbf{z}_t\}_{t=1}^{T} \in \mathcal{X}^T} \left(\sum_{t=1}^{T} f_t(\mathbf{z}_t) + \sum_{t=1}^{T-1} \|\mathbf{z}_{t+1} - \mathbf{z}_t\|^2 \right),$$

and they use $\sum_{t=1}^{T-1} \|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\|$ to bound the regret. Here, $\mathbf{x}_t^* = \operatorname{argmin}_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x})$. It implicitly assumes that the difference between $\mathbf{x}_{t+1}^*$ and $\mathbf{x}_t^*$ are bounded. It is reasonable for a strongly convex function f_t but may not be guaranteed for a general convex function f_t. Additionally, Reference [30] uses $\|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\|^2$ to bound the switching cost, which is more sensitive to the significant change than $\|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\|$. But it is less effective to bound the slight change between them, which is not suitable for many problems such as dynamic control of data centers [32].

4.3 Algorithm

We use mirror descent [5] in the online setting, and present the algorithm MD-OA for the OA setting and the algorithm MD-OCO for the OCO setting, respectively.

As illustrated in Algorithms 1 and 2, both MD-OA and MD-OCO are performed iteratively. For every round, MD-OA first observes the loss function f_t and then makes the decision $\mathbf{x}_t$ at the tth round. But MD-OCO first makes the decision $\mathbf{x}_t$ and then observes the loss function f_t. Therefore, MD-OA usually makes the decision based on the observed f_t for the current round, but MD-OCO has to predict a decision for the next round based on the received f_t.

Note that both MD-OA and MD-OCO need to solve a convex optimizaiton problem to update x. The complexity is dominated by the domain $\mathcal{X}$ and the distance function Φ. Besides, both of them lead to $O(d)$ memory cost. They lead to comparable cost of computation and memory.

5 THEORETICAL ANALYSIS

In this section, we present our main analysis results about the proposed dynamic regret for both MD-OA and MD-OCO and discuss the difference between them.

5.1 New Bounds for Dynamic Regret with Switching Cost

The upper bound of dynamic regret for MD-OA is presented as follows.

Theorem 1. *Choose $\gamma = \min\{\frac{\mu}{L}, T^{-\frac{1}{1+\sigma}} D^{\frac{1}{1+\sigma}}\}$ in Algorithm 1. Under Assumption 1, we have*

$$\sup_{\{f_t\}_{t=1}^{T} \in \mathcal{F}^T} \mathcal{R}_D^{MD\text{-}OA} \lesssim T^{\frac{1}{\sigma+1}} D^{\frac{\sigma}{\sigma+1}} + T^{\frac{1}{\sigma+1}} D^{-\frac{1}{\sigma+1}}.$$

That is, Algorithm 1 yields $O(T^{\frac{1}{\sigma+1}} D^{\frac{\sigma}{\sigma+1}})$ dynamic regret with switching cost.

ALGORITHM 2: MD-OCO: Online Mirror Descent for OCO.

Require: The learning rate η, the number of rounds T, and $\mathbf{x}_0$.

1: **for** $t = 0, 1, \ldots, T-1$ **do**

2: Play $\mathbf{x}_t$. ▷ *Play a decision first before knowing f_t.*

3: Receive a loss function f_t.

4: Query a gradient $\bar{\mathbf{g}}_t \in \nabla f_t(\mathbf{x}_t)$.

5: $\mathbf{x}_{t+1} = \operatorname{argmin}_{\mathbf{x} \in \mathcal{X}} \langle \bar{\mathbf{g}}_t, \mathbf{x} - \mathbf{x}_t \rangle + \frac{1}{\eta} B_\Phi(\mathbf{x}, \mathbf{x}_t)$.

6: **return** $\mathbf{x}_T$

Remark 1. When $\sigma = 1$, MD-OA yields $\mathcal{O}(\sqrt{TD})$ dynamic regret, which achieves the state-of-the-art result in Reference [15]. When $\sigma = 2$, MD-OA yields $\mathcal{O}(T^{\frac{1}{3}} D^{\frac{2}{3}})$ dynamic regret, which is a new result as far as we know.

However, we find different result for MD-OCO. The switching cost does not have an impact on the dynamic regret.

Theorem 2. *Choose $\eta = \min\{\frac{\mu}{4}, \sqrt{\frac{D+G}{T}}\}$ in Algorithm 2. Under Assumption 1, we have*

$$\sup_{\{f_t\}_{t=1}^T \in \mathcal{F}^T} \mathcal{R}_D^{\text{MD-OCO}} \lesssim \sqrt{TD} + \sqrt{T}.$$

That is, Algorithm 2 yields $\mathcal{O}(\sqrt{DT} + \sqrt{T})$ dynamic regret with switching cost.

Remark 2. MD-OCO still yields $\mathcal{O}(\sqrt{TD} + \sqrt{T})$ dynamic regret [20] when there is no switching cost. It shows that the switching cost does not have an impact on the dynamic regret.

Before presenting the discussion, we show that MD-OCO is the optimum for dynamic regret, because the lower bound of the problem matches with the upper bound yielded by MD-OCO.

Theorem 3. *Under Assumption 1, the lower bound of the dynamic regret for the OCO problem is*

$$\inf_{A \in \mathcal{A}} \sup_{\{f_t\}_{t=1}^T \in \mathcal{F}^T} \mathcal{R}_D^A = \Omega\left(\sqrt{TD} + \sqrt{T}\right).$$

Remark 3. When there is no switching cost, the lower bound of dynamic regret for OCO is $\mathcal{O}(\sqrt{TD} + \sqrt{T})$ [50]. Theorem 3 achieves it for the case of switching cost. It implies that the switching cost does not let the online decision making in the OCO setting become more difficult.

5.2 Insights

Switching cost has a significant impact on the dynamic regret for the setting of OA. According to Theorem 1, the switching cost has a significant impact on the dynamic regret of MD-OA. Given a constant D, a small σ leads to a strong dependence on T, and a large σ leads to a weak dependence on T. The reason is that a large σ leads to a large learning rate, which is more effective to follow the dynamics in the environment than a small learning rate.

Switching cost does not have an impact on the dynamic regret for the setting of OCO. According to Theorem 2 and Theorem 3, the dynamic regret yielded by MD-OCO is tight, and MD-OCO is the optimum for the problem. Although the switching cost exists, the dynamic regret yielded by MD-OCO does not have any difference.

As we can see, there is a significant difference between the OA setting and the OCO setting. The reasons are presented as follows.

- MD-OA makes decisions after observing the loss function. It has known the potential operating cost and switching cost for any decision. Thus, it can make decisions to achieve a good tradeoff between the operating cost and switching cost.
- MD-OCO make decisions before observing the loss function. It only knows the historical information and the potential switching cost, and does not know the potential operating cost for any decision at the current round. In the worst case, if the environment provides an adversary loss function to maximize the operating cost based on the decision made by MD-OCO, MD-OCO has to lead to $\mathcal{O}(\sqrt{TD}+\sqrt{T})$ regret even for the case of no switching cost [20]. Although the potential switching cost is known, MD-OCO cannot make a better decision to reduce the regret due to unknown operating cost.

6 EMPIRICAL STUDIES

In this section, we evaluate the total regret and the regret caused by switching cost for settings of both OA and OCO by running online mirror decent. Our experiments show the importance of knowing loss function before making a decision.

6.1 Experimental Settings

We conduct binary classification by using the logistic regression model. Given an instance $\mathbf{a} \in \mathbb{R}^d$ and its label $y \in \{1, -1\}$, the loss function is

$$f(\mathbf{x}) = \log\left(1 + \exp\left(-y\mathbf{a}^\top\mathbf{x}\right)\right).$$

In experiments, we let $\Phi(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|^2$.

We test four methods, including MD-OA, i.e., Algorithm 1, and MD-OCO, i.e., Algorithm 2, online balanced descent [15] denoted by BD-OA in the experiment, and multiple online gradient descent [48] denoted by MGD-OCO in the experiment. Both MD-OA and BD-OA are two variants of online algorithm, and similarily both MD-OCO and MGD-OCO are two variants of online convex optimization. We test those methods on three real datasets: *usenet1*,[3] *usenet2*,[4] and *spam*.[5] The distributions of data streams change over time for those datasets, which is just the dynamic environment as we have discussed. More details about those datasets and its dynamics are presented at: http://mlkd.csd.auth.gr/concept_drift.html.

