Journal of Machine Learning Research 26 (2025) 1-53 Submitted 7/24; Revised 3/25; Published 6/25

Universal Online Convex Optimization Meets Second-order Bounds

Lijun Zhang ZHANGLJ@LAMDA.NJU.EDU.CN
Yibo Wang WANGYB@LAMDA.NJU.EDU.CN
National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
School of Artificial Intelligence, Nanjing University, Nanjing, China

Guanghui Wang GWANG369@GATECH.EDU
College of Computing, Georgia Institute of Technology, Atlanta, USA

Jinfeng Yi YIJINFENG@JD.COM
JD AI Research, Beijing, China

Tianbao Yang TIANBAO-YANG@TAMU.EDU
Department of Computer Science and Engineering, Texas A&M University, College Station, USA

Editor: Vianney Perchet

Abstract

Recently, several universal methods have been proposed for online convex optimization, and attain minimax rates for multiple types of convex functions simultaneously. However, they need to design and optimize one surrogate loss for each type of functions, making it difficult to exploit the structure of the problem and utilize existing algorithms. In this paper, we propose a simple strategy for universal online convex optimization, which avoids these limitations. The key idea is to construct a set of experts to process the *original* online functions, and deploy a meta-algorithm over the *linearized* losses to aggregate predictions from experts. Specifically, the meta-algorithm is required to yield a second-order bound with excess losses, so that it can leverage strong convexity and exponential concavity to control the meta-regret. In this way, our strategy inherits the theoretical guarantee of *any* expert designed for strongly convex functions and exponentially concave functions, up to a double logarithmic factor. As a result, we can plug in off-the-shelf online solvers as black-box experts to deliver problem-dependent regret bounds. For general convex functions, it maintains the minimax optimality and also achieves a small-loss bound. Furthermore, we extend our universal strategy to online composite optimization, where the loss function comprises a time-varying function and a fixed regularizer. To deal with the composite loss functions, we employ a meta-algorithm based on the optimistic online learning framework, which not only enjoys a second-order bound, but also can utilize estimations for upcoming loss functions. With suitable configurations, we show that the additional regularizer does not contribute to the meta-regret, thus ensuring the universality in the composite setting.

Keywords: online convex optimization, online composite optimization, universal algorithms, strongly convex functions, exponentially concave functions

1. Introduction

Online convex optimization (OCO) has become a leading online learning framework, capable of modeling various real-world problems such as online routing and spam filtering (Hazan, 2016). OCO can be seen as a structured repeated game, with the following protocol. At each round t, the online learner chooses $\mathbf{x}_t$ from a convex set $\mathcal{X}$. After committing to this choice, a convex cost function $f_t : \mathcal{X} \mapsto \mathbb{R}$ is revealed, and the loss incurred by the learner is $f_t(\mathbf{x}_t)$. The learner aims to minimize the cumulative loss over T rounds, i.e., $\sum_{t=1}^T f_t(\mathbf{x}_t)$, which is equivalent to minimizing the regret (Cesa-Bianchi and Lugosi, 2006):

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \tag{1}$$

defined as the excess loss of the learner compared to the minimum loss of any fixed choice.

In the literature, there are plenty of algorithms for OCO (Shalev-Shwartz, 2011; Orabona, 2019). For example, online gradient descent (OGD) with $O(1/\sqrt{t})$ step size achieves $O(\sqrt{T})$ regret for general convex functions (Zinkevich, 2003); OGD with $O(1/t)$ step size attains $O(\log T)$ regret for strongly convex functions (Shalev-Shwartz et al., 2007); online Newton step (ONS) enjoys $O(d \log T)$ regret for exponentially concave (abbr. exp-concave) functions, where d is the dimensionality (Hazan et al., 2007). Besides, there exist more powerful online algorithms such as ADAGRAD (Duchi et al., 2011) and its variants (Wan and Zhang, 2022; Yang et al., 2025) that are equipped with problem-dependent regret bounds (Srebro et al., 2010; Chiang et al., 2012; Kingma and Ba, 2015; Mukkamala and Hein, 2017; Reddi et al., 2018), which become tighter when the problem has special structures. Although we have rich theories for OCO, its application requires heavy domain knowledge: Users must know the type of functions in order to select an appropriate algorithm, and when dealing with strongly convex functions and exp-concave functions, they also need to estimate the moduli of strong convexity and exponential concavity.

The lack of universality of previous algorithms motivates the development of universal methods for OCO (Bartlett et al., 2007; Do et al., 2009). One milestone is MetaGrad of van Erven and Koolen (2016), which can handle general convex functions as well as exp-concave functions. Later, Wang et al. (2019) propose Maler, which further supports strongly convex functions explicitly. In a subsequent work, Wang et al. (2020b) develop UFO, which exploits smoothness to deliver small-loss regret bounds, i.e., regret bounds that depend on the minimal loss. However, the three aforementioned methods need to design one surrogate loss for each possible type of functions, which is both tedious and challenging. Furthermore, because they rely on surrogate losses, it is difficult to produce problem-dependent regret bounds, except the small-loss one.

To avoid the above limitations, we propose a simple yet universal strategy for online convex optimization, including both the standard setting where the learner only suffers a time-varying loss $f_t(\mathbf{x}_t)$ in each round, and the composite setting where the learner incurs a composite loss $f_t(\mathbf{x}_t) + r(\mathbf{x}_t)$ with an additional regularizer $r(\mathbf{x})$. The key idea is to build a set of experts to process the *original* online functions, and deploy a meta-algorithm, which ensures a second-order bound with excess losses, to aggregate predictions from experts. Since the expert observes the original functions, it can utilize their structures to deliver problem-dependent regret bounds. The second-order bound of the meta-algorithm facilitates the management for different types of online functions, securing the universality of our strategy. In the following, we specify our strategy for the two settings, respectively.

Standard Online Convex Optimization. In the standard setting, we first create a set of experts to handle the uncertainty of the type of online functions and (possibly) the associated parameters. When facing unknown continuous variables, we discretize them by constructing a geometric series to cover the range of their values. Second, we run a meta-algorithm to track the best expert on the fly, but use the *linearized* losses to measure the performance. To benefit from strong convexity and exponential concavity, we require the meta-algorithm to yield a second-order bound with excess losses, and choose Adapt-ML-Prod (Gaillard et al., 2014) as an example. Specifically, let ℓ_t and ℓ_t^i be the loss of the meta-algorithm and the i-th expert in the t-th round, respectively. The regret of the meta-algorithm satisfies

$$\sum_{t=1}^{T}(\ell_t - \ell_t^i) = O\left(\sqrt{\sum_{t=1}^{T}(\ell_t - \ell_t^i)^2}\right), \ \forall i \tag{2}$$

where for brevity we drop the dependence on the number of experts.

By incorporating existing methods for strongly convex functions, exp-concave functions and general convex functions as experts, we obtain a universal algorithm with the following properties:

- For strongly convex functions, our algorithm is agnostic to the modulus of strong convexity, at the price of maintaining $O(\log T)$ experts. More importantly, it inherits the regret bound of *any* expert designed for strongly convex functions, with a negligible additive factor of $O(\log\log T)$. As a result, we can deliver any problem-dependent or independent regret bound, without prior knowledge of strong convexity;
- For exp-concave functions, the above statements are also true;
- For general convex functions, the theoretical guarantee is a mix of the regret bound of the expert and the second-order bound of Adapt-ML-Prod. When the functions are convex and smooth, it yields a small-loss bound.

Compared to previous universal methods (van Erven and Koolen, 2016; Wang et al., 2019, 2020b), our algorithm has the following advantages:

- It decouples the loss used by the expert-algorithm and that by the meta-algorithm. In this way, we can directly utilize existing online algorithms as black-box subroutines, and do not need to design surrogate losses;
- For strongly convex functions and exp-concave functions, the regret bound of our algorithm achieves *the best of all worlds*, provided that both the domain and gradients are bounded.

Online Composite Optimization. In the composite setting, the loss function is defined as the sum of two functions:

$$f_t(\mathbf{x}) + r(\mathbf{x}) \tag{3}$$

where $f_t(\mathbf{x})$ is a time-varying function, and $r(\mathbf{x})$ is a fixed convex regularizer, such as the ℓ_1-norm and the trace norm (Tibshirani, 1996; Toh and Yun, 2010). Consequently, our goal is to minimize the regret in terms of (3):

$$\sum_{t=1}^{T}\big[f_t(\mathbf{x}_t) + r(\mathbf{x}_t)\big] - \min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\big[f_t(\mathbf{x}) + r(\mathbf{x})\big]. \tag{4}$$

To minimize (4), a natural attempt is to treat the sum of $f_t(\mathbf{x}) + r(\mathbf{x})$ as a new function, and pass it to our universal algorithm designed for the standard setting. However, this approach exhibits several limitations. For example, for an exp-concave function $f_t(\mathbf{x})$ and a convex regularizer $r(\mathbf{x})$, the summation function is not necessarily an exp-concave function (Yang et al., 2018), so that the optimal regret for exp-concave loss functions is unattainable. Moreover, the summation function ignores the presence of the regularizer, thereby failing to leverage the benefit of $r(\mathbf{x})$. Therefore, we need to modify the previous universal strategy to explicitly support composite functions.

In the composite setting, we first construct a set of experts in the same way as the standard setting to process the *original composite* loss functions. Then, we run our meta-algorithm on the *composite linearized* loss, which includes the linearized time-varying function and the regularier. Moreover, instead of utilizing Adapt-ML-Prod as the meta-algorithm, we choose an optimistic online learning method, called Optimistic-Adapt-ML-Prod (Wei et al., 2016), which not only guarantees a second-order bound, but also can exploit estimations for upcoming loss functions. With appropriate estimations, we demonstrate that Optimistic-Adapt-ML-Prod is able to yield the second-order bound that solely depends on the time-varying function $f_t(\mathbf{x})$, so that the strong convexity and exponential concavity of $f_t(\mathbf{x})$ can be utilized to control the meta-regret. By employing previous methods in online composite optimization as experts, our universal algorithm maintains its favorable properties. Specifically, it preserves the regret bound of *any* expert designed for strongly convex time-varying functions and exp-concave time-varying functions, up to a negligible double logarithmic factor. As a result, we can still deliver any problem-dependent or independent regret bounds. For general convex time-varying functions, our method ensures the guarantee that comprises the regret bound of the expert and the second-order bound of Optimistic-Adapt-ML-Prod.

New Regret Bounds. During the analysis, we revisit three existing methods for standard online convex optimization and online composite optimization (Duchi et al., 2010b; Chiang et al., 2012; Scroccaro et al., 2023), and derive *new* theoretical results as byproducts.

- In the standard setting with smooth and strongly convex functions, we extend online extra-gradient descent (OEGD) of Chiang et al. (2012), and derive a novel gradient-variation bound (Theorem 21), i.e., regret bounds that rely on the variation of $\nabla f_t(\mathbf{x})$;
- In the composite setting with smooth and strongly convex time-varying functions, and smooth and general convex time-varying functions, we re-analyze composite objective mirror descent (COMID) of Duchi et al. (2010b), and establish new (pseudo) small-loss bounds (Theorems 24 and 25), i.e., regret bounds that depend on the cumulative loss on $f_t(\mathbf{x})$ suffered by the algorithm itself;
- In the composite setting with smooth and exp-concave time-varying functions, we generalize optimistic composite mirror descent (OCMD) of Scroccaro et al. (2023), and derive a new gradient-variation bound (Theorem 27) and a new (pseudo) small-loss bound (Theorem 28).

