Online Bilevel Optimization: Regret Analysis of Online Alternating Gradient Methods

Davoud Ataee Tarzanagh
University of Pennsylvania

Parvin Nazari
Amirkabir University of Technology

Bojian Hou
University of Pennsylvania

Li Shen
University of Pennsylvania

Laura Balzano
University of Michigan

Abstract

This paper introduces *online bilevel optimization* in which a sequence of time-varying bilevel problems is revealed one after the other. We extend the known regret bounds for single-level online algorithms to the bilevel setting. Specifically, we provide new notions of *bilevel regret*, develop an online alternating time-averaged gradient method that is capable of leveraging smoothness, and give regret bounds in terms of the path-length of the inner and outer minimizer sequences.

1 Introduction

Bilevel optimization (BO) is rapidly evolving due to its wide array of applications in modern machine learning problems, including meta-learning [Bertinetto et al., 2018], hyperparameter optimization [Feurer and Hutter, 2019], neural network architecture search [Liu et al., 2018], data hypercleaning [Shaban et al., 2019], and reinforcement learning [Wu et al., 2020]. A fundamental assumption in BO, which has been adopted by almost all of the relevant literature [Franceschi et al., 2017, Ghadimi and Wang, 2018, Ji et al., 2021b, Nazari et al., 2022a], is that the *inner* and *outer* cost functions do *not* change throughout the horizon over which we seek to optimize. This offline setting may not be suitable to model temporal changes in today's machine learning problems, such as online actor-critic [Vamvoudakis and Lewis, 2010, Zhou et al., 2020a], online meta-learning [Finn et al., 2019], strategic dynamic regression [Harris et al., 2021], and sequential decision-making, for which the objective functions are time-varying and are not available to the decision-maker *a priori*. To address these challenges, this paper introduces an *online bilevel optimization* (OBO) setting in which a sequence of bilevel problems is revealed one after the other, and it studies computationally tractable notions of *bilevel regret* minimization.

1.1 Background: Online Single-Level Optimization

In online single-level optimization, the setup resembles a game between a learner and an adversary [Hazan, 2016a]. In each of the repeated decision rounds ($t \in [T] := 1, \ldots, T$), the learner predicts $\mathbf{x}_t \in \mathcal{X} \subset \mathbb{R}^{d_1}$, an element within a convex decision set. Simultaneously, the adversary selects a loss function $f_t : \mathcal{X} \to \mathbb{R}$, and the learner observes $f_t(\mathbf{x})$, incurring a loss of $f_t(\mathbf{x}_t)$. In the *non-static* setting [Besbes et al., 2015], the learner's performance is measured through its single-level dynamic regret

$$\text{D-Reg}_T := \sum_{t=1}^{T} (f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^*)), \tag{1}$$

where $\mathbf{x}_t^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x})$.

In the case of static regret [Zinkevich, 2003], $\mathbf{x}_t^*$ is replaced by $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$, i.e.,

$$\text{S-Reg}_T := \sum_{t=1}^{T} (f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)). \tag{2}$$

The static regret (2) assumes that the comparators do not change over time. This assumption can be unrealistic in many practical online problems, ranging from motion imagery formation to network analysis, where the underlying environment is dynamic. The parameters $\{\mathbf{x}_t^*\}_{t=1}^{T}$ could correspond to frames in a video or the weights of edges in a social network and, by nature, are variable [Hall and Willett, 2015].

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. Copyright 2024 by the author(s).

1.2 Stackelberg Game and Online Bilevel Optimization

Bilevel optimization, also known as the Stackelberg leader-follower model, involves two players whose choices impact each other's outcomes. One player, the *leader*, possesses knowledge of the other player's objective function, enabling her to predict the follower's choice accurately. Consequently, the leader optimizes her own objective while factoring in the follower's expected response. In contrast, the *follower* is only aware of her own objective and must consider how it is influenced by the leader's decisions [von Stackelberg, 1952].

Online Bilevel Optimization: Let $\mathbf{x}_t \in \mathcal{X} \subset \mathbb{R}^{d_1}$ and $f_t : \mathcal{X} \times \mathbb{R}^{d_2} \to \mathbb{R}$ denote the decision variable and the objective function for the leader, respectively; similarly define $\mathbf{y}_t \in \mathbb{R}^{d_2}$ and $g_t : \mathcal{X} \times \mathbb{R}^{d_2} \to \mathbb{R}$ for the follower [1]. In each round $t \in [T]$, knowing the decision $\mathbf{x}_{t-1}$ of the leader and the objective function g_{t-1} of the follower, the follower has to select $\mathbf{y}_t \in \mathbb{R}^{d_2}$ in an attempt to minimize $g_t(\mathbf{x}_t, \mathbf{y})$ using the information from rounds $t-1, t-2, \ldots, 0$. Being aware of the follower's selection, the leader then moves by selecting $\mathbf{x}_t \in \mathcal{X}$ in an attempt to minimize the *bilevel dynamic regret*, defined as:

$$\text{BD-Reg}_T := \sum_{t=1}^{T} (f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))), \quad (3a)$$

where

$$\begin{aligned} \mathbf{y}_t^*(\mathbf{x}) &\in \arg\min_{\mathbf{y}\in\mathbb{R}^{d_2}} g_t(\mathbf{x}, \mathbf{y}), \quad \text{and} \\ \mathbf{x}_t^* &\in \arg\min_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})). \end{aligned} \quad (3b)$$

Our objective is to design online algorithms with a sublinear bilevel regret, i.e., $\text{BD-Reg}_T = o(T)$.

We also study the framework of regret minimization where $\mathbf{x}_t^*$ in (3) is replaced by $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$. In this case, the goal of the leader is to generate a sequence of decisions $\{\mathbf{x}_t\}_{t=1}^{T}$ so that the following regret can be minimized:

$$\text{BS-Reg}_T := \sum_{t=1}^{T} \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}^*, \mathbf{y}_t^*(\mathbf{x}^*))\right). \quad (4)$$

Note that the above regret is *not* fully static, as the inner optima $\{\mathbf{y}_t^*\}_{t=1}^{T}$ are changing over T.

In general, it is impossible to achieve a sublinear dynamic regret bound due to the arbitrary fluctuations in the time-varying functions [Besbes et al., 2015]. Existing single-level analysis shows that it is indeed possible to bound the dynamic regret in terms of certain regularities of the comparator sequence [Zinkevich, 2003]. Hence, in order to achieve sublinear regret, one has to impose some regularity constraints on the sequence of cost functions. In this work, we define the outer and inner *path-length* (of order p) quantities to capture the regularity of the sequences:

$$\begin{aligned} P_{p,T} &:= \sum_{t=2}^{T} \|\mathbf{x}_{t-1}^* - \mathbf{x}_t^*\|^p, \quad \text{and} \\ Y_{p,T} &:= \sum_{t=2}^{T} \left\|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*)\right\|^p. \end{aligned} \quad (5a)$$

Here, $P_{p,T}$ is the path-length of the outer minimizers and is widely used for analyzing the dynamic regret of single-level non-stationary optimization; see Table 1. $Y_{p,T}$ is a new regularity metric for OBO that measures how fast the minimizers of inner cost functions change. For simplicity of notation, we set

$$S_{p,T} := P_{p,T} + Y_{p,T}. \quad (5b)$$

Besides (5a), other notions of regularity have also been considered in online learning such as function variation [Besbes et al., 2015] $V_T := \sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} |f_{t-1}(\mathbf{x}) - f_t(\mathbf{x})|$ and gradient variation [Chiang et al., 2012] $G_T := \sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} \|\nabla f_{t-1}(\mathbf{x}) - \nabla f_t(\mathbf{x})\|^2$. For the leader's regret analysis in the static and local settings, we respectively define

$$\begin{aligned} \bar{Y}_{p,T} &:= \sum_{t=2}^{T} \left\|\mathbf{y}_{t-1}^*(\mathbf{x}^*) - \mathbf{y}_t^*(\mathbf{x}^*)\right\|^p, \quad \text{and} \\ H_{p,T} &:= \sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathbb{R}^{d_1}} \|\mathbf{y}_{t-1}^*(\mathbf{x}) - \mathbf{y}_t^*(\mathbf{x})\|^p \end{aligned} \quad (6)$$

for capturing the dynamics of the inner problem.

The metrics in (5) and (6) are generally not comparable. Appendix A.1 demonstrates their significant differences across various online problems.

Our Results. Our main contributions lie in developing several new results for OBO, including the first-known regret bound. More specifically, we

- Define new notions of bilevel regret, as given in (3) and (4), which are applicable to a wide class of convex OBO problems. To minimize the proposed regret, we introduce an online alternating gradient descent (OAGD) method capable of leveraging smoothness and provide its regret bounds in terms of the path-length of the inner and/or outer minimizer sequences.

- Present a problem-dependent regret bound on the proposed dynamic regret that depends solely on the outer and inner path-length in Theorem 4. We then

[1] For simplicity of analysis, we use $\mathbb{R}^{d_2}$ as the follower's decision set.

Single-Level Regret Minimization

		Regret
SC-D	Mokhtari et al. [2016]	$\mathcal{O}(1 + P_{1,T})$
	Zhang et al. [2017]	$\mathcal{O}\left(1 + \min\{P_{1,T}, P_{2,T}\}\right)$
SC-S	Hazan et al. [2007]	$\mathcal{O}\left(\log T\right)$
C-D	Besbes et al. [2015]	$\mathcal{O}(1 + G_T^{1/2} P_{1,T}^{1/2})$
	Jadbabaie et al. [2015]	$\mathcal{O}(1 + T^{2/3} {V_T}^{1/3})$
	Yang et al. [2016]	$\mathcal{O}\left(1 + P_{1,T}\right)$
C-S	Zinkevich [2003]	$\mathcal{O}(\sqrt{T})$
NC-L	Hazan et al. [2017]	$\mathcal{O}(T/W^2)$

Bilevel Regret Minimization

		Leader's Regret
SC-D	Theorem 4	$\mathcal{O}\left(1 + \min\{S_{1,T}, S_{2,T}\}\right)$
SC-S	Theorem 6	$\mathcal{O}\left(\log T + \bar{Y}_{2,T}\right)$
C-D	Theorem 7	$\mathcal{O}\left(1 + S_{1,T} + Y_{2,T}\right)$
C-S	Theorem 8	$\mathcal{O}(\sqrt{T} + \bar{Y}_{1,T} + \bar{Y}_{2,T})$
NC-L	Theorem 9	$\mathcal{O}(T/W + H_{1,T} + H_{2,T})$

Table 1: Comparison with prior works on **regret minimization**. Here, $W = \sum_{i=0}^{w-1} u_i$ for $\{u_i\}_{i=0}^{w-1}$ with $1 = u_0 \geq u_1 \ldots u_{w-1} > 0$ (Def. 1); SC-D (-S), C-D(-S), and NC-L denote strongly convex-dynamic (-static), convex-dynamic (-static), and non-convex-local settings, respectively.

establish a lower bound for OBO in Theorem 5 that matches the upper bound we obtain for smooth strongly convex functions. Notably, our bound in the single-level setting ($Y_{1,T} = Y_{2,T} = 0$) aligns with the state-of-the-art result in [Zhang et al., 2017], *without* the need for multiple gradient queries (multiple updates to $\mathbf{x}_t$) as used in their analysis.

• Introduce a novel notion of bilevel local regret, which permits efficient OBO in the non-convex setting. We give an alternating time-averaged gradient method, and prove in Theorem 9 that it achieves sublinear regret according to our proposed bilevel local regret.

Notation. Any notation is defined upon its use and is summarized in Table 2 for reference.

2 Related Work

Static Regret Minimization: *Single-level* static regret (Eq. (2)) is well-studied in the literature of online learning [Shalev-Shwartz et al., 2011]. Zinkevich [2003] shows that online gradient descent (OGD) provides an $\mathcal{O}(\sqrt{T})$ regret bound for convex functions $\{f_t\}_{t=1}^T$. Hazan et al. [2007] improve this bound to $\mathcal{O}(\log T)$ for strongly-convex functions $\{f_t\}_{t=1}^T$.

Dynamic Regret Minimization: *Single-level* dynamic regret forces the player to compete with time-varying comparators and is thus particularly favored in non-stationary environments [Besbes et al., 2015]. There are two kinds of dynamic regret in previous studies: *universal dynamic regret* aims to compare with any feasible comparator sequence [Zinkevich, 2003], while *worst-case dynamic regret* (defined in (1)) specifies the comparator sequence to be the sequence of minimizers of online functions [Besbes et al., 2015]. We compare regret bounds from related works for the latter case in Table 1, as it is the setting studied in this paper.

Local Regret Minimization: There are several approaches to treat online single-level non-convex optimization, including adversarial multi-armed bandit with a continuum of arms [Bubeck et al., 2008, Héliou et al., 2020], and the classical Follow-the-Perturbed-Leader (FPL) algorithm with access to an offline non-convex optimization oracle [Agarwal et al., 2019, Suggala and Netrapalli, 2020]. Complementing this literature, [Hazan et al., 2017] considered a local regret that averages a sliding window of gradients at the current model $\mathbf{x}_t$ and quantifies the objective of predicting points with small gradients on average.

Bilevel Optimization: Since its first formulation by [von Stackelberg, 1952] and the initial mathematical model by [Bracken and McGill, 1973], there has been a steady growth in investigations and applications of offline BO [Liu et al., 2021]. Recently, gradient-based approaches have become popular for their simplicity and efficacy [Franceschi et al., 2017, Ghadimi and Wang, 2018, Ji et al., 2021b, Chen et al., 2021], yet they assume a static cost function, a limitation we overcome by exploring new bilevel optimization algorithms in the online setting. Since our initial submission, several OBO studies have emerged [Lin et al., 2024, Huang et al., 2023], with Lin et al. [2024] introducing an OBO method that updates $\mathbf{x}_t$ based on an average of recent hypergradient estimates, enabling scalable OBO through an approximate Hessian-inverse vector product by solving a linear system.

3 Algorithm and Regret Bounds

In this section, we provide bilevel regret bounds based on the regularities defined in (5) and (6). We first list assumptions for OBO.

Assumption A. *For all $t \in [T]$:*

A1. *f_t is $\ell_{f,0}$-Lipschitz continuous.*

A2. *$g_t(\mathbf{x}, \mathbf{y})$ is μ_g-strongly convex in $\mathbf{y}$ for any $\mathbf{x} \in \mathcal{X}$.*

A3. *$\nabla f_t, \nabla g_t$, and $\nabla^2 g_t$ are respectively $\ell_{f,1}, \ell_{g,1}$, and $\ell_{g,2}$-Lipschitz continuous.*

Assumption A necessitates well-behaved $\{(f_t, g_t)\}_{t=1}^T$, typical in offline BO [Chen et al., 2021]. Throughout, we use $\kappa_g := \ell_{g,1}/\mu_g$ to denote the condition number of online inner functions $\{g_t\}_{t=1}^T$.

Assumption B. *The non-empty closed and convex decision set $\mathcal{X} \subseteq \mathbb{R}^{d_1}$ is bounded, i.e., $\|\mathbf{x} - \mathbf{x}'\| \leq D$ for any $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ and some $D > 0$. Further, $\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| \leq D'$ for some $D' > 0$.*

Assumption B is similar to the existing assumptions on the decision set in single-level online learning [Hazan, 2016a, Zinkevich, 2003].

3.1 OBO with (Hyper-)Gradient Information

Perhaps the simplest algorithm that applies to the most general setting of online (single-level) optimization is OGD [Zinkevich, 2003]: For each $t \in [T]$, play $\mathbf{x}_t \in \mathcal{X}$, observe the function f_t, and set

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}\big(\mathbf{x}_t - \alpha_t \nabla f_t(\mathbf{x}_t)\big), \quad \alpha_t > 0, \qquad \text{(OGD)}$$

where $\Pi_{\mathcal{X}}$ is the projection onto $\mathcal{X}$.

We consider a natural extension of (OGD) to the bilevel setting (containing inner and outer OGD) and demonstrate that it exhibits regret bounds based on the path-length of the inner and/or outer minimizer sequences. To do so, we need to compute the gradient of the outer objective (hypergradient) $\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$ where $\mathbf{y}_t^*(\mathbf{x})$ is defined in (3b). The computation of $\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$ involves Jacobian $\nabla^2_{\mathbf{xy}} g_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$ and Hessian $\nabla^2_{\mathbf{y}} g_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$. More concretely, since $\nabla_{\mathbf{y}} g_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) = 0$, it follows from Assumption A and the implicit function theorem

$$\nabla \mathbf{y}_t^*(\mathbf{x}) \nabla^2_{\mathbf{y}} g_t\left(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})\right) + \nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})\right) = 0,$$

which together with the chain rule gives

$$\begin{aligned}\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) &= \nabla_{\mathbf{x}} f_t\left(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})\right) \\ &\quad + \nabla \mathbf{y}_t^*(\mathbf{x}) \nabla_{\mathbf{y}} f_t\left(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})\right).\end{aligned}$$

The exact gradient $\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$ is generally not available, preventing the use of gradient-type methods for bilevel regret minimization. In this work, inspired by offline bilevel optimization [Ghadimi and Wang, 2018] and online single-level optimization [Hazan et al., 2017, Aydore et al., 2019], we define a new *time-averaged hypergradient* as a surrogate of $\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$ by replacing $\mathbf{y}_t^*(\mathbf{x}_t)$ with $\mathbf{y}_t \in \mathbb{R}^{d_2}$ and using the history of the hypergradients.

Definition 1 (Time-Averaged Hypergradient). *Given a window size $w \in [T]$, let $\{u_i\}_{i=0}^{w-1}$ be a positive decreasing sequence with $u_0 = 1$. Let $F_{t,\mathbf{u}}(\mathbf{x}, \mathbf{y}) := (1/W) \sum_{i=0}^{w-1} u_i f_{t-i}(\mathbf{x}, \mathbf{y})$ with $W = \sum_{i=0}^{w-1} u_i$ and the convention $f_t \equiv 0$ for $t \leq 0$. Let $\mathbf{M}_t(\mathbf{x}, \mathbf{y})$ be the solution of the following linear equation:*

$$\nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x}, \mathbf{y}\right) + \mathbf{M}_t(\mathbf{x}, \mathbf{y}) \nabla^2_{\mathbf{y}} g_t\left(\mathbf{x}, \mathbf{y}\right) = 0.$$

Then, the time-averaged hypergradient is defined as

$$\tilde{\nabla} F_{t,\mathbf{u}}(\mathbf{x}, \mathbf{y}) := \frac{1}{W} \sum_{i=0}^{w-1} u_i \tilde{\nabla} f_{t-i}(\mathbf{x}, \mathbf{y}), \qquad (7)$$

where

$$\tilde{\nabla} f_t(\mathbf{x}, \mathbf{y}) := \nabla_{\mathbf{x}} f_t(\mathbf{x}, \mathbf{y}) + \mathbf{M}_t(\mathbf{x}, \mathbf{y}) \nabla_{\mathbf{y}} f_t(\mathbf{x}, \mathbf{y}). \qquad (8)$$

Algorithm 1 : OAGD for Bilevel Regret Minimization

```
Require: Initial values $(\mathbf{x}_1, \mathbf{y}_1) \in \mathcal{X} \times \mathbb{R}^{d_2}$; parameters $w, T, K_1, K_2, \ldots, K_T \in \mathbb{N}$; stepsizes $\{(\alpha_t, \beta_t) \in \mathbb{R}^2_{++}\}_{t=1}^T$; and weights $\{u_i\}_{i=0}^{w-1}$ with $1 = u_0 \geq u_1 \geq \ldots \geq u_{w-1} > 0$.
for $t = 1$ to $T$ do
    Acquire information about functions $f_t$ and $g_t$
    Set $\mathbf{z}_t^1 \leftarrow \mathbf{y}_t$
    for $k = 1$ to $K_t$ do
        Update $\mathbf{z}_t^{k+1} \leftarrow \mathbf{z}_t^k - \beta_t \nabla_{\mathbf{z}} g_t(\mathbf{x}_t, \mathbf{z}_t^k)$
    end for
    Update $\mathbf{y}_{t+1} \leftarrow \mathbf{z}_t^{K_t+1}$
    Update $\mathbf{x}_{t+1} \leftarrow \Pi_{\mathcal{X}}\left[\mathbf{x}_t - \alpha_t \tilde{\nabla} F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}_{t+1})\right]$
end for
```

Remark 2. *$\tilde{\nabla} F_{t,\mathbf{u}}(\mathbf{x}, \mathbf{y})$ is defined using the hypergradients of the losses from the w recent rounds. By setting $u_i = 1$, it averages a sliding window of online hypergradients at each update. With $u_i = \delta^i$ for $\delta \in (0, 1)$, it emphasizes recent values, giving an exponential average of hypergradients. Although $\tilde{\nabla} F_{t,\mathbf{u}}(\mathbf{x}, \mathbf{y})$ seems computationally intensive for large w, the w terms can be processed in parallel, mitigating the cost.*

The pseudo-code for the online alternating gradient descent (OAGD) method is presented in Algorithm 1. This algorithm is very simple to implement. At each timestep $t \in [T]$, OAGD alternates between the gradient update on $\mathbf{y}_t$ and the time-averaged projected hypergradient on $\mathbf{x}_t$. One can notice that the alternating update in Algorithm 1 serves as a template for running (OGD) on OBO problems. In OAGD, w and u_i are tunable parameters. Remark 2 and Theorem 9 provide suggested values for them. Intuitively, the value of w captures the level of averaging (smoothness) of the hypergradient at round t. We note that OAGD is similar to single-level time-smoothing OGD-type methods for the outer variable update [Hazan et al., 2017]. Also, without the inner variable and by setting the window size $w = 1$, OAGD reduces to (OGD). It should be mentioned that $w > 1$ is not required for our bilevel dynamic and static regret minimization. However, evaluations in Section 4 reveal that Equation (7) with $w > 1$ provides a performance boost over the case $w = 1$. Finally, we note that for $w = 1$, Algorithm 1 is similar to the gradient methods for offline BO [Ghadimi and Wang, 2018, Chen et al., 2021, Ji et al., 2021b].

Lemma 3. *Under Assumption A, for all $t \in [T]$, $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, and $\mathbf{y} \in \mathbb{R}^{d_2}$, we have*

$$\|\mathbf{y}_t^*(\mathbf{x}) - \mathbf{y}_t^*(\mathbf{x}')\| \leq L_{\mathbf{y}} \|\mathbf{x} - \mathbf{x}'\|,$$
$$\|\tilde{\nabla} f_t(\mathbf{x}, \mathbf{y}) - \nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))\| \leq M_f \|\mathbf{y} - \mathbf{y}_t^*(\mathbf{x})\|,$$
$$\|\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) - \nabla f_t(\mathbf{x}', \mathbf{y}_t^*(\mathbf{x}'))\| \leq L_f \|\mathbf{x} - \mathbf{x}'\|.$$

Here, $L_{\mathbf{y}} = \mathcal{O}(\kappa_g)$, $M_f = \mathcal{O}(\kappa_g^2)$, and $L_f = \mathcal{O}(\kappa_g^3)$.

The proof of Lemma 3 is provided in Appendix C.2.1.

3.2 Main Results

This section presents the convergence results of the OAGD algorithm. In Theorems 4–8, we simplify the analysis by setting $w = 1$. For a summary and comparison of these results with the single-level setting, we refer to Table 1. Proofs can be found in Appendix C.

Theorem 4 (**Strongly-Convex Dynamic**). *Suppose Assumptions A–B hold, and $\{f_t\}_{t=1}^T$ are strongly convex with parameter μ_f. Then, Algorithm 1 with*

$$\beta_t = \beta = \frac{2}{\ell_{g,1} + \mu_g}, \quad \alpha_t = \alpha \leq \min\Big\{\frac{1}{\ell_{f,1}}, \frac{\mu_f}{128 M_f^2 L_{\mathbf{y}}^2}\Big\},$$
$$K_t = K \geq \left\lceil 0.25(\kappa_g + 1) \log\big(4(\frac{1}{\alpha \mu_f} + 2)^2\big)\right\rceil,$$

for all $t \in [T]$, satisfies the follwoing

$$\text{BD-Reg}_T \leq \mathcal{O}\Big(1 + \min\Big\{S_{1,T}, \sum_{t=1}^T \|\nabla f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))\|^2 + S_{2,T}\Big\}\Big). \tag{10}$$

Theorem 4 shows that Algorithm 1, using fixed step sizes and $K_t = \tilde{\mathcal{O}}(\kappa_g)$, achieves a problem-dependent regret bound. While it might appear advantageous to increase K_t, our analysis suggests that even as K_t approaches infinity, the regret bound only improves by a constant factor.

In single-level online setting, [Zhang et al., 2017] shows that if $\sum_{t=1}^T \|\nabla f_t(\mathbf{x}_t^*)\|^2 = \mathcal{O}(P_{2,T})$, the dynamic regret bound of (OGD) can be further improved to $\mathcal{O}(1 + \min\{P_{1,T}, P_{2,T}\})$ by allowing multiple gradient queries (resulting in multiple updates to $\mathbf{x}_t$). When $Y_{1,T} = Y_{2,T} = 0$, we achieve a similar dynamic regret bound *without* multiple gradient queries.

If $f_t = f$ and $g_t = g$, then $S_{1,T} = S_{2,T} = 0$ implies a regret bound of $\mathcal{O}(1)$, leading to convergence rates for offline bilevel gradient methods [Ghadimi and Wang, 2018]. If the difference between consecutive inner and outer arguments decreases as $1/t$, then $P_{1,T} = Y_{1,T} = \mathcal{O}(\log T)$, resulting in a logarithmic regret bound.

The following theorem provides the lower bound $\Omega(1 + S_{2,T})$ for OBO.

Theorem 5 (**Lower Bound**). *For any OBO algorithm, there always exists a sequence of smooth and strongly convex functions $\{(f_t, g_t)\}_{t=1}^T$ such that*

$$\text{BD-Reg}_T = \Omega(1 + S_{2,T}).$$

Theorem 5 indicates that the upper bound in Theorem 4 cannot be improved in general.

Theorem 6 (**Strongly-Convex Static**). *Suppose Assumptions A–B hold, and $\{f_t\}_{t=1}^T$ are strongly convex with parameter μ_f. Then, Algorithm 1 with*

$$\beta_t = \beta = \frac{2}{\ell_{g,1} + \mu_g}, \quad \alpha_t = \frac{2}{\mu_f t},$$
$$K_t = K \geq \left\lceil 0.25(\kappa_g + 1) \log\big((\frac{24 L_{\mathbf{y}} M_f}{\mu_f})^2 + 2\big)\right\rceil,$$

for all $t \in [T]$, satisfies the follwoing

$$\text{BS-Reg}_T \leq \mathcal{O}\left(\log T + \bar{Y}_{2,T}\right). \tag{11}$$

Theorem 6 shows that Algorithm 1, with decreasing α_t and $K_t = \tilde{\mathcal{O}}(\kappa_g)$, achieves a problem-dependent regret bound, where the $\log T$ term mirrors single-level static findings [Hazan et al., 2007], and $\bar{Y}_{2,T}$ accounts for the variability in $\{\mathbf{y}_t^*(\mathbf{x}^*)\}_{t=1}^T$ over T.

The following theorem provides the regret bounds for online convex functions $\{f_t\}_{t=1}^T$ in the dynamic setting.