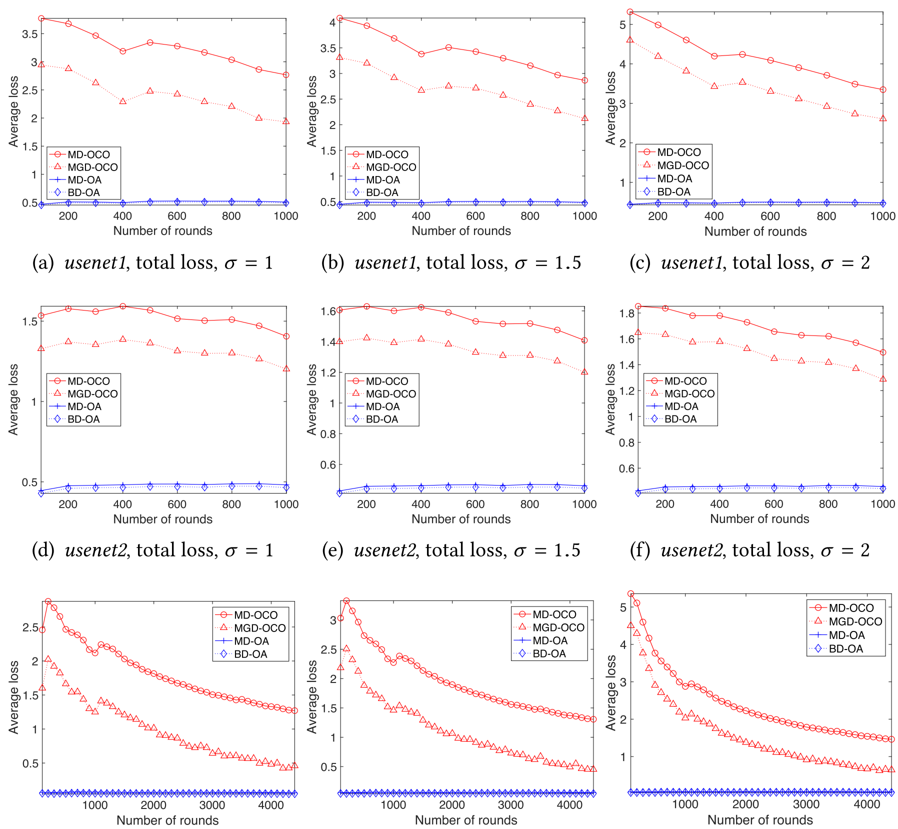
We use the *average loss* to test the regret, because they have the same optimal reference points $\{\mathbf{y}_t^*\}_{l=1}^t$. For the tth round, the average loss is defined by

$$\underbrace{\frac{1}{t}\sum_{l=1}^{t}\log\left(1+\exp\left(-\mathbf{y}_l\mathbf{A}_l^\top\mathbf{x}_l\right)\right)}_{\text{average loss caused by operating cost}} + \underbrace{\frac{1}{t}\sum_{l=0}^{t-1}\|\mathbf{x}_{l+1}-\mathbf{x}_l\|}_{\text{average loss caused by switching cost}},$$

where $\mathbf{A}_l$ is the instance at the lth round and $\mathbf{y}_l$ is its label. Besides, we evaluate the average loss caused by operating cost separately and denote it by OL. Similarly, SL represents the average loss caused by switching cost.

In experiment, we set $D = 10$. Since G, μ, and L are usually not known in practical scenarios, the learning rate is set by the following heuristic rules. We choose the learning rate $\gamma_t = \eta_t = \frac{\delta}{\sqrt{t}}$ for the tth iteration, where δ is a given constants by the following rules. First, we set a large value $\delta = 10$. Then, we iteratively adjust the value of δ by $\delta \leftarrow \delta/2$ when δ cannot let the average loss

[3] http://lpis.csd.auth.gr/mlkd/usenet1.rar.
[4] http://lpis.csd.auth.gr/mlkd/usenet2.rar.
[5] http://lpis.csd.auth.gr/mlkd/concept_drift/spam_data.rar.



(a) *usenet1*, total loss, $\sigma = 1$ (b) *usenet1*, total loss, $\sigma = 1.5$ (c) *usenet1*, total loss, $\sigma = 2$

(d) *usenet2*, total loss, $\sigma = 1$ (e) *usenet2*, total loss, $\sigma = 1.5$ (f) *usenet2*, total loss, $\sigma = 2$

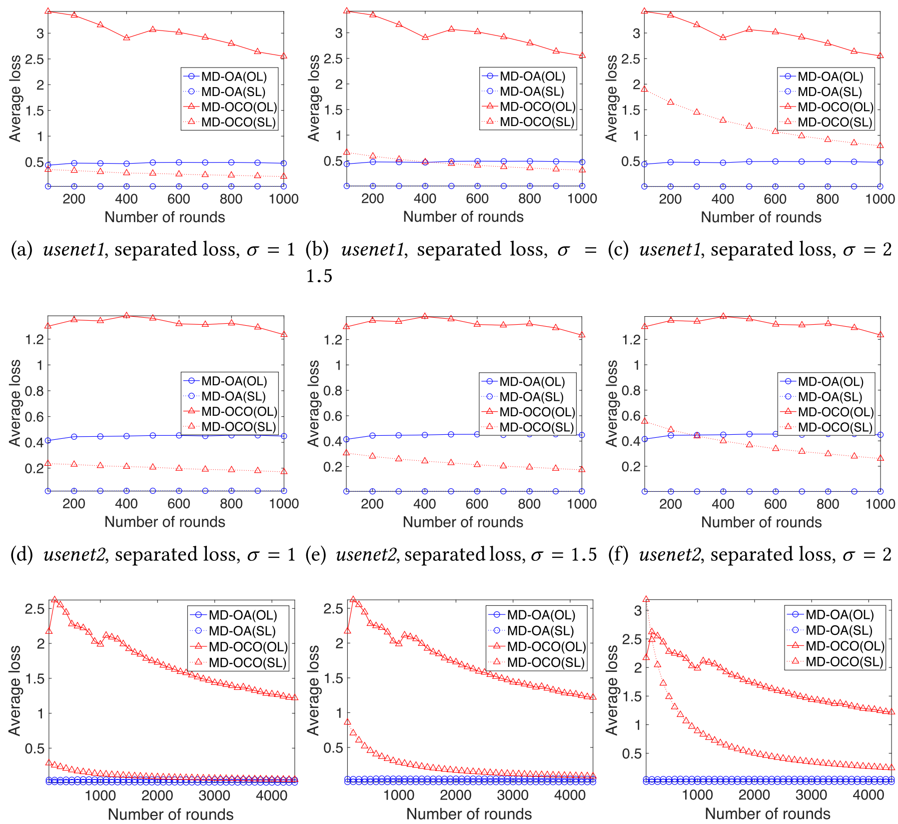
(g) *spam*, total loss, $\sigma = 1$ (h) *spam*, total loss, $\sigma = 1.5$ (i) *spam*, total loss, $\sigma = 2$

Fig. 1. OCO methods leads to large average loss than OA methods.

converge. If the first appropriate δ can let the average loss converge, then it is finally chosen as the optimal learning rate. We use the similar heuristic method to determine other parameters, e.g., the number of inner iterations in MGD-OCO. Finally, the mirror map function is $\frac{1}{2}\|\cdot\|^2$ for BD-OA.