Although the above results are not our primary focus, they further enrich the developments of online convex optimization.

A preliminary version of this paper was presented at the 39th International Conference on Machine Learning (Zhang et al., 2022). In this paper, we expand upon the conference version by further investigating online composite optimization and proposing a new universal algorithm designed for the composite loss functions. Moreover, two analysis byproducts in the composite setting also serve as the extensions.

Organization. The remainder of the paper is structured as follows. Section 2 briefly reviews the related work. Section 3 illustrates our main results for standard online convex optimization, including the specification of our universal strategy and its theoretical guarantees for strongly convex, exp-concave and general convex functions, respectively. In Section 4, we extend the investigations to online composite optimization. Section 5 presents the analysis of all theorems. Section 6 concludes this paper and discusses future work. The appendix details our revisitation as well as new theoretical results for previous methods.

2. Related Work

In this section, we review the related work in OCO, including traditional algorithms, universal algorithms, online composite optimization, and parameter-free algorithms.

2.1 Traditional Algorithms

For general convex functions, the most popular algorithm is online gradient descent (OGD), which attains $O(\sqrt{T})$ regret by setting the step size as $\eta_t = O(1/\sqrt{t})$ (Zinkevich, 2003). For λ-strongly convex functions, the regret bound can be improved to $O(\frac{1}{\lambda}\log T)$ by running OGD with the step size $\eta_t = O(1/[\lambda t])$ (Shalev-Shwartz et al., 2007). For α-exp-concave functions, online Newton step (ONS), with prior knowledge of the parameter α, achieves $O(\frac{d}{\alpha}\log T)$ regret, where d is the dimensionality (Hazan et al., 2007). These regret bounds for general convex functions, strongly convex functions, and exp-concave functions are known to be minimax optimal (Ordentlich and Cover, 1998; Abernethy et al., 2008), which means that they cannot be improved in the worst case. However, these bounds only exhibit the relationship with problem-independent quantities, e.g., the time horizon T and the dimensionality d, and thus do not reflect the property of the online problem at hand.

To exploit the structure of the problem, various problem-dependent (or data-dependent) regret bounds have been established in recent years (Srebro et al., 2010; Duchi et al., 2010a, 2011; Chiang et al., 2012; Orabona et al., 2012; Tieleman and Hinton, 2012; Zeiler, 2012; Yang et al., 2014; Kingma and Ba, 2015; Mukkamala and Hein, 2017; Reddi et al., 2018; Wang et al., 2020a). The problem-dependent bounds reduce to the minimax rates in the worst case, but can be better under favorable conditions.

One well-known result is the small-loss bound which is very popular in the studies of online learning (Littlestone and Warmuth, 1994; Auer et al., 2002; Shalev-Shwartz, 2007; Luo and Schapire, 2015). When the functions are smooth and nonnegative, the regret for general convex functions, λ-strongly convex functions, and α-exp-concave functions can be upper bounded by $O(\sqrt{L_T^*})$, $O(\frac{1}{\lambda}\log L_T^*)$, and $O(\frac{d}{\alpha}\log L_T^*)$ respectively, where

$$L_T^* = \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \tag{5}$$

is the cumulative loss of the best point in $\mathcal{X}$ (Srebro et al., 2010; Orabona et al., 2012; Zhang et al., 2019a; Wang et al., 2020b). These small-loss bounds could be much tighter when L_T^* is small, and still ensure the minimax optimality otherwise. Another problem-dependent guarantee is the gradient-variation bound for smooth functions (Chiang et al., 2012; Yang et al., 2014; Mohri and Yang, 2016), which replaces L_T^* in the upper bounds with the gradient variation:

$$V_T = \sum_{t=1}^{T} \max_{\mathbf{x} \in \mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|_2^2. \tag{6}$$

Then, the regret bounds become smaller if the online functions evolve gradually.

Besides smoothness, it is also possible to exploit other structural properties of the functions, such as the sparsity of gradients. One representative work is ADAGRAD (Duchi et al., 2010a, 2011), which incorporates knowledge of the geometry of the data observed in earlier iterations to perform more informative gradient-based learning. Let $\mathbf{g}_{1:T,j}$ be the vector obtained by concatenating the j-th element of the gradient sequence $\nabla f_1(\mathbf{x}_1), \ldots, \nabla f_T(\mathbf{x}_T)$. ADAGRAD achieves

$$O\left(\sum_{j=1}^{d} \|\mathbf{g}_{1:T,j}\|_2\right) \text{ and } O\left(\frac{1}{\lambda}\sum_{j=1}^{d} \log \|\mathbf{g}_{1:T,j}\|_2\right)$$

regret for general convex functions and λ-strongly convex functions, respectively (Duchi et al., 2010a). These two bounds match the minimax rates in the worst case, and become tighter when gradients are sparse. Since the seminal work of ADAGRAD, a series of problem-dependent online algorithms have been developed, including RMSprop (Tieleman and Hinton, 2012; Mukkamala and Hein, 2017), Adam (Kingma and Ba, 2015; Reddi et al., 2018) and AdamW (Loshchilov and Hutter, 2019).

Although there exist abundant algorithms and theories for OCO, how to choose them in practice is a nontrivial task. To ensure good performance, we not only need to know the type of functions, but also need to estimate the moduli of strong convexity and exponential concavity. The requirement of human intervention restricts the application of OCO to real-world problems, and motivates the development of universal algorithms for OCO.

2.2 Universal Algorithms

The first universal method for OCO is adaptive online gradient descent (AOGD) (Bartlett et al., 2007), which interpolates between the $O(\sqrt{T})$ regret of general convex functions and the $O(\log T)$ regret of strongly convex functions automatically. However, AOGD requires the modulus of strong convexity in each round, and does not support exp-concave functions. Do et al. (2009) develop a proximal variant of AOGD, but it suffers the same limitations.

The studies of universal methods are further advanced by the MetaGrad algorithm of van Erven and Koolen (2016), which adapts to a much broader class of functions, including general convex functions and exp-concave functions. Under the framework of learning with expert advice (Cesa-Bianchi and Lugosi, 2006), MetaGrad is a two-layer algorithm consisting of a set of experts and a meta-algorithm. To handle exponential concavity, each expert minimizes one surrogate loss

$$\ell_{t,\eta}^{exp}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 \left[(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t\right]^2 \tag{7}$$

parameterized by the step size η, where $\mathbf{g}_t = \nabla f_t(\mathbf{x}_t)$. MetaGrad maintains $O(\log T)$ experts to minimize (7) with different step sizes, and combines their predictions with a meta-algorithm. In this way, it attains $O(\frac{d}{\alpha}\log T)$ regret for α-exp-concave functions, without knowing the value of α. At the same time, MetaGrad also achieves an $O(\sqrt{T\log\log T})$ regret bound for general convex functions. Although we can treat strongly convex functions as exp-concave and obtain an $O(d\log T)$ regret bound, there exists an $O(d)$ gap from the minimax rate of strongly convex functions (Abernethy et al., 2008).

To deal with strongly convex functions, Wang et al. (2019) design another surrogate loss

$$\ell_{t,\eta}^{str}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 G^2 \|\mathbf{x}_t - \mathbf{x}\|_2^2 \tag{8}$$

where G is an upper bound of the norm of gradients. Their proposed method, named as Maler, introduces additional $O(\log T)$ experts to optimize (8), and obtains $O(\frac{1}{\lambda}\log T)$ regret for λ-strongly convex functions. Similar to MetaGrad, it gets rid of the priori knowledge of strong convexity. Wang et al. (2019) further propose the following surrogate loss:

$$\ell_{t,\eta}^{con}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 G^2 D^2 \tag{9}$$

where D is an upper bound of the diameter of $\mathcal{X}$, and obtain the optimal $O(\sqrt{T})$ regret for general convex functions. To exploit smoothness, Wang et al. (2020b) propose the following surrogate loss

$$\ell_{t,\eta}^{str,smo}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 \|\mathbf{g}_t\|_2^2 \|\mathbf{x}_t - \mathbf{x}\|_2^2 \tag{10}$$

for strongly convex and smooth functions, reuse the surrogate loss in (7) for exp-concave and smooth functions, and introduce the following surrogate loss

$$\ell_{t,\eta}^{con,smo}(\mathbf{x}) = -\eta(\mathbf{x}_t - \mathbf{x})^\top \mathbf{g}_t + \eta^2 \|\mathbf{g}_t\|_2^2 D^2 \tag{11}$$

for convex and smooth functions. Under the smoothness condition, their algorithm achieves $O(\frac{1}{\lambda}\log L_T^*)$, $O(\frac{d}{\alpha}\log L_T^*)$, and $O(\sqrt{L_T^*})$ bounds for λ-strongly convex functions, α-exp-concave functions, and general convex functions respectively, where L_T^* is defined in (5).

Besides supporting more types of functions, MetaGrad was also extended to avoid the knowledge of the Lipschitz constant G. Specifically, Mhammedi et al. (2019) first design a basic algorithm called MetaGrad+C, which requires an initial estimate of the Lipschitz constant, and then uses a restarting scheme to set this parameter online. In this way, the final algorithm, named as MetaGrad+L, adapts to the Lipschitz hyperparameter automatically. Furthermore, Mhammedi et al. (2019) also remove the need to specify the number of rounds in advance, and van Erven et al. (2021) propose a refined version of MetaGrad+L, which only restarts the meta-algorithm but not the experts.

From the above discussion, we observe that state-of-the-art universal methods in the literature (van Erven and Koolen, 2016; Wang et al., 2019, 2020b) all rely on the construction of surrogate losses, which brings the following two issues:

1. They have to design one surrogate loss for each possible type of functions, which is very challenging. That is because we need to ensure that the regret in terms of surrogate losses can be converted back to the regret of original functions;

2. Since all the experts receive the surrogate losses instead of the original functions, it is difficult to exploit the structure of the particular problem instance. Of course, one can use problem-dependent algorithms to optimize the surrogate losses, but they may not share the same structure with the original functions. Except the small-loss bound (Wang et al., 2020b), it is unclear how to generate other problem-dependent bounds.

2.3 Online Composite Optimization

Online composite optimization aims to minimize the composite loss functions in (3) to generate decisions with certain favorable properties, such as the sparsity introduced by the ℓ_1-norm regularizer (Tibshirani, 1996; Zhang et al., 2015) and the low-rankness encouraged by the trace norm regularizer (Toh and Yun, 2010; Zhang et al., 2019b).

In recent decades, there have been plenty of investigations in online composite optimization (Duchi and Singer, 2009; Xiao, 2009; Duchi et al., 2010b; Yang et al., 2024c). Specifically, the seminal work of Duchi and Singer (2009) proposes the forward backward splitting (FOBOS) method and establishes the $O(\sqrt{T})$ and $O(\frac{1}{\lambda}\log T)$ regret bounds for general convex and λ-strongly convex $f_t(\mathbf{x})$, respectively. Later, based on the primal-dual subgradient framework (Nesterov, 2009), Xiao (2009) develops the regularized dual averaging (RDA) method which ensures the same theoretical guarantees as FOBOS. Duchi et al. (2010b) propose the composite objective mirror descent (COMID) method based on a different mirror descent framework (Beck and Teboulle, 2003), and also achieve the same $O(\sqrt{T})$ and $O(\frac{1}{\lambda}\log T)$ bounds for general convex and λ-strongly convex $f_t(\mathbf{x})$, respectively. Recently, for α-exp-concave $f_t(\mathbf{x})$, Yang et al. (2024c) propose the proximal online Newton step (ProxONS) method and establish an $O(\frac{d}{\alpha}\log T)$ regret bound.