Theorem 7 (**Convex Dynamic**). *Suppose Assumptions A–B hold, functions $\{f_t\}_{t=1}^T$ are convex, and $\exists\, (\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) \in \mathcal{X} \times \mathbb{R}^{d_2}$ such that $\nabla f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) = 0$ for all $t \in [T]$. Then, Algorithm 1 with*

$$\beta_t = \beta = \frac{2}{\ell_{g,1} + \mu_g}, \quad \alpha_t = \alpha \leq \frac{1}{4 L_f},$$
$$K_t \geq \left\lceil 0.25(\kappa_g + 1) \log 4t^2 \right\rceil,$$

for all $t \in [T]$, satisfies the following

$$\text{BD-Reg}_T \leq \mathcal{O}\left(1 + S_{1,T} + Y_{2,T}\right). \tag{12}$$

From Theorem 7 we see that Algorithm 1 achieves an $\mathcal{O}(1 + S_{1,T} + Y_{2,T})$ dynamic regret for a sequence of loss functions that satisfy Assumption A with only gradient feedback. Note that the condition $\nabla f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) = 0$ is referred to as the vanishing gradient condition, which is widely used in the analysis of OGD methods in the single-level convex setting [Yang et al., 2016, Assumption 2].

Theorem 8 (**Convex Static**). *Suppose Assumptions A-B hold and functions $\{f_t\}_{t=1}^T$ are convex. Then, Algorithm 1 with*

$$\beta_t = \beta = \frac{2}{\ell_{g,1} + \mu_g}, \quad \alpha_t = \frac{D}{\ell_{f,0}\sqrt{t}},$$
$$K_t \geq \left\lceil 0.25(\kappa_g + 1) \log 4t^2 \right\rceil,$$

for all $t \in [T]$, satisfies the following

$$\text{BS-Reg}_T \leq \mathcal{O}\left(\sqrt{T} + \bar{Y}_{1,T} + \bar{Y}_{2,T}\right). \tag{13}$$

Theorem 8 provides static bounds when the cost functions are convex. We observe that the term $\sqrt{T}$ is identical to the bound in the single-level static setting [Zinkevich, 2003], and $\bar{Y}_{1,T}$ and $\bar{Y}_{2,T}$ account for the variability in $\{\mathbf{y}_t^*(\mathbf{x}^*)\}_{t=1}^T$ over T. Additionally, we note that $\bar{Y}_{1,T}$ and $\bar{Y}_{2,T}$ are not generally comparable; see Example 1 in Appendix A.1.

3.2.1 Local Regret Minimization

In this section, we consider online bilevel learning with non-convex outer losses. While minimizing the regret (3) makes sense for online convex functions $\{f_t\}_{t=1}^T$, it is not appropriate for general non-convex online costs, as the global minimization of a non-convex objective is generally intractable. We address this issue with a combined approach, leveraging optimality criteria and measures from offline non-convex bilevel analysis, together with smoothing of the online part of the outer objective function similar to [Hazan et al., 2017]. Throughout this section, we set $\mathcal{X} \equiv \mathbb{R}^{d_1}$.

For the sequence $\{u_i\}_{i=0}^{w-1}$ given in Definition 1 and for all $w \in [T]$, we define the following *bilevel local regret*:

$$\text{BL-Reg}_{T,\mathbf{u}} := \sum_{t=1}^{T} \|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2. \tag{14}$$

Here, $\mathbf{y}_t^*(\mathbf{x}) \in \arg\min_{\mathbf{y}\in\mathbb{R}^{d_2}} g_t(\mathbf{x},\mathbf{y})$, and

$$F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) = \frac{1}{W}\sum_{i=0}^{w-1} u_i f_{t-i}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))$$

with the convention $f_t \equiv 0$ for $t \leq 0$. Note that in the single-level setting, (14) with $u_i = 1$ simplifies to the local regret in [Hazan et al., 2017].

For local regret analysis, we utilize $H_{p,T}$, as defined in (6), to measure the variation of $\mathbf{y}_t^*(\mathbf{x})$. We introduce $H_{p,T}$ to account for cases where its value is inherently small. For instance, in the online problem discussed in Section 4, $H_{p,T}$ represents the label variability over T, which can be small for an optimal range of hyperparameters $\mathbf{x} \in \mathbb{R}^{d_1}$.

Assumption C. *For all $t \in [T]$, $|f_t(\mathbf{x},\mathbf{y})| \leq M$ for some finite constant $M > 0$.*

Assumption C is a common assumption in the literature [Hazan et al., 2017]. The following theorem demonstrates OAGD's sublinear local regret.

Theorem 9 (Non-convex Local). *Suppose Assumptions A and C hold. Then, Algorithm 1 with*

$$\beta_t = \beta = \frac{2}{\ell_{g,1}+\mu_g}, \quad K_t = 1,$$
$$\alpha_t = \alpha \leq \min\left\{\frac{1}{8L_f}, \frac{1}{2\sqrt{2}L_{\mathbf{y}}M_f(\kappa_g^2-1)^{1/2}}\right\},$$

for all $t \in [T]$, satisfies the following

$$\text{BL-Reg}_{T,\mathbf{u}} \leq \mathcal{O}\Big(\frac{T}{W} + H_{1,T} + H_{2,T}\Big). \tag{15}$$

The above regret can become sublinear in T provided $H_{1,T} = o(T)$, $H_{2,T} = o(T)$, and the weight w is appropriately chosen so that $W = o(T)$. Theorem 9 aligns closely with the existing bounds in various non-convex optimization contexts. In the OBO setting, it parallels [Lin et al., 2024, Theorem 5.7], yet it does not require monitoring fluctuations between online objective functions. When $H_{p,T} = 0$, as in a single-level setting, Theorem 9 matches the findings of [Hazan et al., 2017] but applies to a broader range of weight sequences $\{u_i\}_{i=0}^{w-1}$. It is important to note that when $u_i = 1$ for all $i \in \{0,\ldots,w-1\}$, it yields a local regret bound of $\mathcal{O}(T/w + H_{1,T} + H_{2,T})$. For the offline case where $f_t = f$, the results provide a convergence guarantee for non-convex BO [Ghadimi and Wang, 2018].

4 Experimental Results

In this section, we conduct preliminary experiments to evaluate OAGD performance, with additional experiments available in Appendix D. Code is available at `https://github.com/BojianHou/OAGD`.

4.1 Online Hyperparameters Learning for Dynamic Regression

Hyperparameter optimization (HO) is the process of finding the best set of hyperparameters that cannot be learned using the training data alone [Franceschi et al., 2018]. An HO problem can be formulated as a BO problem. The outer objective, $f(\mathbf{y}^*(\mathbf{x}); \mathcal{D}^{\text{val}})$, aims to minimize the validation loss concerning the hyperparameters $\mathbf{x}$. Meanwhile, the inner objective, $g(\mathbf{x},\mathbf{y};\mathcal{D}^{\text{tr}})$, minimizes the training loss concerning the model parameters $\mathbf{y}$.

We consider online HO for dynamic regression as follows: At each round or timestep t, new samples $(\mathbf{a}_t, b_t) \in \mathcal{D}_t := \{\mathcal{D}_t^{\text{val}}, \mathcal{D}_t^{\text{tr}}\}$ for all $t \in [T]$ are received, where $\mathbf{a}_t \in \mathbb{R}^{d_2}$ represents the feature vector and $b_t \in \mathbb{R}$ is the corresponding target. It's important to note that the potential correct decision can change abruptly. Specifically, we consider an S-stage scenario where

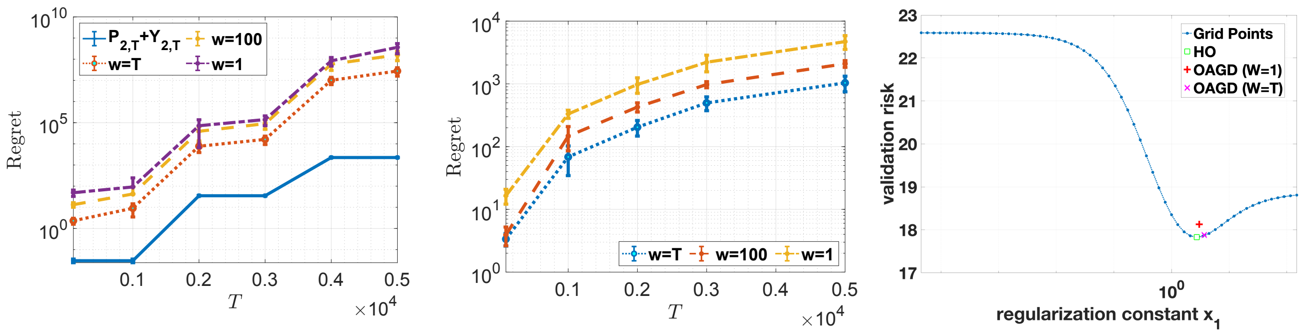

Figure 1: Performance of OAGD in online hyperparameter learning over five runs. The left and middle figures show OBO's regret with three comparators and a fixed comparator, respectively. The right figure illustrates the outer problem's trajectories and the performance of OAGD and offline HO in learning the hyperparameter x_1.

$(\mathbf{x}_s^*, \mathbf{y}_s^*(\mathbf{x}_s^*))$ represents potentially the best decisions for the s-th stage, encompassing all $s \in [S]$:

$$\begin{aligned} \mathbf{x}_s^* &\in \underset{\mathbf{x}\in\mathcal{X}}{\operatorname{argmin}} \ \sum_{t=1}^{T_s} f\left(\mathbf{y}_s^*(\mathbf{x}); \mathcal{D}_t^{\text{val}}\right) \\ \text{s.t. } \mathbf{y}_s^*(\mathbf{x}) &\in \underset{\mathbf{y}\in\mathbb{R}^{d_2}}{\operatorname{argmin}} \ \sum_{t=1}^{T_s} g\left(\mathbf{x},\mathbf{y};\mathcal{D}_t^{\text{tr}}\right). \end{aligned} \tag{16}$$

At each round t of online HO, given a sample $(\mathbf{a}_t, b_t) \in \mathcal{D}_t^{\text{tr}}$, the follower is required to make the prediction by $\mathbf{a}_t^\top \mathbf{y}_t$ based on the learned inner and outer models $(\mathbf{x}_{t-1}, \mathbf{y}_{t-1}) \in \mathcal{X} \times \mathbb{R}^{d_2}$; then, as a consequence the follower suffers a loss $g(\mathbf{x}_{t-1}, \mathbf{y}_t; \mathcal{D}_t^{\text{tr}}) = 1/2(\mathbf{a}_t^\top \mathbf{y}_t - b_t)^2 + \mathbf{y}_t^\top \mathbf{C}(\mathbf{x}_{t-1})\mathbf{y}_t$, where $\mathbf{C}(\mathbf{x}) := \operatorname{diag}(\exp(x_i))_{i=1}^{d_1}$. The leader then receives the feedback of the inner model, i.e., $\mathbf{y}_t$, predicts the new hyperparameter $\mathbf{x}_t$ using a validation sample $(\mathbf{a}_t, b_t) \in \mathcal{D}_t^{\text{val}}$, and suffers the loss $f(\mathbf{y}_t(\mathbf{x}_t); \mathcal{D}_t^{\text{val}}) = 1/2(\mathbf{a}_t^\top \mathbf{y}_t(\mathbf{x}_t) - b_t)^2$. This process repeats across $T = \{T_1, \ldots, T_S\}$ rounds.

Figure 1 (left and middle) shows the variation $P_{2,T} + Y_{2,T}$ and the regret bound of OAGD with three different window sizes $w \in \{1, 100, T\}$ on synthetic data; see Appendix D.1 for further details. We observe that OAGD with $w = T$ performs the best, with a gradual decrease in performance as w decreases to 100 and $w = 1$. Additionally, Figure 1 (right) demonstrates that the performance of OAGD is comparable to the performance of the offline HO [Franceschi et al., 2018].

4.2 Online Parametric Loss Tuning for Imbalanced Data

Imbalanced datasets are common in modern machine learning, posing challenges in generalization and fairness due to underrepresented classes and sensitive attributes. This issue is exacerbated by deep neural networks' tendency to overfit, appearing accurate and fair during training but performing poorly during testing. AutoBalance [Li et al., 2021] addresses this by automatically designing a parametric training loss to balance accuracy and fairness while preventing overfitting. We give an online variant of AutoBalance, demonstrating the enhanced performance of OAGD in this setting.

The bilevel objective function for loss tuning is the same as (16) but the leader's and the follower's loss functions are defined differently. At each round or timestep t, new samples $(\mathbf{a}_t, b_t) \in \mathcal{D}_t := \{\mathcal{D}_t^{\text{val}}, \mathcal{D}_t^{\text{tr}}\}$ for all $t \in [T]$ are received, where $\mathbf{a}_t \in \mathbb{R}^{d_2}$ represents the feature vector and $b_t \in \{1, \ldots, J\}$ represents the corresponding label. For a new sample $(\mathbf{a}_t, b_t)$, the follower suffers from a *parametric* cross-entropy loss:

$$g(\mathbf{x}_{t-1}, \mathbf{y}_t; \mathcal{D}_t^{\text{tr}}) = -\log \frac{e^{\gamma_{b_t}[\mathbf{y}_t(\mathbf{a}_t)]_{b_t} + \Delta_{b_t}}}{\sum_{j=1}^{J} e^{\gamma_j [\mathbf{y}_t(\mathbf{a}_t)]_j + \Delta_j}}, \tag{17a}$$

where $\mathbf{x}_{t-1} := (\Delta_j, \gamma_j)_{j=1}^{J}$ represents the *logits*.

In the outer-level, the leader suffers from a balanced cross entropy loss

$$f(\mathbf{y}_t(\mathbf{x}_t); \mathcal{D}_t^{\text{val}}) = -u_{b_t} \log \frac{e^{[\mathbf{y}_t(\mathbf{a}_t)]_{b_t}}}{\sum_{j=1}^{J} e^{[\mathbf{y}_t(\mathbf{a}_t)]_j}}, \tag{17b}$$

where u_j represents the reciprocal of the proportion of samples from the j-th class to the total number of samples [Li et al., 2021].

There might be one notation abuse in (17) that we need to clarify: $\mathbf{y}_t(\mathbf{x}_t)$ still indicates that the follower $\mathbf{y}_t$ is conditioned on the leader $\mathbf{x}_t$, whereas $[\mathbf{y}_t(\mathbf{a}_t)]_{b_t}$ denotes the predicted logit for class b_t that the follower $\mathbf{y}_t$ makes on sample $\mathbf{a}_t$. Note that the backbone model for $\mathbf{y}_t$ is a 4-layer CNN, resulting in a nonconvex bilevel objective. For more details, refer to Appendix D.

We compare Algorithm 1 with the following baselines:

- **Single-Level OGD** [Zinkevich, 2003]: Updates the model $\mathbf{y}_t$ with fixed hyperparameters $\mathbf{x}$ at each timestep on the newly observed data using gradient descent.
- **AutoBalance** [Li et al., 2021]: An offline bilevel gradient descent framework that updates hyperparameters $\mathbf{x}_t$ and the model $\mathbf{y}_t$ to address imbalance issues.

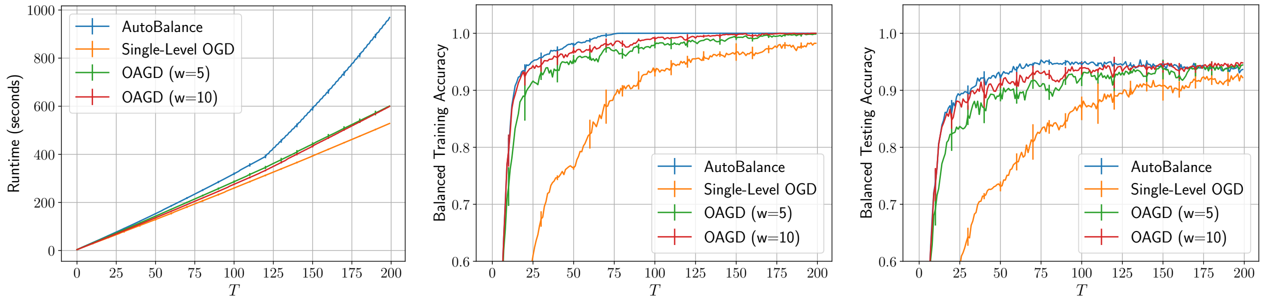

Figure 2: Performance comparison (mean±std) on loss tuning for imbalanced MNIST data across five runs.

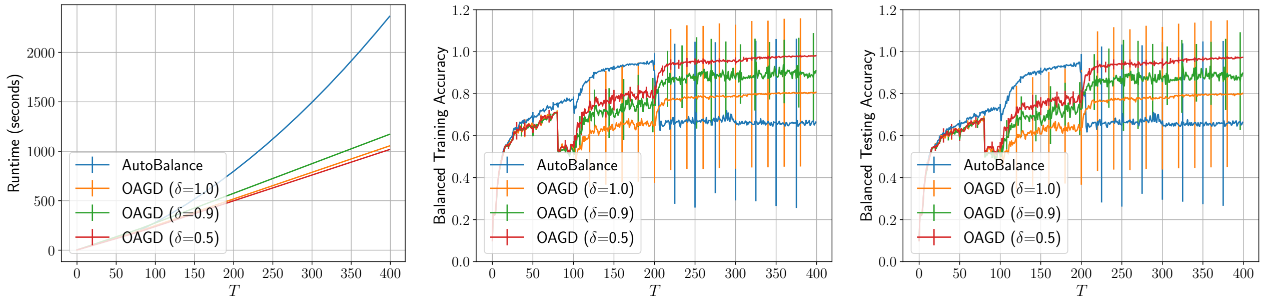

Figure 3: Performance comparison (mean±std) on loss tuning for imbalanced MNIST data across five runs, considering induced distribution shift.

We conducted experiments using the MNIST dataset [LeCun et al., 2010]. To create an imbalanced scenario, we selected samples in proportions of 0.6^i from each class ($i = 0, 1, \ldots, 9$). For online learning, we used a batch size of 128 at each timestep to train our OAGD. If the window size w exceeded 1, we combined the current batch with the previous $w-1$ batches for OAGD training. We evaluated cumulative runtime, along with balanced training and testing accuracy, where balanced accuracy is the class-specific average accuracy: $\frac{1}{J}\sum_{j=1}^{J}\mathbb{P}_{\mathbf{a}_t\sim\mathcal{D}_j}[\arg\max_i([\mathbf{y}_t(\mathbf{a}_t)]_i) = j]$ where $\mathcal{D}_j$ refers to the distribution over samples whose groundtruth class label is j and $\mathbb{P}[A]$ means the probability of event A [Li et al., 2021]. To ensure consistency, we maintained a fixed inner-level learning rate of $\beta = 0.1$ for all bilevel algorithms and single-level OGD, with the outer-level learning rate set at $\alpha = 0.001$.

Figure 2 (left) provides runtime comparisons. The single-level OGD algorithm is the fastest since it lacks an outer-level training step and trains on a single batch of data at each timestep. Our OAGD exhibits similar runtime characteristics, with the runtime increasing as the window size w grows due to more extensive training. In contrast, AutoBalance is the slowest method as it trains on all observed data up to each timestep.

Figure 2 (middle and right) displays balanced training and testing accuracy. AutoBalance quickly achieves high accuracy after 10 timesteps, whereas single-level OGD exhibits slower improvement. OAGD ($w = 5$) and OAGD ($w = 10$) exhibit rapid growth in both testing and training accuracy, eventually outperforming AutoBalance. They benefit from time-smoothing hypergradients. Larger window sizes further enhance OAGD's balanced training and testing accuracy.

We conducted experiments on the MNIST dataset to evaluate performance under time-varying distribution shifts across 400 timesteps, divided into four phases of varying distributions, each lasting 100 timesteps. Initially, the distribution was highly imbalanced, with class proportions set by 0.4^i for classes $i = 0$ to 9. This imbalance gradually lessened over the next two phases, changing from 0.6^i to 0.8^i. The final 100 timesteps featured a balanced distribution across the 10 MNIST classes, ensuring normalized class proportions throughout all phases.

The parameter δ influences the weighting of each window in the "time-averaged hypergradient," per Definition 1. Setting $u_i = 1$ computes the average of online hypergradients in a sliding window. For $u_i = \delta^i$ with $\delta \in (0, 1)$, it weights recent values more through an exponential average. Experiments with δ values of 1, 0.9, and 0.5 show that lower δ gives more emphasis to recent windows.

The results, illustrated in Figure 3, show a significant performance decrease for AutoBalance at timestep $T = 200$, while OAGD remains comparatively stable against distribution shifts. At $T = 80$, OAGD experiences a noticeable drop due to shifting from a single-

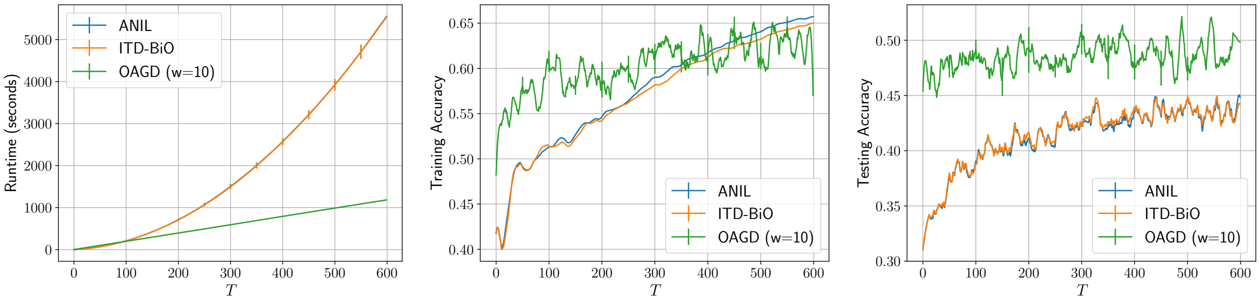

Figure 4: Performance comparison (mean±std) on meta-learning for FC100 data across five runs.

level method with fixed hyperparameters (warm-up phase) to a bilevel framework that starts adjusting hyperparameters at $T = 80$, causing a short disruption. This warm-up phase is standard in bilevel optimization [Li et al., 2021, Lorraine et al., 2020]. Although larger w are beneficial for gradient accuracy, they can produce outdated gradients if data distribution changes, negatively affecting model updates. In contrast, a smaller δ can improve adaptability and performance by focusing on recent data, even with large window sizes w.

4.3 Online Meta-Learning

Meta-learning aims to bootstrap from a set of given tasks to learn faster on future tasks [Finn et al., 2017, Balcan et al., 2019]. A popular formulation is online meta-learning (OML) where agents sequentially face tasks and apply methods such as classical FPL [Finn et al., 2019] or mirror descent [Denevi et al., 2019] for enhanced meta-learning. We consider an implicit gradient-based OML setting: for each task $\mathcal{T}_t$ and some $\beta > 0$, the follower adapts the leader's model $\mathbf{w}_t \in \mathcal{W} \subset \mathbb{R}^d$ using training data $\mathcal{D}_t^{\text{tr}}$ an inner OGD:

$$\mathbf{u}_t^*(\mathbf{w}_t) \in \underset{\mathbf{u}\in\mathbb{R}^d}{\arg\min} \left\langle \mathbf{u}, \nabla f(\mathbf{w}_t; \mathcal{D}_t^{tr}) \right\rangle + \frac{1}{2\beta} \|\mathbf{u} - \mathbf{w}_t\|^2 .$$

Then, the test data $\mathcal{D}_t^{\text{ts}}$ will be revealed to the leader for evaluating the performance of the follower's model $\mathbf{u}_t^*(\mathbf{w}_t)$. The loss observed at this timestep, denoted as $f(\mathbf{u}_t^*(\mathbf{w}_t); \mathcal{D}_t^{\text{ts}})$, can then be fed into the leader's algorithm (outer OGD) to update $\mathbf{w}_t$. Despite the convex nature of the loss function, which is a cross-entropy loss, where $\mathbf{w}$ represents a 4-layer CNN, ultimately rendering the outer problem non-convex. Further, the inner problem for loss tuning involves training a CNN and is non-convex, showing our implementation's wide scope. We compare our OAGD with the following meta-learning methods:

- **ANIL** [Raghu et al., 2019]: A widely used meta-learning algorithm, which simplifies MAML by removing the inner loop for all parts of the MAML-trained network except for the task-specific head.

- **ITD-BiO** [Ji et al., 2021b]: A gradient-based stochastic bilevel optimization framework based on iterative differentiation (ITD).

We evaluate our model on the FC100 (Fewshot-CIFAR100) dataset [Oreshkin et al., 2018] through a 5-way 5-shot task. In the online setting, each timestep presents one task with 25 training and 25 testing samples. If the window size exceeds 1, data from up to $w - 1$ previous tasks are included. In contrast, the offline setting allows baselines to use all observed data up to the current timestep. The inner and outer learning rates for ANIL and ITD-BiO are 0.01 and $5e - 5$, respectively, while our OAGD employs learning rates of 0.1 and $1e - 4$. These experiments were conducted on a P100 GPU equipped with 12 GB of memory.

In Figure 4, we provide the performance comparison in terms of runtime, training accuracy, and testing accuracy. We compare only our OAGD ($w = 10$) to other baselines to enhance the precision of the figure. For the sensitivity analysis concerning the window size, please refer to the Appendix. From the left figure in Figure 4, it's noticeable that the two baselines consume a similar longer time with an exponential trend, while our OAGD requires the least time, following a linear trend. However, as indicated by the middle and right figures, our OAGD demonstrates competitive training accuracy and even better testing accuracy across all the timesteps, highlighting its superiority.

5 Conclusion

This paper studies online bilevel optimization and provides regret guarantees under different convexity assumptions on the time-varying objective functions. In particular, we propose a new class of online bilevel algorithms capable of leveraging smoothness and providing regret bound in terms of problem-dependent quantities, such as the path-length of the comparator sequence.

Acknowledgements

This work was supported in part by NSF CAREER award CCF1845076, AFOSR YIP award FA9550-19-1-

0026, ARO YIP award W911NF1910027, and NIH grants U01 AG066833, U01 AG068057, and RF1 AG063481.

References

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

N. Agarwal, A. Gonen, and E. Hazan. Learning in non-convex games with an optimization oracle. In *Conference on Learning Theory*, pages 18–29. PMLR, 2019.

E. Aiyoshi and K. Shimizu. A solution method for the static constrained stackelberg problem via penalty method. *IEEE Transactions on Automatic Control*, 29(12):1111–1114, 1984.

F. A. Al-Khayyal, R. Horst, and P. M. Pardalos. Global optimization of concave functions subject to quadratic constraints: an application in nonlinear bilevel programming. *Annals of Operations Research*, 34(1):125–147, 1992.

S. M. Arnold, P. Mahajan, D. Datta, I. Bunner, and K. S. Zarkias. learn2learn: A library for meta-learning research. *arXiv preprint arXiv:2008.12284*, 2020.

S. Aydore, T. Zhu, and D. P. Foster. Dynamic local regret for non-convex online forecasting. *Advances in Neural Information Processing Systems*, 32, 2019.

D. Baby and Y.-X. Wang. Online forecasting of total-variation-bounded sequences. In *Advances in Neural Information Processing Systems 32*, page 11071–11081, 2019.

M.-F. Balcan, M. Khodak, and A. Talwalkar. Provable guarantees for gradient-based meta-learning. In *International Conference on Machine Learning*, pages 424–433. PMLR, 2019.

B. Becker and R. Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

L. Bertinetto, J. F. Henriques, P. Torr, and A. Vedaldi. Meta-learning with differentiable closed-form solvers. In *International Conference on Learning Representations*, 2018.

O. Besbes, Y. Gur, and A. Zeevi. Non-stationary stochastic optimization. *Operations research*, 63(5): 1227–1244, 2015.

O. Bousquet and M. K. Warmuth. Tracking a small set of experts by mixing past posteriors. *Journal of Machine Learning Research*, 3(Nov):363–396, 2002.

J. Bracken and J. T. McGill. Mathematical programs with optimization problems in the constraints. *Operations Research*, 21(1):37–44, 1973.

S. Bubeck, G. Stoltz, C. Szepesvári, and R. Munos. Online optimization in x-armed bandits. *Advances in Neural Information Processing Systems*, 21, 2008.