6.2 Numerical Results

As shown in Figure 1, both MD-OA and BD-OA are much more effcetive than MD-OCO and MGD-OCO to decrease the average loss during a few rounds of begining. Those OA methods yield much smaller average loss than OCO methods. The reason is that OA knows the loss function f_t before making decision $\mathbf{x}_t$. But, OCO has to make decision before know the loss function. Benefiting from knowing the loss function f_t, OA reduces the average loss more efffectively than OCO. It matches with our theoretical analysis. That is, Algorithm 1 leads to $O(T^{\frac{1}{1+\sigma}} D^{\frac{\sigma}{1+\sigma}})$ regret, but Algorithm 2 leads to $O(\sqrt{TD} + \sqrt{T})$ regret. When $\sigma \geq 1$, OA tends to lead to smaller regret than OCO. The reason is that OA knows the potential loss before making a decision for every round. But, OCO works in an adversary environment, and it has to make a decision before knowing the potential loss. Thus, OA is able to make a better decision than OCO to decrease the loss. Additionally, we observe that both MD-OA and BD-OA reduce much more average loss than MD-OCO and MGD-OCO for a large σ, which validates our theoretical results nicely. It means that OA is more effective to reduce the switching cost than OCO for a large σ. Specifically, as shown in Figure 2, the average



(a) *usenet1*, separated loss, $\sigma = 1$ (b) *usenet1*, separated loss, $\sigma = 1.5$ (c) *usenet1*, separated loss, $\sigma = 2$

(d) *usenet2*, separated loss, $\sigma = 1$ (e) *usenet2*, separated loss, $\sigma = 1.5$ (f) *usenet2*, separated loss, $\sigma = 2$

(g) *spam*, separated loss, $\sigma = 1$ (h) *spam*, separated loss, $\sigma = 1.5$ (i) *spam*, separated loss, $\sigma = 2$

Fig. 2. Comparing with MD-OCO. The superiority of MD-OA becomes significant for a large σ.

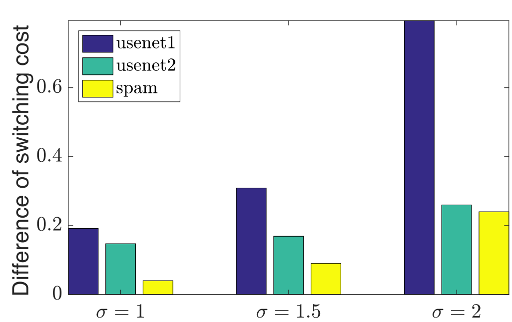


Fig. 3. MD-OCO leads to more average loss caused by switching cost than MD-OA, especially for a large σ.

loss caused by switching cost of OA methods, i.e., MD-OA(SL), has unsignificant changes, but that of OCO methods, i.e., MD-OCO(SL), has remarkable increase for a large σ.

When handling the whole dataset, the final difference of switching cost between MD-OA and MD-OCO is shown in Figure 3. Here, the difference of switching cost is measured by using *average loss caused by switching cost* of MD-OCO minus corresponding *average loss caused by switching cost* of MD-OA. As we can see, it highlights that OA is more effective to decrease the switching cost. The superiority becomes significant for a large σ, which verifies our theoretical results nicely again.

7 CONCLUSION AND FUTURE WORK

We have proposed a new dynamic regret with switching cost and a new analysis framework for both online algorithms and online convex optimization. We find that the switching cost significantly impacts on the regret yielded by OA methods but does not have an impact on the regret yielded by OCO methods. Empirical studies have validated our theoretical result.

Moreover, the switching cost in the article is measured by using the norm of the difference between two successive decisions, that is, $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|$. It is interesting to investigate whether the work can be extended to a more general distance measure function such as Bregman divergence $d_B(\mathbf{x}_{t+1}, \mathbf{x}_t)$ or Mahalanobis distance $d_M(\mathbf{x}_{t+1}, \mathbf{x}_t)$. Specifically, if the Bregman divergence[6] is used, then the switching cost is thus $d_B(\mathbf{x}_{t+1}, \mathbf{x}_t) = \psi(\mathbf{x}_{t+1}) - \psi(\mathbf{x}_t) - \langle \nabla\psi(\mathbf{x}_t), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle$, where $\psi(\cdot)$ is a differentiable distance function. If the Mahalanobis distance[7] is used, then the switching cost is thus $d_M(\mathbf{x}_{t+1}, \mathbf{x}_t) = \sqrt{(\mathbf{x}_{t+1} - \mathbf{x}_t)^\top \mathrm{S}(\mathbf{x}_{t+1} - \mathbf{x}_t)}$, where S is the given covariance matrix. We leave the potential extension as the future work.

Besides, our analysis provides regret bound for any given budget of dynamics D. It is a good direction to extend the work in the parameter-free setting, where analysis is adaptive to the dynamics D of environment. Some previous work, such as Reference [45], has proposed the adaptive online method and analysis framework. But Reference [45] works in the expert setting, not a general setting of online convex optimization. It is still unknown whether their method can be used to extend our analysis.

APPENDIX

PROOFS

Lemma 1. *Given any vectors* $\mathbf{g}$, $\mathbf{u}_t \in \mathcal{X}$, $\mathbf{u}^* \in \mathcal{X}$ *, and a constant scalar* $\lambda > 0$, *if*

$$\mathbf{u}_{t+1} = \mathop{argmin}_{\mathbf{u} \in \mathcal{X}} \langle \mathbf{g}, \mathbf{u} - \mathbf{u}_t \rangle + \frac{1}{\lambda} B_\Phi(\mathbf{u}, \mathbf{u}_t),$$

we have

$$\langle \mathbf{g}, \mathbf{u}_{t+1} - \mathbf{u}^* \rangle \le \frac{1}{\lambda} \left(B_\Phi(\mathbf{u}^*, \mathbf{u}_t) - B_\Phi(\mathbf{u}^*, \mathbf{u}_{t+1}) - B_\Phi(\mathbf{u}_{t+1}, \mathbf{u}_t) \right).$$

Proof. Denote $h(\mathbf{u}) = \langle \mathbf{g}, \mathbf{u} - \mathbf{u}_t \rangle + \frac{1}{\lambda} B_\Phi(\mathbf{u}, \mathbf{u}_t)$, and $\mathbf{u}_\tau = \mathbf{u}_{t+1} + \tau(\mathbf{u}^* - \mathbf{u}_{t+1})$. According to the optimality of $\mathbf{x}_t$, we have

$$\begin{aligned}
& 0 \le h(\mathbf{u}_\tau) - h(\mathbf{u}_{t+1}) \\
= & \langle \mathbf{g}, \mathbf{u}_\tau - \mathbf{u}_{t+1} \rangle + \frac{1}{\lambda} \left(B_\Phi(\mathbf{u}_\tau, \mathbf{u}_t) - B_\Phi(\mathbf{u}_{t+1}, \mathbf{u}_t) \right) \\
= & \langle \mathbf{g}, \tau(\mathbf{u}^* - \mathbf{u}_{t+1}) \rangle + \frac{1}{\lambda} \left(\Phi(\mathbf{u}_\tau) - \Phi(\mathbf{u}_{t+1}) + \langle \nabla\Phi(\mathbf{u}_t), \tau(\mathbf{u}_{t+1} - \mathbf{u}^*) \rangle \right) \\
\le & \langle \mathbf{g}, \tau(\mathbf{u}^* - \mathbf{u}_{t+1}) \rangle + \frac{1}{\lambda} \langle \nabla\Phi(\mathbf{u}_{t+1}), \tau(\mathbf{u}^* - \mathbf{u}_{t+1}) \rangle + \frac{1}{\lambda} \langle \nabla\Phi(\mathbf{u}_t), \tau(\mathbf{u}_{t+1} - \mathbf{u}^*) \rangle \\
= & \langle \mathbf{g}, \tau(\mathbf{u}^* - \mathbf{u}_{t+1}) \rangle + \frac{1}{\lambda} \langle \nabla\Phi(\mathbf{u}_t) - \Phi(\mathbf{u}_{t+1}), \tau(\mathbf{u}_{t+1} - \mathbf{u}^*) \rangle .
\end{aligned}$$

Thus, we have

$$\begin{aligned}
& \langle \mathbf{g}, \mathbf{u}_{t+1} - \mathbf{u}^* \rangle \le \frac{1}{\lambda} \langle \nabla\Phi(\mathbf{u}_t) - \Phi(\mathbf{u}_{t+1}), \mathbf{u}_{t+1} - \mathbf{u}^* \rangle \\
= & \frac{1}{\lambda} \left(B_\Phi(\mathbf{u}^*, \mathbf{u}_t) - B_\Phi(\mathbf{u}^*, \mathbf{u}_{t+1}) - B_\Phi(\mathbf{u}_{t+1}, \mathbf{u}_t) \right).
\end{aligned}$$

[6] See details in https://en.wikipedia.org/wiki/Bregman_divergence.
[7] See details in https://en.wikipedia.org/wiki/Mahalanobis_distance.