Besides the above methods, there are other efforts investigating problem-dependent regret bounds (Yang et al., 2014; Mohri and Yang, 2016; Joulani et al., 2020; Scroccaro et al., 2023), by exploiting the special properties of the problem, such as the smoothness of $f_t(\mathbf{x})$. Specifically, based on the optimistic online learning framework (Rakhlin and Sridharan, 2013), Mohri and Yang (2016) introduce the adaptive optimistic FTRL (AO-FTRL) method, which is further generalized by Joulani et al. (2020). The key idea is to utilize an estimation of the upcoming loss function for decision updating. If the estimation is accurate, the regret can be very small; otherwise, the minimax rate is preserved. Based on similar ideas, Scroccaro et al. (2023) develop the optimistic composite mirror descent (OptCMD) method, and achieve $O(\sqrt{V_T})$ and $O(\frac{1}{\lambda}\log V_T)$ regret bounds for smooth and general convex, and smooth and λ-strongly convex $f_t(\mathbf{x})$, respectively.

Despite extensive methods in online composite optimization, applying them in practical problems still requires to know the type of loss functions in advance. Moreover, in the literature, there is no universal algorithm that explicitly handles composite losses.

2.4 Parameter-Free Algorithms

A parallel line of research is parameter-free online learning, which aims to design algorithms to handle unconstrained domains with unavailable bound D. The motivation is that traditional online algorithms, such as OGD (Zinkevich, 2003), require some prior knowledge about the domain to set their parameters, e.g., the step size. To avoid this limitation, parameter-free algorithms are designed to automatically adapt to the norms of comparators rather than the domain itself (Mcmahan and Streeter, 2012, 2013; Orabona, 2014; Orabona and Pál, 2016; Cutkosky and Boahen, 2016, 2017; Foster et al., 2017; Orabona and Pál, 2018; Cutkosky and Orabona, 2018; Cutkosky, 2019; Mhammedi and Koolen, 2020). The study of parameter-free algorithms mainly focuses on general convex functions, and is complementary to the development of universal algorithms.

2.5 Follow-Up Literature Review

Since the publication of our first version (Zhang et al., 2022), there have been a number of significant developments in universal online learning, which we briefly discuss here.

In the literature, Yan et al. (2023) extend our universal strategy by establishing the gradient-variation bound for general convex functions, and achieve the first universal gradient-variation guarantee for three types of functions. But their method relies on a complex three-layer structure with challenging theoretical analysis, and the derived bounds are suboptimal for general convex functions. Later, Wang et al. (2024a) extend the strategy of Yan et al. (2023) to the composite stochastically extended adversarial setting, but their extension still suffers from the same limitations. To address these issues, Yan et al. (2024) retain the core idea of our universal strategy, and introduce a simpler two-layer universal method that can ensure minimax gradient-variation bounds for three types of functions. Recently, Xie et al. (2024) relax the smoothness assumption for gradient-variation bounds, and establish novel theoretical guarantees under the generalized smoothness condition (Li et al., 2023).

Additionally, Yang et al. (2024a) consider the non-stationary environment in universal online learning, and establish universal small-loss bounds for adaptive regret—a classical performance metric for non-stationary environment—based on our strategy. Yang et al. (2024b) reduce the number of projection operations in our universal algorithm from $O(\log T)$ per round to $O(1)$, significantly improving the computational efficiency.

In summary, these follow-up studies demonstrate the broad impact of our universal strategy across multiple topics of online learning, including gradient-variation bounds, adaptive regret minimization and others, validating the versatility and significance of our work.

3. Standard Online Convex Optimization

In this section, we first introduce necessary preliminaries, and then explain the motivation of our method. Next, we elaborate the proposed strategy for the standard OCO, and then present theoretical guarantees for strongly convex functions, exp-concave functions and general convex functions. Finally, we discuss the extension to unknown gradient bound.

3.1 Preliminaries

We start with two common assumptions of OCO (Zinkevich, 2003; van Erven and Koolen, 2016; Zhao et al., 2020; Zhang et al., 2021).

Assumption 1 *The gradients of all functions are bounded by G, i.e.,*

$$\max_{\mathbf{x}\in\mathcal{X}} \|\nabla f_t(\mathbf{x})\|_2 \leq G, \ \forall t \in [T]. \tag{12}$$

To facilitate presentations, we assume the value of G is known beforehand, and will return to this issue in Section 3.7.

Assumption 2 *The diameter of the domain $\mathcal{X}$ is bounded by D, i.e.,*

$$\max_{\mathbf{x},\mathbf{y}\in\mathcal{X}} \|\mathbf{x}-\mathbf{y}\|_2 \leq D. \tag{13}$$

Then, we introduce the definitions of strongly convex functions and exp-concave functions (Boyd and Vandenberghe, 2004; Cesa-Bianchi and Lugosi, 2006).

Definition 1 *A function $f : \mathcal{X} \mapsto \mathbb{R}$ is λ-strongly convex if*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle + \frac{\lambda}{2}\|\mathbf{y}-\mathbf{x}\|_2^2, \tag{14}$$

for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.

Definition 2 *A function $f : \mathcal{X} \mapsto \mathbb{R}$ is α-exp-concave if $\exp(-\alpha f(\cdot))$ is concave over $\mathcal{X}$.*

In the analysis, we will make use of the following property of exp-concave functions (Hazan et al., 2007, Lemma 3).

Lemma 3 *Suppose $f : \mathcal{X} \mapsto \mathbb{R}$ is α-exp-concave. Under Assumptions 1 and 2, we have*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle + \frac{\beta}{2}\langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle^2, \tag{15}$$

for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, where $\beta = \frac{1}{2}\min\{\frac{1}{4GD}, \alpha\}$.

At the end, we formally state the problem setting of universal OCO, including its definition and the classical algorithmic framework.

Definition 4 *A strategy is considered universal for online convex optimization if, without knowing the function type, it satisfies the following conditions:*

(i) For general convex functions, it ensures a sublinear regret bound of $O(\sqrt{T})$;

(ii) For strongly convex functions with an unknown strong convexity modulus λ, it ensures a logarithmic regret bound of $O(\frac{1}{\lambda}\log T)$;

(iii) For exp-concave functions with an unknown exponential concavity modulus α, it ensures a logarithmic regret bound of $O(\frac{d}{\alpha}\log T)$.

As we summarize in Section 2.2, the classical framework for universal OCO (van Erven and Koolen, 2016; Wang et al., 2019, 2020b) typically includes the following two components:

1. A set of experts, each of which handles a specific type of function with a corresponding modulus. For example, the expert handling strongly convex functions is instantiated with an existing online algorithm designed for strongly convex functions, e.g., OGD (Shalev-Shwartz et al., 2007), and is configured with parameters that relate to a specific strong convexity modulus;
2. A meta-algorithm that can combine the predictions from all experts to achieve desired regret bounds in Definition 4 for unknown online functions.

Remark: It is important to note that a straightforward approach for universal OCO is to employ existing prediction with expert advice (PEA) methods as the meta-algorithm to aggregate predictions from experts by using *original* functions. However, this approach can lead to an $O(\sqrt{T})$ meta-regret bound in the worst case, which dominates the overall bounds for all three types of functions, thereby undermining the goal of universal online learning.

3.2 Motivations

Similar to existing universal methods (van Erven and Koolen, 2016; Wang et al., 2019, 2020b), we follow the framework of prediction with expert advice. Specifically, we construct a set of experts for each type of functions and use a meta-algorithm to aggregate their predictions. The difference is that our universal strategy adopts two novel ingredients:

(i) the experts process the *original* functions so that they can exploit the structure of the problem instance to deliver problem-dependent regret bounds;

(ii) the meta-algorithm chooses the *linearized* losses and makes use of second-order bounds to control the meta-regret.

In the following, we take strongly convex functions as an example to explain the motivation.

To simplify notations, we assume that experts in the set $\mathcal{E}$ are ordered, and use E^i to denote the i-th expert. Let $\mathbf{x}_t$ and $\mathbf{x}_t^i$ be the output of the meta-algorithm and the expert E^i in the t-th round, respectively. The regret of the meta-algorithm can be decomposed as the sum of the meta-regret and the expert-regret:

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = \underbrace{\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^i)}_{:=\text{meta-regret}} + \underbrace{\sum_{t=1}^{T} f_t(\mathbf{x}_t^i) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})}_{:=\text{expert-regret}}.$$

To bound the expert-regret, we can directly utilize theoretical guarantees of the expert-algorithm. So, the key is how to ensure a small meta-regret.

Instead of using the original function, our meta-algorithm chooses the linearized loss

$$l_t(\mathbf{x}) = \langle \nabla f_t(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle \tag{16}$$

to measure the performance of the expert. From Definition 1, we have the following relationship between the meta-regret in terms of $f_t(\cdot)$ and the meta-regret in terms of $l_t(\cdot)$:

$$\begin{aligned} \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^i) &\overset{(14)}{\leq} \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle - \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2 \\ &\overset{(16)}{=} \sum_{t=1}^{T} \big(l_t(\mathbf{x}_t) - l_t(\mathbf{x}_t^i) \big) - \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2. \end{aligned} \tag{17}$$

Since we require the meta-algorithm to yield a second-order bound in the form of (2), the meta-regret in terms of $l_t(\cdot)$ becomes

$$\begin{aligned} \sum_{t=1}^{T} \big(l_t(\mathbf{x}_t) - l_t(\mathbf{x}_t^i) \big) &\overset{(2)}{=} O\left(\sqrt{\sum_{t=1}^{T} \big(l_t(\mathbf{x}_t) - l_t(\mathbf{x}_t^i) \big)^2} \right) \\ &\overset{(16)}{=} O\left(\sqrt{\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle^2} \right). \end{aligned} \tag{18}$$

From Assumption 1, we further have

$$\begin{aligned}\sum_{t=1}^{T}\left(l_t(\mathbf{x}_t)-l_t(\mathbf{x}_t^i)\right) \overset{(12),(18)}{=} & O\left(\sqrt{G^2\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_t^i\|_2^2}\right)\\ = & O\left(\frac{G^2}{\lambda}\right)+\frac{\lambda}{2}\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_t^i\|_2^2,\end{aligned} \tag{19}$$

where the last step follows from the fact that $\lambda > 0$, and the basic inequality $2\sqrt{ab} \leq a+b, \forall a,b \geq 0$ with the choices of $a=\frac{G^2}{2\lambda}$ and $b=\frac{\lambda}{2}\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_t^i\|_2^2$. Combining (17) and (19), we have the following meta-regret

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t)-\sum_{t=1}^{T} f_t(\mathbf{x}_t^i)=O\left(\frac{G^2}{\lambda}\right)$$

which is small and thus does not affect the optimality of the algorithm.

These discussions deliver the following conclusion: It is the second-order bound in (18) that makes it possible to exploit the negative term in (17), which arises from the strong convexity. Based on Lemma 3, we observe a similar phenomenon for exp-concave functions.

3.3 Our Universal Strategy

Our universal strategy for standard online convex optimization (USC) is summarized in Algorithm 1. We will consider three types of convex functions: strongly convex functions, exp-concave functions, and general convex functions.