T.-J. Chang and S. Shahrampour. On online optimization: Dynamic regret analysis of strongly convex and smooth problems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6966–6973, 2021.

T. Chen, Y. Sun, and W. Yin. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. *Advances in Neural Information Processing Systems*, 34, 2021.

T. Chen, Y. Sun, and W. Yin. A single-timescale stochastic bilevel optimization method. *AISTATS*, 2022.

C.-K. Chiang, T. Yang, C.-J. Lee, M. Mahdavi, C.-J. Lu, R. Jin, and S. Zhu. Online optimization with gradual variations. In *Conference on Learning Theory*, pages 6–1. JMLR Workshop and Conference Proceedings, 2012.

A. Daniely, A. Gonen, and S. Shalev-Shwartz. Strongly adaptive online learning. In *International Conference on Machine Learning*, pages 1405–1411, 2015.

G. Denevi, D. Stamos, C. Ciliberto, and M. Pontil. Online-within-online meta-learning. *Advances in Neural Information Processing Systems*, 32, 2019.

J. Domke. Generic methods for optimization-based modeling. In *Artificial Intelligence and Statistics*, pages 318–326. PMLR, 2012.

T. A. Edmunds and J. F. Bard. Algorithms for nonlinear bilevel mathematical programs. *IEEE transactions on Systems, Man, and Cybernetics*, 21(1):83–89, 1991.

M. Feurer and F. Hutter. Hyperparameter optimization. In *Automated machine learning*, pages 3–33. Springer, Cham, 2019.

C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

C. Finn, A. Rajeswaran, S. Kakade, and S. Levine. Online meta-learning. In *International Conference on Machine Learning*, pages 1920–1930. PMLR, 2019.

L. Franceschi, M. Donini, P. Frasconi, and M. Pontil. Forward and reverse gradient-based hyperparameter optimization. In *International Conference on Machine Learning*, pages 1165–1173. PMLR, 2017.

L. Franceschi, P. Frasconi, S. Salzo, R. Grazzi, and M. Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*, pages 1568–1577. PMLR, 2018.

S. Ghadimi and M. Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.

R. Grazzi, L. Franceschi, M. Pontil, and S. Salzo. On the iteration complexity of hypergradient computation. In *International Conference on Machine Learning*, pages 3748–3758. PMLR, 2020.

Z. Guo, Y. Xu, W. Yin, R. Jin, and T. Yang. On stochastic moving-average estimators for non-convex optimization. *arXiv preprint arXiv:2104.14840*, 2021.

E. C. Hall and R. M. Willett. Online convex optimization in dynamic environments. *IEEE Journal of Selected Topics in Signal Processing*, 9(4):647–662, 2015.

N. Hallak, P. Mertikopoulos, and V. Cevher. Regret minimization in stochastic non-convex learning via a proximal-gradient approach. In *International Conference on Machine Learning*, pages 4008–4017. PMLR, 2021.

P. Hansen, B. Jaumard, and G. Savard. New branch-and-bound rules for linear bilevel programming. *SIAM Journal on scientific and Statistical Computing*, 13(5): 1194–1217, 1992.

K. Harris, H. Heidari, and S. Z. Wu. Stateful strategic regression. *Advances in Neural Information Processing Systems*, 34:28728–28741, 2021.

E. Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016a. URL http://ocobook.cs.princeton.edu/OCObook.pdf.

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

E. Hazan and C. Seshadhri. Adaptive algorithms for online decision problems. In *Electronic colloquium on computational complexity (ECCC)*, volume 14, 2007.

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

E. Hazan, K. Singh, and C. Zhang. Efficient regret minimization in non-convex games. In *International Conference on Machine Learning*, pages 1433–1441. PMLR, 2017.

A. Héliou, M. Martin, P. Mertikopoulos, and T. Rahier. Online non-convex optimization with imperfect feedback. *Advances in Neural Information Processing Systems*, 33:17224–17235, 2020.

A. Héliou, M. Martin, P. Mertikopoulos, and T. Rahier. Zeroth-order non-convex learning via hierarchical dual averaging. In *International Conference on Machine Learning*, pages 4192–4202. PMLR, 2021.

M. Herbster and M. K. Warmuth. Tracking the best expert. *Machine learning*, 32(2):151–178, 1998.

M. Herbster and M. K. Warmuth. Tracking the best linear predictor. *Journal of Machine Learning Research*, 1(281-309):10–1162, 2001.

S. C. Hoi, D. Sahoo, J. Lu, and P. Zhao. Online learning: A comprehensive survey. *Neurocomputing*, 459:249–289, 2021.

M. Hong, H.-T. Wai, Z. Wang, and Z. Yang. A two-timescale stochastic algorithm framework for bilevel optimization: Complexity analysis and application to actor-critic. *SIAM Journal on Optimization*, 33(1): 147–180, 2023.

F. Huang and H. Huang. Biadam: Fast adaptive bilevel optimization methods. *arXiv preprint arXiv:2106.11396*, 2021.

Y. Huang, Y. Cheng, Y. Liang, and L. Huang. Online min-max problems with non-convexity and non-stationarity. *Transactions on Machine Learning Research*, 2023.

A. Jadbabaie, A. Rakhlin, S. Shahrampour, and K. Sridharan. Online optimization: Competing with dynamic comparators. In *Artificial Intelligence and Statistics*, pages 398–406, 2015.

K. Ji, J. Yang, and Y. Liang. Provably faster algorithms for bilevel optimization and applications to meta-learning. In *International Conference on Machine Learning*, 2021a.

K. Ji, J. Yang, and Y. Liang. Bilevel optimization: Convergence analysis and enhanced design. In *International Conference on Machine Learning*, pages 4882–4892. PMLR, 2021b.

R. Kleinberg, A. Slivkins, and E. Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 681–690, 2008.

W. Krichene, M. Balandat, C. Tomlin, and A. Bayen. The hedge algorithm on a continuum. In *International Conference on Machine Learning*, pages 824–832. PMLR, 2015.

Y. LeCun, C. Cortes, and C. Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

J. Li, B. Gu, and H. Huang. Improved bilevel model: Fast and optimal algorithm with theoretical guarantee. *arXiv preprint arXiv:2009.00690*, 2020.

M. Li, X. Zhang, C. Thrampoulidis, J. Chen, and S. Oymak. Autobalance: Optimized loss functions for imbalanced data. *Advances in Neural Information Processing Systems*, 34:3163–3177, 2021.

Y. Liang et al. Lower bounds and accelerated algorithms for bilevel optimization. *Journal of Machine Learning Research*, 24(22):1–56, 2023.

S. Lin, D. Sow, K. Ji, Y. Liang, and N. Shroff. Non-convex bilevel optimization with time-varying objective functions. *Advances in Neural Information Processing Systems*, 36, 2024.

H. Liu, K. Simonyan, and Y. Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.

R. Liu, P. Mu, X. Yuan, S. Zeng, and J. Zhang. A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In *International Conference on Machine Learning*, pages 6305–6315. PMLR, 2020.

R. Liu, J. Gao, J. Zhang, D. Meng, and Z. Lin. Investigating bi-level optimization for learning and vision from a unified perspective: A survey and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):10045–10067, 2021.

J. Lorraine, P. Vicol, and D. Duvenaud. Optimizing millions of hyperparameters by implicit differentiation. In *International conference on artificial intelligence and statistics*, pages 1540–1552. PMLR, 2020.

Y. Lv, T. Hu, G. Wang, and Z. Wan. A penalty function method based on kuhn–tucker condition for solving linear bilevel programming. *Applied Mathematics and Computation*, 188(1):808–813, 2007.

D. Maclaurin, D. Duvenaud, and R. Adams. Gradient-based hyperparameter optimization through reversible learning. In *International conference on machine learning*, pages 2113–2122. PMLR, 2015.

R. V. Marinescu, N. P. Oxtoby, A. L. Young, E. E. Bron, A. W. Toga, M. W. Weiner, F. Barkhof, N. C. Fox, P. Golland, S. Klein, et al. Tadpole challenge: Accurate alzheimer's disease prediction through crowdsourced forecasting of future data. In *Predictive Intelligence in Medicine: Second International Workshop, PRIME 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Proceedings 2*, pages 1–10. Springer, 2019.

A. Mokhtari, S. Shahrampour, A. Jadbabaie, and A. Ribeiro. Online optimization in dynamic environments: Improved regret rates for strongly convex problems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 7195–7201. IEEE, 2016.

G. M. Moore. *Bilevel programming algorithms for machine learning model selection.* Rensselaer Polytechnic Institute, 2010.

P. Nazari and E. Khorram. Dynamic regret analysis for online meta-learning. *arXiv preprint arXiv:2109.14375*, 2021.

P. Nazari, E. Khorram, and D. A. Tarzanagh. Adaptive online distributed optimization in dynamic environments. *Optimization Methods and Software*, pages 1–25, 2019.

P. Nazari, A. Mousavi, D. A. Tarzanagh, and G. Michailidis. A penalty-based method for communication-efficient decentralized bilevel programming. *arXiv preprint arXiv:2211.04088*, 2022a.

P. Nazari, D. A. Tarzanagh, and G. Michailidis. Dadam: A consensus-based distributed adaptive gradient method for online optimization. *IEEE Transactions on Signal Processing*, 2022b.

Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

B. Oreshkin, P. Rodríguez López, and A. Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. *Advances in neural information processing systems*, 31, 2018.

F. Pedregosa. Hyperparameter optimization with approximate gradient. In *International conference on machine learning*, pages 737–746. PMLR, 2016.

A. Raghu, M. Raghu, S. Bengio, and O. Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of maml. *arXiv preprint arXiv:1909.09157*, 2019.

A. Shaban, C.-A. Cheng, N. Hatch, and B. Boots. Truncated back-propagation for bilevel optimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1723–1732. PMLR, 2019.

S. Shalev-Shwartz. *Online learning: Theory, algorithms, and applications*. Hebrew University, 2007.

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

C. Shi, J. Lu, and G. Zhang. An extended kuhn–tucker approach for linear bilevel programming. *Applied Mathematics and Computation*, 162(1):51–63, 2005.

A. Sinha, P. Malo, and K. Deb. A review on bilevel optimization: from classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2):276–295, 2017.

N. Srebro, K. Sridharan, and A. Tewari. Smoothness, low noise and fast rates. *Advances in neural information processing systems*, 23, 2010.

A. S. Suggala and P. Netrapalli. Online non-convex learning: Following the perturbed leader is optimal. In *Algorithmic Learning Theory*, pages 845–861. PMLR, 2020.

M. Sugiyama and M. Kawanabe. *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. MIT press, 2012.

K. G. Vamvoudakis and F. L. Lewis. Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica*, 46(5): 878–888, 2010.

O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

H. von Stackelberg. *Theory of the market economy*. United Kingdom: William Hodge, 1952.

C.-Y. Wei, Y.-T. Hong, and C.-J. Lu. Tracking the best expert in non-stationary stochastic environments. *Advances in neural information processing systems*, 29:3972–3980, 2016.

Y. F. Wu, W. Zhang, P. Xu, and Q. Gu. A finite-time analysis of two time-scale actor-critic methods. *Advances in Neural Information Processing Systems*, 33:17617–17628, 2020.

T. Yang, L. Zhang, R. Jin, and J. Yi. Tracking slowly moving clairvoyant: Optimal dynamic regret of online learning with true and noisy gradient. In *International Conference on Machine Learning*, pages 449–457. PMLR, 2016.

L. Zhang, T. Yang, J. Yi, J. Rong, and Z.-H. Zhou. Improved dynamic regret for non-degenerate functions. In *NIPS*, 2017.

L. Zhang, S. Lu, and Z.-H. Zhou. Adaptive online learning in dynamic environments. In *Advances in neural information processing systems*, pages 1323–1333, 2018a.

L. Zhang, T. Yang, Z.-H. Zhou, et al. Dynamic regret of strongly adaptive methods. In *International Conference on Machine Learning*, pages 5882–5891, 2018b.

L. Zhang, T.-Y. Liu, and Z.-H. Zhou. Adaptive regret of convex and smooth functions. In *International Conference on Machine Learning*, pages 7414–7423, 2019.

L. Zhang, S. Lu, and T. Yang. Minimizing dynamic regret and adaptive regret simultaneously. In *International Conference on Artificial Intelligence and Statistics*, pages 309–319. PMLR, 2020.

P. Zhao and L. Zhang. Improved analysis for dynamic regret of strongly convex and smooth functions. In *Learning for Dynamics and Control*, pages 48–59. PMLR, 2021.

P. Zhao, Y.-J. Zhang, L. Zhang, and Z.-H. Zhou. Dynamic regret of convex and smooth functions. *Advances in Neural Information Processing Systems*, 33: 12510–12520, 2020.

K. Zheng, H. Luo, I. Diakonikolas, and L. Wang. Equipping experts/bandits with long-term memory. *Advances in neural information processing systems*, 2019.

W. Zhou, Y. Li, Y. Yang, H. Wang, and T. Hospedales. Online meta-critic learning for off-policy actor-critic methods. *Advances in Neural Information Processing Systems*, 33:17662–17673, 2020a.

Y. Zhou, V. Sanches Portella, M. Schmidt, and N. Harvey. Regret bounds without lipschitz continuity: online learning with relative-lipschitz losses. *Advances in Neural Information Processing Systems*, 33:15823–15833, 2020b.

M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936, 2003.

Checklist

A. For all models and algorithms presented, check if you include:

(a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]

(b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]

(c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

B. For any theoretical claim, check if you include:

(a) Statements of the full set of assumptions of all theoretical results. [Yes]

(b) Complete proofs of all theoretical results. [Yes]

(c) Clear explanations of any assumptions. [Yes]

C. For all figures and tables that present empirical results, check if you include:

(a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]

(b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]

(c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]

(d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

D. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

(a) Citations of the creator If your work uses existing assets. [Yes]

(b) The license information of the assets, if applicable. [Yes]

(c) New assets either in the supplemental material or as a URL, if applicable. [Yes]

(d) Information about consent from data providers/curators. [Yes]

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

E. If you used crowdsourcing or conducted research with human subjects, check if you include:

(a) The full text of instructions given to participants and screenshots. [Not Applicable]

(b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary Materials for Online Bilevel Optimization: Regret Analysis of Online Alternating Gradient Methods

Roadmap. The appendix is organized as follows:

- Appendix A provides some preliminaries on online optimization and a summary of notations used in the appendix.
- Appendix B discusses additional related work on online single-level optimization and offline bilevel optimization.
- Appendix C gives an addendum to Section 3:
  - Appendix C.1 provides the lower bound of OBO (proof of Theorem 5).
  - Appendix C.2 gives the proof for strongly convex OBO with partial information in both dynamic (proof of Theorem 4) and static (proof of Theorem 6) settings.
  - Appendix C.3 gives the proof for convex OBO with partial information in both dynamic (proof of Theorem 7) and static (proof of Theorem 8) settings.
  - Appendix C.4 provides the proof for non-convex OBO with partial information (proof of Theorem 9).
- Appendix D details the implementation and includes additional experiments:
  - Appendix D.1 gives details on hyperparameters learning for dynamic regression.
  - Appendix D.2 gives details on online parametric loss tuning experiments as well as additional experiments.
  - Appendix D.3 provides details of online meta-learning experiments as well as additional experiments.
  - Appendix D.4 presents the numerical sensitivity of algorithms to window size and learning rate.

Table 2: Summary of the Notations

Notation	**Description**
t	Time (round) index
K_t	The number of inner iterations at each round t
T	The total number of rounds
α	Outer stepsize
β	Inner stepsize
$\mathbf{x}_t$	Leader's decision at round t
f_t	Leader's objective at round t
$\mathbf{y}_t$	Follower's decision at round t
g_t	Follower's objective at round t
$\mathbf{x}_t^*$	Leader's optimal decision in the dynamic setting at round t: $\mathbf{x}_t^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x})$
$\mathbf{x}^*$	Leader's optimal decision in static setting: $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$
$\mathbf{y}_t^*(\mathbf{x})$	Follower's optimal decision at round t for a given $\mathbf{x}$
∇h_t, $\nabla^2_{\mathbf{xy}} h_t$, $\nabla^2_{\mathbf{y}} h_t$	Gradient, Jacobian, and Hessian of h_t
$\tilde{\nabla} f_t(\mathbf{x}, \mathbf{y})$	An approximation of the hypergradient $\nabla f_t(\mathbf{x}, \mathbf{y})$
W, w	$W = \sum_{i=0}^{w-1} u_i$ for $\{u_i\}_{i=0}^{w-1}$ with $1 = u_0 \geq u_1 \ldots u_{w-1} > 0$ and window size $w \in [T]$
$\tilde{\nabla} F_{t,\mathbf{u}}(\mathbf{x}, \mathbf{y})$	Approximate time-averaged hypergradient: $\tilde{\nabla} F_{t,\mathbf{u}}(\mathbf{x}, \mathbf{y}) = \frac{1}{W} \sum_{i=0}^{w-1} u_i \tilde{\nabla} f_{t-i}(\mathbf{x}, \mathbf{y})$
$\nabla F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))$	Exact time-averaged hypergradient: $\nabla F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) = \frac{1}{W} \sum_{i=0}^{w-1} u_i \nabla f_{t-i}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))$
$\lVert\cdot\rVert$	The Euclidean norm
$\mathbb{E}[x]$	Expectation of the random variable x
D	The (2-norm) diameter of $\mathcal{X}$: $D = \max_{\mathbf{x},\mathbf{x}'\in\mathcal{X}} \lVert \mathbf{x} - \mathbf{x}' \rVert$
D'	Upper bound on follower's initialization: $\lVert \mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1) \rVert \leq D'$
M	Upper bound on the outer function: $\lvert f_t \rvert \leq M$
M_f	Difference between $\tilde{\nabla} f_t(\mathbf{x}, \mathbf{y}_t)$ and $\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$ w.r.t. $\lVert \mathbf{y}_t^*(\mathbf{x}) - \mathbf{y}_t \rVert$
$L_{\mathbf{y}}$	Lipschitz constant of $\mathbf{y}_t^*(\mathbf{x})$
L_f	Lipschitz constant of $\nabla f_t(\mathbf{x})$
F_T	Outer function value at the optimum: $\sum_{t=1}^{T} f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))$
$P_{p,T}$	Path-length of the outer minimizers: $\sum_{t=2}^{T} \lVert \mathbf{x}_{t-1}^* - \mathbf{x}_t^* \rVert^p$
$Y_{p,T}$	Path-length of the inner minimizers: $\sum_{t=2}^{T} \left\lVert \mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*) \right\rVert^p$
$S_{p,T}$	The summation of the inner and outer path-lengths as $S_{p,T} = P_{p,T} + Y_{p,T}$
$\bar{Y}_{p,T}$	The static variant of $Y_{p,T}$: $\sum_{t=2}^{T} \left\lVert \mathbf{y}_{t-1}^*(\mathbf{x}^*) - \mathbf{y}_t^*(\mathbf{x}^*) \right\rVert^p$
$H_{p,T}$	Inner minimizer function variation: $\sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathbb{R}^{d_1}} \lVert \mathbf{y}_{t-1}^*(\mathbf{x}) - \mathbf{y}_t^*(\mathbf{x}) \rVert^p$
V_T	Online functions variation: $\sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} \lvert f_{t-1}(\mathbf{x}) - f_t(\mathbf{x}) \rvert$
G_T	Online gradients variation: $\sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} \lVert \nabla f_{t-1}(\mathbf{x}) - \nabla f_t(\mathbf{x}) \rVert^2$
D-Reg$_T$	(single-level) dynamic regret: $\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^*)$
S-Reg$_T$	(single-level) static regret: $\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$
L-Reg$_T$	(single-level) local regret: $\sum_{t=1}^{T} \lVert \nabla F_{t,\mathbf{u}}(\mathbf{x}_t) \rVert^2$
BD-Reg$_T$	Bilevel dynamic regret: $\sum_{t=1}^{T} f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))$
BS-Reg$_T$	Bilevel (outer) static regret: $\sum_{t=1}^{T} f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$
BL-Reg$_T$	Bilevel local regret: $\sum_{t=1}^{T} \lVert \nabla F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) \rVert^2$

A Addendum to Section 1: Preliminaries and Notations

We provide several technical lemmas used in the proofs. We start by assembling some well-known facts about convex and smooth functions.

(F1) (**Smoothness**): Suppose $f(\mathbf{x})$ is L-smooth. Then, by definition, the following inequalities hold for any two points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$:

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|,$$

$$f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \mathbf{y} - \mathbf{x}, \nabla f(\mathbf{x})\rangle + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2.$$

Further, if $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x})$, then

$$\|\nabla f(\mathbf{y})\|^2 \leq 2L(f(\mathbf{y}) - f(\mathbf{x}^*)).$$

(F2) (**Smoothness and Convexity**): Suppose $f(\mathbf{x})$ is L-smooth and convex. Then, the following holds for any two points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$:

$$\langle \nabla f(\mathbf{y}) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle \geq \frac{1}{L}\|\nabla f(\mathbf{y}) - \nabla f(\mathbf{x})\|^2.$$

(F3) (**Strong-Convexity**): Suppose $f(\mathbf{x})$ is μ-strongly convex. Then, by definition, the following inequality holds for any two points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$:

$$f(\mathbf{y}) - f(\mathbf{x}) \geq \langle \mathbf{y} - \mathbf{x}, \nabla f(\mathbf{x})\rangle + \frac{\mu}{2}\|\mathbf{y} - \mathbf{x}\|^2.$$

Using the above inequality, one can conclude that

$$\langle \nabla f(\mathbf{y}) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle \geq \mu\|\mathbf{y} - \mathbf{x}\|^2.$$

The following lemma provides the self-bounding property of smooth functions.

Lemma 10. *[Srebro et al., 2010, Lemma 3.1] For a non-negative and L–smooth function $f : \mathcal{X} \to \mathbb{R}$, we have*

$$\|\nabla f(\mathbf{x})\| \leq \sqrt{4Lf(\mathbf{x})}, \ \forall \mathbf{x} \in \mathcal{X}.$$

Lemma 11. *[Nesterov, 2003, Theorem 2.1.11] Let $g : \mathbb{R}^d \to \mathbb{R}$ be a function that is smooth, μ_g-strongly convex, and L_g-gradient Lipschitz continuous on an open convex set $\mathcal{Y} \subseteq \mathbb{R}^d$. Suppose that g has a global minimizer $\mathbf{y}^*$ over $\mathcal{Y}$. Then, the sequence $\{\mathbf{y}_t\}_{t=1}^T$ generated by the gradient descent method $\mathbf{y}_{t+1} = \mathbf{y}_t - \beta\nabla g(\mathbf{y}_t)$ with stepsize $\beta \in (0, 2/(\mu_g + L_g)]$ satisfies*

$$\|\mathbf{y}_{t+1} - \mathbf{y}^*\|^2 \leq \left(1 - \frac{2\beta\mu_g L_g}{\mu_g + L_g}\right)\|\mathbf{y}_t - \mathbf{y}^*\|^2.$$

If $\beta = 2/(\mu_g + L_g)$, then

$$\|\mathbf{y}_{t+1} - \mathbf{y}^*\|^2 \leq \left(\frac{\kappa_g - 1}{\kappa_g + 1}\right)^2\|\mathbf{y}_t - \mathbf{y}^*\|^2,$$

where $\kappa_g = L_g/\mu_g$.

Lemma 12. *For any set of vectors $\{\mathbf{x}_i\}_{i=1}^m$ with $\mathbf{x}_i \in \mathbb{R}^d$, we have*

$$\left\|\sum_{i=1}^m \mathbf{x}_i\right\|^2 \leq m\sum_{i=1}^m \|\mathbf{x}_i\|^2.$$

Lemma 13. *For all $T \in \mathbb{N}$,*

I. $\log(T) + \frac{1}{T} \leq \sum_{t=1}^T \frac{1}{t} \leq \log(T) + 1$;

II. $\sum_{t=1}^T \frac{1}{\sqrt{t}} \leq 2\sqrt{T}$;

III. *If* $1 < s < \infty$, *then* $\xi(s) = \sum_{t=1}^{T} 1/t^s$ *is called the Riemann* ξ*-function and we have*

$$\xi(2n) = (-1)^{n+1}\frac{(2\pi)^{2n}B_{2n}}{2(2n)!}, \qquad n = 1, 2, 3, \ldots,$$

where the coefficients B_{2n} *are the Bernoulli numbers.*

Lemma 14. *For any* $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^d$, *the following holds for any* $c > 0$:

$$\|\mathbf{x} + \mathbf{y}\|^2 \leq (1 + c)\|\mathbf{x}\|^2 + \left(1 + \frac{1}{c}\right)\|\mathbf{y}\|^2, \quad \text{and}$$
$$\|\mathbf{x} - \mathbf{y}\|^2 \geq (1 - c)\|\mathbf{x} - \mathbf{z}\|^2 + \left(1 - \frac{1}{c}\right)\|\mathbf{z} - \mathbf{y}\|^2.$$

Lemma 15. *[Shalev-Shwartz et al., 2011, Lemma 2.8] Let* $\mathcal{X} \subseteq \mathbb{R}^d$ *be a nonempty convex set. Let* $f(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$ *be a* μ_f*-strongly convex function over* $\mathcal{X}$. *Let* $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathcal{X}}\{f(\mathbf{x})\}$. *Then, for any* $\mathbf{y} \in \mathcal{X}$, *we have*

$$f(\mathbf{x}^*) - f(\mathbf{y}) \leq -\frac{\mu_f}{2}\|\mathbf{y} - \mathbf{x}^*\|^2.$$

A.1 On the Comparability of Dynamic Metrics

The following example shows that $P_{p,T}$, $Y_{p,T}$, and $\bar{Y}_{p,T}$ are not comparable in general and all three measures play a key role in OBO.

Example 1. *Let* $x \in \mathcal{X} = [-1, 1] \subset \mathbb{R}$, $y \in \mathbb{R}$ *and consider a sequence of quadratic cost functions*

$$f_t(x, y) = \frac{1}{2}\left(x + 2a_t^{(1)}\right)^2 + \frac{1}{2}\left(y - a_t^{(2)}\right)^2 + a_t^{(3)},$$
$$g_t(x, y) = \frac{1}{2}y^2 - \left(x - a_t^{(2)}\right)y + a_t^{(4)},$$

for all $t \in [T]$, *where* $\{a_t^{(i)}\}_{i=1}^4$ *are some time-varying constants.*

It follows from (3b) *that*

$$y_t^*(x_t) = x_t - a_t^{(2)}, \qquad x_t^* = -a_t^{(1)} + a_t^{(2)}, \qquad y_t^*(x_t^*) = a_t^{(1)}.$$

Let $a_t^{(2)} = (-1)^t/\sqrt{t}$ *for all* $t \in [T]$.

- *If* $a_t^{(1)} = a_t^{(2)}$, *then* $P_{1,T} = P_{2,T} = 0$, $Y_{1,T} = \bar{Y}_{1,T} = \mathcal{O}(\sqrt{T})$, *and* $Y_{2,T} = \bar{Y}_{2,T} = \mathcal{O}(\log T)$.
- *If* $a_t^{(1)} = 0$, *then* $P_{1,T} = \mathcal{O}(\sqrt{T})$, $P_{2,T} = \mathcal{O}(\log T)$, *and* $Y_{1,T} = Y_{2,T} = 0$.

This shows that $S_{1,T} = P_{1,T} + Y_{2,T}$, $S_{2,T} = P_{2,T} + Y_{2,T}$ *are not comparable in general. Similarly, static metrics* $\bar{Y}_{1,T}$ *and* $\bar{Y}_{2,T}$ *are not comparable.*

B Addendum to Section 2: Additional Related Work

Online learning and stochastic optimization are closely related. The key difference between them is that at each round t of the online optimization, the loss function can be arbitrarily chosen by the adversary. Given the vastness of the online and stochastic optimization literature, we do not strive to provide an exhaustive review. Instead, we mainly focus on a few representative works on online static and *worst-case* dynamic regret minimization, as well as bilevel optimization. Refer to [Hazan, 2016a, Hoi et al., 2021] and [Liu et al., 2021, Sinha et al., 2017] for surveys on online and bilevel optimization, respectively.