It completes the proof. □

Lemma 2. *For any* $\mathbf{x} \in \mathcal{X}$, *we have*

$$B_\Phi(\mathbf{y}_{t+1}^*, \mathbf{x}) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}) \le 2G \left\| \mathbf{y}_{t+1}^* - \mathbf{y}_t^* \right\|. \tag{1}$$

Proof. According to the third-point identity of the Bregman divergence, we have

$$\begin{aligned}
& B_\Phi(\mathbf{y}_{t+1}^*, \mathbf{x}) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}) \\
=& \left\langle \nabla\Phi(\mathbf{y}_{t+1}^*) - \nabla\Phi(\mathbf{x}), \mathbf{y}_{t+1}^* - \mathbf{y}_t^* \right\rangle - B_\Phi(\mathbf{y}_t^*, \mathbf{y}_{t+1}^*) \\
\overset{①}{\le}& \left\langle \nabla\Phi(\mathbf{y}_{t+1}^*) - \nabla\Phi(\mathbf{x}), \mathbf{y}_{t+1}^* - \mathbf{y}_t^* \right\rangle \\
\le& \left\| \nabla\Phi(\mathbf{y}_{t+1}^*) - \nabla\Phi(\mathbf{x}) \right\| \left\| \mathbf{y}_{t+1}^* - \mathbf{y}_t^* \right\| \\
\le& \left(\left\| \nabla\Phi(\mathbf{y}_{t+1}^*) \right\| + \left\| \nabla\Phi(\mathbf{x}) \right\| \right) \left\| \mathbf{y}_{t+1}^* - \mathbf{y}_t^* \right\| \\
\le& 2G \left\| \mathbf{y}_{t+1}^* - \mathbf{y}_t^* \right\|.
\end{aligned} \tag{2}$$

① holds, because $B_\Phi(\mathbf{u}, \mathbf{v}) \ge 0$ holds for any vectors $\mathbf{u}$ and $\mathbf{v}$. It completes the proof. □

Lemma 3. *Given* $\mathbf{x}_{t-1} \in \mathcal{X}$ *and* $\hat{\mathbf{g}}_t$, *if* $\mathbf{x}_t = \mathrm{argmin}_{\mathbf{x} \in \mathcal{X}} \left\langle \hat{\mathbf{g}}_t, \mathbf{x} - \mathbf{x}_{t-1} \right\rangle + \frac{1}{\gamma} B_\Phi(\mathbf{x}, \mathbf{x}_{t-1})$, *we have*

$$\left\| \mathbf{x}_t - \mathbf{x}_{t-1} \right\| \le \frac{2G\gamma}{\mu}.$$

Proof.

$$\left\langle \hat{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{x}_{t-1} \right\rangle + \frac{\mu}{2\gamma} \left\| \mathbf{x}_t - \mathbf{x}_{t-1} \right\|^2 \overset{①}{\le} \left\langle \hat{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{x}_{t-1} \right\rangle + \frac{1}{\gamma} B_\Phi(\mathbf{x}_t, \mathbf{x}_{t-1}) \overset{②}{\le} 0.$$

① holds due to Φ is μ-strongly convex, and ② holds due to the optimality of $\mathbf{x}_t$. Thus,

$$\frac{\mu}{2\gamma} \left\| \mathbf{x}_t - \mathbf{x}_{t-1} \right\|^2 \le \left\langle \hat{\mathbf{g}}_t, -\mathbf{x}_t + \mathbf{x}_{t-1} \right\rangle \le \left\| \hat{\mathbf{g}}_t \right\| \left\| -\mathbf{x}_t + \mathbf{x}_{t-1} \right\| \le G \left\| -\mathbf{x}_t + \mathbf{x}_{t-1} \right\|.$$

That is,

$$\left\| \mathbf{x}_t - \mathbf{x}_{t-1} \right\| \le \frac{2G\gamma}{\mu}.$$

It completes the proof. □

Proof to Theorem 1:

Proof.

$$\begin{aligned}
& f_t(\mathbf{x}_t) - f_t(\mathbf{y}_t^*) \\
=& f_t(\mathbf{x}_t) - f_t(\mathbf{x}_{t-1}) + f_t(\mathbf{x}_{t-1}) - f_t(\mathbf{y}_t^*) \\
\le& f_t(\mathbf{x}_t) - f_t(\mathbf{x}_{t-1}) + \left\langle \hat{\mathbf{g}}_t, \mathbf{x}_{t-1} - \mathbf{y}_t^* \right\rangle \\
=& f_t(\mathbf{x}_t) - f_t(\mathbf{x}_{t-1}) - \left\langle \hat{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{x}_{t-1} \right\rangle + \left\langle \hat{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{y}_t^* \right\rangle \\
\overset{①}{\le}& \frac{L}{2} \left\| \mathbf{x}_{t-1} - \mathbf{x}_t \right\|^2 + \left\langle \hat{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{y}_t^* \right\rangle \\
\overset{②}{\le}& \frac{L}{2} \left\| \mathbf{x}_{t-1} - \mathbf{x}_t \right\|^2 + \frac{1}{\gamma} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t-1}) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{x}_t, \mathbf{x}_{t-1}) \right) \\
\overset{③}{\le}& \frac{L\gamma - \mu}{2\gamma} \left\| \mathbf{x}_{t-1} - \mathbf{x}_t \right\|^2 + \frac{1}{\gamma} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t-1}) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) \right) \\
\overset{④}{\le}& \frac{1}{\gamma} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t-1}) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) \right).
\end{aligned} \tag{3}$$

① holds, because f_t has L-Lipschitz gradient. ② holds due to Lemma 1 by setting $\mathbf{g} = \hat{\mathbf{g}}_t$, $\mathbf{u}_t = \mathbf{x}_{t-1}$, $\mathbf{u}_{t+1} = \mathbf{x}_t$, $\mathbf{u}^* = \mathbf{y}_t^*$, and $\lambda = \gamma$. ③ holds, because that Φ is μ-strongly convex, that is, $B_\Phi(\mathbf{x}_t, \mathbf{x}_{t-1}) \geq \frac{\mu}{2}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2$. ④ holds due to $\gamma \leq \frac{\mu}{L}$.

Thus, we have

$$
\begin{aligned}
&\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t) - f_t(\mathbf{y}_t^*) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma\right) - \sum_{t=1}^{T}\|\mathbf{y}_t^* - \mathbf{y}_{t-1}^*\|^\sigma \\
\leq &\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t) - f_t(\mathbf{y}_t^*) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma\right) \\
\overset{①}{\leq} &\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma + \frac{1}{\gamma}\sum_{t=1}^{T}\left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t-1}) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t)\right) \\
= &\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma + \frac{1}{\gamma}\left(B_\Phi(\mathbf{y}_1^*, \mathbf{x}_0) - B_\Phi(\mathbf{y}_T^*, \mathbf{x}_T)\right) + \frac{1}{\gamma}\sum_{t=1}^{T-1}\left(B_\Phi(\mathbf{y}_{t+1}^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t)\right) \\
\overset{②}{\leq} &\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma + \frac{2G}{\gamma}\sum_{t=1}^{T-1}\|\mathbf{y}_{t+1}^* - \mathbf{y}_t^*\| + \frac{1}{\gamma}\left(B_\Phi(\mathbf{y}_1^*, \mathbf{x}_0) - B_\Phi(\mathbf{y}_T^*, \mathbf{x}_T)\right) \\
\leq &\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma + \frac{2G}{\gamma}\sum_{t=1}^{T-1}\|\mathbf{y}_{t+1}^* - \mathbf{y}_t^*\| + \frac{1}{\gamma}B_\Phi(\mathbf{y}_1^*, \mathbf{x}_0) \\
\leq &\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma + \frac{2GD}{\gamma} + \frac{R^2}{\gamma} \\
\overset{③}{\leq} &\left(\frac{2G}{\mu}\right)^\sigma \gamma^\sigma T + \frac{2GD + R^2}{\gamma}.
\end{aligned}
$$

① holds due to (3). ② holds due to

$$B_\Phi(\mathbf{y}_{t+1}^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) \leq 2G\|\mathbf{y}_{t+1}^* - \mathbf{y}_t^*\|$$

according to Lemma 2. ③ holds due to Lemma 3.