Let $\mathcal{A}_{str}$ be the set of candidate algorithms designed for strongly convex functions, and $\mathcal{P}_{str}$ be the set of possible values of the modulus of strong convexity. Note that although the modulus of strong convexity is continuous, we can construct a finite set $\mathcal{P}_{str}$ to approximate its value, which will be elaborated in Section 3.4. For each algorithm $A \in \mathcal{A}_{str}$ and each $\lambda \in \mathcal{P}_{str}$, we create an expert $E(A,\lambda)$ by invoking algorithm A with parameter λ, and add it to the set $\mathcal{E}$ consisting of experts. Similarly, let $\mathcal{A}_{exp}$ be the set of algorithms designed for exp-concave functions, and $\mathcal{P}_{exp}$ be the set of values of the modulus of exponential concavity. For each algorithm $A \in \mathcal{A}_{exp}$ and each $\alpha \in \mathcal{P}_{exp}$, we instantiate an expert $E(A,\alpha)$ by running algorithm A with parameter α, and add it to $\mathcal{E}$. General convex functions can be handled more easily, and we denote the set of algorithms designed for them by $\mathcal{A}_{con}$. Then, we simply create an expert $E(A)$ for each $A \in \mathcal{A}_{con}$ and add it to $\mathcal{E}$. For notational simplicity, we denote $E(A,\emptyset)=E(A)$. Detailed steps for constructing experts are shown in Steps 3-10.

Next, we deploy a meta-algorithm to track the best expert on the fly. Here, one can use any method that enjoys second-order bounds with excess losses (Koolen and Erven, 2015; Mhammedi et al., 2019). We choose Adapt-ML-Prod (Gaillard et al., 2014) because it is simpler and already satisfies our requirements. In the t-th round, we denote by p_t^i the weight assigned to E^i, and $\mathbf{x}_t^i$ the prediction of E^i. The weights are determined according to Adapt-ML-Prod (Step 12). After receiving predictions from all experts in $\mathcal{E}$ (Step 13), USC submits the weighted average (Step 14):

$$\mathbf{x}_t=\sum_{i=1}^{|\mathcal{E}|} p_t^i \mathbf{x}_t^i. \tag{20}$$

Algorithm 1 A Universal Strategy for Online Convex Optimization (USC)

1: **Input:** $\mathcal{A}_{str}$, $\mathcal{A}_{exp}$, $\mathcal{A}_{con}$, $\mathcal{P}_{str}$ and $\mathcal{P}_{exp}$
2: Initialize $\mathcal{E} = \emptyset$
3: **for** each $(\mathcal{A}, \mathcal{P})$ in $\{(\mathcal{A}_{str}, \mathcal{P}_{str}), (\mathcal{A}_{exp}, \mathcal{P}_{exp}), (\mathcal{A}_{con}, \emptyset)\}$ **do**
4:     **for** each algorithm $A \in \mathcal{A}$ **do**
5:         **for** each parameter $\theta \in \mathcal{P}$ **do**
6:             Create an expert $E(A, \theta)$
7:             $\mathcal{E} = \mathcal{E} \cup E(A, \theta)$
8:         **end for**
9:     **end for**
10: **end for**
11: **for** $t = 1$ **to** T **do**
12:     Calculate the weight p_t^i of each expert E^i by

$$p_t^i = \frac{\eta_{t-1}^i w_{t-1}^i}{\sum_{j=1}^{|\mathcal{E}|} \eta_{t-1}^j w_{t-1}^j}$$

13:     Receive $\mathbf{x}_t^i$ from each expert E^i in $\mathcal{E}$
14:     Output the weighted average $\mathbf{x}_t = \sum_{i=1}^{|\mathcal{E}|} p_t^i \mathbf{x}_t^i$
15:     Observe the loss function $f_t(\cdot)$
16:     Send the required information of $f_t(\cdot)$ to each expert in $\mathcal{E}$
17: **end for**

Then, it observes the loss function $f_t(\cdot)$ and sends the required information to all experts so that they can update their predictions (Steps 15-16). If the expert is a first-order algorithm, we only need to send the gradient of $f_t(\cdot)$. Thus, USC may query the gradient *multiple* times in each round. It is worth to mention that allowing the online learner to observe multiple gradients does not affect the minimax rates in the full-information setting (Abernethy et al., 2008), where the learner can observe the entire function. Therefore, we can still use existing minimax rates to verify the optimality of USC.

Finally, we briefly explain how to calculate the weights of experts, i.e., p_t^i's. As we explained before, the meta-algorithm uses the linearized loss in (16) to measure the performance of each expert. In particular, the loss of E^i is given by

$$l_t^i := l_t(\mathbf{x}_t^i) = \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t^i - \mathbf{x}_t \rangle.$$

Under Assumptions 1 and 2, we have

$$|l_t^i| \leq \|\nabla f_t(\mathbf{x}_t)\|_2 \|\mathbf{x}_t^i - \mathbf{x}_t\|_2 \overset{(12),(13)}{\leq} GD.$$

Because Adapt-ML-Prod requires the loss to lie in $[0, 1]$, we normalize l_t^i in the following way

$$\ell_t^i = \frac{\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t^i - \mathbf{x}_t \rangle + GD}{2GD} \in [0, 1]. \tag{21}$$

Then, the loss of the meta-algorithm suffered in the t-th round becomes

$$\ell_t = \sum_{i=1}^{|\mathcal{E}|} p_t^i \ell_t^i \overset{(20),(21)}{=} \frac{1}{2}. \tag{22}$$

According to Adapt-ML-Prod (Gaillard et al., 2014), the weight of expert E^i is determined by

$$p_t^i = \frac{\eta_{t-1}^i w_{t-1}^i}{\sum_{j=1}^{|\mathcal{E}|} \eta_{t-1}^j w_{t-1}^j} \tag{23}$$

where

$$\eta_{t-1}^i = \min\left\{\frac{1}{2}, \sqrt{\frac{\ln|\mathcal{E}|}{1+\sum_{s=1}^{t-1}(\ell_s - \ell_s^i)^2}}\right\},\ t \geq 1, \tag{24}$$

$$w_{t-1}^i = \left(w_{t-2}^i\big(1+\eta_{t-2}^i(\ell_{t-1} - \ell_{t-1}^i)\big)\right)^{\frac{\eta_{t-1}^i}{\eta_{t-2}^i}},\ t \geq 2.$$

In the beginning, we set $w_0^i = 1/|\mathcal{E}|$. As indicated by (24), Gaillard et al. (2014) use an adaptive way to set multiple time-varying learning rates.

To formally state the meta-regret of USC, we introduce the following theorem.

Theorem 5 *Let Adapt-ML-Prod be the meta-algorithm that evaluates expert performance using the linearized loss in* (21)*. Then, for the λ-strongly convex function $f_t(\cdot)$, the meta-regret is bounded by*

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^i) = O\left(\frac{G^2}{\lambda}\right).$$

For the β-exp-concave function $f_t(\cdot)$, the meta-regret is bounded by

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^i) = O\left(\frac{1}{\beta}\right).$$

For the general convex function $f_t(\cdot)$, the meta-regret is bounded by

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^i) = O(GD\sqrt{T}).$$

In the following, we dig into the algorithmic details of our universal strategy.

3.4 Strongly Convex Functions

We present the regret bound of our strategy when encountering strongly convex functions. To apply USC in Algorithm 1, we need to specify $\mathcal{A}_{str}$, the set of candidate algorithms, and $\mathcal{P}_{str}$, the set of possible values of the modulus of strong convexity. To build $\mathcal{A}_{str}$, we can utilize any existing algorithm for online strongly convex optimization, such as

- OGD for strongly convex functions (SC-OGD) (Shalev-Shwartz et al., 2007);

- ADAGRAD for strongly convex functions (Duchi et al., 2010a);
- Online extra-gradient descent (OEGD) for strongly convex and smooth functions (Chiang et al., 2012);
- SC-RMSProp (Mukkamala and Hein, 2017);
- SAdam (Wang et al., 2020a);
- S^2OGD for strongly convex and smooth functions (Wang et al., 2020b).

It should be noted that Chiang et al. (2012) only investigate OEGD for exp-concave functions and general convex functions, under the smoothness condition. In Appendix A, we extend OEGD to strongly convex functions and obtain a gradient-variation bound of order $O(\log V_T)$, which may be of independent interest.[1]

We now construct $\mathcal{P}_{str}$. Without loss of generality, we assume the unknown modulus λ is both lower bounded and upper bounded. In particular, we assume $\lambda \in [1/T, 1]$, because there is no need to explicitly consider the cases $\lambda < 1/T$ and $\lambda > 1$, as explained below:

1. The regret bound for strongly convex functions scales inversely with λ.Thus, if $\lambda < 1/T$, the bound becomes at least $\Omega(T)$, which is vacuous. In this case, we cannot benefit from strong convexity and should treat these functions as general convex.[2]
2. From Definition 1, we know that λ-strongly convex functions with $\lambda > 1$ are also 1-strongly convex. Therefore, they can be handled as 1-strongly convex, and the resulting bound is optimal up to a constant (i.e., λ) factor.

Based on the interval $[1/T, 1]$, we set $\mathcal{P}_{str}$ to be an exponentially spaced grid with a ratio of 2:

$$\mathcal{P}_{str} = \left\{ \frac{1}{T}, \frac{2}{T}, \frac{2^2}{T}, \cdots, \frac{2^N}{T} \right\}, \ N = \lceil \log_2 T \rceil. \tag{25}$$

$\mathcal{P}_{str}$ can approximate λ well in the sense that for any $\lambda \in [1/T, 1]$, there must exist a $\widehat{\lambda} \in \mathcal{P}_{str}$ such that $\widehat{\lambda} \leq \lambda \leq 2\widehat{\lambda}$.

In the following, we denote by $R(A, \widehat{\lambda})$ the regret bound, predicted by theory, of expert $E(A, \widehat{\lambda})$ in Algorithm 1. Note that the expert $E(A, \widehat{\lambda})$ assumes the online functions are $\widehat{\lambda}$-strongly convex, which is true since $\widehat{\lambda} \leq \lambda$. Thus, the regret bound $R(A, \widehat{\lambda})$ is *valid*, and it is also *tight* because $\lambda \leq 2\widehat{\lambda}$. We have the following theoretical guarantee.

Theorem 6 *Under Assumptions 1 and 2, if the online functions are λ-strongly convex with $\lambda \in [1/T, 1]$, Algorithm 1 satisfies*

$$\begin{aligned} \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \leq & \min_{A \in \mathcal{A}_{str}} R(A, \widehat{\lambda}) + 2\Gamma G D \left(2 + \frac{1}{\sqrt{\ln |\mathcal{E}|}} \right) + \frac{\Gamma^2 G^2}{2\lambda \ln |\mathcal{E}|} \\ = & \min_{A \in \mathcal{A}_{str}} R(A, \widehat{\lambda}) + O\left(\frac{\log \log T}{\lambda} \right) \end{aligned}$$

1. Compared to our conference version (Zhang et al., 2022), we adopt a simpler step size from Chen et al. (2023, §4.3).
2. More precisely, when $O(GD\sqrt{T}) < O(G^2 \log T/\lambda)$, i.e., $\lambda < G \log T/(D\sqrt{T})$, we should treat λ-strongly convex functions as general convex ones. For simplicity and clarity of analysis, we choose $1/T$ as the threshold, which is more conservative since $1/T < G \log T/(D\sqrt{T})$ when T is sufficiently large, allowing for a wider grid range of λ.

where $\widehat{\lambda} \in \mathcal{P}_{str}$, $\widehat{\lambda} \leq \lambda \leq 2\widehat{\lambda}$, *and*

$$\Gamma = 3\ln|\mathcal{E}| + \ln\left(1 + \frac{|\mathcal{E}|}{2e}\big(1+\ln(T+1)\big)\right) \overset{(27)}{=} O(\log\log T). \tag{26}$$

Remark: To reveal the order of the upper bound, we assume the number of candidate algorithms is small, so $|\mathcal{A}_{str}|$, $|\mathcal{A}_{exp}|$ and $|\mathcal{A}_{con}|$ are all small constants. Thus,

$$|\mathcal{E}| = |\mathcal{A}_{str}| \cdot |\mathcal{P}_{str}| + |\mathcal{A}_{exp}| \cdot |\mathcal{P}_{exp}| + |\mathcal{A}_{con}| \overset{(25),(28)}{=} O(\log T) \tag{27}$$

which is used in (26). When both the domain and gradients are bounded, Theorem 6 shows that USC achieves *the best of all worlds* for strongly convex functions, up to an additive factor of $O(\log\log T)$.