Static Regret Minimization: In single-level online optimization, the goal of the player (learner) is to choose a sequence $\{\mathbf{x}_t\}_{t=1}^T$ such that their regret is minimized. There are different notions of regret in the literature, including static, dynamic (defined in (1)), and adaptive [Hazan, 2016b, Shalev-Shwartz, 2007, Shalev-Shwartz et al., 2011]. In the case of static regret, $\mathbf{x}_t^*$ is replaced by $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^T f_t(\mathbf{x})$. This type of regret is well-studied in the literature of online learning [Hazan, 2016b, Shalev-Shwartz, 2007, Shalev-Shwartz et al., 2011]. Zinkevich [2003] shows that online gradient descent (OGD) provides an $\mathcal{O}(\sqrt{T})$ regret bound for convex (possibly nonsmooth) functions. Hazan et al. [2007] improve this bound to $\mathcal{O}(\log T)$ for strongly-convex functions. These results were also shown to be minimax optimal [Abernethy et al., 2008]. Zhou et al. [2020b] provide regret bounds for online learning algorithms under relative Lipschitz and/or relative strongly-convexity assumptions.

In addition to exploiting the convexity of online functions, recent studies have focused on improving static regret by incorporating smoothness [Chiang et al., 2012, Srebro et al., 2010]. These problem-dependent bounds can safeguard the worst-case minimax rate, yet they can be much better in easy cases of online learning problems (e.g., loss functions with a small deviation). For instance, [Srebro et al., 2010] shows that for convex smooth non-negative functions, OGD can achieve an $\mathcal{O}(1+\sqrt{F_T})$ small-loss regret bound, where $F_T = \sum_{t=1}^T f_t(\mathbf{x}^*)$ and $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^T f_t(\mathbf{x})$. For convex smooth functions, [Chiang et al., 2012] establishes an $\mathcal{O}(1+\sqrt{G_T})$ bound, where $G_T = \sum_{t=2}^T \sup_{\mathbf{x}\in\mathcal{X}} \|\nabla f_{t-1}(\mathbf{x}) - \nabla f_t(\mathbf{x})\|^2$ is the gradient variation. These bounds are particularly favored in slowly changing environments in which the online functions evolve gradually [Zhao et al., 2020].

Dynamic Regret Minimization: *Single-level* dynamic regret forces the player to compete with time-varying comparators, and thus is particularly favored in non-stationary environments [Sugiyama and Kawanabe, 2012]. The notion of dynamic regret is also referred to as tracking regret or shifting regret in the prediction with expert advice setting [Bousquet and Warmuth, 2002, Herbster and Warmuth, 1998, 2001, Wei et al., 2016, Zheng et al., 2019]. There are two kinds of dynamic regret in previous studies: The *universal dynamic regret* aims to compare with any feasible comparator sequence [Zhang et al., 2018a, Zhao et al., 2020, Zinkevich, 2003], while the *worst-case dynamic regret* (defined in (1)) specifies the comparator sequence to be the sequence of minimizers of online functions [Aydore et al., 2019, Besbes et al., 2015, Jadbabaie et al., 2015, Mokhtari et al., 2016, Yang et al., 2016, Zhang et al., 2017, Nazari and Khorram, 2021]. We present related works for the latter case as it is the setting studied in this paper.

It is known that in the worst case, sublinear dynamic regret is not attainable unless one imposes regularity of some form on the comparator sequence or the function sequence [Besbes et al., 2015, Hall and Willett, 2015, Jadbabaie et al., 2015]. Yang et al. [2016] shows that OGD enjoys an $\mathcal{O}(1+\sqrt{TP_{1,T}})$ worst-case dynamic regret bound for convex functions when the path-length $P_{1,T}$ is known. For strongly convex and smooth functions, [Mokhtari et al., 2016] shows that an $\mathcal{O}(1+P_{1,T})$ dynamic regret bound is achievable. Chang and Shahrampour [2021] proves that OGD can achieve an $\mathcal{O}(1+P_{2,T})$ regret bound without the bounded gradient assumption. Zhang et al. [2017] further proposes the online multiple gradient descent algorithm and proves that the algorithm enjoys an $\mathcal{O}(1+\min\{P_{1,T}, P_{2,T}\})$ regret bound; this bound has been recently enhanced to $\mathcal{O}(1+\min\{P_{1,T}, P_{2,T}, V_T\})$ by an improved analysis [Zhao and Zhang, 2021], where $V_T = \sum_{t=2}^T \sup_{\mathbf{x}\in\mathcal{X}} |f_{t-1}(\mathbf{x}) - f_t(\mathbf{x})|$. Yang et al. [2016] further shows that the $\mathcal{O}(1+P_{2,T})$ rate is attainable for convex and smooth functions, provided that all the minimizers $\mathbf{x}_t^*$ lie in the interior of the domain $\mathcal{X}$. The above results use path-length (or squared path-length) as the regularity, which is in terms of the trajectory of the comparator sequence. Nazari et al. [2019, 2022b] extend the above results to the distributed settings and provide dynamic regret bounds in terms of the ℓ_1 path-length. Besbes et al. [2015] shows that OGD with a restarting strategy attains an $\mathcal{O}(1+T^{2/3}{V_T}^{1/3})$ regret for convex functions when V_T is available, which has been recently improved to $\mathcal{O}(1+T^{1/3}{V_T}^{2/3})$ for the square loss [Baby and Wang, 2019].

Adaptive Regret: Adaptive regret [Daniely et al., 2015, Hazan and Seshadhri, 2007, Zhang et al., 2019, 2020, 2018b] is also used to capture the dynamics in the environment. Specifically, it characterizes a local version of static regret, where

$$\text{Regret}_T([r,s]) \triangleq \sum_{t=r}^{s} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=r}^{s} f_t(\mathbf{x}),$$

for each interval $[r,s] \subseteq [T]$. Zhang et al. [2018b] provide a connection between strongly adaptive regret and dynamic regret and proposes an adaptive algorithm that can bound the dynamic regret without prior knowledge of the functional variation. Zhang et al. [2020] develop a new algorithm that can minimize the dynamic regret and the adaptive regret simultaneously.

Local Regret Minimization: Non-convex online optimization is a more challenging setting than the convex case. Some notable works in the non-convex literature include adversarial multi-armed bandit with a continuum of arms [Bubeck et al., 2008, Hériou et al., 2020, 2021, Krichene et al., 2015] and classical Follow-the-Perturbed-Leader algorithm with access to an offline non-convex optimization oracle [Agarwal et al., 2019, Kleinberg et al., 2008, Suggala and Netrapalli, 2020]. Hazan et al. [2017] introduces a local regret measure based on gradients of the loss to address intractable non-convex online models. Their regret is local in the sense that it averages a sliding window of gradients and quantifies the objective of predicting points with small gradients on average. They are motivated by a game-theoretic perspective, where an adversary reveals observations from an unknown static loss. The gradients of the loss functions from the w most recent rounds of play are evaluated at the current model parameters $\mathbf{x}_t$, and these gradients are then averaged. The motivation behind averaging is two-fold: (i) a randomly selected update has a small time-averaged gradient in expectation if an algorithm incurs local regret sublinear in T, and (ii) for any online algorithm, an adversarial sequence of loss functions can force the local regret incurred to scale with T as $\mathcal{O}(T/w^2)$. Hallak et al. [2021] extends the local regret minimization to online, non-smooth, non-convex problems. These arguments, presented in [Aydore et al., 2019, Hallak et al., 2021, Hazan et al., 2017, Nazari et al., 2022b], inspire our use of local regret for OBO.

(Offline) Bilevel Optimization: Since its first formulation by Stackelberg [von Stackelberg, 1952] and the first mathematical model by Bracken and McGill [Bracken and McGill, 1973], there has been significant growth in the applications and developments of bilevel programming. Existing works either reduce the problem to a single-level optimization problem [Aiyoshi and Shimizu, 1984, Al-Khayyal et al., 1992, Edmunds and Bard, 1991, Hansen et al., 1992, Lv et al., 2007, Moore, 2010, Shi et al., 2005, Sinha et al., 2017], or apply (alternating) optimization methods to solve the original problem. The single-level formulations, which employ the Karush-Kuhn-Tucker (KKT) conditions or penalty approaches, are generally difficult to solve [Sinha et al., 2017].

Gradient-based approaches are more attractive for bilevel programming due to their simplicity and effectiveness. This type of approach estimates the hypergradients for iterative updates, and can generally be divided into two categories: approximate implicit differentiation (AID) and iterative differentiation (ITD) classes. ITD-based approaches [Finn et al., 2017, Franceschi et al., 2017, Grazzi et al., 2020, Maclaurin et al., 2015] estimate the hypergradient either in reverse (automatic differentiation) or forward manner. AID-based approaches [Domke, 2012, Grazzi et al., 2020, Ji et al., 2021b, Pedregosa, 2016] estimate the hypergradient via implicit differentiation. Franceschi et al. [2018] characterized the asymptotic convergence of a backpropagation-based approach as one of ITD-based algorithms by assuming the inner-level problem is strongly convex. Shaban et al. [2019] provided a similar analysis for a truncated backpropagation scheme. Li et al. [2020], Liu et al. [2020] analyzed the asymptotic performance of ITD-based approaches when the inner-level problem is convex.

Finite-time complexity analysis for bilevel optimization has also been explored. Ghadimi and Wang [2018] provided a finite-time convergence analysis for an AID-based algorithm under various loss geometries: the outer function being strongly convex, convex, or non-convex, while the inner function remains strongly convex. Ji et al. [2021b] provided an improved finite-time analysis for both AID- and ITD-based algorithms under the nonconvex-strongly-convex geometry. Liang et al. [2023] provided the lower bounds on complexity as well as upper bounds under these two geometries. When the objective functions can be expressed in an expected or finite-time form, [Ghadimi and Wang, 2018, Hong et al., 2023, Ji et al., 2021b] developed stochastic bilevel algorithms and provided the finite-time analysis. There have been subsequent studies on accelerating SGD-type bilevel optimization via momentum and variance reduction techniques [Chen et al., 2022, Guo et al., 2021, Huang and Huang, 2021, Ji et al., 2021a] as well. However, a fundamental assumption in all the aforementioned works is that the cost function does *not* change throughout the horizon over which we seek to optimize it.

C Addendum to Section 3: Proof of Main Theorems

C.1 Proof of Theorem 5

Proof. We randomly generate a sequence of functions $\{(f_t, g_t)\}_{t=1}^T$ and show that there exists a distribution of online functions such that for any bilevel algorithm $\mathcal{A}$, we have $\mathbb{E}\left[\text{BD-Reg}_T\right] \geq \mathbb{E}[S_{2,T}]$. Specifically, for any bilevel algorithm $\mathcal{A}$ that generates a sequence of $(x_t, y_t^*) \in \mathbb{R} \times \mathbb{R}$ for all $t \in [T]$, we consider the expected regret as follows:

$$\mathbb{E}\left[\text{BD-Reg}_T\right] = \mathbb{E}\left[\sum_{t=1}^{T} f_t(x_t, y_t^*(x_t)) - \sum_{t=1}^{T} f_t(x_t^*, y_t^*(x_t^*))\right].$$

For each round t, we randomly sample $a_t^{(1)}, a_t^{(2)} \in \mathbb{R}$ from the Gaussian distribution $\mathcal{N}(0,1)$. For all $t \in [T]$, let

$$\begin{aligned} & f_t(x, y_t^*(x)) = 6\left(y_t^*(x) - \left(a_t^{(1)} + a_t^{(2)}\right)\right)^2 + 6\left(x - a_t^{(1)}\right)^2, \\ & \text{s.t.} \quad y_t^*(x) \in \underset{y\in\mathbb{R}}{\arg\min}\, g_t(x,y) = \frac{1}{2}y^2 - \left(x + a_t^{(2)}\right)y. \end{aligned}$$

It follows from (3b) that

$$y_t^*(x_t) = x_t + a_t^{(2)}, \quad x_t^* = a_t^{(1)}, \quad \text{and} \quad y_t^*(x_t^*) = a_t^{(1)} + a_t^{(2)}.$$

Notice that x_t is independent from $a_t^{(1)}$. Hence,

$$\begin{aligned} \mathbb{E}\left[\text{BD-Reg}_T\right] &= 6\sum_{t=1}^{T} \mathbb{E}\left[\left(x_t - a_t^{(1)}\right)^2\right] \\ &= 6\sum_{t=1}^{T}\left(\mathbb{E}\left[\left(a_t^{(1)}\right)^2\right] + \mathbb{E}\left[x_t^2\right]\right) \\ &\geq 6T. \end{aligned} \tag{20}$$

For $T \geq 2$, we obtain

$$\begin{aligned} \mathbb{E}\left[S_{2,T}\right] = \mathbb{E}[P_{2,T}] + \mathbb{E}[Y_{2,T}] &= \sum_{t=2}^{T}\mathbb{E}\left[\left(x_{t-1}^* - x_t^*\right)^2\right] + \sum_{t=2}^{T}\mathbb{E}\left[\left(y_{t-1}^*(x_{t-1}^*) - y_t^*(x_t^*)\right)^2\right] \\ &= \sum_{t=2}^{T}\mathbb{E}\left[\left(a_t^{(1)} - a_{t-1}^{(1)}\right)^2\right] + \sum_{t=2}^{T}\mathbb{E}\left[\left(a_t^{(1)} + a_t^{(2)} - (a_{t-1}^{(1)} + a_{t-1}^{(2)})\right)^2\right] \\ &= \sum_{t=2}^{T} 2\left(\mathbb{E}\left[\left(a_t^{(1)}\right)^2\right] + \mathbb{E}\left[\left(a_{t-1}^{(1)}\right)^2\right]\right) + \mathbb{E}\left[\left(a_t^{(2)}\right)^2\right] + \mathbb{E}\left[\left(a_{t-1}^{(2)}\right)^2\right] \\ &\leq 6(T-1). \end{aligned} \tag{21}$$

Here, the third equality follows from the independence of $a_t^{(1)}$ and $a_t^{(2)}$ for all $t \in [T]$.

Now, it follows from (20) and (21) that $\mathbb{E}\left[\text{BD-Reg}_T\right] \geq \mathbb{E}[S_{2,T}]$. This completes the proof of Theorem 5. ■

C.2 Proof for Strongly Convex OBO with Partial Information

In this section, we provide the dynamic regret bound for strongly convex OBO with partial information. Specifically, we derive a problem-dependent regret bound for Algorithm 1.

C.2.1 Auxiliary Lemmas

Lemma 16 (Restatement of Lemma 3). *Under Assumption A, for all $t \in [T]$, $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, and $\mathbf{y} \in \mathbb{R}^{d_2}$, we have*

$$\left\|\mathbf{y}_t^*(\mathbf{x}) - \mathbf{y}_t^*(\mathbf{x}')\right\| \leq L_{\mathbf{y}}\left\|\mathbf{x} - \mathbf{x}'\right\|, \tag{22}$$

$$\left\|\tilde{\nabla} f_t(\mathbf{x}, \mathbf{y}) - \nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))\right\| \leq M_f\left\|\mathbf{y} - \mathbf{y}_t^*(\mathbf{x})\right\|, \tag{23}$$

$$\left\|\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) - \nabla f_t(\mathbf{x}', \mathbf{y}_t^*(\mathbf{x}'))\right\| \leq L_f\left\|\mathbf{x} - \mathbf{x}'\right\|. \tag{24}$$

Here, $L_{\mathbf{y}}$, M_f, and L_f are defined in (25), (30), and (32), respectively.

Proof. The proof is an adaptation of the proof from [Ghadimi and Wang, 2018, Lemma 2.2] to the online setting. We first show (22). Since $\mathbf{y}_t^*(\mathbf{x}) \in \arg\min_{\mathbf{y}\in\mathbb{R}^{d_2}} g_t(\mathbf{x},\mathbf{y})$, we have

$$\nabla_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)=0, \quad \text{and} \quad \nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)=0.$$

This together with the chain rule implies that

$$\nabla \mathbf{y}_t^*(\mathbf{x})\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)+\nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)=0.$$

It follows from Assumption A2. that $\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)$ is positive definite. Hence,

$$\nabla \mathbf{y}_t^*(\mathbf{x})=-\nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right)^{-1}.$$

Now, from Assumption A3., we get

$$\begin{aligned}\|\nabla \mathbf{y}_t^*(\mathbf{x})\| &= \left\|\nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right)^{-1}\right\| \\ &\le \left\|\nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right\|\left\|\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right)^{-1}\right\| \\ &\le \frac{\ell_{g,1}}{\mu_g} =: L_{\mathbf{y}}.\end{aligned} \tag{25}$$

Next, we show (23). Let $\mathbf{M}_t(\mathbf{x},\mathbf{y}) := \nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}\right)\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}\right)\right)^{-1}$. Define

$$\begin{aligned}
\Delta_t &:= \nabla f_t(\mathbf{x},\mathbf{y})-\nabla f_t(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})),\\
\Delta_t^1 &:= \nabla_{\mathbf{x}} f_t\left(\mathbf{x},\mathbf{y}\right)-\nabla_{\mathbf{x}} f_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right),\\
\Delta_t^2 &:= \mathbf{M}_t(\mathbf{x},\mathbf{y})\nabla_{\mathbf{y}} f_t\left(\mathbf{x},\mathbf{y}\right)-\mathbf{M}_t(\mathbf{x},\mathbf{y}_t^*(\mathbf{x}))\nabla_{\mathbf{y}} f_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right),\\
\Delta_t^3 &:= \mathbf{M}_t(\mathbf{x},\mathbf{y})\{\nabla_{\mathbf{y}} f_t\left(\mathbf{x},\mathbf{y}\right)-\nabla f_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\},\\
\Delta_t^4 &:= \{\mathbf{M}_t(\mathbf{x},\mathbf{y})-\mathbf{M}_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\}\nabla_{\mathbf{y}} f_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right),\\
\Delta_t^5 &:= \left\{\nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}\right)-\nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right\}\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}\right)\right)^{-1},\\
\Delta_t^6 &:= \nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\left\{\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}\right)\right)^{-1}-\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right)^{-1}\right\},
\end{aligned}$$

which implies that

$$\Delta_t=\Delta_t^1-\Delta_t^2=\Delta_t^1-\Delta_t^3-\Delta_t^4=\Delta_t^1-\Delta_t^3-(\Delta_t^5+\Delta_t^6)\nabla_{\mathbf{y}} f_t(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})). \tag{26}$$

From Assumption A, we have

$$\|\Delta_t^1\|\le \ell_{f,1}\left\|\mathbf{y}-\mathbf{y}_t^*(\mathbf{x})\right\|, \quad \|\Delta_t^3\|\le \frac{\ell_{f,1}\ell_{g,1}}{\mu_g}\left\|\mathbf{y}-\mathbf{y}_t^*(\mathbf{x})\right\|, \quad \|\Delta_t^5\|\le \frac{\ell_{g,2}}{\mu_g}\left\|\mathbf{y}-\mathbf{y}_t^*(\mathbf{x})\right\|. \tag{27}$$

Note that, for any invertible matrices $\mathbf{A}_1$ and $\mathbf{A}_2$, we have

$$\|\mathbf{A}_2^{-1}-\mathbf{A}_1^{-1}\|=\|\mathbf{A}_1^{-1}(\mathbf{A}_1-\mathbf{A}_2)\mathbf{A}_2^{-1}\|\le\|\mathbf{A}_1^{-1}\|\|\mathbf{A}_2^{-1}\|\|\mathbf{A}_1-\mathbf{A}_2\|,$$

which implies that

$$\begin{aligned}\|\Delta_t^6\| &\le \left\|\nabla^2_{\mathbf{xy}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right\|\left\|\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}\right)\right)^{-1}-\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right)^{-1}\right\| \\ &\le \ell_{g,1}\left\|\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}\right)\right)^{-1}\right\|\left\|\left(\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right)^{-1}\right\|\left\|\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}\right)-\nabla^2_{\mathbf{y}} g_t\left(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})\right)\right\| \\ &\le \frac{\ell_{g,1}\ell_{g,2}}{\mu_g^2}\left\|\mathbf{y}-\mathbf{y}_t^*(\mathbf{x})\right\|.\end{aligned} \tag{28}$$

Therefore, by substitution (27) and (28) into (26), we have

$$\begin{aligned}\|\nabla f_t(\mathbf{x},\mathbf{y})-\nabla f_t(\mathbf{x},\mathbf{y}_t^*(\mathbf{x}))\| &\le \|\Delta_t^1\|+\|\Delta_t^3\|+\|\Delta_t^5+\Delta_t^6\|\|\nabla_{\mathbf{y}} f_t(\mathbf{x},\mathbf{y}_t^*(\mathbf{x}))\| \\ &\le \|\Delta_t^1\|+\|\Delta_t^3\|+\|\Delta_t^5+\Delta_t^6\|\ell_{f,0} \\ &\le M_f\left\|\mathbf{y}-\mathbf{y}_t^*(\mathbf{x})\right\|,\end{aligned} \tag{29}$$

where

$$M_f := \ell_{f,1} + \frac{\ell_{g,1}\ell_{f,1}}{\mu_g} + \frac{\ell_{f,0}}{\mu_g}\Big(\ell_{g,2} + \frac{\ell_{g,1}\ell_{g,2}}{\mu_g}\Big). \tag{30}$$

Next, we show (24). Note that

$$\begin{aligned}\|\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) - \nabla f_t(\mathbf{x}', \mathbf{y}_t^*(\mathbf{x}'))\| \leq & \left\|\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) - \tilde{\nabla} f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}'))\right\| \\ & + \left\|\tilde{\nabla} f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}')) - \nabla f_t(\mathbf{x}', \mathbf{y}_t^*(\mathbf{x}'))\right\|.\end{aligned} \tag{31}$$

We then study each terms separately. From (29) and (22), we get

$$\left\|\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) - \tilde{\nabla} f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}'))\right\| \leq M_f \|\mathbf{y}_t^*(\mathbf{x}) - \mathbf{y}_t^*(\mathbf{x}')\| \leq M_f L_{\mathbf{y}} \|\mathbf{x} - \mathbf{x}'\|.$$

Moreover, by similar argument to (29), we obtain

$$\|\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}')) - \nabla f_t(\mathbf{x}', \mathbf{y}_t^*(\mathbf{x}'))\| \leq \left(\ell_{f,1} + \frac{\ell_{g,1}\ell_{f,1}}{\mu_g} + \frac{\ell_{f,0}}{\mu_g}\left(\ell_{g,2} + \frac{\ell_{g,1}\ell_{g,2}}{\mu_g}\right)\right)\|\mathbf{x} - \mathbf{x}'\|.$$

By substituting the above two inequalities into (31), we have

$$\|\nabla f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) - \nabla f_t(\mathbf{x}', \mathbf{y}_t^*(\mathbf{x}'))\| \leq L_f \|\mathbf{x} - \mathbf{x}'\|,$$

where

$$L_f := \ell_{f,1} + \frac{\ell_{g,1}(\ell_{f,1} + M_f)}{\mu_g} + \frac{\ell_{f,0}}{\mu_g}\left(\ell_{g,2} + \frac{\ell_{g,1}\ell_{g,2}}{\mu_g}\right). \tag{32}$$

■

The following lemma characterizes the inner estimation error $\sum_{t=1}^T \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^p$, where $\mathbf{y}_{t+1}$ is the inner variable updated via Algorithm 1. It shows that by applying inner OGD multiple times at each round t, we are able to extract more information from each inner function and, therefore, are more likely to obtain a tight bound for the inner error in terms of the path-length $Y_{p,T}$.

Lemma 17. *Suppose Assumption A holds. In Algorithm 1, choose*

$$\beta_t = \beta = \frac{2}{\ell_{g,1} + \mu_g}, \quad \text{and} \quad K_t = \left\lceil \frac{(\kappa_g + 1)\log \rho_t^{-2}}{4} \right\rceil$$

for some positive deceasing sequence $\{\rho_t\}_{t=1}^T$. Then, Algorithm 1 guarantees the following.

L1. *If $\rho_1 < \sqrt{1/2}$, we have*

$$\begin{aligned}\sum_{t=1}^T \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq & \frac{\rho_1^2}{1 - 2\rho_1^2}\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 \\ & + \frac{6}{1 - 2\rho_1^2}\left(2L_{\mathbf{y}}^2 \sum_{t=1}^T \rho_t^2 \|\mathbf{x}_t - \mathbf{x}_t^*\|^2 + \sum_{t=2}^T \rho_t^2 \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*)\|^2\right).\end{aligned}$$

L2. *If $\rho_1 < 1$, we get*

$$\begin{aligned}\sum_{t=1}^T \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| \leq & \frac{\rho_1}{1 - \rho_1}\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| \\ & + \frac{1}{1 - \rho_1}\left(2L_{\mathbf{y}} \sum_{t=1}^T \rho_t \|\mathbf{x}_t - \mathbf{x}_t^*\| + \sum_{t=2}^T \rho_t \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*)\|\right).\end{aligned}$$

Proof. We show L1.. The proof of L2. follows similarly. Since $\beta = 2/(\ell_{g,1} + \mu_g)$, from Lemma 11, we have

$$\|\mathbf{z}_t^{K_t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq \left(1 - \frac{2}{\kappa_g + 1}\right)^2 \|\mathbf{z}_t^{K_t} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2,$$

which implies that

$$\|\mathbf{z}_t^{K_t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq \left(1 - \frac{2}{\kappa_g + 1}\right)^{2K_t} \|\mathbf{z}_t^{1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2. \tag{33}$$

By our assumption $K_t = \lceil 0.25(\kappa_g + 1) \log \rho_t^{-2} \rceil$ which implies that

$$\left(1 - \frac{2}{\kappa_g + 1}\right)^{2K_t} \leq \exp\left(-\frac{4K_t}{\kappa_g + 1}\right) \leq \rho_t^2. \tag{34}$$

Then, using (33) and (34), we have

$$\|\mathbf{z}_t^{K_t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 = \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq \rho_t^2 \|\mathbf{y}_t - \mathbf{y}_t^*(\mathbf{x}_t)\|^2.$$

Hence,

$$\sum_{t=1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq \rho_1^2 \|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 + \sum_{t=2}^{T} \rho_t^2 \|\mathbf{y}_t - \mathbf{y}_t^*(\mathbf{x}_t)\|^2, \tag{35}$$

which implies that

$$\begin{aligned}\sum_{t=2}^{T} \rho_t^2 \|\mathbf{y}_t - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 &\leq 2 \sum_{t=2}^{T} \rho_t^2 \left(\|\mathbf{y}_t - \mathbf{y}_{t-1}^*(\mathbf{x}_{t-1})\|^2 + \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}) - \mathbf{y}_t^*(\mathbf{x}_t)\|^2\right) \\ &\leq 2 \sum_{t=1}^{T} \rho_t^2 \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 + 2 \sum_{t=2}^{T} \rho_t^2 \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}) - \mathbf{y}_t^*(\mathbf{x}_t)\|^2.\end{aligned} \tag{36}$$

It follows from Lemma 12 that

$$\begin{aligned}\rho_t^2 \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}) - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 &\leq 3\rho_t^2 \|\mathbf{y}_t^*(\mathbf{x}_t) - \mathbf{y}_t^*(\mathbf{x}_t^*)\|^2 \\ &+ 3\rho_t^2 \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}) - \mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*)\|^2 \\ &+ 3\rho_t^2 \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*)\|^2 \\ &\leq 3L_{\mathbf{y}}^2 \rho_{t-1}^2 \|\mathbf{x}_{t-1} - \mathbf{x}_{t-1}^*\|^2 \\ &+ 3L_{\mathbf{y}}^2 \rho_t^2 \|\mathbf{x}_t - \mathbf{x}_t^*\|^2 \\ &+ 3\rho_t^2 \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*)\|^2,\end{aligned} \tag{37}$$

where the second inequality uses the assumption that $\rho_t \leq \rho_{t-1}$ for all $t \in [T]$.