Choose $\gamma = \min\{\frac{\mu}{L}, T^{-\frac{1}{1+\sigma}}D^{\frac{1}{1+\sigma}}\}$. We have

$$
\begin{aligned}
&\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t) - f_t(\mathbf{y}_t^*) + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma\right) - \sum_{t=1}^{T}\|\mathbf{y}_t^* - \mathbf{y}_{t-1}^*\|^\sigma \\
\leq &\left(\frac{2G}{\mu}\right)^\sigma T^{\frac{1}{\sigma+1}}D^{\frac{\sigma}{\sigma+1}} + \max\left\{\frac{L(2GD + R^2)}{\mu}, T^{\frac{1}{\sigma+1}}\left(2GD^{\frac{\sigma}{\sigma+1}} + R^2D^{-\frac{1}{\sigma+1}}\right)\right\} \\
\lesssim &T^{\frac{1}{\sigma+1}}D^{\frac{\sigma}{\sigma+1}} + T^{\frac{1}{\sigma+1}}D^{-\frac{1}{\sigma+1}}.
\end{aligned}
$$

Since it holds for any seqence $\{f_t\}_{t=1}^T \in \mathcal{F}^T$, we finally obtain

$$\sup_{\{f_t\}_{t=1}^T \in \mathcal{F}^T} \mathcal{R}_D^{\text{MD-OA}} \lesssim T^{\frac{1}{\sigma+1}}D^{\frac{\sigma}{\sigma+1}} + T^{\frac{1}{\sigma+1}}D^{-\frac{1}{\sigma+1}}.$$

It completes the proof. □

Proof to Theorem 2:

Proof.

$$
\begin{aligned}
& f_t(\mathbf{x}_t) - f_t(\mathbf{y}_t^*) + \|\mathbf{x}_t - \mathbf{x}_{t+1}\|^\sigma - \|\mathbf{y}_t^* - \mathbf{y}_{t+1}^*\|^\sigma \\
\le & \langle \bar{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{y}_t^* \rangle + \|\mathbf{x}_t - \mathbf{x}_{t+1}\|^\sigma \\
= & \langle \bar{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{x}_{t+1} \rangle + \langle \bar{\mathbf{g}}_t, \mathbf{x}_{t+1} - \mathbf{y}_t^* \rangle + \|\mathbf{x}_t - \mathbf{x}_{t+1}\|^\sigma \\
\overset{①}{\le} & \langle \bar{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{x}_{t+1} \rangle + \frac{1}{\eta} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t+1}) - B_\Phi(\mathbf{x}_{t+1}, \mathbf{x}_t) \right) + \|\mathbf{x}_t - \mathbf{x}_{t+1}\|^\sigma \\
\overset{②}{\le} & \langle \bar{\mathbf{g}}_t, \mathbf{x}_t - \mathbf{x}_{t+1} \rangle - \frac{\mu}{2\eta} \|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 + \frac{1}{\eta} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t+1}) \right) + \|\mathbf{x}_t - \mathbf{x}_{t+1}\|^\sigma \\
\overset{③}{\le} & \frac{\eta}{\mu} \|\bar{\mathbf{g}}_t\|^2 + \left(-\frac{\mu}{4\eta} \|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 + \|\mathbf{x}_{t+1} - \mathbf{x}_t\|^\sigma \right) + \frac{1}{\eta} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t+1}) \right) \\
\le & \frac{\eta G^2}{\mu} + \left(-\left(\frac{\sigma}{2}\right)^{\frac{2}{2-\sigma}} \left(\frac{4\eta}{\mu}\right)^{\frac{\sigma}{2-\sigma}} + \left(\frac{\sigma}{2}\right)^{\frac{\sigma}{2-\sigma}} \left(\frac{4\eta}{\mu}\right)^{\frac{\sigma}{2-\sigma}} \right) + \frac{1}{\eta} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t+1}) \right) \\
\le & \frac{\eta G^2}{\mu} + \left(\frac{\sigma}{2}\right)^{\frac{\sigma}{2-\sigma}} \left(\frac{4\eta}{\mu}\right) + \frac{1}{\eta} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t+1}) \right).
\end{aligned}
$$

① holds due to Lemma 1 by setting $\mathbf{g} = \bar{\mathbf{g}}_t$, $\mathbf{u}_t = \mathbf{x}_t$, $\mathbf{u}_{t+1} = \mathbf{x}_{t+1}$, $\mathbf{u}^* = \mathbf{y}_t^*$, and $\lambda = \eta$. ② holds due to Φ is μ-strongly convex. ③ holds, because $\langle \mathbf{u}, \mathbf{v} \rangle \le \frac{a}{2}\|\mathbf{u}\|^2 + \frac{1}{2a}\|\mathbf{v}\|^2$ holds for any $\mathbf{u}$, $\mathbf{v}$, and $a > 0$. The last inequality holds due to $\eta \le \frac{\mu}{4}$ and $1 \le \sigma \le 2$.

Telescoping it over t, we have

$$
\begin{aligned}
& \sum_{t=1}^{T} (f_t(\mathbf{x}_t) - f_t(\mathbf{y}_t^*)) + \sum_{t=1}^{T-1} \left(\|\mathbf{x}_t - \mathbf{x}_{t+1}\|^\sigma - \|\mathbf{y}_t^* - \mathbf{y}_{t+1}^*\|^\sigma \right) \\
\le & \frac{T\eta G^2}{\mu} + \frac{1}{\eta} \sum_{t=1}^{T} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t+1}) \right) + \left(\frac{\sigma}{2}\right)^{\frac{\sigma}{2-\sigma}} \left(\frac{4\eta}{\mu}\right) \\
= & \frac{T\eta G^2}{\mu} + \frac{1}{\eta} \left(\sum_{t=2}^{T} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_{t-1}^*, \mathbf{x}_t) \right) \right) + \frac{1}{\eta} \left(B_\Phi(\mathbf{y}_1^*, \mathbf{x}_1) - B_\Phi(\mathbf{y}_T^*, \mathbf{x}_{T+1}) \right) \\
& + \left(\frac{\sigma}{2}\right)^{\frac{\sigma}{2-\sigma}} \left(\frac{4\eta}{\mu}\right) \\
\le & \frac{T\eta G^2}{\mu} + \frac{1}{\eta} \left(\sum_{t=2}^{T} \left(B_\Phi(\mathbf{y}_t^*, \mathbf{x}_t) - B_\Phi(\mathbf{y}_{t-1}^*, \mathbf{x}_t) \right) \right) + \frac{1}{\eta} B_\Phi(\mathbf{y}_1^*, \mathbf{x}_1) + \left(\frac{\sigma}{2}\right)^{\frac{\sigma}{2-\sigma}} \left(\frac{4\eta}{\mu}\right) \\
\overset{①}{\le} & \frac{T\eta G^2}{\mu} + \frac{2G}{\eta} \sum_{t=1}^{T-1} \|\mathbf{y}_{t+1}^* - \mathbf{y}_t^*\| + \frac{1}{\eta} B_\Phi(\mathbf{y}_1^*, \mathbf{x}_1) + \left(\frac{\sigma}{2}\right)^{\frac{\sigma}{2-\sigma}} \left(\frac{4\eta}{\mu}\right) \\
\le & \frac{T\eta G^2}{\mu} + \frac{2GD}{\eta} + \frac{R^2}{\eta} + \left(\frac{\sigma}{2}\right)^{\frac{\sigma}{2-\sigma}} \left(\frac{4\eta}{\mu}\right) \\
\lesssim & \sqrt{TD} + \sqrt{T}.
\end{aligned}
$$

① holds due to

$$
B_\Phi(\mathbf{y}_{t+1}^*, \mathbf{x}_{t+1}) - B_\Phi(\mathbf{y}_t^*, \mathbf{x}_{t+1}) \le 2G \|\mathbf{y}_{t+1}^* - \mathbf{y}_t^*\|
$$

according to Lemma 2. The last inequality holds by setting $\eta = \min\{\sqrt{\frac{D+G}{T}}, \frac{\mu}{4}\}$.