Remark: The computational complexity of an expert per iteration is generally independent from T, but may depend on the dimensionality d (Duchi et al., 2010a). To simplify discussions, we hide the dependence on d, and assume the complexity is $O(1)$ per iteration. Since USC maintains $|\mathcal{E}| = O(\log T)$ experts, its computational complexity is $O(\log T)$ per iteration, which is the same as that of previous methods (van Erven and Koolen, 2016; Wang et al., 2019, 2020b).

To be more concrete, we use the small-loss bound and the gradient-variation bound for smooth functions to give an example. To this end, we need an additional assumption (Srebro et al., 2010).

Assumption 3 *All the online functions are nonnegative, and H-smooth over the entire space that contains $\mathcal{X}$.*

By using OEGD (Chiang et al., 2012) and S^2OGD (Wang et al., 2020b) as experts, we have the following corollary.

Corollary 7 *Under Assumptions 1, 2, and 3, if the online functions are λ-strongly convex with $\lambda \in [1/T, 1]$, we have*

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = O\left(\frac{1}{\lambda}\Big(\min(\log L_T^*, \log V_T) + \log\log T\Big)\right)$$

where L_T^* and V_T are defined in (5) and (6) respectively, provided OEGD, S^2OGD $\in \mathcal{A}_{str}$.

3.5 Exp-Concave Functions

We move to exp-concave functions, and use the following algorithms to build $\mathcal{A}_{exp}$:

- Online Newton step (ONS) (Hazan et al., 2007);
- ONS for exp-concave and smooth functions (Orabona et al., 2012);
- OEGD for exp-concave and smooth functions (Chiang et al., 2012).

Following the same arguments as in Section 3.4, we also assume the modulus of exponential concavity α lies in $[1/T, 1]$, and use the same geometric series to construct $\mathcal{P}_{exp}$ as

$$\mathcal{P}_{exp} = \left\{\frac{1}{T}, \frac{2}{T}, \frac{2^2}{T}, \cdots, \frac{2^N}{T}\right\}, \ N = \lceil \log_2 T \rceil. \tag{28}$$

Then, for any $\alpha \in [1/T, 1]$, there must exist an $\widehat{\alpha} \in \mathcal{P}_{exp}$ such that $\widehat{\alpha} \leq \alpha \leq 2\widehat{\alpha}$.

We denote by $R(A, \widehat{\alpha})$ the regret bound of expert $E(A, \widehat{\alpha})$ in Algorithm 1. Similarly, $R(A, \widehat{\alpha})$ is both valid and tight. We have the following guarantee for exp-concave functions, which is analogous to Theorem 6.

Theorem 8 *Under Assumptions 1 and 2, if the online functions are α-exp-concave with $\alpha \in [1/T, 1]$, Algorithm 1 satisfies*

$$\begin{aligned}\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) &\leq \min_{A\in\mathcal{A}_{exp}} R(A, \widehat{\alpha}) + 2\Gamma GD \left(2 + \frac{1}{\sqrt{\ln|\mathcal{E}|}}\right) + \frac{\Gamma^2}{2\beta \ln|\mathcal{E}|} \\ &= \min_{A\in\mathcal{A}_{exp}} R(A, \widehat{\alpha}) + O\left(\frac{\log\log T}{\alpha}\right)\end{aligned}$$

where $\widehat{\alpha} \in \mathcal{P}_{exp}$, $\widehat{\alpha} \leq \alpha \leq 2\widehat{\alpha}$, $\beta = \frac{1}{2}\min\{\frac{1}{4GD}, \alpha\}$, and Γ is defined in (26).

Remark: Similar to the case of strongly convex functions, USC inherits the regret bound of *any* expert designed for exp-concave functions, with a negligible double logarithmic factor. By using ONS (Orabona et al., 2012) and OEGD (Chiang et al., 2012) as experts, we obtain the best of the small-loss bound and the gradient-variation bound, up to a double logarithmic factor.

Corollary 9 *Under Assumptions 1, 2, and 3, if the online functions are α-exp-concave with $\alpha \in [1/T, 1]$, we have*

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = O\left(\frac{1}{\alpha}\Big(d\min(\log L_T^*, \log V_T) + \log\log T\Big)\right)$$

where L_T^* and V_T are defined in (5) and (6) respectively, provided ONS, OEGD $\in \mathcal{A}_{exp}$.

3.6 General Convex Functions

Finally, we study general convex functions, and in this case, we have various algorithms to construct $\mathcal{A}_{con}$, such as OGD (Zinkevich, 2003), ADAGRAD (Duchi et al., 2011), OEGD for convex and smooth functions (Chiang et al., 2012), RMSprop (Tieleman and Hinton, 2012), ADADELTA (Zeiler, 2012), Adam (Kingma and Ba, 2015), AO-FTRL (Mohri and Yang, 2016), and SOGD (Zhang et al., 2019a).

Let $R(A)$ be the regret bound of expert $E(A)$ in Algorithm 1. The theoretical guarantee of USC for general convex functions is stated below.

Theorem 10 *Under Assumptions 1 and 2, for any sequence of convex functions, Algorithm 1 satisfies*

$$\begin{aligned}\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) &\leq \min_{A\in\mathcal{A}_{con}} R(A) + 4\Gamma GD + \frac{\Gamma D}{\sqrt{\ln|\mathcal{E}|}} \sqrt{4G^2 + \sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_2^2} \\ &= \min_{A\in\mathcal{A}_{con}} R(A) + O\left(\sqrt{T\log\log T}\right)\end{aligned}$$

where Γ is defined in (26).

Remark: The above theorem is weaker than those for strongly convex functions and exp-concave functions. That is because we cannot eliminate the regret of the meta-algorithm, and the final regret is the sum of the expert-regret and the meta-regret. Nevertheless, Theorem 10 still implies a small-loss bound for smooth functions, when SOGD (Zhang et al., 2019a) is used as the expert.

Corollary 11 *Under Assumptions 1, 2, and 3, for any sequence of convex functions, we have*

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) = O\left(\sqrt{L_T^* \log\log T}\right)$$

where L_T^* is defined in (5), provided SOGD $\in \mathcal{A}_{con}$.

3.7 Extension to Unknown G

We note that USC requires the prior knowledge of the gradient norm bound G, which is not always readily available in practical applications and thus motivates us to investigate an extension of USC with unknown G. Generally speaking, thanks to the highly flexible framework of USC, we can conveniently substitute both the meta-algorithm and expert-algorithm with any suitable online methods that can adapt to the unknown G. In the following, we discuss how to replace both the meta-algorithm and expert-algorithm in USC to avoid prior knowledge of G.

First, for the meta-algorithm, we note that G is employed to normalize the linearized loss in (21), enabling the application of the meta-algorithm (i.e., Adapt-ML-Prod). When G is unknown, we can opt for more advanced methods which not only deliver second-order bounds but also adapt to the unknown loss range automatically, e.g., Squint+L (Mhammedi et al., 2019). Consequently, we can directly use the (unnormalized) linearized loss in (16), maintaining the same guarantee for meta-regret.

Second, in expert-algorithms, G is utilized to tune their own parameters. When G is unknown, we can select experts that do not need to know G in hindsight. For example, for strongly convex functions, we can use OGD with the step size $\eta_t = 1/(\lambda t)$ (Shalev-Shwartz et al., 2007) and AOGD (Bartlett et al., 2007). For exp-concave functions, we can choose the exponentially weighted online optimization (EWOO) algorithm (Hazan et al., 2007) and MetaGrad+L (Mhammedi et al., 2019). For general convex functions, we can use AOGD with $\lambda = 0$ (Bartlett et al., 2007). We note that in parameter-free online learning, some methods designed for unconstrained domains can also adapt to unknown G, such as scale-free online learning (Orabona and Pál, 2018) and FreeGrad (Mhammedi and Koolen, 2020). Therefore, these methods can also serve as expert-algorithms for general convex functions.

4. Online Composite Optimization

In this section, we extend our universal strategy to online composite optimization.

4.1 Preliminaries

First, we introduce the following standard assumptions in online composite optimization (Duchi and Singer, 2009; Duchi et al., 2010b).

Assumption 4 *The regularization function $r(\cdot)$ is convex over $\mathcal{X}$.*

Assumption 5 *The regularization function $r(\cdot)$ is non-negative and upper bounded by a known constant C, i.e., $\forall \mathbf{x} \in \mathcal{X}$, $0 \leq r(\mathbf{x}) \leq C$.*

We assume that in the composite loss function (3), $r(\cdot)$ is convex and known, but the function $f_t(\cdot)$ could be general convex, strongly convex, or exp-concave. In the literature, there have been plenty of studies for each specific setting (Duchi and Singer, 2009; Xiao, 2009; Duchi et al., 2010b; Yang et al., 2024c), but all of them require the prior knowledge about $f_t(\cdot)$. In this paper, we aim to design a universal algorithm for online composite optimization, and attain the optimal regret for each possible case. To this end, one may attempt to treat the sum of $f_t(\mathbf{x}) + r(\mathbf{x})$ as a new function, and pass it to USC. However, this approach introduces following issues. First, the sum of an exp-concave function and a convex function is not necessarily an exp-concave function (Yang et al., 2018), implying that USC cannot deliver the optimal regret when $f_t(\cdot)$ is exp-concave. Second, due to the additional component $r(\cdot)$, the summation function $f_t(\mathbf{x}) + r(\mathbf{x})$ may lose certain special properties of $f_t(\cdot)$, e.g., the smoothness which is essential for delivering many problem-dependent bounds. Third, ignoring the composite structure fails to harness the power of the regularizer, such as incorporating prior knowledge about the problem at hand. Therefore, additional modifications are necessary to handle composite loss functions.

4.2 Motivations

Our universal algorithm for online composite optimization (USC-Comp) is still based on the framework of learning with expert advice: building a set of experts for each type of functions and employing a meta-algorithm to aggregate their predictions. Similar to USC, the experts are running over the original composite functions so that the properties of the loss functions can be utilized to establish problem-dependent regret bounds. To handle the composite losses, we modify the meta-algorithm in the following way:

(i) we run the meta-algorithm over a *composite linearized* loss, which combines the linearized loss and the regularization function, i.e.,

$$l_t(\mathbf{x}) := \langle \nabla f_t(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t \rangle + r(\mathbf{x})$$

to measure the performance of experts;

(ii) instead of employing Adapt-ML-Prod as the meta-algorithm, we utilize a different algorithm, called Optimistic-Adapt-ML-Prod (Wei et al., 2016), which, with appropriate configurations, can deliver a second-order bound that exclusively depends on the time-varying functions $f_t(\cdot)$ so that the strong convexity and exponential concavity of $f_t(\cdot)$ can be utilized to control the meta-regret.