Now, combining (35), (36), and (37), we obtain

$$\begin{aligned}\sum_{t=1}^{T} \left(1 - 2\rho_t^2\right) \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq \rho_1^2 \|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 + 12L_{\mathbf{y}}^2 \sum_{t=1}^{T} \rho_t^2 \|\mathbf{x}_t - \mathbf{x}_t^*\|^2 \\ + 6 \sum_{t=2}^{T} \rho_t^2 \|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*)\|^2,\end{aligned}$$

which together with our assumption that $\rho_t \leq \rho_{t-1}$ completes the proof. ■

The following lemma is an extension of [Mokhtari et al., 2016, Proposition 2] to (online) bilevel optimization, characterizing the dynamics of the tracking error $\|\mathbf{x}_t - \mathbf{x}_t^*\|^2$. Specifically, it shows that $\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\|^2$ can be upper bounded in terms of $\|\mathbf{x}_t - \mathbf{x}_t^*\|^2$ and $\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2$.

Lemma 18. *Suppose Assumption A holds and $\alpha_t = \alpha \leq 1/\ell_{f,1}$. Further, assume functions $\{f_t\}_{t=1}^T$ are strongly convex with parameter μ_f. Then, for the sequence $\{(\mathbf{x}_t, \mathbf{y}_t)\}_{t=1}^T$ generated by Algorithm 1, we have*

$$\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\|^2 \leq (1-\gamma)\|\mathbf{x}_t - \mathbf{x}_t^*\|^2 + \frac{2M_f^2\alpha}{(1+\frac{\mu_f}{2}\alpha)\mu_f} \left\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\right\|^2, \tag{38}$$

where

$$\gamma := \frac{3\mu_f}{\frac{2}{\alpha} + \mu_f} \in (0,1]. \tag{39}$$

Proof. From μ_f-strong convexity of f_t, we get

$$\begin{aligned} f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) &\geq f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) + \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x} - \mathbf{x}_t \rangle + \frac{\mu_f}{2}\left\|\mathbf{x} - \mathbf{x}_t\right\|^2 \\ &= f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) + \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle \\ &+ \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x} - \mathbf{x}_{t+1} \rangle + \frac{\mu_f}{2}\left\|\mathbf{x} - \mathbf{x}_t\right\|^2. \end{aligned} \tag{40}$$

According to the optimality condition of the update rule $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}\left[\mathbf{x}_t - \alpha\tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1})\right]$, we have

$$\langle \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}) + \frac{1}{\alpha}(\mathbf{x}_{t+1} - \mathbf{x}_t), \mathbf{x} - \mathbf{x}_{t+1} \rangle \geq 0,$$

which is equivalent to

$$\langle \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}) - \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) + \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x} - \mathbf{x}_{t+1} \rangle \geq \frac{1}{\alpha}\langle \mathbf{x}_t - \mathbf{x}_{t+1}, \mathbf{x} - \mathbf{x}_{t+1} \rangle.$$

Hence,

$$\begin{aligned} \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x} - \mathbf{x}_{t+1} \rangle &\geq \frac{1}{\alpha}\langle \mathbf{x}_t - \mathbf{x}_{t+1}, \mathbf{x} - \mathbf{x}_{t+1} \rangle \\ &\quad + \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}), \mathbf{x} - \mathbf{x}_{t+1} \rangle. \end{aligned}$$

Substituting this inequality in (40), we get

$$\begin{aligned} f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x})) \geq f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) &+ \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle + \frac{1}{\alpha}\langle \mathbf{x}_t - \mathbf{x}_{t+1}, \mathbf{x} - \mathbf{x}_{t+1} \rangle \\ &+ \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}), \mathbf{x} - \mathbf{x}_{t+1} \rangle + \frac{\mu_f}{2}\left\|\mathbf{x} - \mathbf{x}_t\right\|^2. \end{aligned} \tag{41}$$

In addition, $\ell_{f,1}$-smoothness of f_t (Assumption A3.) gives

$$\begin{aligned} f_t(\mathbf{x}_{t+1}, \mathbf{y}_t^*(\mathbf{x}_{t+1})) &\leq f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) + \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle + \frac{\ell_{f,1}}{2}\left\|\mathbf{x}_{t+1} - \mathbf{x}_t\right\|^2 \\ &\leq f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) + \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle + \frac{1}{2\alpha}\left\|\mathbf{x}_{t+1} - \mathbf{x}_t\right\|^2, \end{aligned}$$

where the inequality is by $\alpha \leq 1/\ell_{f,1}$.

Thus,

$$\begin{aligned} &f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) + \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle \\ &\geq f_t(\mathbf{x}_{t+1}, \mathbf{y}_t^*(\mathbf{x}_{t+1})) - \frac{1}{2\alpha}\left\|\mathbf{x}_{t+1} - \mathbf{x}_t\right\|^2 \\ &\geq f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) + \frac{\mu_f}{2}\left\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\right\|^2 - \frac{1}{2\alpha}\left\|\mathbf{x}_{t+1} - \mathbf{x}_t\right\|^2, \end{aligned} \tag{42}$$

where the second inequality holds since from Lemma 15, we have

$$f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) \leq f_t(\mathbf{x}_{t+1}, \mathbf{y}_t^*(\mathbf{x}_{t+1})) - \frac{\mu_f}{2}\left\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\right\|^2.$$

Combining (41) and (42), we get

$$\begin{aligned}f_t(\mathbf{x},\mathbf{y}_t^*(\mathbf{x})) \geq f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*)) + \frac{\mu_f}{2}\left\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\right\|^2 - \frac{1}{2\alpha}\left\|\mathbf{x}_{t+1}-\mathbf{x}_t\right\|^2 + \frac{1}{\alpha}\langle \mathbf{x}_t-\mathbf{x}_{t+1},\mathbf{x}-\mathbf{x}_{t+1}\rangle \\ + \langle \nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1}), \mathbf{x}-\mathbf{x}_{t+1}\rangle + \frac{\mu_f}{2}\left\|\mathbf{x}-\mathbf{x}_t\right\|^2.\end{aligned}$$

By setting $\mathbf{x}=\mathbf{x}_t^*$, we have

$$\begin{aligned}f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*)) \geq{}& f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*)) + \frac{\mu_f}{2}\left\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\right\|^2 \\ &- \frac{1}{2\alpha}\left\|\mathbf{x}_{t+1}-\mathbf{x}_t\right\|^2 + \frac{1}{\alpha}\langle \mathbf{x}_t-\mathbf{x}_{t+1},\mathbf{x}_t^*-\mathbf{x}_{t+1}\rangle \\ &+ \langle \nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1}), \mathbf{x}_t^*-\mathbf{x}_{t+1}\rangle + \frac{\mu_f}{2}\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2.\end{aligned}$$

Since $\langle u,v\rangle \geq -\frac{c}{2}\|u\|^2 - \frac{1}{2c}\|v\|^2$, $\forall u,v\in\mathbb{R}^n$, $\forall c>0$, we obtain

$$\begin{aligned}0 \geq{}& -\frac{1}{2\alpha}\left\|\mathbf{x}_{t+1}-\mathbf{x}_t\right\|^2 + \frac{1}{\alpha}\langle \mathbf{x}_t-\mathbf{x}_{t+1},\mathbf{x}_t^*-\mathbf{x}_t\rangle \\ &+ \frac{1}{\alpha}\langle \mathbf{x}_t-\mathbf{x}_{t+1},\mathbf{x}_t-\mathbf{x}_{t+1}\rangle - \frac{1}{2c}\|\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})\|^2 \\ &+ \left(\frac{\mu_f}{2}-\frac{c}{2}\right)\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\|^2 + \frac{\mu_f}{2}\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2.\end{aligned}$$

After rearranging, we obtain

$$\begin{aligned}&\langle \mathbf{x}_t-\mathbf{x}_{t+1},\mathbf{x}_t^*-\mathbf{x}_t\rangle \\ &\leq \frac{1}{2}\left\|\mathbf{x}_{t+1}-\mathbf{x}_t\right\|^2 - \frac{\mu_f}{2}\alpha\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2 - \left\|\mathbf{x}_t-\mathbf{x}_{t+1}\right\|^2 + (\frac{c}{2}-\frac{\mu_f}{2})\alpha\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\|^2 \\ &+ \frac{\alpha}{2c}\|\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})\|^2. \end{aligned} \tag{43}$$

Note that

$$\begin{aligned}\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\|^2 &= \|\mathbf{x}_{t+1}-\mathbf{x}_t+\mathbf{x}_t-\mathbf{x}_t^*\|^2 \\ &= \|\mathbf{x}_{t+1}-\mathbf{x}_t\|^2 + \|\mathbf{x}_t-\mathbf{x}_t^*\|^2 + 2\langle \mathbf{x}_{t+1}-\mathbf{x}_t,\mathbf{x}_t-\mathbf{x}_t^*\rangle \\ &\leq (1-\mu_f\alpha)\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2 + (c-\mu_f)\alpha\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\|^2 \\ &+ \frac{\alpha}{c}\|\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})\|^2,\end{aligned} \tag{44}$$

where the inequality follows from (43).

From Lemma 3, we obtain

$$\|\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})\|^2 \leq M_f^2\left\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\right\|^2. \tag{45}$$

Inserting (45) into (44) implies

$$(1-(c-\mu_f)\alpha)\left\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\right\|^2 \leq (1-\mu_f\alpha)\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2 + \frac{M_f^2}{c}\alpha\left\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\right\|^2.$$

By setting $c=\mu_f/2$, we get

$$\left(1+\frac{\mu_f}{2}\alpha\right)\left\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\right\|^2 \leq (1-\mu_f\alpha)\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2 + \frac{2M_f^2}{\mu_f}\alpha\left(\|\mathbf{y}_t-\mathbf{y}_t^*(\mathbf{x}_t)\|^2 + \|\mathbf{v}_t-\mathbf{v}_t^*(\mathbf{x}_t)\|^2\right).$$

Finally, dividing both sides of the above inequality by $\left(1+\frac{\mu_f}{2}\alpha\right)$, we obtain

$$\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\|^2 \leq \gamma\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2 + \frac{2M_f^2\alpha}{\left(1+\frac{\mu_f}{2}\alpha\right)\mu_f}\left(\|\mathbf{y}_t-\mathbf{y}_t^*(\mathbf{x}_t)\|^2 + \|\mathbf{v}_t-\mathbf{v}_t^*(\mathbf{x}_t)\|^2\right),$$

where γ is defined in (39).

Since the strong convexity constant μ_f is smaller than the constant of gradient Lipschitz continuity $\ell_{f,1}$, and the constant α is chosen such that $\alpha \leq 1/\ell_{f,1}$, we have $\alpha \leq 1/\mu_f$, which implies that $\gamma \leq 1$. ■

The following lemma plays a key role in the proof of OAGD in the strongly convex setting. It basically shows that under certain conditions on inner and outer step sizes, $\sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\|^p$ can be bounded in terms of $P_{p,T}$ and $Y_{p,T}$.

Lemma 19. *Suppose Assumption A holds. In Algorithm 1, for all $t \in [T]$, choose*

$$\begin{aligned} \beta_t &= \beta = \frac{2}{\ell_{g,1}+\mu_g}, \\ \alpha_t &= \alpha \leq \min\left\{\frac{1}{\ell_{f,1}}, \frac{\mu_f}{128 M_f^2 L_{\mathbf{y}}^2}\right\}, \quad \text{and} \\ K_t &\geq \left\lceil \frac{(\kappa_g+1)\log \rho_t^{-2}}{4} \right\rceil. \end{aligned}$$

Then, Algorithm 1 guarantees the following.

H1. *If $\rho_t = \rho \leq \frac{\gamma}{1+\gamma}$, then*

$$\begin{aligned} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\| \leq{}& \frac{4}{\gamma}\left(\|\mathbf{x}_1 - \mathbf{x}_1^*\| + P_{1,T}\right) \\ &+ \frac{1}{2L_{\mathbf{y}}}\left(\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| + Y_{1,T}\right). \end{aligned}$$

H2. *If $\rho_t = \rho \leq \frac{\sqrt{\gamma}}{\sqrt{2}\sqrt{\gamma+1}}$, then*

$$\begin{aligned} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\|^2 \leq{}& \frac{64}{23\gamma}\left(\|\mathbf{x}_1 - \mathbf{x}_1^*\|^2 + (1+\frac{2}{\gamma})P_{2,T}\right) \\ &+ \frac{3}{92L_{\mathbf{y}}^2}\left(\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 + 6Y_{2,T}\right). \end{aligned}$$

Here, γ is defined in (39); $P_{p,T}$ and $Y_{p,T}$ are defined in (5a).

Proof. We first show H1.. It follows from the triangle inequality that

$$\begin{aligned} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\| &= \|\mathbf{x}_1 - \mathbf{x}_1^*\| + \sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\| \\ &\leq \|\mathbf{x}_1 - \mathbf{x}_1^*\| + \sum_{t=2}^{T}\left(\|\mathbf{x}_t - \mathbf{x}_{t-1}^*\| + \|\mathbf{x}_{t-1}^* - \mathbf{x}_t^*\|\right) \\ &\leq \|\mathbf{x}_1 - \mathbf{x}_1^*\| + \sum_{t=1}^{T} \|\mathbf{x}_{t+1} - \mathbf{x}_t^*\| + P_{1,T}. \end{aligned} \tag{46}$$

Next, we provide an upper bound for the second term on the right-hand side of (46). Note that our choice of the stepsize α_t in the statement of Lemma 19 satisfies the condition of Lemma 18. Hence, from Lemma 18 and the inequality $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ for $a, b \geq 0$, we get

$$\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\| \leq \sqrt{1-\gamma}\|\mathbf{x}_t - \mathbf{x}_t^*\| + M_f\sqrt{\frac{2\alpha}{\mu_f}}\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|.$$

Summing both sides of the above inequality from $t = 1$ to T, we get

$$\begin{aligned} \sum_{t=1}^{T} \|\mathbf{x}_{t+1} - \mathbf{x}_t^*\| &\leq \sqrt{1-\gamma}\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_t^*\| + M_f\sqrt{\frac{2\alpha}{\mu_f}}\sum_{t=1}^{T}\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| \\ &\leq (1-\frac{\gamma}{2})\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_t^*\| + M_f\sqrt{\frac{2\alpha}{\mu_f}}\sum_{t=1}^{T}\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|. \end{aligned} \tag{47}$$

Here, $\gamma = \frac{3\mu_f}{\frac{2}{\alpha}+\mu_f}$, and the second inequality follows since $\sqrt{1-a} \leq 1 - \frac{a}{2}$ for any $a \leq 1$.

Note that our assumption on ρ in the statement of Lemma 19–H1. satisfies the requirement of Lemma 17–L2.. Hence, we have

$$\sum_{t=1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| \leq \frac{\rho_1}{1-\rho_1}\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| \\ + \frac{1}{1-\rho_1}\left(2L_{\mathbf{y}}\sum_{t=1}^{T}\rho_t\|\mathbf{x}_t - \mathbf{x}_t^*\| + \sum_{t=2}^{T}\rho_t\|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*)\|\right). \tag{48}$$

Substituting (48) into (47), we get

$$\sum_{t=1}^{T}\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\| \leq \sum_{t=1}^{T}\left(1 - \frac{\gamma}{2} + \frac{M_f L_{\mathbf{y}} 2\sqrt{2}}{\sqrt{\mu_f}}\sqrt{\alpha}\frac{\rho_t}{1-\rho_1}\right)\|\mathbf{x}_t - \mathbf{x}_t^*\| \\ + M_f\sqrt{\frac{2\alpha}{\mu_f}}\left(\frac{\rho_1}{1-\rho_1}\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| + \frac{1}{1-\rho_1}\sum_{t=2}^{T}\rho_t\|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*) - \mathbf{y}_t^*(\mathbf{x}_t^*)\|\right). \tag{49}$$

By setting $\rho_t = \rho \leq \frac{\gamma}{1+\gamma}$, we have

$$\frac{M_f L_{\mathbf{y}} 2\sqrt{2}}{\sqrt{\mu_f}}\sqrt{\alpha}\frac{\rho}{1-\rho} \leq \frac{M_f L_{\mathbf{y}} 2\sqrt{2}}{\sqrt{\mu_f}}\sqrt{\alpha}\gamma \leq \frac{\gamma}{4}, \tag{50}$$

where the second inequality holds due to our assumption on the outer stepsize, i.e., $\alpha \leq \frac{\mu_f}{128 M_f^2 L_{\mathbf{y}}^2}$.

Combining the above two inequalities (50) and (49), we conclude that

$$\sum_{t=1}^{T}\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\| \leq \left(1 - \frac{\gamma}{4}\right)\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_t^*\| + M_f\sqrt{\frac{2\alpha}{\mu_f}}\left(\frac{\rho}{1-\rho}\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| + \frac{\rho}{1-\rho}Y_{1,T}\right) \\ \leq \left(1 - \frac{\gamma}{4}\right)\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_t^*\| + \frac{\gamma}{8L_{\mathbf{y}}}\left(\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| + Y_{1,T}\right),$$

where the second inequality is by $\alpha \leq \frac{\mu_f}{128 M_f^2 L_{\mathbf{y}}^2}$ and $\rho \leq \frac{\gamma}{1+\gamma}$.
Plugging this into (46) yields

$$\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_t^*\| \leq \|\mathbf{x}_1 - \mathbf{x}_1^*\| + \left(1 - \frac{\gamma}{4}\right)\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_t^*\| \\ + \frac{\gamma}{8L_{\mathbf{y}}}\left(\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| + Y_{1,T}\right) + P_{1,T}.$$

Rearranging terms in the above inequality finishes the proof.

We now show part H2. of the lemma.

First, note that

$$\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{x}_t^*\|^2 = \|\mathbf{x}_1 - \mathbf{x}_1^*\|^2 + \sum_{t=2}^{T}\|\mathbf{x}_t - \mathbf{x}_t^*\|^2 \\ \leq \|\mathbf{x}_1 - \mathbf{x}_1^*\|^2 + (1 + \frac{\gamma}{2})\sum_{t=2}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}^*\|^2 + (1 + \frac{2}{\gamma})\sum_{t=2}^{T}\|\mathbf{x}_{t-1}^* - \mathbf{x}_t^*\|^2 \\ \leq \|\mathbf{x}_1 - \mathbf{x}_1^*\|^2 + (1 + \frac{\gamma}{2})\sum_{t=1}^{T}\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\|^2 + (1 + \frac{2}{\gamma})P_{2,T}, \tag{51}$$

where the first inequality follows from Lemma 14.

Similar to the previous case, we provide an upper bound for the second term on the right-hand side of (51).

Since our assumption on α in the statement of Lemma 19 satisfies the requirement of Lemma 18, we have

$$\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\|^2 \le (1-\gamma)\|\mathbf{x}_t-\mathbf{x}_t^*\|^2 + \frac{2M_f^2\alpha}{(1+\frac{\mu_f}{2}\alpha)\mu_f}\left\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\right\|^2,$$

which implies

$$\begin{aligned}(1+\frac{\gamma}{2})\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\|^2 &\le (1+\frac{\gamma}{2})(1-\gamma)\|\mathbf{x}_t-\mathbf{x}_t^*\|^2 + (1+\frac{\gamma}{2})\frac{2M_f^2\alpha}{(1+\frac{\mu_f}{2}\alpha)\mu_f}\left\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\right\|^2\\ &\le (1-\frac{\gamma}{2})\|\mathbf{x}_t-\mathbf{x}_t^*\|^2 + (1+\frac{\gamma}{2})\frac{2M_f^2\alpha}{\mu_f}\left\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\right\|^2,\end{aligned}$$

where the second inequality is due to $(1+a/2)(1-a)\le(1-a/2-a^2/2)\le 1-a/2$.

Summing both sides of the above inequality from $t=1$ to T, we get

$$\begin{aligned}(1+\frac{\gamma}{2})\sum_{t=1}^T\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\|^2 &\le (1-\frac{\gamma}{2})\sum_{t=1}^T\|\mathbf{x}_t-\mathbf{x}_t^*\|^2 + (1+\frac{\gamma}{2})\frac{2M_f^2\alpha}{\mu_f}\sum_{t=1}^T\left\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\right\|^2\\ &\le (1-\frac{\gamma}{2})\sum_{t=1}^T\|\mathbf{x}_t-\mathbf{x}_t^*\|^2 + \frac{3M_f^2\alpha}{\mu_f}\sum_{t=1}^T\left\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\right\|^2, \qquad (52)\end{aligned}$$

where the second inequality follows since $\gamma=\frac{3\mu_f}{\frac{2}{\alpha}+\mu_f}\le 1$; see (39).

Since our assumption on ρ in the statement of Lemma 19–H2. satisfies the requirement of Lemma 17–L1. , we have

$$\begin{aligned}\sum_{t=1}^T\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2 \le& \frac{\rho_1^2}{1-2\rho_1^2}\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2\\ &+\frac{6}{1-2\rho_1^2}\left(2L_{\mathbf{y}}^2\sum_{t=1}^T\rho_t^2\|\mathbf{x}_t-\mathbf{x}_t^*\|^2+\sum_{t=2}^T\rho_t^2\|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*)-\mathbf{y}_t^*(\mathbf{x}_t^*)\|^2\right).\end{aligned}$$

This together with (52) gives

$$\begin{aligned}(1+\frac{\gamma}{2})\sum_{t=1}^T\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\|^2 \le& \left(1-\frac{\gamma}{2}+\frac{36M_f^2L_{\mathbf{y}}^2\alpha}{\mu_f}\frac{\rho_t^2}{(1-2\rho_1^2)}\right)\sum_{t=1}^T\|\mathbf{x}_t-\mathbf{x}_t^*\|^2\\ &+\frac{3M_f^2\alpha}{\mu_f}\left(\frac{\rho_1^2}{1-2\rho_1^2}\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2+\frac{6}{1-2\rho_1^2}\sum_{t=2}^T\rho_t^2\|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*)-\mathbf{y}_t^*(\mathbf{x}_t^*)\|^2\right). \qquad (53)\end{aligned}$$

Since by our setting $\rho_t=\rho\le\frac{\sqrt{\gamma}}{\sqrt{2}\sqrt{\gamma+1}}$, we have

$$\frac{36M_f^2L_{\mathbf{y}}^2\alpha}{\mu_f}\frac{\rho^2}{(1-2\rho^2)}\le\frac{36M_f^2L_{\mathbf{y}}^2\alpha}{\mu_f}\frac{\gamma}{2}\le\frac{9\gamma}{64},$$

where the second inequality uses our assumption on the stepsize, i.e., $\alpha\le\frac{\mu_f}{128M_f^2L_{\mathbf{y}}^2}$.

Combining the above inequality with (53), we obtain

$$\begin{aligned}(1+\frac{\gamma}{2})\sum_{t=1}^T\|\mathbf{x}_{t+1}-\mathbf{x}_t^*\|^2 \le& \left(1-\frac{23\gamma}{64}\right)\sum_{t=1}^T\|\mathbf{x}_t-\mathbf{x}_t^*\|^2\\ &+\frac{3M_f^2\alpha}{\mu_f}\left(\frac{\rho^2}{1-2\rho^2}\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2+\frac{6\rho^2}{1-2\rho^2}Y_{2,T}\right)\\ \le& \left(1-\frac{23\gamma}{64}\right)\sum_{t=1}^T\|\mathbf{x}_t-\mathbf{x}_t^*\|^2+\frac{3\gamma}{256L_{\mathbf{y}}^2}\left(\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2+6Y_{2,T}\right), \qquad (54)\end{aligned}$$

where the second inequality follows from $\alpha \leq \frac{\mu_f}{128 M_f^2 L_{\mathbf{y}}^2}$ and $\rho \leq \frac{\sqrt{\gamma}}{\sqrt{2}\sqrt{\gamma+1}}$.
Further, plugging (54) into (51) yields:

$$\sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\|^2 \leq \|\mathbf{x}_1 - \mathbf{x}_1^*\|^2 + \left(1 - \frac{23\gamma}{64}\right) \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\|^2 + (1 + \frac{2}{\gamma})P_{2,T} + \frac{3\gamma}{256 L_{\mathbf{y}}^2} \left(\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 + 6Y_{2,T}\right).$$

Rearranging the inequality gives H2.. ■

C.2.2 Proof of Theorem 4

Proof. Assumption A1. implies that $\|\nabla f_t(\mathbf{x}, \mathbf{y}(\mathbf{x}))\| \leq \ell_{f,0}$ for any $t \in [T]$ and any $\mathbf{x} \in \mathcal{X}$. Thus, we get

$$\sum_{t=1}^{T} \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))\right) \leq \ell_{f,0} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\|.$$

Note that our choices of the stepsize α_t and K_t in the theorem statement can be rewritten as

$$\alpha_t = \alpha \leq \min\left\{\frac{1}{\ell_{f,1}}, \frac{\mu_f}{128 M_f^2 L_{\mathbf{y}}^2}\right\}, \quad \text{and}$$
$$K_t \geq \left\lceil \frac{(\kappa_g + 1)\log \rho_t^{-2}}{4} \right\rceil, \quad \text{with} \quad \rho_t = \rho \leq \frac{\gamma}{3(1+\gamma)}. \tag{55}$$

These choices satisfy the condition of Lemma 19–H1.. Hence, from Lemma 19–H1., we get

$$\sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\| \leq \frac{4}{\gamma}\left(\|\mathbf{x}_1 - \mathbf{x}_1^*\| + P_{1,T}\right) + \frac{1}{2L_{\mathbf{y}}}\left(\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| + Y_{1,T}\right),$$

which, in conjunction with Assumption B, implies

$$\begin{aligned}\sum_{t=1}^{T} \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))\right) &\leq \frac{4\ell_{f,0}}{\gamma}\left(\|\mathbf{x}_1 - \mathbf{x}_1^*\| + P_{1,T}\right) + \frac{\ell_{f,0}}{2L_{\mathbf{y}}}\left(\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| + Y_{1,T}\right)\\
&\leq \frac{4\ell_{f,0}}{\gamma}\left(D + P_{1,T}\right) + \frac{\ell_{f,0}}{2L_{\mathbf{y}}}\left(D' + Y_{1,T}\right)\\
&= \mathcal{O}\left(1 + S_{1,T}\right).\end{aligned} \tag{56}$$

In the following, we show that the dynamic regret can also be upper bounded by $S_{2,T} = P_{2,T} + Y_{2,T}$.
It follows from Lemma 3 that

$$\begin{aligned}f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) &\leq \langle \nabla f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)), \mathbf{x}_t - \mathbf{x}_t^* \rangle + \frac{L_f}{2}\|\mathbf{x}_t - \mathbf{x}_t^*\|^2\\
&\leq \frac{1}{2}\|\nabla f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))\|^2 + \frac{1}{2}(1 + L_f)\|\mathbf{x}_t - \mathbf{x}_t^*\|^2.\end{aligned} \tag{57}$$

Summing the inequality (57) over $t \in [T]$, we get

$$\begin{aligned}&\sum_{t=1}^{T} \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))\right)\\
&\leq \frac{1}{2}\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*))\|^2 + \frac{1}{2}(1 + L_f)\sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^*\|^2.\end{aligned} \tag{58}$$