Since it holds for any seqence of $f_t \in \mathcal{F}$, we finally obtain

$$\sup_{\{f_t\}_{t=1}^T \in \mathcal{F}^T} \mathcal{R}_D^{\text{MD-OCO}} \lesssim \sqrt{TD} + \sqrt{T}.$$

It completes the proof. □

Proof to Theorem 3:

Proof. This proof is inspired by Reference [50], but our new analysis generalizes [50] to the case of switching cost.

Construct the function $f_t(\mathbf{x}_t) = \langle \mathbf{v}_t, \mathbf{x}_t \rangle$ for any $t \in [T]$. Here, $\mathbf{v}_t \in \{1, -1\}^d$, and every element $\mathbf{v}_t(j)$ with $j \in [d]$ is a random variable, which is sampled from a Rademacher distribution independently. For any online method $A \in \mathcal{A}$, its regret is bounded as follows:

$$\begin{aligned}
&\sup_{\{f_t\}_{t=1}^T} \mathcal{R}_D^A \geq \mathcal{R}_D^A \\
&= \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{t=1}^T f_t(\mathbf{x}_t) + \sum_{t=1}^T \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma - \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \min_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \left(\sum_{t=1}^T f_t(\mathbf{y}_t) + \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \right) \\
&= \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \left(\sum_{t=1}^T f_t(\mathbf{x}_t) + \sum_{t=1}^T \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma \right) + \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \left(- \sum_{t=1}^T f_t(\mathbf{y}_t) - \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \right) \\
&= \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \left(f_t(\mathbf{x}_t) - f_t(\mathbf{y}_t) - \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \right) + \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{t=1}^T \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma \\
&= \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \left(\langle \mathbf{v}_t, \mathbf{x}_t - \mathbf{y}_t \rangle - \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \right) + \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{t=1}^T \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma .
\end{aligned} \tag{4}$$

For any optimal sequence of $\{\mathbf{y}_t^*\}_{t=1}^T$,

$$\mathop{\mathbb{E}}_{\mathbf{v}_t} \langle \mathbf{v}_t, \mathbf{x}_{t-1} - \mathbf{y}_{t-1}^* \rangle = \left\langle \mathop{\mathbb{E}}_{\mathbf{v}_t} \mathbf{v}_t, \mathbf{x}_{t-1} - \mathbf{y}_{t-1}^* \right\rangle = \langle \mathbf{0}, \mathbf{x}_{t-1} - \mathbf{y}_{t-1}^* \rangle = 0.$$

Thus, for any optimal sequence of $\{\mathbf{y}_t^*\}_{t=1}^T$, we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \left(\langle \mathbf{v}_t, \mathbf{x}_t - \mathbf{y}_t \rangle - \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \right) \\
&= \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \left(\sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{x}_t - \mathbf{y}_t^* \rangle - \sum_{t=1}^T \|\mathbf{y}_t^* - \mathbf{y}_{t-1}^*\|^\sigma \right) \\
&= \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{x}_t - \mathbf{x}_{t-1} + \mathbf{y}_{t-1}^* - \mathbf{y}_t^* \rangle - \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}^*\|^\sigma \\
&= \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{x}_t - \mathbf{x}_{t-1} \rangle + \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \left(\sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{y}_{t-1}^* - \mathbf{y}_t^* \rangle - \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}^*\|^\sigma \right) \\
&= \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{x}_t - \mathbf{x}_{t-1} \rangle + \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \left(\sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{y}_{t-1} - \mathbf{y}_t \rangle - \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \right)
\end{aligned}$$

Substituting it into Equation (6), we have

$$
\begin{aligned}
&\sup_{\{f_t\}_{t=1}^T} \mathcal{R}_D^{\mathcal{A}} \\
\geq\ & \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \left(\sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{x}_t - \mathbf{x}_{t-1} \rangle + \sum_{t=1}^T \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma \right) \\
&+ \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \left(\sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{y}_{t-1} - \mathbf{y}_t \rangle - \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \right) \\
\overset{①}{\geq}\ & \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \left(\sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{y}_{t-1} - \mathbf{y}_t \rangle - \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \right) \\
\geq\ & \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{y}_{t-1} - \mathbf{y}_t \rangle - \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \\
\overset{②}{\geq}\ & \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{y}_{t-1} - \mathbf{y}_t \rangle - D^\sigma \\
\overset{③}{=}\ & \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \langle \mathbf{v}_t, -\mathbf{y}_t \rangle - D^\sigma \\
\overset{④}{=}\ & \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \langle \mathbf{v}_t, \mathbf{y}_t \rangle - D^\sigma.
\end{aligned}
$$

① holds due to

$$
\mathop{\mathbb{E}}_{\mathbf{v}_t} \left(\langle \mathbf{v}_t, \mathbf{x}_t - \mathbf{x}_{t-1} \rangle + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma \right) = \left\langle \mathop{\mathbb{E}}_{\mathbf{v}_t} \mathbf{v}_t, \mathbf{x}_t - \mathbf{x}_{t-1} \right\rangle + \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma = \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^\sigma \geq 0.
$$

② holds, because, for any sequence $\{\mathbf{y}_t\}_{t=1}^T$, $\sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\| \leq D$. Thus,

$$
\max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|^\sigma \leq \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \left(\sum_{t=1}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\| \right)^\sigma \leq D^\sigma.
$$

③ holds, because, for any vector $\mathbf{y}_{t-1}$,

$$
\mathop{\mathbb{E}}_{\mathbf{v}_t} \langle \mathbf{v}_t, \mathbf{y}_{t-1} \rangle = \left\langle \mathop{\mathbb{E}}_{\mathbf{v}_t} \mathbf{v}_t, \mathbf{y}_{t-1} \right\rangle = \langle \mathbf{0}, \mathbf{y}_{t-1} \rangle = 0.
$$

④ holds, because the domain of $\mathbf{v}_t$ is symmetric.

Furthermore, we construct a sequence $\{\mathbf{y}_t\}_{t=1}^T$ as follows.

(1) Evenly split $\{\mathbf{y}_t\}_{t=1}^T$ into two subsets: $\{\mathbf{y}_t\}_{t=1}^{T_1}$ and $\{\mathbf{y}_{T_1+t}\}_{t=1}^{T_2}$. Here, $T_1 = T_2 = \frac{T}{2}$.

(2) After that, evenly split $\{\mathbf{y}_t\}_{t=1}^{T_1}$ into $N := \min\{\frac{D}{R}, T_1\}$ subsets, that is, $\{\mathbf{y}_t\}_{t=1}^{\frac{T_1}{N}}$, $\{\mathbf{y}_t\}_{t=\frac{T_1}{N}+1}^{\frac{2T_1}{N}}$, $\{\mathbf{y}_t\}_{t=\frac{2T_1}{N}+1}^{\frac{3T_1}{N}}, \ldots, \{\mathbf{y}_t\}_{t=\frac{(N-1)T_1}{N}+1}^{T_1}$.

(3) For the ith subset of the sequence $\{\mathbf{y}_t\}_{t=1}^{T_1}$, let the values in it be same, and denote it by $\mathbf{u}_i$ with $\|\mathbf{u}_i\| \leq \frac{R}{2}$. For the whole sequence $\{\mathbf{y}_{T_1+t}\}_{t=1}^{T_2}$, let all the values be same, namely $\mathbf{u}_N$.