To make it clearer, we take the exponential concavity case, i.e., $f_t(\cdot)$ is exp-concave, as an example. In this case, the meta-algorithm should yield a second-order regret bound that solely depends on $f_t(\cdot)$. In other words, even though the regularization function $r(\cdot)$ is present in the meta-regret, it should not contribute to the upper bound. This is achievable due to the following fact:

> Prior to determining the weights for the experts, the meta-algorithm has already observed the part of the loss function that is related to the regularizer.

The above fact facilitates the possibility to set suitable configurations of Optimistic-Adapt-ML-Prod to eliminate the influence of $r(\cdot)$ on the meta-regret.

Specifically, let $\mathbf{x}_t$ and $\mathbf{x}_t^i$ be the output of the meta-algorithm and the expert E^i in the t-th round, respectively. Then, the regret in (4) can be decomposed as the sum of the meta-regret and the expert-regret:

$$\begin{aligned}&\sum_{t=1}^{T}\big[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\big]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\big[f_t(\mathbf{x})+r(\mathbf{x})\big]\\ =&\underbrace{\sum_{t=1}^{T}[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)]-\sum_{t=1}^{T}\big[f_t(\mathbf{x}_t^i)+r(\mathbf{x}_t^i)\big]}_{:=\text{meta-regret}}+\underbrace{\sum_{t=1}^{T}[f_t(\mathbf{x}_t^i)+r(\mathbf{x}_t^i)]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}[f_t(\mathbf{x})+r(\mathbf{x})]}_{:=\text{expert-regret}}.\end{aligned}$$

The expert-regret can be upper bounded by leveraging theoretical results of existing methods in online composite learning. Therefore, we only need to focus on the meta-regret.

To deal with the regularizer $r(\cdot)$, the meta-algorithm exploits the composite linearized loss

$$l_t^i := l_t(\mathbf{x}_t^i)=\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t^i-\mathbf{x}_t\rangle+r(\mathbf{x}_t^i) \tag{29}$$

to measure the performance of E^i, where $\mathbf{x}_t=\sum_{i=1}^{|\mathcal{E}|}p_t^i\mathbf{x}_t^i$ denotes the decision made by the meta-algorithm and p_t^i denotes the weight assigned to the expert E^i. With the above composite linearized loss, the meta-regret is upper bounded by

$$\begin{aligned}\text{meta-regret}\leq&\sum_{t=1}^{T}\big[f_t(\mathbf{x}_t)-f_t(\mathbf{x}_t^i)\big]+\sum_{t=1}^{T}\left[\sum_{i=1}^{|\mathcal{E}|}p_t^i r(\mathbf{x}_t^i)-r(\mathbf{x}_t^i)\right]\\ \overset{(15)}{\leq}&\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle-\frac{\beta}{2}\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle^2+\sum_{t=1}^{T}\left[\sum_{i=1}^{|\mathcal{E}|}p_t^i r(\mathbf{x}_t^i)-r(\mathbf{x}_t^i)\right]\\ \overset{(29)}{=}&\sum_{t=1}^{T}\left(\sum_{i=1}^{|\mathcal{E}|}p_t^i l_t^i-l_t^i\right)-\frac{\beta}{2}\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle^2=\sum_{t=1}^{T}\hat{R}_t^i-\frac{\beta}{2}\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle^2,\end{aligned} \tag{30}$$

where the first step is due to Jensen's inequality, i.e., $r(\mathbf{x}_t)=r(\sum_{i=1}^{|\mathcal{E}|}p_t^i\mathbf{x}_t^i)\leq\sum_{i=1}^{|\mathcal{E}|}p_t^i r(\mathbf{x}_t^i)$, and for brevity, we define

$$\hat{R}_t^i=\sum_{i=1}^{|\mathcal{E}|}p_t^i l_t^i-l_t^i=\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle+\sum_{i=1}^{|\mathcal{E}|}p_t^i r(\mathbf{x}_t^i)-r(\mathbf{x}_t^i). \tag{31}$$

Note that to manage the meta-regret, it is essential to constrain $\sum_{t=1}^{T}\hat{R}_t^i$ by a second-order bound which only depends on $f_t(\cdot)$, so that the negative term in (30) can be exploited.

To this end, we employ Optimistic-Adapt-ML-Prod as the meta-algorithm, which utilizes an optimistic estimation (also called optimism) for the upcoming loss function to update p_t^i

Algorithm 2 A Universal Strategy for Online Composite Optimization (USC-Comp)

Input: $\mathcal{A}_{str}$, $\mathcal{A}_{exp}$, $\mathcal{A}_{con}$, $\mathcal{P}_{str}$ and $\mathcal{P}_{exp}$
Initialize $\mathcal{E}$ by following Steps 3-10 in Algorithm 1
Observe the regularization term $r(\cdot)$
for $t = 1$ **to** T **do**
    Receive $\mathbf{x}_t^i$ from each expert E^i in $\mathcal{E}$
    Compute the optimism m_t^i of each expert E^i by (35)
    Calculate the weight p_t^i of each expert E^i by (36)
    Output the weighted average $\mathbf{x}_t = \sum_{i=1}^{|\mathcal{E}|} p_t^i \mathbf{x}_t^i$
    Observe the loss function $f_t(\cdot)$
    Send the required information of $f_t(\cdot)$ and $r(\cdot)$ to each expert in $\mathcal{E}$
end for

for each expert E^i. In our algorithm, we set the estimation as

$$\hat{m}_t^i = \sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i) - r(\mathbf{x}_t^i). \tag{32}$$

We emphasize that although $\hat{m}_t^i$ relies on p_t^i which in turn depends on $\hat{m}_t^i$, we can still compute $\hat{m}_t^i$ in each round t before updating p_t^i. The reasons lie in that (i) the regularizer term $r(\cdot)$ is fixed and has been already revealed to the meta-algorithm; (ii) the decision $\mathbf{x}_t^i$ can be sent to the meta-algorithm before computing $\hat{m}_t^i$; (iii) $\sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i)$ can be approximated efficiently with little sacrifice in meta-regret bounds, which will be elaborated in Section 4.3.

Then, according to Optimistic-Adapt-ML-Prod (Wei et al., 2016, Theorem 3.4), we have

$$\begin{aligned}\sum_{t=1}^T \hat{R}_t^i =& O\left(\sqrt{\sum_{t=1}^T \left(\hat{R}_t^i - \hat{m}_t^i\right)^2}\right) \overset{(31),(32)}{=} O\left(\sqrt{\sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i\rangle^2}\right)\\ =& O\left(\frac{1}{\beta}\right) + \frac{\beta}{2}\sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i\rangle^2.\end{aligned} \tag{33}$$

Substituting (33) into (30), we obtain the following meta-regret bound:

$$\sum_{t=1}^T [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \sum_{t=1}^T \left[f_t(\mathbf{x}_t^i) + r(\mathbf{x}_t^i)\right] = O\left(\frac{1}{\beta}\right),$$

which is small and thus does not affect the overall regret bound of the algorithm.

4.3 Our Universal Strategy

We summarize our universal strategy for online composite optimization (USC-Comp) in Algorithm 2. Our new strategy is designed for three cases, in which the regularization term $r(\cdot)$ is convex and $f_t(\cdot)$ could be strongly convex, exp-concave, or general convex.

To handle the above cases simultaneously, we build the expert set $\mathcal{E}$ based on three candidate algorithm sets (i.e., $\mathcal{A}_{str}$, $\mathcal{A}_{exp}$ and $\mathcal{A}_{con}$), each of which deals with one case, and two modulus value sets (i.e., $\mathcal{P}_{str}$ and $\mathcal{P}_{exp}$), each of which contains possible values of the modulus of strong convexity or exponential concavity. The process of constructing $\mathcal{E}$ follows UCS in Algorithm 1. First, we initialize $\mathcal{E}$ with experts designed for the strongly convex function $f_t(\cdot)$. Specifically, for each algorithm $A \in \mathcal{A}_{str}$ and each $\lambda \in \mathcal{P}_{str}$, we create an expert $E(A, \lambda)$ by running the algorithm A with the parameter λ, and add it to the expert set $\mathcal{E}$. Then, for the exp-concave function $f_t(\cdot)$ and general convex function $f_t(\cdot)$, we repeat similar steps, adding an expert $E(A, \alpha)$ for each algorithm $A \in \mathcal{A}_{exp}$ and each $\alpha \in \mathcal{P}_{exp}$, and an expert $E(A)$ for each algorithm $A \in \mathcal{A}_{con}$.

Next, we employ Optimistic-Adapt-ML-Prod (Wei et al., 2016) to track the best expert on the fly. In the beginning, we observe the regularization term $r(\cdot)$ (Step 3). Then, at the t-th round, after receiving the predictions from all experts (Step 5), we are able to compute the estimation m_t^i and the weight p_t^i of each expert E^i (Steps 6-7), and submit the weighted average $\mathbf{x}_t = \sum_{i=1}^{|\mathcal{E}|} p_t^i \mathbf{x}_t^i$ (Step 8). Finally, we observe the loss function $f_t(\cdot)$ and $r(\cdot)$, and send the required information to each expert in $\mathcal{E}$ (Steps 9-10).

The above process is similar to USC, and the distinction lies in the update of weight for each expert E^i. Specifically, the meta-algorithm employs the composite linearized loss in (29) to measure the performance of each expert. Because Optimistic-Adapt-ML-Prod requires $|\hat{R}_t^i - \hat{m}_t^i| \leq 2$, we normalize $\hat{R}_t^i$ and $\hat{m}_t^i$ in the following way

$$R_t^i = \frac{1}{GD}\hat{R}_t^i \overset{(31)}{=} \frac{1}{GD}\left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle + \sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i) - r(\mathbf{x}_t^i)\right) \tag{34}$$

$$m_t^i = \frac{1}{GD}\hat{m}_t^i \overset{(32)}{=} \frac{1}{GD}\left(\sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i) - r(\mathbf{x}_t^i)\right). \tag{35}$$

From (34) and (35), we can verify that

$$\left|R_t^i - m_t^i\right| = \frac{1}{GD}\left|\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle\right| \leq \frac{1}{GD}\|\nabla f_t(\mathbf{x}_t)\|_2 \|\mathbf{x}_t - \mathbf{x}_t^i\|_2 \overset{(12),(13)}{\leq} 2.$$

According to Optimistic-Adapt-ML-Prod (Wei et al., 2016), the weight of expert E^i is computed by

$$p_t^i = \frac{\eta_{t-1}^i \widetilde{w}_{t-1}^i}{\sum_{j=1}^{|\mathcal{E}|} \eta_{t-1}^j \widetilde{w}_{t-1}^j} \tag{36}$$

where $\widetilde{w}_{t-1}^i = w_{t-1}^i \exp\left(\eta_{t-1}^i m_t^i\right)$, and η_{t-1}^i and w_{t-1}^i are determined by

$$\eta_{t-1}^i = \min\left\{\frac{1}{4}, \sqrt{\frac{\ln|\mathcal{E}|}{1 + \sum_{s=1}^{t-1}(R_s^i - m_s^i)^2}}\right\},\ t \geq 1, \tag{37}$$

$$w_{t-1}^i = \left(w_{t-2}^i \exp\left(\eta_{t-2}^i R_{t-1}^i - \left(\eta_{t-2}^i (R_{t-1}^i - m_{t-1}^i)\right)^2\right)\right)^{\frac{\eta_{t-1}^i}{\eta_{t-2}^i}},\ t \geq 2.$$

In the beginning, we initialize $w_0^i = 1/|\mathcal{E}|$.