The choices of α_t and K_t in (55) satisfy the condition of Lemma 19–H2. as well. Hence, from Lemma 19–H2. and Assumption B, we get

$$\begin{aligned}\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_t^*\|^2 &\leq \frac{64}{23\gamma}\left(\|\mathbf{x}_1-\mathbf{x}_1^*\|^2+(1+\frac{2}{\gamma})P_{2,T}\right)\\ &+\frac{3}{92L_{\mathbf{y}}^2}\left(\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2+6Y_{2,T}\right)\\ &\leq \frac{64}{23\gamma}\left(D^2+(1+\frac{2}{\gamma})P_{2,T}\right)+\frac{3}{92L_{\mathbf{y}}^2}\left({D'}^2+6Y_{2,T}\right).\end{aligned}\tag{59}$$

Putting together (59) and (58), we get

$$\begin{aligned}&\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\right)\\ &\leq \frac{1}{2}\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\|^2+\frac{32}{23\gamma}(1+L_f)\left(D^2+(1+\frac{2}{\gamma})P_{2,T}\right)\\ &+\frac{3}{184L_{\mathbf{y}}^2}(1+L_f)\left({D'}^2+6Y_{2,T}\right)\\ &=\mathcal{O}\left(1+\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\|^2+P_{2,T}+Y_{2,T}\right).\end{aligned}\tag{60}$$

Now, from (56) and (60), we have

$$\begin{aligned}&\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\right)\\ &\leq \mathcal{O}\left(1+\min\left\{S_{1,T},\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\|^2+S_{2,T}\right\}\right).\end{aligned}$$

This completes the proof. ■

C.2.3 Proof of Theorem 6

Proof. Recall the update rule of Algorithm 1 (with $w=1$): $\mathbf{x}_{t+1}=\Pi_{\mathcal{X}}\left[\mathbf{x}_t-\alpha_t\tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})\right]$. From the Pythagorean theorem, we get

$$\begin{aligned}\|\mathbf{x}_{t+1}-\mathbf{x}^*\|^2 &\leq \left\|\mathbf{x}_t-\alpha_t\tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})-\mathbf{x}^*\right\|^2\\ &=\|\mathbf{x}_t-\mathbf{x}^*\|^2-2\alpha_t\langle\tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1}),\mathbf{x}_t-\mathbf{x}^*\rangle+\alpha_t^2\left\|\tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})\right\|^2\\ &\leq \|\mathbf{x}_t-\mathbf{x}^*\|^2-2\alpha_t\langle\tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1}),\mathbf{x}_t-\mathbf{x}^*\rangle\\ &+2\alpha_t^2\|\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+2\alpha_t^2\left\|\tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})-\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\right\|^2,\end{aligned}$$

where the second inequality uses Lemma 14 with $c=1$.
Rearranging the above inequality yields

$$\begin{aligned}\langle\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)),\mathbf{x}_t-\mathbf{x}^*\rangle &\leq \frac{1}{2\alpha_t}\|\mathbf{x}_t-\mathbf{x}^*\|^2-\frac{1}{2\alpha_t}\|\mathbf{x}_{t+1}-\mathbf{x}^*\|^2+\alpha_t\|\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2\\ &+\alpha_t\left\|\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-\tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1})\right\|^2\\ &+\left\langle\nabla f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-\tilde{\nabla} f_t(\mathbf{x}_t,\mathbf{y}_{t+1}),\mathbf{x}_t-\mathbf{x}^*\right\rangle.\end{aligned}\tag{61}$$

From Lemma 3, for any $c > 0$, we have

$$\begin{aligned}\left\langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}), \mathbf{x}_t - \mathbf{x}^* \right\rangle &\leq \frac{c}{2} \|\mathbf{x}_t - \mathbf{x}^*\|^2 + \frac{1}{2c} \left\| \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}) \right\|^2 \\ &\leq \frac{c}{2} \|\mathbf{x}_t - \mathbf{x}^*\|^2 + \frac{M_f^2}{2c} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2, \qquad (62) \\ \left\| \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}) \right\|^2 &\leq M_f^2 \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2.\end{aligned}$$

Combining (61) and (62), we get

$$\begin{aligned}\langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_t - \mathbf{x}^* \rangle \leq{}& \frac{1}{2}\left(\frac{1}{\alpha_t} + c\right) \|\mathbf{x}_t - \mathbf{x}^*\|^2 - \frac{1}{2\alpha_t} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 \\ &+ \alpha_t \|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2 + M_f^2 \left(\alpha_t + \frac{1}{2c}\right) \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2.\end{aligned} \tag{63}$$

Applying the definition of μ_f-strong convexity to the pair of points $\{\mathbf{x}_t,\mathbf{x}^*\}$, we have

$$2\left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}^*, \mathbf{y}_t^*(\mathbf{x}^*))\right) \leq 2\langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_t - \mathbf{x}^* \rangle - \mu_f \|\mathbf{x}_t - \mathbf{x}^*\|^2. \tag{64}$$

From (63) and (64), we get

$$\begin{aligned}2\left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}^*, \mathbf{y}_t^*(\mathbf{x}^*))\right) &\leq 2\langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_t - \mathbf{x}^* \rangle - \mu_f \|\mathbf{x}_t - \mathbf{x}^*\|^2 \\ &\leq \left(\frac{1}{\alpha_t} + c - \mu_f\right) \|\mathbf{x}_t - \mathbf{x}^*\|^2 - \frac{1}{\alpha_t} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 \\ &+ 2\alpha_t \|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2 + 2M_f^2 \left(\alpha_t + \frac{1}{c}\right) \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2.\end{aligned}$$

Summing from $t = 1$ to T, we have

$$\begin{aligned}&2\sum_{t=1}^{T} \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}^*, \mathbf{y}_t^*(\mathbf{x}^*))\right) \\ &\leq \left(\frac{1}{\alpha_1} + c - \mu_f\right) \|\mathbf{x}_1 - \mathbf{x}^*\|^2 + \sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{x}^*\|^2 \left(\frac{1}{\alpha_t} - \frac{1}{\alpha_{t-1}} + c - \mu_f\right) \\ &+ 2\sum_{t=1}^{T} \alpha_t \|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2 + 2M_f^2 \sum_{t=1}^{T} \left(\alpha_t + \frac{1}{c}\right) \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2.\end{aligned} \tag{65}$$

Next, we bound the last term in the right-hand side of (65). To proceed, note that our choice of K_t as

$$K_t = \left\lceil \frac{(\kappa_g + 1) \log \rho_t^{-2}}{4} \right\rceil, \quad \text{with} \quad \rho_t = \rho \leq \frac{1}{\sqrt{2}} \sqrt{\frac{\theta}{1+\theta}}, \quad \text{and} \quad \theta := \frac{c}{12 L_{\mathbf{y}}^2 M_f^2 (\alpha_1 + \frac{1}{c})}$$

satisfies the condition required in Lemma 17–L1.. Hence, from Lemma 17–L1., we obtain

$$\begin{aligned}&2M_f^2 \sum_{t=1}^{T} \left(\alpha_t + \frac{1}{c}\right) \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \\ &\leq 2M_f^2 \left(\alpha_1 + \frac{1}{c}\right) \frac{6\rho^2}{1 - 2\rho^2} \left(\frac{1}{6} \|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 + 2L_{\mathbf{y}}^2 \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}^*\|^2 + \bar{Y}_{2,T}\right).\end{aligned} \tag{66}$$

Since $\rho \leq \frac{1}{\sqrt{2}} \sqrt{\frac{\theta}{1+\theta}}$, we have

$$4L_{\mathbf{y}}^2 M_f^2 \left(\alpha_1 + \frac{1}{c}\right) \frac{6\rho^2}{1 - 2\rho^2} \leq c,$$

which, in conjunction with Eq. (66), yields

$$\begin{aligned}&2M_f^2\sum_{t=1}^{T}\left(\alpha_t+\frac{1}{c}\right)\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2\\ &\leq \frac{c}{12L_{\mathbf{y}}^2}\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2+c\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}^*\|^2+\frac{c}{2L_{\mathbf{y}}^2}\bar{Y}_{2,T}. \end{aligned} \tag{67}$$

Thus, combining (67) and (65) and using Assumption A1., we obtain

$$\begin{aligned}2\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-f_t(\mathbf{x}^*,\mathbf{y}_t^*(\mathbf{x}^*))\right)\leq&\left(\frac{1}{\alpha_1}+c-\mu_f\right)\|\mathbf{x}_1-\mathbf{x}^*\|^2\\ &+\sum_{t=2}^{T}\|\mathbf{x}_t-\mathbf{x}^*\|^2\left(\frac{1}{\alpha_t}-\frac{1}{\alpha_{t-1}}+2c-\mu_f\right)\\ &+2\ell_{f,0}^2\sum_{t=1}^{T}\alpha_t+\frac{c}{12L_{\mathbf{y}}^2}\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2+\frac{c}{2L_{\mathbf{y}}^2}\bar{Y}_{2,T}.\end{aligned} \tag{68}$$

By setting $c=\mu_f/4$ and $\alpha_t=2/(\mu_f t)$ and utilizing Assumption B, we have

$$2\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-f_t(\mathbf{x}^*,\mathbf{y}_t^*(\mathbf{x}^*))\right)\leq\frac{4\ell_{f,0}^2}{\mu_f}\sum_{t=1}^{T}\frac{1}{t}+\frac{\mu_f}{48L_{\mathbf{y}}^2}D'^2+\frac{\mu_f}{8L_{\mathbf{y}}^2}\bar{Y}_{2,T}. \tag{69}$$

Let

$$e_1:=\frac{\mu_f}{48L_{\mathbf{y}}^2}D'^2,\quad e_2:=\frac{\mu_f}{8L_{\mathbf{y}}^2},\qquad e_3:=\frac{4\ell_{f,0}^2}{\mu_f}. \tag{70}$$

Combining Lemma 13–I. and (70) with (69), we obtain

$$\text{BS-Reg}_T\leq e_3\log T+e_2\bar{Y}_{2,T}+e_1.$$

■

Corollary 20. *Under the same setting as Theorem 4,*

(I) *If function f_t is non-negative for each $t\in[T]$, then*

$$\text{BD-Reg}_T\leq\mathcal{O}\left(1+\min\{S_{1,T},F_T+S_{2,T}\}\right), \tag{71}$$

where $F_T:=\sum_{t=1}^{T}f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))$.

(II) *If $\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\|=\mathcal{O}(S_{2,T})$, then*

$$\text{BD-Reg}_T\leq\mathcal{O}\left(1+\min\{S_{1,T},S_{2,T}\}\right). \tag{72}$$

Proof. (i) If $f_t\geq 0$ for all $t\in[T]$, then it follows from Lemma 10 that $\|\nabla f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\|\leq\sqrt{4\ell_{f,1}f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))}$. This together with (10) gives the desired result.

(ii) If $\mathbf{x}_t^*\in\arg\min_{\mathbf{x}\in\mathcal{X}}f_t(\mathbf{x})$ for all $t\in[T]$ and the minimizers $\{\mathbf{x}_t^*\}_{t=1}^T$ lie in the interior of the domain $\mathcal{X}$, we have $\|\nabla f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\|=0$ which together with (10) gives the desired result. If $\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\|=\mathcal{O}(S_{2,T})$, then (72) follows from (10).

■

Corollary 20 naturally interpolates between single-level and bilevel regret. In the case when $Y_{1,T}=Y_{2,T}=0$, Eq. (71) gives a single-level regret for strongly convex, smooth, and non-negative losses, similar to [Srebro et al., 2010, Zhao et al., 2020]. We note that if the minimizers $\{\mathbf{x}_t^*\}_{t=1}^T$ lie in the interior of the domain $\mathcal{X}$, we have $\|\nabla f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\|=0$ for all $t\in[T]$, which implies the $\mathcal{O}\left(1+\min\{S_{1,T},S_{2,T}\}\right)$ regret bound.

C.3 Proof for Convex OBO with Partial Information

C.3.1 Proof of Theorem 7

Proof. From the update rule of Algorithm 1 (with $w = 1$), we have $\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}\left[\mathbf{x}_t - \alpha\tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1})\right]$. Now, from the Pythagorean theorem, we get

$$\begin{aligned}
\frac{1}{2}\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\|^2 &\leq \frac{1}{2}\left\|\mathbf{x}_t - \alpha\tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}) - \mathbf{x}_t^*\right\|^2 \\
&= \frac{1}{2}\|\mathbf{x}_t - \mathbf{x}_t^*\|^2 - \alpha\langle\tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}), \mathbf{x}_t - \mathbf{x}_t^*\rangle + \frac{\alpha^2}{2}\left\|\tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1})\right\|^2 \\
&\leq \frac{1}{2}\|\mathbf{x}_t - \mathbf{x}_t^*\|^2 - \alpha\langle\tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}), \mathbf{x}_t - \mathbf{x}_t^*\rangle + \alpha^2\|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2 \\
&+ \alpha^2\left\|\tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}) - \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\right\|^2. \qquad (73)
\end{aligned}$$

Here, the second inequality holds because of the Lemma 14 by setting $c = 1$.
Rearranging the above inequality and summing over $t \in [T]$, we obtain

$$\begin{aligned}
\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_t - \mathbf{x}_t^*\rangle &\leq \sum_{t=1}^{T}\left(\frac{1}{2\alpha}\|\mathbf{x}_t - \mathbf{x}_t^*\|^2 - \frac{1}{2\alpha}\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\|^2\right) && \text{(74a)}\\
&+ \alpha\sum_{t=1}^{T}\left\|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1})\right\|^2 && \text{(74b)}\\
&+ \sum_{t=1}^{T}\left\langle\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}), \mathbf{x}_t - \mathbf{x}_t^*\right\rangle && \text{(74c)}\\
&+ \alpha\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2. && \text{(74d)}
\end{aligned}$$

Next, we upper bound each term of (74).
• **Bounding (74a):** Observe that

$$\begin{aligned}
\text{(74a)} &\leq \frac{1}{2\alpha}\|\mathbf{x}_1 - \mathbf{x}_1^*\|^2 - \frac{1}{2\alpha}\|\mathbf{x}_{T+1} - \mathbf{x}_T^*\|^2 + \frac{1}{2\alpha}\sum_{t=2}^{T}\left(\|\mathbf{x}_t - \mathbf{x}_t^*\|^2 - \|\mathbf{x}_t - \mathbf{x}_{t-1}^*\|^2\right)\\
&\leq \frac{1}{2\alpha}\|\mathbf{x}_1 - \mathbf{x}_1^*\|^2 + \frac{1}{2\alpha}\sum_{t=2}^{T}\|\mathbf{x}_t^* - \mathbf{x}_t + \mathbf{x}_{t-1}^* - \mathbf{x}_t\|\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\|\\
&\leq \frac{D^2}{2\alpha} + \frac{D}{2\alpha}\sum_{t=2}^{T}\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\|, \qquad \text{(75a)}
\end{aligned}$$

where the second inequality follows since

$$\begin{aligned}
\|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_{t-1}^* - \mathbf{x}_t\|^2 &= \left\langle\mathbf{x}_t^* - \mathbf{x}_t + \mathbf{x}_{t-1}^* - \mathbf{x}_t, \mathbf{x}_t^* - \mathbf{x}_t - (\mathbf{x}_{t-1}^* - \mathbf{x}_t)\right\rangle\\
&\leq \|\mathbf{x}_t^* - \mathbf{x}_t + \mathbf{x}_{t-1}^* - \mathbf{x}_t\|\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\|,
\end{aligned}$$

and the last inequality follows from Assumption B.
• **Bounding (74b) and (74c):** It follows from Lemma 3 and Assumption B that

$$\begin{aligned}
\sum_{t=1}^{T}\left\langle\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}), \mathbf{x}_t - \mathbf{x}_t^*\right\rangle &\leq \sum_{t=1}^{T}\|\mathbf{x}_t^* - \mathbf{x}_t\|\|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1})\|\\
&\leq DM_f\sum_{t=1}^{T}\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|,\\
\sum_{t=1}^{T}\left\|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1})\right\|^2 &\leq M_f^2\sum_{t=1}^{T}\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2.
\end{aligned}$$

Hence,

$$(74b) + (74c) \leq DM_f \sum_{t=1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| + \alpha M_f^2 \sum_{t=1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 . \tag{75b}$$

- **Bounding (74d):** By the smoothness of $\phi_t(\mathbf{x}) = f_t(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))$, for any $\mathbf{x} \in \mathbb{R}^{d_1}$, we have

$$\phi_t(\mathbf{x}) - \phi_t(\mathbf{x}_t) \leq \langle \nabla \phi_t(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle + \frac{L_f}{2} \|\mathbf{x} - \mathbf{x}_t\|^2.$$

Let $\mathbf{x} = \mathbf{x}_t' = \mathbf{x}_t - \frac{1}{L_f} \nabla \phi_t(\mathbf{x}_t)$ in the above inequality, we have $\phi_t(\mathbf{x}_t') - \phi_t(\mathbf{x}_t) \leq -\frac{\|\nabla \phi_t(\mathbf{x}_t)\|^2}{2L_f}$.

It follows from the convexity of $f_t(\mathbf{x}, \cdot)$ that

$$\phi_t(\mathbf{x}_t') \geq \phi_t(\mathbf{x}_t^*) + \langle \nabla \phi_t(\mathbf{x}_t^*), \mathbf{x}_t' - \mathbf{x}_t^* \rangle = \phi_t(\mathbf{x}_t^*),$$

where the equality follows from the vanishing gradient condition ($\exists(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) \in \mathcal{X} \times \mathbb{R}^{d_2}$ such that $\nabla f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) = 0$ for all $t \in [T]$).

Hence,

$$\phi_t(\mathbf{x}_t^*) - \phi_t(\mathbf{x}_t) \leq \phi_t(\mathbf{x}_t') - \phi_t(\mathbf{x}_t) \leq -\frac{\|\nabla \phi_t(\mathbf{x}_t)\|^2}{2L_f},$$

which implies that

$$(74d) \leq 2\alpha L_f \sum_{t=1}^{T} \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) \right). \tag{75c}$$

- **Bounding $\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_t - \mathbf{x}_t^* \rangle$:** Substituting (75a)–(75c) into (74), we get

$$\begin{aligned}\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_t - \mathbf{x}_t^* \rangle \leq \sum_{t=1}^{T} \Big(&\alpha M_f^2 \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 + DM_f \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| \\ &+ 2\alpha L_f \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) \right) \Big) \\ &+ \frac{D^2}{2\alpha} + \frac{D}{2\alpha} \sum_{t=2}^{T} \|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\|.\end{aligned} \tag{76}$$

- **Completing the proof of Theorem 7:** By the convexity of f_t and (76), we obtain

$$\begin{aligned}&\sum_{t=1}^{T} \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) \right) \\ &\leq \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_t - \mathbf{x}_t^* \rangle \\ &\leq \sum_{t=1}^{T} \Big(\alpha M_f^2 \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 + DM_f \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| \\ &+ 2\alpha L_f \big(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}_t^*, \mathbf{y}_t^*(\mathbf{x}_t^*)) \big) \Big) + \frac{D^2}{2\alpha} + \frac{D}{2\alpha} P_{1,T}.\end{aligned} \tag{77}$$

Note that our choice of K_t in the theorem statement as

$$K_t = \left\lceil \frac{(\kappa_g + 1) \log \rho_t^{-2}}{4} \right\rceil, \quad \text{with} \quad \rho_t = \frac{1}{2t^2} \tag{78}$$

satisfies the condition of Lemma 17–L1.. Moreover, using Lemma 13–III., we have

$$\sum_{t=1}^{T} \rho_t = \frac{\pi^2}{12} \quad \text{and} \quad \sum_{t=1}^{T} \rho_t^2 = \frac{\pi^4}{360}.$$

This, together with Lemma 17–L1. and Assumption B, gives

$$\begin{aligned}\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2 &\leq \frac{\rho_1^2}{1-2\rho_1^2}\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2\\ &+\frac{6}{1-2\rho_1^2}\left(2L_{\mathbf{y}}^2\sum_{t=1}^{T}\rho_t^2\|\mathbf{x}_t-\mathbf{x}_t^*\|^2+\sum_{t=2}^{T}\rho_t^2\|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*)-\mathbf{y}_t^*(\mathbf{x}_t^*)\|^2\right)\\ &\leq \frac{1}{2}D'^2+\frac{\pi^4}{30}L_{\mathbf{y}}^2D^2+\frac{\pi^4}{60}L_{\mathbf{y}}^2Y_{2,T}.\end{aligned} \tag{79a}$$

Similarly, we obtain

$$\begin{aligned}\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\| &\leq \frac{\rho_1}{1-\rho_1}\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|\\ &+\frac{1}{1-\rho_1}\left(2L_{\mathbf{y}}\sum_{t=1}^{T}\rho_t\|\mathbf{x}_t-\mathbf{x}_t^*\|+\sum_{t=2}^{T}\rho_t\|\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1}^*)-\mathbf{y}_t^*(\mathbf{x}_t^*)\|\right)\\ &\leq D'+\pi^2L_{\mathbf{y}}D+\frac{\pi^2}{2}L_{\mathbf{y}}Y_{1,T}.\end{aligned} \tag{79b}$$

Now, let

$$\dot{E}_1(\alpha):=\alpha M_f^2\left(\frac{1}{2}D'^2+\frac{\pi^4}{30}L_{\mathbf{y}}^2D^2\right)+DM_f\left(D'+\pi^2L_{\mathbf{y}}D\right).$$

Substituting (79a) and (79b) into (77), we have

$$\begin{aligned}&(1-2\alpha L_f)\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-f_t(\mathbf{x}_t^*,\mathbf{y}_t^*(\mathbf{x}_t^*))\right)\\ &\leq \dot{E}_1(\alpha)+\frac{D^2}{2\alpha}+\frac{D}{2\alpha}P_{1,T}+\alpha M_f^2\frac{\pi^4}{60}L_{\mathbf{y}}^2Y_{2,T}+DM_f\frac{\pi^2}{2}L_{\mathbf{y}}Y_{1,T}.\end{aligned} \tag{80}$$

Let $\alpha\leq 1/(4L_f)$ and

$$\begin{aligned}\dot{c}_1(\alpha)&:=\frac{D}{2\alpha(1-2\alpha L_f)},\\ \dot{c}_2(\alpha)&:=\frac{1}{1-2\alpha L_f}DM_f\frac{\pi^2}{2}L_{\mathbf{y}},\\ \dot{c}_3(\alpha)&:=\frac{1}{1-2\alpha L_f}\alpha M_f^2\frac{\pi^4}{60}L_{\mathbf{y}}^2,\\ \dot{c}_4(\alpha)&:=\frac{\dot{E}_1(\alpha)}{1-2\alpha L_f}+\frac{D^2}{2\alpha(1-2\alpha L_f)}.\end{aligned}$$

The above definitions together with (80) implies

$$\text{BD-Reg}_T\leq \dot{c}_1(\alpha)P_{1,T}+\dot{c}_2(\alpha)Y_{1,T}+\dot{c}_3(\alpha)Y_{2,T}+\dot{c}_4(\alpha).$$

This gives the desired result in (12). ■

C.3.2 Proof of Theorem 8

Proof. The proof is similar to Theorem 7. From the update rule of Algorithm 1 (with $w=1$), we have $\mathbf{x}_{t+1}=\Pi_{\mathcal{X}}\left[\mathbf{x}_t-\alpha_t\tilde{\nabla}f_t(\mathbf{x}_t,\mathbf{y}_{t+1})\right]$. Now, applying the Pythagorean theorem and employing an argument identical to (73) and (74), we obtain

$$\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)), \mathbf{x}_t - \mathbf{x}^* \rangle \leq \sum_{t=1}^{T} \left(\frac{1}{2\alpha_t} \left\| \mathbf{x}_t - \mathbf{x}^* \right\|^2 - \frac{1}{2\alpha_t} \left\| \mathbf{x}_{t+1} - \mathbf{x}^* \right\|^2 \right) \tag{81a}$$

$$+ \sum_{t=1}^{T} \alpha_t \left\| \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}) \right\|^2 \tag{81b}$$

$$+ \sum_{t=1}^{T} \left\langle \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - \tilde{\nabla} f_t(\mathbf{x}_t, \mathbf{y}_{t+1}), \mathbf{x}_t - \mathbf{x}^* \right\rangle \tag{81c}$$

$$+ \sum_{t=1}^{T} \alpha_t \left\| \nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) \right\|^2 . \tag{81d}$$

Next, we upper bound each term of (81).

From Assumption B, we have

$$\begin{aligned}
(81a) &= \sum_{t=1}^{T} \left(\frac{1}{2\alpha_t} \|\mathbf{x}_t - \mathbf{x}^*\|^2 - \frac{1}{2\alpha_{t+1}} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 \right) \\
&+ \sum_{t=1}^{T} \left(\frac{1}{2\alpha_{t+1}} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 - \frac{1}{2\alpha_t} \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 \right) \\
&\leq \frac{D^2}{2\alpha_1} - \frac{\|\mathbf{x}_{T+1} - \mathbf{x}^*\|^2}{2\alpha_{T+1}} + D^2 \sum_{t=1}^{T} \left(\frac{1}{2\alpha_{t+1}} - \frac{1}{2\alpha_t} \right) \\
&\leq \frac{D^2}{2\alpha_{T+1}}.
\end{aligned} \tag{82a}$$

Using Lemma 3 and Assumption B, and following similar steps as in the derivation of (75b), we obtain

$$\begin{aligned}
(81b) + (81c) &\leq D M_f \sum_{t=1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| + M_f^2 \sum_{t=1}^{T} \alpha_t \left\| \mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t) \right\|^2 \\
&\leq D M_f \sum_{t=1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| + M_f^2 \alpha_1 \sum_{t=1}^{T} \left\| \mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t) \right\|^2 .
\end{aligned} \tag{82b}$$

Further, it follows from Assumption A1. that

$$(81d) = \sum_{t=1}^{T} \alpha_t \|\nabla f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2 \leq \ell_{f,0}^2 \sum_{t=1}^{T} \alpha_t . \tag{82c}$$

Substituting (82a)–(82c) into (81) gives

$$\begin{aligned}
&\sum_{t=1}^{T} \left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}^*, \mathbf{y}_t^*(\mathbf{x}^*)) \right) \\
&\leq \frac{D^2}{2\alpha_{T+1}} + \sum_{t=1}^{T} \left(M_f^2 \alpha_1 \left\| \mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t) \right\|^2 + D M_f \left\| \mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t) \right\| + \ell_{f,0}^2 \alpha_t \right),
\end{aligned} \tag{83}$$

By Lemma 17–L1., (78) and Assumption B, we have

$$\begin{aligned}
\sum_{t=1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 &\leq \frac{\rho_1^2}{1 - 2\rho_1^2} \|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 \\
&+ \frac{6}{1 - 2\rho_1^2} \left(2L_{\mathbf{y}}^2 \sum_{t=1}^{T} \rho_t^2 \|\mathbf{x}_t - \mathbf{x}^*\|^2 + \sum_{t=2}^{T} \rho_t^2 \|\mathbf{y}_{t-1}^*(\mathbf{x}^*) - \mathbf{y}_t^*(\mathbf{x}^*)\|^2 \right) \\
&\leq \frac{1}{2} D'^2 + \frac{\pi^4}{30} L_{\mathbf{y}}^2 D^2 + \frac{\pi^4}{60} L_{\mathbf{y}}^2 \bar{Y}_{2,T},
\end{aligned} \tag{84a}$$

where $\bar{Y}_{2,T} = \sum_{t=2}^{T} \left\|\mathbf{y}_{t-1}^*(\mathbf{x}^*) - \mathbf{y}_t^*(\mathbf{x}^*)\right\|^2$.