(4) For the sequence of $\{\mathbf{y}_t\}_{t=1}^{T_1}$, elements in different subsets are different such that $\|\mathbf{u}_{i+1} - \mathbf{u}_i\| \leq \|\mathbf{u}_{i+1}\| + \|\mathbf{u}_i\| \leq R$. Thus,

$$\begin{aligned}
&\sum_{t=1}^{T-1} \|\mathbf{y}_{t+1} - \mathbf{y}_t\| = \sum_{t=1}^{T_1-1} \|\mathbf{y}_{t+1} - \mathbf{y}_t\| + \sum_{t=T_1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t\| \\
=& \sum_{i=1}^{N-1} \|\mathbf{u}_{i+1} - \mathbf{u}_i\| + 0 \\
\leq& (N-1)R \\
\leq& D.
\end{aligned}$$

The last inequality holds due to $(N-1)R \leq D$. It implies that $\{\mathbf{y}_t\}_{t=1}^{T}$ under our construction is feasible.

Then, we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \sum_{t=1}^{T} \langle \mathbf{v}_t, \mathbf{y}_t \rangle = \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{L}_D^T} \left(\sum_{t=1}^{T_1} \langle \mathbf{v}_t, \mathbf{y}_t \rangle + \sum_{t=T_1+1}^{T} \langle \mathbf{v}_t, \mathbf{y}_t \rangle \right) \\
=& \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{i=1}^{N} \max_{\|\mathbf{u}_i\| \leq \frac{R}{2}} \left\langle \sum_{t=1+\frac{T(i-1)}{N}}^{\frac{Ti}{N}} \mathbf{v}_t, \mathbf{u}_i \right\rangle + \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\|\mathbf{u}_N\| \leq \frac{R}{2}} \left\langle \sum_{t=T_1+1}^{T} \mathbf{v}_t, \mathbf{u}_N \right\rangle \\
\overset{①}{=}& \frac{R}{2} \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{i=1}^{N} \left\| \sum_{t=1+\frac{T(i-1)}{N}}^{\frac{Ti}{N}} \mathbf{v}_t \right\| + \frac{R}{2} \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \left\| \sum_{t=T_1+1}^{T} \mathbf{v}_t \right\| \\
\overset{②}{\geq}& \frac{R}{2\sqrt{d}} \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{i=1}^{N} \sum_{j=1}^{d} \left| \sum_{t=1+\frac{T(i-1)}{N}}^{\frac{Ti}{N}} \mathbf{v}_t(j) \right| + \frac{R}{2\sqrt{d}} \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \sum_{j=1}^{d} \left| \sum_{t=T_1+1}^{T} \mathbf{v}_t(j) \right| \\
\overset{③}{=}& \frac{\sqrt{d}NR}{2} \cdot \Omega\left(\sqrt{\frac{T}{N}}\right) + \frac{R\sqrt{d}}{2} \cdot \Omega\left(\sqrt{\frac{T}{2}}\right) \\
=& \Omega\left(\sqrt{R}\sqrt{TNR} + \sqrt{T}\right) \\
\overset{④}{=}& \Omega\left(\sqrt{TD} + \sqrt{T}\right).
\end{aligned}$$

① holds, because the maximum is obtained at the boundary of the domain. ② holds, because, for any $\mathbf{v} \in \mathbb{R}^d$, $\|\mathbf{v}\|_1 \leq \sqrt{d}\,\|\mathbf{v}\|_2$. ③ holds due to a classic result [23], that is,

$$\mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \left| \sum_{t=1+\frac{T(i-1)}{N}}^{\frac{Ti}{N}} \mathbf{v}_t(j) \right| = \Omega\left(\sqrt{\frac{T}{N}}\right).$$

④ holds due to $D - R \leq NR \leq D + R$, which implies that $NR \lesssim D$ holds for $D > 0$.

Therefore, we obtain

$$\sup_{\{f_t\}_{t=1}^T} \mathcal{R}_D^A \geq \mathop{\mathbb{E}}_{\mathbf{v}_{1:T}} \max_{\{\mathbf{y}_t\}_{t=1}^T \in \mathcal{X}^T} \sum_{t=1}^{T} \langle \mathbf{v}_t, \mathbf{y}_t \rangle - D^\sigma = \Omega\left(\sqrt{TD} + \sqrt{T}\right).$$

The last equality holds, because D^σ is a constant, and it does not increase over T.

Since it holds for any online algorithm $A \in \mathcal{A}$, we finally have

$$\inf_{A \in \mathcal{A}} \sup_{\{f_t\}_{t=1}^T \in \mathcal{F}^T} = \Omega\left(\sqrt{TD} + \sqrt{T}\right).$$

It completes the proof. □

REFERENCES

[1] Jacob Abernethy, Peter L. Bartlett, Niv Buchbinder, and Isabelle Stanton. 2010. A regularization approach to metrical task systems. In *Proceedings of the 21st International Conference on Algorithmic Learning Theory (ALT'10)*. Springer-Verlag, Berlin, 270–284.

[2] Lachlan Andrew, Siddharth Barman, Katrina Ligett, Minghong Lin, Adam Meyerson, Alan Roytman, and Adam Wierman. 2013. A tale of two metrics: Simultaneous bounds on competitiveness and regret. In *Proceedings of the ACM International Conference on Measurement and Modeling of Computer Systems*. 329–330.

[3] Antonios Antoniadis, Kevin Schewior, and Rudolf Fleischer. 2018. A tight lower bound for online convex optimization with switching costs. In *Approximation and Online Algorithms*. Springer International Publishing, Cham, 164–175.

[4] Nikhil Bansal, Niv Buchbinder, and Joseph Naor. 2010. Metrical task systems and the K-server problem on HSTs. In *Proceedings of the 37th International Colloquium Conference on Automata, Languages and Programming*.

[5] Amir Beck and Marc Teboulle. 2003. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operat. Res. Lett.* 31, 3 (2003), 167–175.

[6] Andrey Bernstein, Shie Mannor, and Nahum Shimkin. 2010. Online classification with specificity constraints. In *Proceedings of Advances in Neural Information Processing Systems (NIPS'10)*, J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta (Eds.). 190–198.

[7] Omar Besbes, Yonatan Gur, and Assaf J. Zeevi. 2015. Non-stationary stochastic optimization. *Operations Research* 63, 5 (2015), 1227–1244.

[8] Avrim Blum and Carl Burch. 2000. On-line learning and the metrical task system problem. *Mach. Learn.* 39, 1 (Apr. 2000), 35–58.

[9] Sébastien Bubeck. 2015. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning* 8, 3 (Nov 2015), 231–357.

[10] Sébastien Bubeck, Michael B. Cohen, James R. Lee, and Yin Tat Lee. 2019. Metrical task systems on trees via mirror descent and unfair gluing. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA'19)*.

[11] Sébastien Bubeck, Michael B. Cohen, Yin Tat Lee, James R. Lee, and Aleksander Mkadry. 2018. K-server via multiscale entropic regularization. In *Proceedings of the 50th Annual ACM Symposium on Theory of Computing (STOC'18)*. ACM, New York, NY, 3–16.

[12] Niv Buchbinder, Shahar Chen, Joshep (Seffi) Naor, and Ohad Shamir. 2012. Unified algorithms for online learning and competitive analysis. In *Proceedings of the 25th Annual Conference on Learning Theory (COLT'12)*, Shie Mannor, Nathan Srebro, and Robert C. Williamson (Eds.), Vol. 23, 5.1–5.18.

[13] Niangjun Chen, Anish Agarwal, Adam Wierman, Siddharth Barman, and Lachlan L. H. Andrew. 2015. Online convex optimization using predictions. In *Proceedings of the ACM International Conference on Measurement and Modeling of Computer Systems*. 191–204.

[14] Niangjun Chen, Joshua Comden, Zhenhua Liu, Anshul Gandhi, and Adam Wierman. 2016. Using predictions in online optimization: Looking forward with an eye on the past. In *Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science*. 193–206.

[15] Niangjun Chen, Gautam Goel, and Adam Wierman. 2018. Smoothed online convex optimization in high dimensions via online balanced descent. In *Proceedings of the 31st Conference On Learning Theory (COLT'18)*, Vol. 75. 1574–1594.

[16] Tianyi Chen, Qing Ling, and Georgios B. Giannakis. 2017. An online convex optimization approach to proactive network resource allocation. *IEEE Trans. Sign. Process.* 65, 24 (2017), 6350–6364.