Now, let's explain why the quantity $\sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i)$ of m_t^i in (35) is computable before updating p_t^i. The key idea is to demonstrate that $\sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i)$ can be regarded as the fixed point of a continuous function. Given the value of $\gamma = \sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i)$, we have

$$m_t^i \overset{(35)}{=} \frac{1}{GD}\left(\sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i) - r(\mathbf{x}_t^i)\right) = \frac{1}{GD}\left(\gamma - r(\mathbf{x}_t^i)\right).$$

Correspondingly, $\widetilde{w}_{t-1}^i$ and p_t^i can be viewed as the function of γ, i.e.,

$$\widetilde{w}_{t-1}^i(\gamma) = w_{t-1}^i \exp\left(\frac{\eta_{t-1}^i}{GD}\left(\gamma - r(\mathbf{x}_t^i)\right)\right) \text{ and } p_t^i(\gamma) = \frac{\eta_{t-1}^i \widetilde{w}_{t-1}^i(\gamma)}{\sum_{j=1}^{|\mathcal{E}|} \eta_{t-1}^j \widetilde{w}_{t-1}^j(\gamma)}.$$

Thus, the calculation of γ reduces to finding the fixed point of $F(\gamma) = \sum_{i=1}^{|\mathcal{E}|} p_t^i(\gamma) r(\mathbf{x}_t^i)$ satisfying $F(\gamma) = \gamma$. It can be verified that $F(\gamma)$ is continuous and bounded in $[0, C]$, which implies that there must exist some fixed point $\gamma \in [0, C]$. To find the point γ, we can deploy the binary-search strategy, which only suffers $1/T$ error in $\log T$ iterations and therefore, does not affect the regret bound (Wei et al., 2016).

To formally state the relationship between the optimism and composite losses, we introduce the following theorem.

Theorem 12 *Let Optimistic-Adapt-ML-Prod be the meta-algorithm that evaluates expert performance using the composite linearized loss in* (29)*, and is configured with the optimism in* (35)*. Then, for the λ-strongly convex function $f_t(\cdot)$, the meta-regret is bounded by*

$$\sum_{t=1}^{T} [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \sum_{t=1}^{T} \left[f_t(\mathbf{x}_t^i) + r(\mathbf{x}_t^i)\right] = O\left(\frac{G^2}{\lambda}\right).$$

For the β-exp-concave function $f_t(\cdot)$, the meta-regret is bounded by

$$\sum_{t=1}^{T} [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \sum_{t=1}^{T} \left[f_t(\mathbf{x}_t^i) + r(\mathbf{x}_t^i)\right] = O\left(\frac{1}{\beta}\right).$$

For the general convex function $f_t(\cdot)$, the meta-regret is bounded by

$$\sum_{t=1}^{T} [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \sum_{t=1}^{T} \left[f_t(\mathbf{x}_t^i) + r(\mathbf{x}_t^i)\right] = O(GD\sqrt{T}).$$

In the following, we provide the algorithmic details of our universal strategy.

4.4 Strongly Convex Functions

For the strongly convex $f_t(\cdot)$ and convex $r(\cdot)$, we employ algorithms below to build $\mathcal{A}_{str}$:

- FOBOS for strongly convex functions (SC-FOBOS) (Duchi and Singer, 2009);
- COMID for strongly convex functions (SC-COMID) (Duchi et al., 2010b);
- OCMD for strongly convex functions (SC-OptCMD) (Scroccaro et al., 2023).

Note that Duchi et al. (2010b) only provide the worst-case regret bounds of SC-COMID for strongly convex time-varying functions. In Appendix B, we further utilize the smoothness of $f_t(\cdot)$ and establish a *new* (pseudo) small-loss bound of $O(\log \tilde{L}_T)$ for smooth and strongly convex $f_t(\cdot)$, where

$$\tilde{L}_T = \sum_{t=1}^{T} f_t(\tilde{\mathbf{x}}_t) \tag{38}$$

is the cumulative loss of SC-COMID involving $f_t(\cdot)$.

Remark: We emphasize that $O(\log \tilde{L}_T)$ is not the standard small-loss bound, since $\tilde{L}_T$ depends on not only the time-varying function $f_t(\cdot)$ but also the prediction $\tilde{\mathbf{x}}_t$ chosen by the algorithm itself. Due to the additional regularization term $r(\cdot)$ in (4), it remains unclear how to establish a standard small-loss bound that solely depends on $f_t(\cdot)$.

Following the same configuration as in Section 3.4, we assume the modulus of strong convexity $\lambda \in [1/T, 1]$, and construct $\mathcal{P}_{str}$ in the same way as (25), according to which there must exist a $\widehat{\lambda} \in \mathcal{P}_{str}$ such that $\widehat{\lambda} \leq \lambda \leq 2\widehat{\lambda}$ for any $\lambda \in [1/T, 1]$. We denote by $R(A, \widehat{\lambda})$ the regret bound of expert $E(A, \widehat{\lambda})$ in Algorithm 2. Since $\widehat{\lambda} \leq \lambda \leq 2\widehat{\lambda}$, $R(A, \widehat{\lambda})$ is both valid and tight. We have the following theorem for strongly convex functions.

Theorem 13 *Under Assumptions 1, 2, 4 and 5, if the time-varying function $f_t(\cdot)$ is λ-strongly convex with $\lambda \in [1/T, 1]$, Algorithm 2 satisfies*

$$\begin{aligned}
&\sum_{t=1}^{T} [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} [f_t(\mathbf{x}) + r(\mathbf{x})] \\
\leq &\min_{A \in \mathcal{A}_{str}} R(A, \widehat{\lambda}) + GD\left(\Xi + \frac{\Psi}{\sqrt{\ln |\mathcal{E}|}}\right) + \frac{\Psi^2 G^2}{2\lambda \ln |\mathcal{E}|} = \min_{A \in \mathcal{A}_{str}} R(A, \widehat{\lambda}) + O\left(\frac{\log \log T}{\lambda}\right)
\end{aligned}$$

where $\widehat{\lambda} \in \mathcal{P}_{str}$, $\widehat{\lambda} \leq \lambda \leq 2\widehat{\lambda}$, and

$$\Psi = \ln |\mathcal{E}| + \ln\left(1 + \frac{|\mathcal{E}|}{e}\big(1 + \ln(T+1)\big)\right) \overset{(27)}{=} O(\log \log T), \tag{39}$$

$$\Xi = \frac{1}{4}\Psi + 2\sqrt{\ln |\mathcal{E}|} + 16 \ln |\mathcal{E}| \overset{(27)}{=} O(\log \log T). \tag{40}$$

Remark: The above theorem indicates that Algorithm 2 preserves the regret bound of *any* expert designed for strongly convex functions, up to a negligible double logarithmic factor. Specifically, when the time-varying function $f_t(\cdot)$ is smooth, we can achieve the following problem-dependent bound, by using SC-COMID (Duchi et al., 2010b) and SC-OptCMD (Scroccaro et al., 2023) as experts.

Corollary 14 *Under Assumptions 1, 2, 3, 4 and 5, if the time-varying function $f_t(\cdot)$ is λ-strongly convex with $\lambda \in [1/T, 1]$, and SC-COMID, SC-OptCMD $\in \mathcal{A}_{str}$, we have*

$$\sum_{t=1}^{T} [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} [f_t(\mathbf{x}) + r(\mathbf{x})] = O\left(\frac{1}{\lambda}\Big(\min\{\log \tilde{L}_T, \log V_T\} + \log \log T\Big)\right)$$

where $\tilde{L}_T$ and V_T are defined in (38) *and* (6) *respectively.*

4.5 Exp-Concave Functions

For the exp-concave $f_t(\cdot)$ and convex $r(\cdot)$, we employ algorithms below to build $\mathcal{A}_{\text{exp}}$:

- ProxONS (Yang et al., 2024c);
- OCMD for exp-concave functions (Exp-OptCMD) (Scroccaro et al., 2023).

Note that Scroccaro et al. (2023) have investigated OCMD for general convex and strongly convex time-varying functions, under the smoothness condition of $f_t(\cdot)$. In Appendix C, we further extend OCMD to exp-concave time-varying functions, and establish *new* problem-dependent bounds of $O(\log V_T)$ and $O(\log \tilde{L}_T)$.

Similar to Section 3.5, we assume the exponential concavity modulus α lies in $[1/T, 1]$, and use the same geometric series to construct $\mathcal{P}_{exp}$ as (28). Then, for $\forall \alpha \in [1/T, 1]$, there must exist an $\widehat{\alpha} \in \mathcal{P}_{exp}$ such that $\widehat{\alpha} \le \alpha \le 2\widehat{\alpha}$. We denote by $R(A, \widehat{\alpha})$ the regret bound of expert $E(A, \widehat{\alpha})$ in Algorithm 2, and have the following guarantee for exp-concave functions.

Theorem 15 *Under Assumptions 1, 2, 4 and 5, if the time-varying function $f_t(\cdot)$ is α-exp-concave with $\alpha \in [1/T, 1]$, Algorithm 2 satisfies*

$$\begin{aligned}&\sum_{t=1}^{T} [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} [f_t(\mathbf{x}) + r(\mathbf{x})] \\ \le &\min_{A \in \mathcal{A}_{exp}} R(A, \widehat{\alpha}) + GD\left(\Xi + \frac{\Psi}{\sqrt{\ln|\mathcal{E}|}}\right) + \frac{\Psi^2}{2\beta \ln|\mathcal{E}|} = \min_{A \in \mathcal{A}_{exp}} R(A, \widehat{\alpha}) + O\left(\frac{\log\log T}{\alpha}\right)\end{aligned}$$

where $\widehat{\alpha} \in \mathcal{P}_{exp}$, $\widehat{\alpha} \le \alpha \le 2\widehat{\alpha}$, $\beta = \frac{1}{2}\min\{\frac{1}{4GD}, \alpha\}$, and Ψ and Ξ are defined in (39) *and* (40)*, respectively.*

Remark: Similar to the strongly convex case, Algorithm 2 also inherits the regret bound of any expert designed for exp-concave functions. By incorporating Exp-OptCMD into $\mathcal{A}_{\text{exp}}$, Algorithm 2 ensures the following problem-dependent bound.

Corollary 16 *Under Assumptions 1, 2, 3, 4 and 5, if the time-varying function $f_t(\cdot)$ is α-exp-concave with $\alpha \in [1/T, 1]$, we have*

$$\sum_{t=1}^{T} [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} [f_t(\mathbf{x}) + r(\mathbf{x})] = O\left(\frac{1}{\alpha}\Big(d \min\{\log \tilde{L}_T, \log V_T\} + \log\log T\Big)\right)$$

where $\tilde{L}_T$ and V_T are defined in (38) *and* (6) *respectively, provided Exp-OptCMD $\in \mathcal{A}_{str}$.*

4.6 General Convex Functions

For the general convex $f_t(\cdot)$ and convex $r(\cdot)$, we employ algorithms below to build $\mathcal{A}_{con}$:

- FOBOS (Duchi and Singer, 2009);
- COMID (Duchi et al., 2010b);
- CAO-FTRL (Mohri and Yang, 2016);
- Composite-objective AO-FTRL (Joulani et al., 2020);
- OCMD (Scroccaro et al., 2023).