Similarly, we obtain

$$\begin{aligned}\sum_{t=1}^{T} \|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\| &\le \frac{\rho_1}{1-\rho_1}\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\| \\ &+ \frac{1}{1-\rho_1}\left(2L_{\mathbf{y}}\sum_{t=1}^{T}\rho_t\|\mathbf{x}_t - \mathbf{x}^*\| + \sum_{t=2}^{T}\rho_t\|\mathbf{y}_{t-1}^*(\mathbf{x}^*) - \mathbf{y}_t^*(\mathbf{x}^*)\|\right) \\ &\le D' + \pi^2 L_{\mathbf{y}} D + \frac{\pi^2}{2}L_{\mathbf{y}}\bar{Y}_{1,T},\end{aligned} \tag{84b}$$

where $\bar{Y}_{1,T} = \sum_{t=2}^{T} \left\|\mathbf{y}_{t-1}^*(\mathbf{x}^*) - \mathbf{y}_t^*(\mathbf{x}^*)\right\|$.

Let

$$\dot{E}_1(\alpha_1) := \alpha_1 M_f^2\left(\frac{1}{2}D'^2 + \frac{\pi^4}{30}L_{\mathbf{y}}^2 D^2\right) + DM_f\left(D' + \pi^2 L_{\mathbf{y}} D\right).$$

By substituting (84a) and (84b) into (83) and using our choice of the stepsize $\alpha_t = D/(\ell_{f,0}\sqrt{t})$, we obtain

$$\begin{aligned}&\sum_{t=1}^{T}\left(f_t(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t)) - f_t(\mathbf{x}^*, \mathbf{y}_t^*(\mathbf{x}^*))\right) \\ &\le \frac{1}{2}D\ell_{f,0}\sqrt{T} + \dot{E}_1(\alpha_1) + \alpha_1 M_f^2\frac{\pi^4}{60}L_{\mathbf{y}}^2\bar{Y}_{2,T} + DM_f\frac{\pi^2}{2}L_{\mathbf{y}}\bar{Y}_{1,T} + D\ell_{f,0}\sum_{t=1}^{T}\frac{1}{\sqrt{t}},\end{aligned} \tag{85}$$

Let

$$\dot{e}_1 := \frac{3}{2}D\ell_{f,0}, \quad \dot{e}_2 := DM_f\frac{\pi^2}{2}L_{\mathbf{y}}, \quad \dot{e}_3 := \alpha_1 M_f^2\frac{\pi^4}{60}L_{\mathbf{y}}^2.$$

From (85) and using Lemma 13–II., we get

$$\text{BS-Reg}_T \le \dot{e}_1\sqrt{T} + \dot{E}_1(\alpha_1) + \dot{e}_3\bar{Y}_{2,T} + \dot{e}_2\bar{Y}_{1,T}.$$

This completes the proof of the theorem and gives (13). ■

C.3.3 Discussion on the number of inner iterations and the window size

As mentioned before, by using inner gradient descent multiple times, we are able to get more information from each inner function and obtain a tight bound for the dynamic regret in terms of $Y_{p,T}$. However, according to our analysis in Theorems 4 and 7, even for sufficiently large K_t and $w > 1$, the dynamic regret bound can only be improved by a constant factor. A related question is whether we can reduce the value of K_t by using, for example, the smoothness of $\nabla\mathbf{y}_t(\mathbf{x})$, similar to offline bilevel optimization [Chen et al., 2021], or by adopting more advanced optimization techniques, such as acceleration or momentum-type gradient methods for both inner and outer updates [Nesterov, 2003]. These are open problems for us and will be investigated as future work.

C.4 Proof for Non-convex OBO with Partial Information

This section gives regret bounds for OBO in the non-convex setting.

C.4.1 Auxiliary Lemmas

Lemma 21. *Under Assumption A, for all $t \in [T]$ and $\mathbf{x} \in \mathbb{R}^d$, we have*

$$\left\|\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}, \mathbf{y}_{t+1}) - \nabla F_{t,\mathbf{u}}(\mathbf{x}, \mathbf{y}_t^*(\mathbf{x}))\right\|^2 \le M_f^2\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x})\|^2, \tag{86}$$

where $\tilde{\nabla}F_{t,\mathbf{u}}$ is defined in (7) and M_f is given in Lemma 3.

Proof. From (7), we get

$$
\begin{aligned}
&\left\|\tilde{\nabla} F_{t,\mathbf{u}}(\mathbf{x},\mathbf{y}_{t+1}) - \nabla F_{t,\mathbf{u}}(\mathbf{x},\mathbf{y}_t^*(\mathbf{x}))\right\|^2 \\
&= \left\|\frac{1}{W}\sum_{i=0}^{w-1} u_i \left(\tilde{\nabla} f_{t-i}(\mathbf{x},\mathbf{y}_{t+1}) - \nabla f_{t-i}(\mathbf{x},\mathbf{y}_t^*(\mathbf{x}))\right)\right\|^2 \\
&\leq \frac{1}{2W^2}\sum_{i=0}^{w-1}\sum_{j=0}^{w-1} u_i u_j \left\|\tilde{\nabla} f_{t-i}(\mathbf{x},\mathbf{y}_{t+1}) - \nabla f_{t-i}(\mathbf{x},\mathbf{y}_t^*(\mathbf{x}))\right\|^2 \\
&+ \frac{1}{2W^2}\sum_{i=0}^{w-1}\sum_{j=0}^{w-1} u_i u_j \left\|\tilde{\nabla} f_{t-j}(\mathbf{x},\mathbf{y}_{t+1}) - \nabla f_{t-j}(\mathbf{x},\mathbf{y}_t^*(\mathbf{x}))\right\|^2 \\
&= \frac{1}{W^2}\sum_{j=0}^{w-1} u_j \sum_{i=0}^{w-1} u_i \left\|\tilde{\nabla} f_{t-i}(\mathbf{x},\mathbf{y}_{t+1}) - \nabla f_{t-i}(\mathbf{x},\mathbf{y}_t^*(\mathbf{x}))\right\|^2 \\
&\leq \frac{M_f^2}{W^2}\sum_{j=0}^{w-1} u_j \sum_{i=0}^{w-1} u_i \left\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x})\right\|^2 \\
&= M_f^2 \left\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x})\right\|^2 .
\end{aligned}
$$

Here, the first inequality uses Lemma 14 with $c = 1$; the second inequality uses Lemma 3; and the last equality follows since $(1/W)\sum_{i=0}^{w-1} u_i = 1$. ■

Similar to Lemma 17, the following lemma characterizes the inner estimation error $\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|$, where $\mathbf{y}_{t+1}$ is the inner variable update via Algorithm 1. In particular, it shows that by applying inner gradient descent at each round t, we are able to obtain an error bound in terms of the local regret $\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2$ and the inner solution variation $H_{2,T} = \sum_{t=2}^{T}\sup_{\mathbf{x}\in\mathbb{R}^{d_1}}\|\mathbf{y}_{t-1}^*(\mathbf{x}) - \mathbf{y}_t^*(\mathbf{x})\|^2$.

Lemma 22. *Suppose Assumption A holds. If we choose the stepsizes as*

$$
\beta_t = \beta = \frac{2}{\ell_{g,1}+\mu_g}, \quad \text{and} \quad \alpha_t = \alpha \leq \frac{1}{2\sqrt{2}L_{\mathbf{y}}M_f(\kappa_g^2-1)^{1/2}},
$$

for all $t \in [T]$, then the sequence $\{(\mathbf{x}_t,\mathbf{y}_t)\}_{t=1}^T$ generated by Algorithm 1 satisfy

$$
\begin{aligned}
\sum_{t=1}^{T}\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq{}& \frac{(\kappa_g-1)^2}{2(\kappa_g+1)}\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 \\
&+ \frac{1}{2M_f^2}(\frac{\kappa_g-1}{\kappa_g+1})\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2 + 2(\kappa_g-1)^2 H_{2,T}.
\end{aligned}
\tag{87}
$$

Here, $H_{2,T} = \sum_{t=2}^{T}\sup_{\mathbf{x}\in\mathbb{R}^{d_1}}\|\mathbf{y}_{t-1}^*(\mathbf{x}) - \mathbf{y}_t^*(\mathbf{x})\|^2$.

Proof. Since $\beta = 2/(\ell_{g,1}+\mu_g)$, from Lemma 11, we have

$$
\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq \left(1 - \frac{2}{\kappa_g+1}\right)^2 \|\mathbf{y}_t - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 ,
$$

which implies that

$$
\begin{aligned}
\sum_{t=1}^{T}\|\mathbf{y}_{t+1} - \mathbf{y}_t^*(\mathbf{x}_t)\|^2 \leq{}& \left(1 - \frac{2}{\kappa_g+1}\right)^2 \|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2 \\
&+ \left(1 - \frac{2}{\kappa_g+1}\right)^2 \sum_{t=2}^{T}\|\mathbf{y}_t - \mathbf{y}_t^*(\mathbf{x}_t)\|^2.
\end{aligned}
\tag{88}
$$

From Lemma 14, we have

$$\begin{aligned}\sum_{t=2}^{T}\|\mathbf{y}_t-\mathbf{y}_t^*(\mathbf{x}_t)\|^2 &\le \left(1+\frac{1}{\kappa_g+1}\right)\sum_{t=2}^{T}\|\mathbf{y}_t-\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1})\|^2\\ &+(1+\kappa_g+1)\sum_{t=2}^{T}\|\mathbf{y}_t^*(\mathbf{x}_t)-\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1})\|^2\\ &\le \left(1+\frac{1}{\kappa_g+1}\right)\sum_{t=2}^{T}\|\mathbf{y}_t-\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1})\|^2\\ &+2(2+\kappa_g)\sum_{t=2}^{T}\|\mathbf{y}_t^*(\mathbf{x}_t)-\mathbf{y}_t^*(\mathbf{x}_{t-1})\|^2\\ &+2(2+\kappa_g)\sum_{t=2}^{T}\|\mathbf{y}_t^*(\mathbf{x}_{t-1})-\mathbf{y}_{t-1}^*(\mathbf{x}_{t-1})\|^2\\ &\le \left(1+\frac{1}{\kappa_g+1}\right)\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2\\ &+2(2+\kappa_g)\sum_{t=1}^{T}\|\mathbf{y}_{t+1}^*(\mathbf{x}_{t+1})-\mathbf{y}_{t+1}^*(\mathbf{x}_t)\|^2+2(2+\kappa_g)H_{2,T}. \qquad (89)\end{aligned}$$

From Lemma 3 and the update rule of $\mathbf{x}_t$, we obtain

$$\begin{aligned}\|\mathbf{y}_{t+1}^*(\mathbf{x}_{t+1})-\mathbf{y}_{t+1}^*(\mathbf{x}_t)\|^2 &\le L_{\mathbf{y}}^2\|\mathbf{x}_t-\mathbf{x}_{t+1}\|^2\\ &= L_{\mathbf{y}}^2\alpha^2\left\|\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})\right\|^2\\ &\le 2L_{\mathbf{y}}^2\alpha^2\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2\\ &+2L_{\mathbf{y}}^2\alpha^2\left\|\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})-\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\right\|^2\\ &\le 2L_{\mathbf{y}}^2\alpha^2\left(\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+M_f^2\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2\right),\end{aligned}$$

where the second inequality holds due to Lemma 14 and the last inequality follows from Lemma 21.

Now, substituting the above bound into (89), we obtain

$$\begin{aligned}\sum_{t=2}^{T}\|\mathbf{y}_t-\mathbf{y}_t^*(\mathbf{x}_t)\|^2 &\le \left(1+\frac{1}{\kappa_g+1}+4(2+\kappa_g)L_{\mathbf{y}}^2\alpha^2M_f^2\right)\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2\\ &+4\left(2+\kappa_g\right)L_{\mathbf{y}}^2\alpha^2\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+2\left(2+\kappa_g\right)H_{2,T}. \qquad (90)\end{aligned}$$

Substituting (90) into (88), we get

$$\begin{aligned}\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2 &\le \left(1-\frac{2}{\kappa_g+1}\right)^2\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2+A(\alpha)\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2\\ &+B(\alpha)\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+2\left(2+\kappa_g\right)\left(1-\frac{2}{\kappa_g+1}\right)^2H_{2,T}, \qquad (91)\end{aligned}$$

where

$$\begin{aligned}A(\alpha) &:= \left(1+\frac{1}{\kappa_g+1}\right)\left(1-\frac{2}{\kappa_g+1}\right)^2+4\left(2+\kappa_g\right)\left(1-\frac{2}{\kappa_g+1}\right)^2L_{\mathbf{y}}^2\alpha^2M_f^2, \quad \text{and}\\ B(\alpha) &:= 4\left(2+\kappa_g\right)\left(1-\frac{2}{\kappa_g+1}\right)^2L_{\mathbf{y}}^2\alpha^2.\end{aligned}$$

We now proceed to bound terms $A(\alpha)$ and $B(\alpha)$, respectively. Let's bound term $A(\alpha)$ first as

$$\begin{aligned}
A(\alpha) &= \left(1-\frac{2}{\kappa_g+1}\right)\left((1+\frac{1}{\kappa_g+1})(1-\frac{2}{\kappa_g+1})+4(2+\kappa_g)(1-\frac{2}{\kappa_g+1})L_{\mathbf{y}}^2\alpha^2M_f^2\right)\\
&\leq \left(1-\frac{2}{\kappa_g+1}\right)\left(1-\frac{1}{\kappa_g+1}+8(1+\kappa_g)(1-\frac{2}{\kappa_g+1})L_{\mathbf{y}}^2\alpha^2M_f^2\right)\\
&= \left(1-\frac{2}{\kappa_g+1}\right)\left(1-\frac{1}{\kappa_g+1}+8(\kappa_g-1)L_{\mathbf{y}}^2\alpha^2M_f^2\right)\\
&\leq \left(1-\frac{2}{\kappa_g+1}\right),
\end{aligned}$$

where the first inequality is by the inequality $(1+a/2)(1-a)\leq(1-a/2-a^2/2)\leq 1-a/2$ and the second inequality is due to the assumption that $\alpha^2\leq\frac{1}{8(\kappa_g^2-1)L_{\mathbf{y}}^2M_f^2}$.

Next, we bound $B(\alpha)$ as follows

$$B(\alpha)\leq 8(1+\kappa_g)\left(1-\frac{2}{\kappa_g+1}\right)^2L_{\mathbf{y}}^2\alpha^2=\frac{8(\kappa_g-1)^2}{\kappa_g+1}L_{\mathbf{y}}^2\alpha^2\leq\frac{\kappa_g-1}{(\kappa_g+1)^2M_f^2},$$

where the last inequality holds because $\alpha^2\leq\frac{1}{8(\kappa_g^2-1)L_{\mathbf{y}}^2M_f^2}$.

Inserting the above two bounds for $A(\alpha)$ and $B(\alpha)$ into (91) gives

$$\begin{aligned}
\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2 &\leq \left(\frac{\kappa_g-1}{\kappa_g+1}\right)^2\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2+\left(1-\frac{2}{\kappa_g+1}\right)\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2\\
&+\frac{\kappa_g-1}{(\kappa_g+1)^2M_f^2}\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+\frac{4(\kappa_g-1)^2}{\kappa_g+1}H_{2,T}.
\end{aligned}$$

Rearranging the above terms gives (87) and completes the proof. ■

The following lemma shows that the difference between the time-averaged function $F_{t,\mathbf{u}}$ computed at $(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))$ and $(\mathbf{x}_{t+1},\mathbf{y}_t^*(\mathbf{x}_{t+1}))$ is bounded. This extends the single-level setting to the generic weight sequence $\{u_i\}_{i=0}^{w-1}$, and the proof utilizes the ideas from [Aydore et al., 2019, Lemmas 3.2, 3.3] and [Hazan et al., 2017, Theorem 3].

Lemma 23. *Let $\{(f_t,g_t)\}_{t=1}^T$ be the sequence of functions presented to Algorithm 1, satisfying Assumption C. Then, we have*

$$\sum_{t=1}^{T}(F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-F_{t,\mathbf{u}}(\mathbf{x}_{t+1},\mathbf{y}_t^*(\mathbf{x}_{t+1})))\leq\frac{2TM}{W}+2M+\ell_{f,0}H_{1,T}, \tag{92}$$

where $H_{1,T}=\sum_{t=2}^{T}\sup_{\mathbf{x}\in\mathbb{R}^{d_1}}\|\mathbf{y}_t^*(\mathbf{x})-\mathbf{y}_{t-1}^*(\mathbf{x})\|$ and M is defined in Assumption C.

Proof. Observe that

$$\sum_{t=1}^{T}(F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-F_{t,\mathbf{u}}(\mathbf{x}_{t+1},\mathbf{y}_t^*(\mathbf{x}_{t+1})))$$

$$=\sum_{t=1}^{T}\left(F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-F_{t,\mathbf{u}}(\mathbf{x}_{t+1},\mathbf{y}_{t+1}^*(\mathbf{x}_{t+1}))\right) \tag{93a}$$

$$+\sum_{t=1}^{T}\left(F_{t,\mathbf{u}}(\mathbf{x}_{t+1},\mathbf{y}_{t+1}^*(\mathbf{x}_{t+1}))-F_{t,\mathbf{u}}(\mathbf{x}_{t+1},\mathbf{y}_t^*(\mathbf{x}_{t+1}))\right). \tag{93b}$$

In the following, we bound the terms (93a) and (93b) separately.

For (93b), we have

$$\begin{aligned}
\text{(93b)} &= \sum_{t=1}^{T} \frac{1}{W} \sum_{i=0}^{w-1} u_i \left(f_{t-i}(\mathbf{x}_{t+1}, \mathbf{y}^*_{t+1}(\mathbf{x}_{t+1})) - f_{t-i}(\mathbf{x}_{t+1}, \mathbf{y}^*_t(\mathbf{x}_{t+1})) \right) \\
&\le \frac{\ell_{f,0}}{W} \sum_{i=0}^{w-1} u_i \sum_{t=1}^{T} \|\mathbf{y}^*_{t+1}(\mathbf{x}_{t+1}) - \mathbf{y}^*_t(\mathbf{x}_{t+1})\| \\
&= \ell_{f,0} \sum_{t=1}^{T} \|\mathbf{y}^*_{t+1}(\mathbf{x}_{t+1}) - \mathbf{y}^*_t(\mathbf{x}_{t+1})\| \\
&\le \ell_{f,0} H_{1,T},
\end{aligned} \tag{94}$$

where the first inequality is due to Assumption A1. and the second equality follows since $(1/W)\sum_{i=0}^{w-1} u_i = 1$.

For the term (93a), we have

$$\begin{aligned}
\text{(93a)} &= \sum_{t=2}^{T} \Big(F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - F_{t-1,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) \Big) \\
&\quad + F_{1,\mathbf{u}}(\mathbf{x}_1, \mathbf{y}^*_1(\mathbf{x}_1)) - F_{T,\mathbf{u}}(\mathbf{x}_{T+1}, \mathbf{y}^*_{T+1}(\mathbf{x}_{T+1})) \\
&= \sum_{t=2}^{T} \frac{1}{W} \sum_{i=0}^{w-1} u_i \left(f_{t-i}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - f_{t-1-i}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) \right) \\
&\quad + f_1(\mathbf{x}_1, \mathbf{y}^*_1(\mathbf{x}_1)) - \frac{1}{W} \sum_{i=0}^{w-1} u_i f_{T-i}(\mathbf{x}_{T+1}, \mathbf{y}^*_{T+1}(\mathbf{x}_{T+1})).
\end{aligned}$$

Since $\{u_i\}_{i=0}^{w-1}$ is the weight sequence with $1 = u_0 \ge u_1 \dots u_{w-1} > 0$, given in Definition 1, we have

$$\begin{aligned}
&\sum_{i=0}^{w-1} u_i \left(f_{t-i}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - f_{t-1-i}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) \right) \\
&= u_0 f_t(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) + u_1 f_{t-1}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) + \cdots + u_{w-1} f_{t-w+1}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) \\
&- u_0 f_{t-1}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - u_1 f_{t-2}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - \cdots - u_{w-1} f_{t-w}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) \\
&= u_0 f_t(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - u_{w-1} f_{t-w}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) + \sum_{i=1}^{w-1} (u_i - u_{i-1}) f_{t-i}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) \\
&\le u_0 f_t(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - u_{w-1} f_{t-w}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) + \sum_{i=1}^{w-1} (u_{i-1} - u_i) \left| f_{t-i}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) \right| \\
&\le u_0 f_t(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - u_{w-1} f_{t-w}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) + \max_i |f_{t-i}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t))| (u_0 - u_{w-1}),
\end{aligned}$$

which implies that

$$\begin{aligned}
\text{(93a)} &\le \sum_{t=2}^{T} \frac{1}{W} \Big(u_0 f_t(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) - u_{w-1} f_{t-w}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t)) + \max_i |f_{t-i}(\mathbf{x}_t, \mathbf{y}^*_t(\mathbf{x}_t))| (u_0 - u_{w-1}) \Big) \\
&+ f_1(\mathbf{x}_1, \mathbf{y}^*_1(\mathbf{x}_1)) - \frac{1}{W} \sum_{i=0}^{w-1} u_i f_{T-i}(\mathbf{x}_{T+1}, \mathbf{y}^*_{T+1}(\mathbf{x}_{T+1})) \\
&\le \frac{2TM\,(u_0 - u_{w-1})}{W} + M + M \\
&\le \frac{2TM}{W} + M + M,
\end{aligned} \tag{95}$$

where the second inequality is by Assumption C.

Combining (94) with (95), we get (92). ■

C.4.2 Proof of Theorem 9

Proof. Lemma 3 implies that (24) still holds by replacing f_t with F_t. Hence,

$$\begin{aligned}
&F_{t,\mathbf{u}}(\mathbf{x}_{t+1},\mathbf{y}_t^*(\mathbf{x}_{t+1}))-F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\\
&\le \langle\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)),\mathbf{x}_{t+1}-\mathbf{x}_t\rangle+\frac{L_f}{2}\|\mathbf{x}_{t+1}-\mathbf{x}_t\|^2\\
&\le -\alpha\left\langle\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)),\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})\right\rangle+\frac{L_f\alpha^2}{2}\|\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})\|^2. \qquad (96)
\end{aligned}$$

By Lemma 21, we have

$$\begin{aligned}
-\left\langle\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)),\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})\right\rangle &= -\langle\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)),\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\rangle\\
&\quad-\left\langle\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t)),\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})-\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\right\rangle\\
&\le -\frac{1}{2}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+\frac{1}{2}\left\|\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})-\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\right\|^2 \qquad (97a)\\
&\le -\frac{1}{2}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+\frac{M_f^2}{2}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2,
\end{aligned}$$

and

$$\begin{aligned}
\|\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})\|^2 &\le 2\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+2\left\|\tilde{\nabla}F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_{t+1})-\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\right\|^2\\
&\le 2\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2+2M_f^2\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2. \qquad (97b)
\end{aligned}$$

Substituting (97a) and (97b) into (96), rearranging terms and summing up from $t=1$ to $t=T$, we obtain

$$\begin{aligned}
&\left(\frac{\alpha}{2}-L_f\alpha^2\right)\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2\\
&\le\sum_{t=1}^{T}(F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))-F_{t,\mathbf{u}}(\mathbf{x}_{t+1},\mathbf{y}_t^*(\mathbf{x}_{t+1})))+M_f^2\left(\frac{\alpha}{2}+L_f\alpha^2\right)\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2\\
&\le\frac{2TM}{W}+2M+\ell_{f,0}H_{1,T}+M_f^2\left(\frac{\alpha}{2}+L_f\alpha^2\right)\sum_{t=1}^{T}\|\mathbf{y}_{t+1}-\mathbf{y}_t^*(\mathbf{x}_t)\|^2, \qquad (98)
\end{aligned}$$

where the second inequality follows from Lemma 23.

Note that our choices of the stepsizes α_t and β_t as

$$\beta_t=\beta=\frac{2}{\ell_{g,1}+\mu_g},\quad \text{and}\quad \alpha_t=\alpha\le\min\left\{\frac{1}{8L_f},\frac{1}{2\sqrt{2}L_{\mathbf{y}}M_f(\kappa_g^2-1)^{1/2}}\right\},$$

satisfy the condition of Lemma 22. Hence, from Lemma 22, we get

$$\begin{aligned}
&\left(\frac{\alpha}{2}-L_f\alpha^2\right)\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2\\
&\le\frac{2TM}{W}+2M+\ell_{f,0}H_{1,T}+\frac{1}{2}\left(\frac{\alpha}{2}+L_f\alpha^2\right)\left(\frac{\kappa_g-1}{\kappa_g+1}\right)\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2\\
&+M_f^2\left(\frac{\alpha}{2}+L_f\alpha^2\right)(\kappa_g-1)^2\left(\frac{\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2}{2(\kappa_g+1)}+2H_{2,T}\right).
\end{aligned}$$

Rearranging the terms leads to

$$\begin{aligned}
&\frac{1}{2(\kappa_g+1)}\left(\frac{(\kappa_g+3)\alpha}{2}-3(\kappa_g+\frac{1}{3})L_f\alpha^2\right)\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t,\mathbf{y}_t^*(\mathbf{x}_t))\|^2\\
&\le\frac{2TM}{W}+2M+\ell_{f,0}H_{1,T}+M_f^2\left(\frac{\alpha}{2}+L_f\alpha^2\right)(\kappa_g-1)^2\left(\frac{\|\mathbf{y}_1-\mathbf{y}_1^*(\mathbf{x}_1)\|^2}{2(\kappa_g+1)}+2H_{2,T}\right). \qquad (99)
\end{aligned}$$

Since $\alpha \leq 1/(8L_f)$, we have

$$\frac{(\kappa_g+3)\alpha}{2} - 3(\kappa_g + \frac{1}{3})L_f\alpha^2 \geq \frac{(\kappa_g+1)\alpha}{2} - 3(\kappa_g+1)L_f\alpha^2 \geq \frac{\alpha}{8}(\kappa_g+1), \quad \text{and}$$
$$\frac{\alpha}{2} + L_f\alpha^2 \leq \frac{5}{8}\alpha.$$

Substituting the above observations into (99) gives

$$\begin{aligned}\frac{\alpha}{16}\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2 &\leq \frac{2TM}{W} + 2M + \ell_{f,0}H_{1,T} \\ &\quad + \frac{5}{8}\alpha M_f^2(\kappa_g-1)^2\left(\frac{\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2}{2(\kappa_g+1)} + 2H_{2,T}\right).\end{aligned}$$

Therefore, we get the following bound

$$\begin{aligned}\sum_{t=1}^{T}\|\nabla F_{t,\mathbf{u}}(\mathbf{x}_t, \mathbf{y}_t^*(\mathbf{x}_t))\|^2 &\leq \frac{16}{\alpha}\left(\frac{2TM}{W} + 2M + \ell_{f,0}H_{1,T}\right) \\ &\quad + 10M_f^2(\kappa_g-1)^2\left(\frac{\|\mathbf{y}_1 - \mathbf{y}_1^*(\mathbf{x}_1)\|^2}{2(\kappa_g+1)} + 2H_{2,T}\right) \\ &= \mathcal{O}\left(\frac{T}{W} + H_{1,T} + H_{2,T}\right).\end{aligned}$$

This completes the proof. ■

D Addendum to Section 4: Implementation Details and Additional Experiments

D.1 Details on Online Hyperparameter Learning for Dynamic Regression

The synthetic data are generated as follows: To simulate the distribution changes, we generate the output according to $b_t = \mathbf{a}_t^\top \mathbf{y}_s^*(\mathbf{x}_s^*) + \epsilon_t$, where $(\mathbf{x}_s^*, \mathbf{y}_s^*(\mathbf{x}_s^*)) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$ is the underlying model for s-th stage, and $\epsilon_t \in [0, 0.1]$ is the random noise. We consider two setups for the underlying model: (i) there are three changes ($S = 3$) in the minimizers $(\mathbf{x}_s^*, \mathbf{y}_s^*(\mathbf{x}_s^*))$, and (ii) the underlying model is fixed ($S = 1$), i.e., $(\mathbf{x}^*, \mathbf{y}^*(\mathbf{x}^*)) = (\mathbf{x}_s^*, \mathbf{y}_s^*(\mathbf{x}_s^*))$ for all $t \in [T]$. The time horizon, the outer and inner dimensions are set to $T = 5000$, $d_1 = 1$ and $d_2 = 5$, respectively.