[17] Chao Kai Chiang, Tianbao Yang, Chia Jung Lee, Mehrdad Mahdavi, Chi Jen Lu, Rong Jin, and Shenghuo Zhu. 2012. Online optimization with gradual variations. In *Proceedings of the 25th Annual Conference on Learning Theory (COLT'12)*. 1–20.

[18] Koby Crammer, Jaz Kandola, and Yoram Singer. 2004. Online classification on a budget. In *Proceedings of Advances in Neural Information Processing Systems (NIPS'04)*. 225–232.

[19] Xiand Gao, Xiaobo Li, and Shuzhong Zhang. 2018. Online learning with non-convex losses and non-stationary regret. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics (AISTATS'18)*, Amos Storkey and Fernando Perez-Cruz (Eds.), Vol. 84. 235–243.

[20] András György and Csaba Szepesvári. 2016. Shifting regret, mirror descent, and matrices. In *Proceedings of the 33rd International Conference on Machine Learning (ICML'16)*. 2943–2951.

[21] Eric C. Hall and Rebecca Willett. 2013. Dynamical models and tracking regret in online convex programming. In *Proceedings of International Conference on International Conference on Machine Learning (ICML'13)*.
[22] Eric C. Hall and Rebecca M. Willett. 2015. Online convex optimization in dynamic environments. *IEEE J. Select. Top. Sign. Process.* 9, 4 (2015), 647–662.
[23] Elad Hazan. 2016. Introduction to online convex optimization. *Found. Trends Optimiz.* 2, 3–4 (2016), 157–325.
[24] Ali Jadbabaie, Alexander Rakhlin, Shahin Shahrampour, and Karthik Sridharan. 2015. Online optimization: Competing with dynamic comparators. In *Proceedings of International Conference on Artificial Intelligence and Statistics (AISTATS'15)*. 398–406.
[25] Rodolphe Jenatton, Jim Huang, and Cedric Archambeau. 2016. Adaptive algorithms for online convex optimization with long-term constraints. In *Proceedings of the 33rd International Conference on Machine Learning (ICML'16)*, Vol. 48. 402–411.
[26] James R. Lee. 2018. Fusible HSTs and the randomized k-server conjecture.. In *Proceedings of the IEEE 59th Annual Symposium on Foundations of Computer Science*.
[27] Bin Li and Steven C. H. Hoi. 2014. Online portfolio selection: A survey. *Comput. Surv.* 46, 3 (2014), 35:1–35:36.
[28] Bin Li, Steven C. H. Hoi, Peilin Zhao, and Vivekanand Gopalkrishnan. 2013. Confidence weighted mean reversion strategy for online portfolio selection. *ACM Trans. Knowl. Discov. Data* 7, 1 (Mar. 2013), 4:1–4:38.
[29] C. Li, P. Zhou, L. Xiong, Q. Wang, and T. Wang. 2018. Differentially private distributed online learning. *IEEE Trans. Knowl. Data Eng.* 30, 8 (Aug. 2018), 1440–1453.
[30] Yingying Li, Guannan Qu, and Na Li. 2018. Online optimization with predictions and switching costs: Fast algorithms and the fundamental limit. In *Proceedings of the Annual American Control Conference (ACC'18)*. 3008–3013.
[31] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska. 2011. Dynamic right-sizing for power-proportional data centers. In *Proceedings of IEEE International Conference on Computer Communications (INFOCOMM'11)*. 1098–1106.
[32] Minghong Lin, Adam Wierman, Alan Roytman, Adam Meyerson, and Lachlan L. H. Andrew. 2012. Online optimization with switching cost. *SIGMETRICS Perf. Eval. Rev.* 40, 3 (2012), 98–100.
[33] T. Lu, M. Chen, and L. L. H. Andrew. 2013. Simple and effective dynamic provisioning for power-proportional data centers. *IEEE Trans. Parallel Distrib. Syst.* 24, 6 (Jun. 2013), 1161–1171.
[34] Aryan Mokhtari, Shahin Shahrampour, Ali Jadbabaie, and Alejandro Ribeiro. 2016. Online optimization in dynamic environments: Improved regret rates for strongly convex problems. In *Proceedings of IEEE Conference on Decision and Control (CDC'16)*. IEEE, 7195–7201.
[35] Reetabrata Mookherjee, Benjamin F. Hobbs, Terry Lee Friesz, and Matthew A. Rigdon. 2008. Dynamic oligopolistic competition on an electric power network with ramping costs and joint sales constraints. *J. Industr. Manage. Optimiz.* 4, 3 (11 2008), 425–452.
[36] Manfred Morari and Jay H. Lee. 1999. Model predictive control: Past, present and future. *Comput. Chem. Eng.* 23, 4 (1999), 667–682.
[37] Marc P. Renault and Adi Rosén. 2012. On online algorithms with advice for the k-server problem. In *Approximation and Online Algorithms*, Roberto Solis-Oba and Giuseppe Persiano (Eds.). Springer, Berlin, 198–210.
[38] Shai Shalev-Shwartz. 2012. Online learning and online convex optimization. *Found. Trends Mach. Learn.* 4, 2 (2012), 107–194.
[39] Y. Sun, K. Tang, L. L. Minku, S. Wang, and X. Yao. 2016. Online ensemble learning of data streams with gradually evolved classes. *IEEE Trans. Knowl. Data Eng.* 28, 6 (Jun. 2016), 1532–1545.
[40] Hao Wang, Jianwei Huang, Xiaojun Lin, and Hamed Mohsenian-Rad. 2014. Exploring smart grid and data center interactions for electric power load balancing. *SIGMETRICS Perf. Eval. Rev.* 41, 3 (Jan. 2014), 89–94.
[41] Liang Wang, Kuang-chih Lee, and Quan Lu. 2016. Improving advertisement recommendation by enriching user browser cookie attributes. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM'16)*. 2401–2404.
[42] M. Wang, C. Xu, X. Chen, H. Hao, L. Zhong, and S. Yu. 2019. Differential privacy oriented distributed online learning for mobile social video prefetching. *IEEE Trans. Multimedia* 21, 3 (Mar. 2019), 636–651.
[43] Haiqin Yang, Michael R. Lyu, and Irwin King. 2013. Efficient online learning for multitask feature selection. *ACM Trans. Knowl. Discov. Data* 7, 2 (Aug. 2013), 6:1–6:27.
[44] Tianbao Yang, Lijun Zhang, Rong Jin, and Jinfeng Yi. 2016. Tracking slowly moving clairvoyant—Optimal dynamic regret of online learning with true and noisy gradient. In *Proceedings of the 34th International Conference on Machine Learning (ICML'16)*.
[45] Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. 2018. Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.). 1323–1333.
[46] Lijun Zhang, Tianbao Yang, rong jin, and Zhi-Hua Zhou. 2018. Dynamic regret of strongly adaptive methods. In *Proceedings of the 35th International Conference on Machine Learning (ICML'18)*. 5882–5891.

[47] Lijun Zhang, Tianbao Yang, Jinfeng Yi, Rong Jin, and Zhi-Hua Zhou. 2017. Improved dynamic regret for non-degenerate functions. In *Proceedings of the Conference on Neural Information Processing Systems (NIPS'17)*.
[48] Lijun Zhang, Tianbao Yangt, Jinfeng Yi, Rong Jin, and Zhi-Hua Zhou. 2017. Improved dynamic regret for non-degenerate functions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 732–741.
[49] Q. Zhang, Q. Zhu, M. F. Zhani, and R. Boutaba. 2012. Dynamic service placement in geographically distributed clouds. In *Proceedings of the IEEE 32nd International Conference on Distributed Computing Systems (ICDCS'12)*. 526–535.
[50] Yawei Zhao, Shuang Qiu, and Ji Liu. 2018. Proximal online gradient is optimum for dynamic regret. *CoRR* cs.LG (Oct. 2018). *arXiv:cs.LG/1810.03594*.
[51] Martin Zinkevich. 2003. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of International Conference on Machine Learning (ICML'03)*. 928–935.

Received June 2019; revised November 2019; accepted December 2019