We used a grid-search of parameters in our experiments in Subsection 4.1. For the grid-search setting, we select the best performing parameters K_t, α, β from a grid $\{5, 10\} \times \{0.001, 0.01, 0.1, 0.5\} \times \{0.001, 0.01, 0.1, 0.5\}$. The smoothing (averaging) parameter δ is set to 0.9. All algorithms have been run on a Mac machine equipped with a 1.8 GHz Intel Core i5 processor and 8 GB RAM.

D.2 Details and Additional Experiments on Online Parametric Loss Tuning Implementation

This subsection provides details on implementing online parametric loss tuning tailored for imbalanced data. Additionally, it includes extra experiments conducted on two other datasets: Tadpole and Adult.

D.2.1 Dataset Specifications and Model Architectures

MNIST The MNIST image dataset [LeCun et al., 2010] comprises 10 classes of human-written numbers ranging from 0 to 9. The dataset contains a total of 60,000 training images and 10,000 testing images, each sized at 28×28. Consequently, there are approximately 6,000 training images and 1,000 testing images for each class. To introduce imbalance into the training and validation data, we randomly selected 5000×0.6^i, $i = 0, 1, \ldots 9$ samples from the original training data for each class. These samples were then divided into new training and validation datasets at a 4:1 ratio. We employed a 4-layer convolutional neural network (CNN) for all comparison algorithms. Each convolutional block in the network consists of a 3 × 3 convolution (with padding=1 and stride=1), batch normalization, ReLU activation, and 2 × 2 max pooling. The CNN has 64 filters in every convolutional layer.

Tadpole In the Appendix, we conduct additional experiments on the Tadpole dataset [Marinescu et al., 2019]. The Tadople dataset is introduced in Grand Challenge [2], a platform for end-to-end development of machine learning solutions in biomedical imaging. Tadpole is an abbreviation for The Alzheimer's Disease Prediction Of Longitudinal Evolution (TADPOLE), a subset of the Alzheimer's Disease Neuroimaging Initiative (ADNI) [3], which constitutes an extensive data collection for Alzheimer's disease (AD). Initially, Tadpole contains 12,741 samples and 1,907 features. Its classes include individuals classified as cognitively normal (CN), mild cognitive impairment (MCI) or Alzheimer's disease (AD). We only select 17 commonly-used features, including 'CDRSB', 'ADAS11', 'MMSE', 'RAVLT_immediate', 'Hippocampus', 'WholeBrain', 'Entorhinal', 'MidTemp', 'FDG', 'AV45', 'ABETA_UPENNBIOMK9_04_19_17', 'TAU_UPENNBIOMK9_04_19_17', 'PTAU_UPENNBIOMK9_04_19_17', 'APOE4', 'AGE', 'ADAS13', 'Ventricles'. We exclusively select classes MCI and AD to form the two-class classification task. The two classes are already imbalanced, with 2,106 samples in the AD class and 4,044 in the MCI class. To further imbalance the dataset, we only select half of the samples from AD. We utilize a 2-layer multilayer perceptron (MLP) with ReLU as the activation function and employ Dropout for regularization.

Adult We also conduct additional experiments on the Adult dataset [Becker and Kohavi, 1996], aiming to predict an individual's annual income based on various factors, including the individual's education level, age, gender, occupation and more. The dataset originally comprises 48,842 samples and 15 features. After removing samples with missing values and duplicated features following the process introduced in Kaggle [4], we have 45,175 samples and 11 features. The two classes are defined as follows: income less than or equal to $50K (class 0) and income greater than $50K (class 1). The original distribution is already imbalanced (0 vs 1 is 3:1). Thus, we do not modify it further. We also employ a 2-layer multilayer perceptron (MLP) with ReLU as the activation function and Dropout for regularization.

D.2.2 Baselines and Setting Details

In our experiments, we compare our method to two baselines: one being AutoBalance [Li et al., 2021], and the other being Single-Level OGD [Zinkevich, 2003].

AutoBalance AutoBalance [Li et al., 2021] is an offline bilevel gradient descent framework that updates hyperparameters $\mathbf{x}_t$ and the model $\mathbf{y}_t$ to address imbalance issues. We essentially adopt all the settings from the Autobalance study. Specifically, in all three datasets, the inner-level optimization trains the CNN model using a learning rate of 0.1, momentum of 0.9, and weight decay of $1e-4$. However, to adapt it to the online environment and ensure a fair comparison, AutoBalance will, at each timestep, utilize all the observed data until the current timestep to train the model instead of employing a fixed number of batches, as in the original setting of AutoBalance. This will allow AutoBalance to run quickly at the beginning but progressively slower as time passes. At the outer level, AutoBalance does not initiate training from the beginning. Instead, AutoBalance usually initiates the outer level after the network achieves near-zero loss. For MNIST, AutoBalance starts outer-level training at the 120th timestep, while it starts at the 80th and 40th timesteps for Tadpole and Adult, respectively. The learning rate for the outer level is 0.001 on all the three datasets. Our OAGD follows the same setting of AutoBalance for both the inner- and outer-level training.

Single-Level OGD The Single-Level OGD [Zinkevich, 2003] updates the model, $\mathbf{y}_t$, with fixed hyperparameters, $\mathbf{x}$, at each timestep solely based on the newly observed data using gradient descent. Specifically, the hyperparameters include adjustments in multiplicative and additive logits, along with the inverse class weight. For Single-Level OGD, the multiplicative logits adjustment is 1, the additive logits adjustment is 0, and the inverse class weight is 1, resulting in a vanilla cross-entropy loss. The learning rate is 0.1 on all the three datasets.

D.2.3 Additional Experiments

Figure 5 provides a performance comparison (mean±std) for parametric loss tuning on imbalanced **Tadpole** data across five runs. We compare our OAGD ($w = 5, 10$) with AutoBalance and Single-Level OGD. OAGD achieves a comparable balanced testing accuracy to AutoBalance but with a reduced runtime. AutoBalance outperforms

[2] https://tadpole.grand-challenge.org/Data/

[3] https://adni.loni.usc.edu/

[4] https://www.kaggle.com/code/amirhosseinzinati/adult-income-k-nearest-neighbors-knn

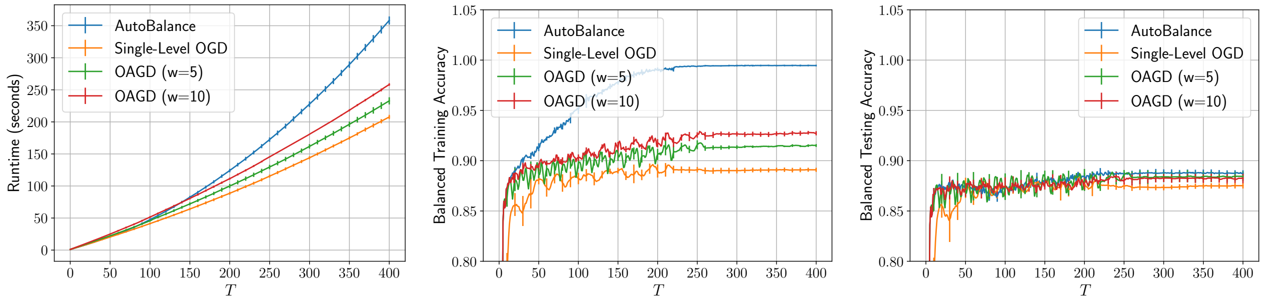

Figure 5: Performance comparison (mean±std) on parametric loss tuning for imbalanced **Tadpole** data over five runs. We compare our OAGD ($w = 5, 10$) with AutoBalance and Single-Level OGD. OAGD achieves comparable balanced testing accuracy to AutoBalance but with a reduced runtime.

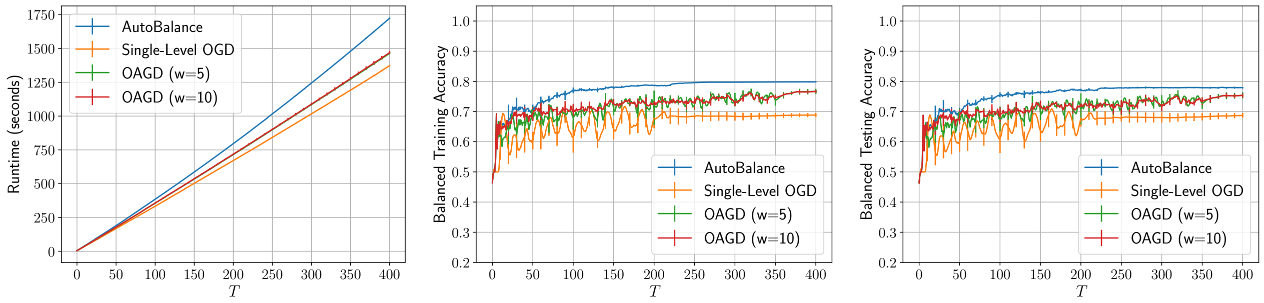

Figure 6: Performance comparison (mean±std) on parametric loss tuning for imbalanced **Adult** data over five runs. We compare our OAGD ($w = 5, 10$) with AutoBalance and Single-Level OGD. OAGD achieves comparable balanced testing accuracy to AutoBalance but with a reduced runtime.

OAGD and Single-Level OGD in terms of balanced training accuracy. This is because AutoBalance utilizes many more samples than OAGD and OGD. This allows AutoBalance to have a higher chance of overfitting the training data, resulting in high training accuracy, while still obtaining similar balanced testing accuracy compared to other methods.

Figure 6 provides a performance comparison (mean±std) on parametric loss tuning for imbalanced **Adult** data over five runs. We compare our OAGD ($w = 5, 10$) with AutoBalance and Single-Level OGD. OAGD achieves comparable balanced accuracy to AutoBalance but with a reduced runtime.

D.3 Details and Additional Experiments on Online Meta Learning

This subsection provides the implementation details and additional experiments on miniImageNet of online meta learning.

D.3.1 Datasets and Model Architectures

FC100 FC100 [Oreshkin et al., 2018] is a dataset derived from CIFAR100, containing 100 classes, with each class comprising 600 images of size 32. Following [Oreshkin et al., 2018], the 100 classes are divided as follows: 60 classes for meta-training, 20 classes for meta-validation, and 20 classes for meta-testing. There are 36,000, 12,000, 12,000 samples in the original training, validation and testing datasets, respectively. We transform them into 20,000, 600 and 600 training, validation and testing tasks using the *TaskDataset* tool from learn2learn [Arnold et al., 2020]. For all comparison algorithms, we employ a 4-layer convolutional neural networks(CNN) comprising four convolutional blocks. Each block consists of a 3×3 convolution (padding=1, stride=2), batch normalization, ReLU activation, and 2×2 max pooling. Additionally, each convolutional layer contains 64 filters.

MiniImageNet The miniImageNet dataset [Vinyals et al., 2016] is derived from ImageNet and comprises 100 classes, each containing 600 images sized at 84×84. Following the repository, we partition these classes into 64 classes for meta-training, 16 classes for meta-validation, and 20 classes for meta-testing. Following the

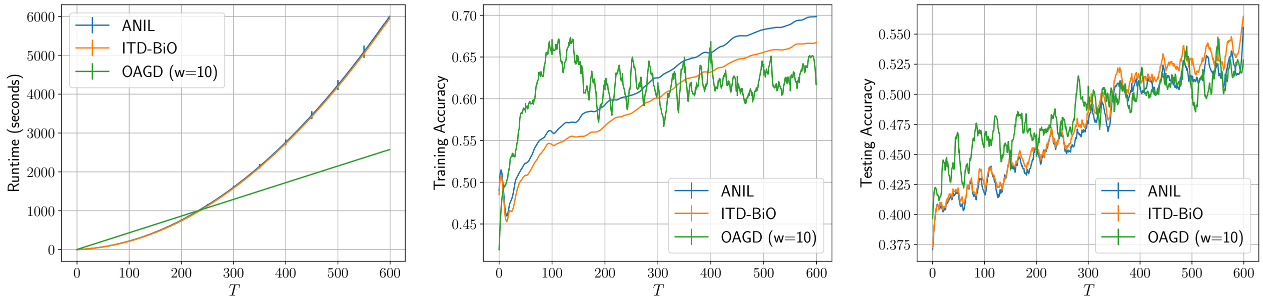

Figure 7: Performance comparison (mean±std) for online meta-learning on the **miniImageNet** dataset across five runs. We compare our OAGD ($w = 10$) with ANIL and ITD-BiO. OAGD achieves comparable accuracy to the baselines while significantly reducing runtime.

repository, we use a four-layer CNN with four convolutional blocks, where each block sequentially consists of a 3×3 convolution, batch normalization, ReLU activation, and 2×2 max pooling. Each convolutional layer has 64 filters.

D.3.2 Baselines and Setting Details

In our experiments, we compare our method to three baselines: MAML [Finn et al., 2017], ANIL [Raghu et al., 2019], and ITD-BiO [Ji et al., 2021b]. Notably, these three methods are all implemented using iterative differentiation within the PyTorch framework. They leverage two modules, the *features* and the *head*, albeit in distinct ways. The *features* are used to process the raw input, such as the CNN which processes the image in our case, while the *head* is responsible for the final classification. To ensure a fair comparison, we set the inner learning rate $\beta = 0.1$, the outer learning rate α=0.001, and inner step $K = 20$ for all the methods.

MAML MAML stands as the foundational work of meta-learning, in which meta-parameters are learned in the outer loop, while task-specific models are learned in the inner loop using only a small amount of data from the current task. In the implementation, MAML combines *features* and the *head* into the meta model, which is then cloned by the local model to perform local adaptation. We update all parameters of the meta model, including those of the *features* and *head*.

ANIL ANIL stands as a widely used meta-learning algorithm that simplifies MAML by eliminating the inner loop for all parts of the MAML-trained network except the task-specific head. In its implementation, only the *head* of the meta model, cloned by the local model, is employed for subsequent local adaptation. Nonetheless, the *features* remain utilized for data processing. Ultimately, both the parameters of the *features* and the *head* undergo updates.

ITD-BiO ITD-BiO is a gradient-based stochastic bilevel optimization framework relying on iterative differentiation (ITD). In its implementation, the meta model cloned by the local model comprises solely the *head* for subsequent local adaptation. Nevertheless, we continue to utilize the *features* for data processing. Ultimately, only the parameters of the *features* undergo updates.

It is worth noting that we do not adapt all the baselines to the online environment; instead, we directly utilize the implementation from [Ji et al., 2021b]. In the online environment, data is observed in batches at a time, and the offline method updates the model by incorporating all the data observed up until the current timestep. This will initially speed up the training, but it will progressively slow down over time. Eventually, the entire process will become very time-consuming. Instead, the implementation uses only a fixed number of batches (tasks) randomly sampled from all available tasks at each timestep, which, in our case, is 32. We only use a specific window-size number of batches (tasks) and load them sequentially.

D.3.3 Results on Additional Dataset

We conducted our experiments on an additional dataset, **miniImageNet**. Figure 7 provides a performance comparison (mean±std) for meta-learning on the miniImageNet dataset across five runs. We compare our OAGD ($w = 10$) with ANIL and ITD-BiO. OAGD achieves comparable accuracy but with a shorter runtime.

D.4 Sensitivity Analysis

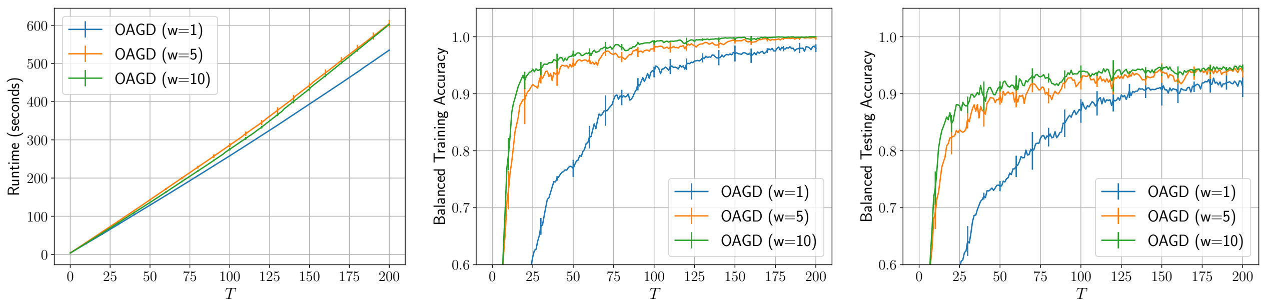

Figure 8: Performance comparison (mean±std) for parametric loss tuning on imbalanced **MNIST** data across five runs. We compare our OAGD across different window sizes ($w = 1, 5, 10$). The larger the window size, the better the accuracy and the longer the runtime.

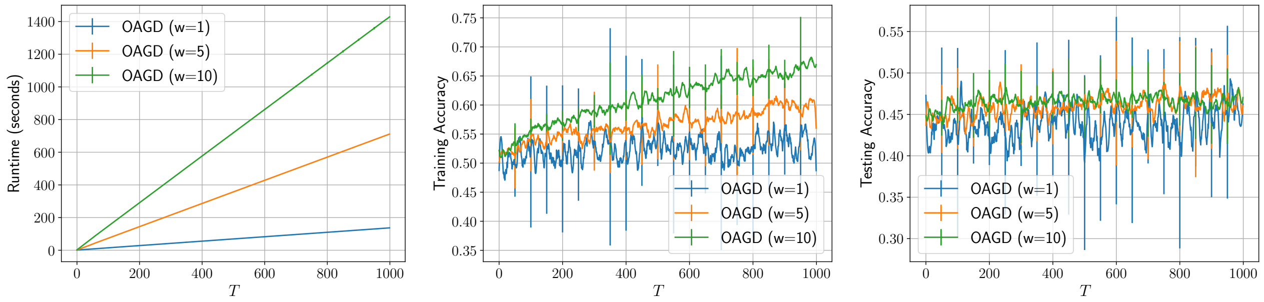

Figure 9: Performance comparison (mean±std) for meta-learning on **FC100** data across five runs. We compare our OAGD with different window sizes ($w = 1, 5, 10$). The larger the window size, the better the accuracy and the longer the runtime.

In this section, we perform sensitivity analysis on the window size w, learning rate (both inner β and outer α), and inner optimization step K to acquire a comprehensive understanding of our method.

D.4.1 Sensitivity Analysis of Window Size w

Figures 8 and 9 illustrate the results of our OAGD on MNIST and FC100 using different window sizes ($w = 1, 5, 10$). As observed, the larger the window size, the higher the accuracy and the longer the runtime. This is expected, as a larger window size allows for the use of more information from previous timesteps. Consequently, the gradient can be approximated with greater accuracy, leading to improved results. However, this also requires more computations for gradient calculation. It is important to note that we do not merely record previous gradients. Instead, we leverage previous data and the current model to compute the gradients, an approach that has been proven to be more effective.

D.4.2 Sensitivity Analysis of Learning Rates α and β

Figures 10 and 11 display the sensitivity results for the inner and outer learning rates, β and α, on the Tadpole dataset. Likewise, Figures 12 and 13 exhibit the sensitivity results for the inner and outer learning rates on the Adult dataset. Specifically, when analyzing the sensitivity to the inner learning rate, we fix the outer learning rate at 0.001 and experiment with different inner learning rates ($\beta = 0.9, 0.5, 0.1, 0.01$). Conversely, when examining sensitivity to the outer learning rate, we set the inner learning rate to 0.1 and test different outer learning rates ($\alpha = 0.0001, 0.001, 0.01, 0.1$). Our observations indicate that our OAGD is not particularly sensitive to changes in either the inner or outer learning rates, as all cases demonstrate consistently high accuracy.

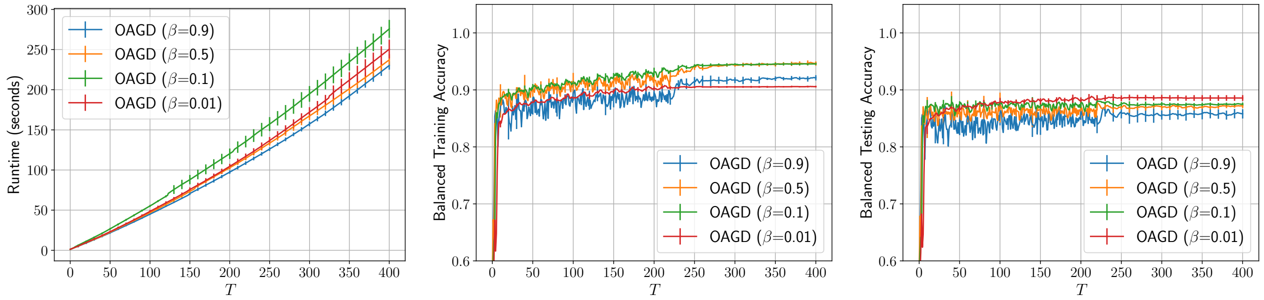

Figure 10: Performance comparison (mean±std) on parametric loss tuning for imbalanced **Tadpole** data over five runs. We compare our OAGD ($w = 10$) using various inner learning rates ($\beta = 0.01, 0.1, 0.5, 0.9$) while keeping the outer learning rate fixed at 0.001. Our OAGD is not significantly affected by changes in the inner learning rate.

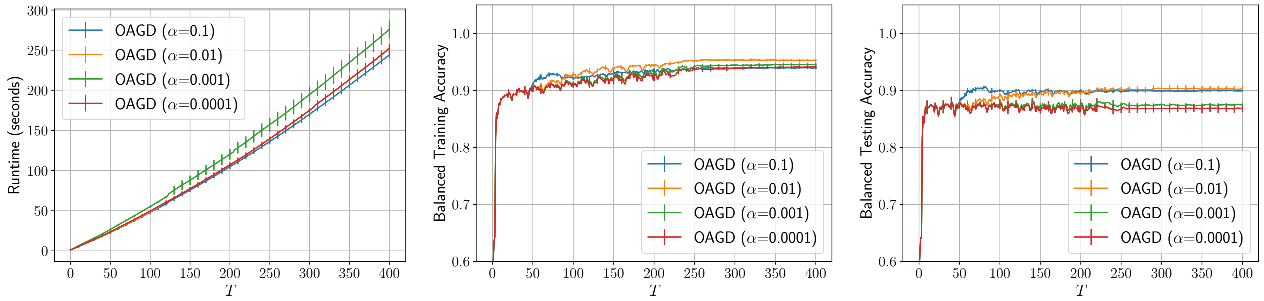

Figure 11: Performance comparison (mean±std) on parametric loss tuning for imbalanced **Tadpole** data over five runs. We compare our OAGD ($w = 10$) using various outer learning rates ($\alpha = 0.0001, 0.001, 0.01, 0.1$) while maintaining a fixed inner learning rate of 0.1. Our OAGD is not sensitive to the outer learning rate.

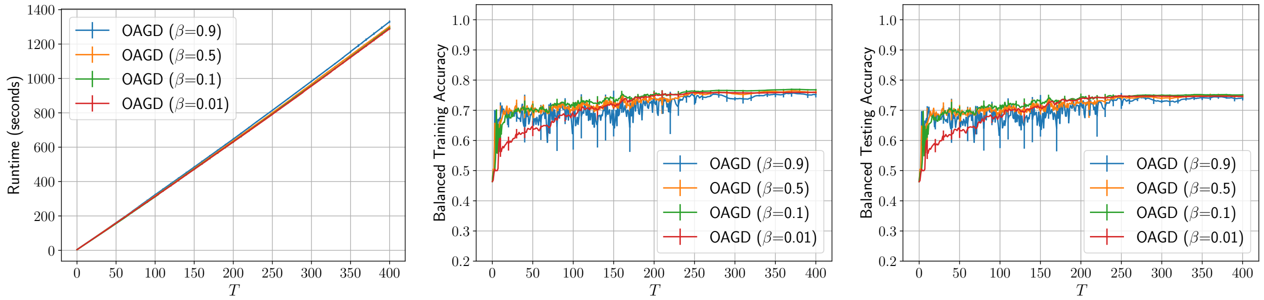

Figure 12: Performance comparison (mean±std) on parametric loss tuning for imbalanced **Adult** data over five runs. We compare our OAGD ($w = 10$) using various inner learning rates ($\beta = 0.01, 0.1, 0.5, 0.9$) while keeping the outer learning rate fixed at 0.001. Our OAGD is not sensitive to the inner learning rate.

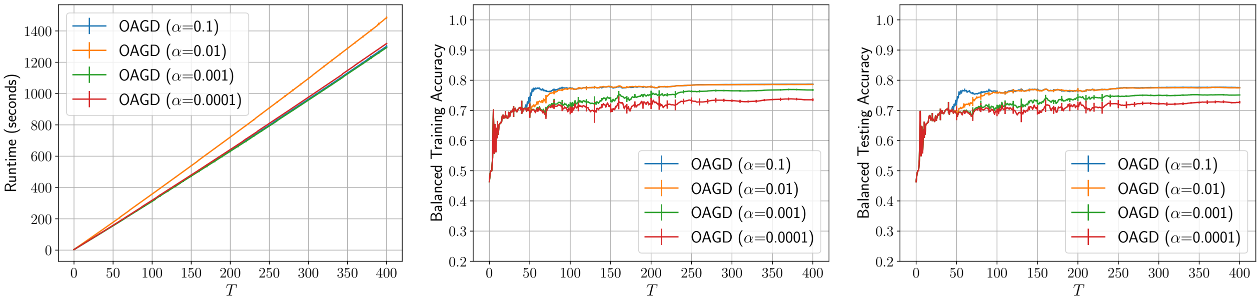

Figure 13: Performance comparison (mean±std) on parametric loss tuning for imbalanced **Adult** data across five runs. We compare our OAGD ($w = 10$) using different outer learning rates ($\alpha = 0.0001, 0.001, 0.01, 0.1$) while keeping the inner learning rate fixed at 0.1. Our OAGD is not sensitive to the outer learning rate.

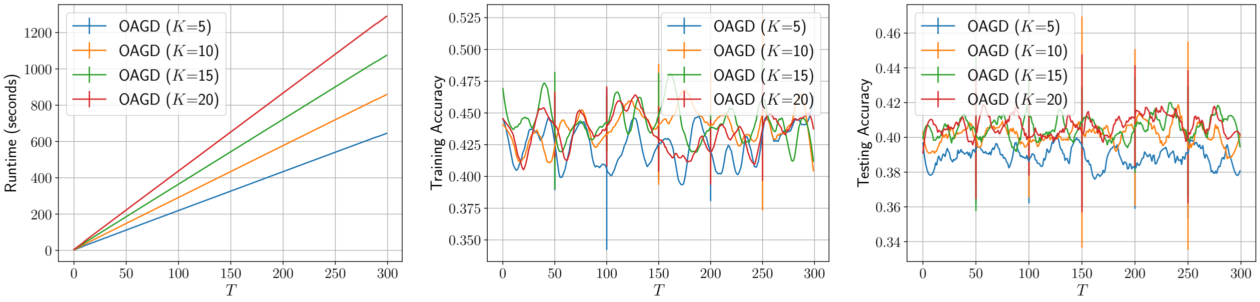

Figure 14: Performance comparison (mean±std) on online meta-learning for **miniImageNet** data across five runs. We compare our OAGD ($w = 10$) with varying inner steps for inner optimization. A larger inner step may yield improved and more stable accuracy but leads to longer runtime.

D.4.3 Sensitivity Analysis to Inner Step K

Figure 14 provides the sensitivity analysis results to different numbers of inner optimization steps $K_t = K$ for each round t. We can see that a larger number of inner optimization steps can make the accuracy better and more stable. However, this will take longer runtime.