During the analysis, we develop a *new* (pseudo) small-loss bound of $O(\sqrt{\tilde{L}_T})$ for COMID when the time-varying function $f_t(\cdot)$ is smooth and general convex. More details can be found in Appendix B.

Let $R(A)$ be the regret bound of expert $E(A)$ in $\mathcal{A}_{con}$. The theoretical guarantee of Algorithm 2 for general convex functions is stated below.

Theorem 17 *Under Assumptions 1, 2, 4 and 5, for any sequence of convex time-varying functions, Algorithm 2 satisfies*

$$\sum_{t=1}^{T}[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}[f_t(\mathbf{x})+r(\mathbf{x})]$$

$$\leq \min_{A\in\mathcal{A}_{con}} R(A)+\Xi GD+\frac{\Psi D}{\sqrt{\ln|\mathcal{E}|}}\sqrt{G^2+\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2}=\min_{A\in\mathcal{A}_{con}} R(A)+O\left(\sqrt{T\log\log T}\right)$$

where Ψ *and* Ξ *is defined in* (39) *and* (40)*, respectively.*

Remark: The above bound remains weaker than those in Theorems 13 and 15, due to the uneliminated part $O(\sqrt{\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2})$. Under the smoothness of $f_t(\cdot)$, by employing COMID as an expert in $\mathcal{A}_{con}$, Theorem 17 implies the following problem-dependent bound.

Corollary 18 *Under Assumptions 1, 2, 3, 4 and 5, for any sequence of convex time-varying functions, we have*

$$\sum_{t=1}^{T}[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}[f_t(\mathbf{x})+r(\mathbf{x})]=O\left(\sqrt{\tilde{L}_T^{meta}\log\log T}+\sqrt{\tilde{L}_T}\right)$$

where $\tilde{L}_T^{meta}$ *denotes the cumulative loss in* (38) *of the meta-algorithm, and* $\tilde{L}_T$ *denotes that of COMID, provided COMID* $\in\mathcal{A}_{con}$*.*

Currently, our universal strategy for online composite optimization requires prior knowledge of G, which may not be available in practical applications. To address this limitation, a natural approach is to employ methods that can adapt to the unknown G as the meta-algorithm and expert-algorithms, as discussed in Section 3.7. In this direction, there are several interesting research avenues, such as extending Optimistic-Adapt-ML-Prod (Wei et al., 2016) to adapt to G, which we leave for future work.

5. Analysis

In this section, we present the analysis of all theorems.

5.1 Proof of Theorem 5

The proof of Theorem 5 can be derived from the subsequent analysis for Theorems 6, 8 and 10. Specifically, the meta-regret bounds can be obtained from (43) for strongly convex functions, from (49) for exp-concave functions, and from (52) for general convex functions.

5.2 Proof of Theorem 6

We first analyze the meta-regret of our strategy. According to the theoretical guarantee of Adapt-ML-Prod (Gaillard et al., 2014, Corollary 4), we have

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

for all expert $E^i \in \mathcal{E}$, where Γ is given in (26). Combining with the definitions of ℓ_t^i and ℓ_t in (21) and (22), we arrive at

$$\begin{aligned}\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle \leq & 4\Gamma G D + \frac{\Gamma}{\sqrt{\ln |\mathcal{E}|}} \sqrt{4G^2D^2 + \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle^2} \\ \leq & 2\Gamma G D \left(2 + \frac{1}{\sqrt{\ln |\mathcal{E}|}}\right) + \frac{\Gamma}{\sqrt{\ln |\mathcal{E}|}} \sqrt{\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle^2}\end{aligned} \tag{41}$$

where the last step follows from the basic inequality $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$, $\forall a, b \geq 0$.

To utilize the property of strong convexity in (14), we proceed in the following way:

$$\begin{aligned}&\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle \\ \leq & 2\Gamma G D \left(2 + \frac{1}{\sqrt{\ln |\mathcal{E}|}}\right) + \frac{\Gamma^2 G^2}{2\lambda \ln |\mathcal{E}|} + \frac{\lambda}{2G^2} \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle^2 \\ \leq & 2\Gamma G D \left(2 + \frac{1}{\sqrt{\ln |\mathcal{E}|}}\right) + \frac{\Gamma^2 G^2}{2\lambda \ln |\mathcal{E}|} + \frac{\lambda}{2G^2} \sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_2^2 \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2 \\ \overset{(12)}{\leq} & 2\Gamma G D \left(2 + \frac{1}{\sqrt{\ln |\mathcal{E}|}}\right) + \frac{\Gamma^2 G^2}{2\lambda \ln |\mathcal{E}|} + \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2\end{aligned} \tag{42}$$

where the first step follows from the basic inequality $2\sqrt{ab} \leq a + b$, $\forall a, b \geq 0$. According to Definition 1, the meta-regret in terms of $f_t(\cdot)$ is given by

$$\begin{aligned}\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^i) \overset{(14)}{\leq} & \sum_{t=1}^{T} \left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle - \frac{\lambda}{2} \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2 \right) \\ \overset{(42)}{\leq} & 2\Gamma G D \left(2 + \frac{1}{\sqrt{\ln |\mathcal{E}|}}\right) + \frac{\Gamma^2 G^2}{2\lambda \ln |\mathcal{E}|}.\end{aligned} \tag{43}$$

Next, we study the expert-regret. Let E^i be the expert $E(A, \widehat{\lambda})$ where $A \in \mathcal{A}_{str}$, $\widehat{\lambda} \in \mathcal{P}_{str}$, and $\widehat{\lambda} \leq \lambda \leq 2\widehat{\lambda}$. Since λ-strongly convex functions are also $\widehat{\lambda}$-strongly convex, expert $E(A, \widehat{\lambda})$ makes a right assumption, and the following inequality is true

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t^i) - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \leq R(A, \widehat{\lambda}). \tag{44}$$

Combining (43) and (44), we have

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \leq R(A,\widehat{\lambda}) + 2\Gamma GD\left(2+\frac{1}{\sqrt{\ln|\mathcal{E}|}}\right) + \frac{\Gamma^2 G^2}{2\lambda \ln|\mathcal{E}|}. \tag{45}$$

We complete the proof by noticing that (45) holds for any $A \in \mathcal{A}_{str}$.

5.3 Proof of Corollary 7

From the theoretical guarantee of OEGD for strongly convex and smooth functions in Theorem 21, we have

$$R(\text{OEGD},\widehat{\lambda}) = O\left(\frac{\log V_T}{\widehat{\lambda}}\right) \overset{\lambda\leq 2\widehat{\lambda}}{=} O\left(\frac{\log V_T}{\lambda}\right). \tag{46}$$

Similarly, from the regret bound of S^2OGD (Wang et al., 2020b, Theorem 1), we have

$$R(\text{S}^2\text{OGD},\widehat{\lambda}) = O\left(\frac{\log L_T^*}{\widehat{\lambda}}\right) \overset{\lambda\leq 2\widehat{\lambda}}{=} O\left(\frac{\log L_T^*}{\lambda}\right). \tag{47}$$

We obtain the corollary by substituting (46) and (47) into Theorem 6.

5.4 Proof of Theorem 8

The analysis is similar to that of Theorem 6. To make use of the property of exponential concavity in (15), we change (42) as follows:

$$\begin{aligned}&\sum_{t=1}^{T}\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i\rangle \\ \leq& 2\Gamma GD\left(2+\frac{1}{\sqrt{\ln|\mathcal{E}|}}\right) + \frac{\Gamma^2}{2\beta\ln|\mathcal{E}|} + \frac{\beta}{2}\sum_{t=1}^{T}\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i\rangle^2.\end{aligned} \tag{48}$$

According to Lemma 3, the meta-regret in terms of $f_t(\cdot)$ can be bounded by

$$\begin{aligned}\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^i) \overset{(15)}{\leq}& \sum_{t=1}^{T}\left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i\rangle - \frac{\beta}{2}\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i\rangle^2\right) \\ \overset{(48)}{\leq}& 2\Gamma GD\left(2+\frac{1}{\sqrt{\ln|\mathcal{E}|}}\right) + \frac{\Gamma^2}{2\beta\ln|\mathcal{E}|}.\end{aligned} \tag{49}$$

The rest of the proof is identical to that of Theorem 6.

5.5 Proof of Corollary 9

From the theoretical guarantee of ONS for exp-concave and smooth functions (Orabona et al., 2012, Theorem 1), we have

$$R(\text{ONS},\widehat{\alpha}) = O\left(\frac{d\log L_T^*}{\widehat{\alpha}}\right) \overset{\alpha\leq 2\widehat{\alpha}}{=} O\left(\frac{d\log L_T^*}{\alpha}\right). \tag{50}$$

Similarly, from the regret bound of OEGD (Chiang et al., 2012, Theorem 15), we have

$$R(\text{OEGD}, \widehat{\alpha}) = O\left(\frac{d \log V_T}{\widehat{\alpha}}\right) \overset{\alpha \leq 2\widehat{\alpha}}{=} O\left(\frac{d \log V_T}{\alpha}\right). \tag{51}$$

We obtain the corollary by substituting (50) and (51) into Theorem 8.

5.6 Proof of Theorem 10

From the first-order condition of convex functions, we bound the meta-regret by

$$\begin{aligned}
\sum_{t=1}^{T} f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^i) \leq & \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle \\
\overset{(41)}{\leq} & 4\Gamma G D + \frac{\Gamma}{\sqrt{\ln |\mathcal{E}|}} \sqrt{4G^2D^2 + \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle^2} \\
\leq & 4\Gamma G D + \frac{\Gamma}{\sqrt{\ln |\mathcal{E}|}} \sqrt{4G^2D^2 + \sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_2^2 \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2} \\
\overset{(13)}{\leq} & 4\Gamma G D + \frac{\Gamma D}{\sqrt{\ln |\mathcal{E}|}} \sqrt{4G^2 + \sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_2^2}.
\end{aligned} \tag{52}$$

We complete the proof by combining the above inequality with that of the expert-regret:

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t^i) - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \leq R(A), \ \forall A \in \mathcal{A}_{con}.$$

5.7 Proof of Corollary 11

We need the self-bounding property of smooth functions (Srebro et al., 2010, Lemma 3.1).

Lemma 19 *Let the function $f : \mathcal{X} \mapsto \mathbb{R}$ be nonnegative, and H-smooth over the entire space that contains $\mathcal{X}$. Then, we have*

$$\|\nabla f(\mathbf{x})\| \leq \sqrt{4Hf(\mathbf{x})}, \ \forall \mathbf{x} \in \mathcal{X}. \tag{53}$$

Combining Lemma 19 and Theorem 10, we have

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \leq \min_{A \in \mathcal{A}_{con}} R(A) + 4\Gamma G D + \frac{\Gamma D}{\sqrt{\ln |\mathcal{E}|}} \sqrt{4G^2 + 4H \sum_{t=1}^{T} f_t(\mathbf{x}_t)}. \tag{54}$$

From the theoretical guarantee of SOGD for convex and smooth functions (Zhang et al., 2019a, Theorem 2), we have

$$R(\text{SOGD}) = 8HD^2 + D\sqrt{2\delta + 8HL_T^*} \tag{55}$$

where $\delta > 0$ can be any small constant. Substituting (55) into (54), we obtain

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - L_T^* \leq 8HD^2 + D\sqrt{2\delta + 8HL_T^*} + 4\Gamma GD + \frac{\Gamma D}{\sqrt{\ln|\mathcal{E}|}}\sqrt{4G^2 + 4H\sum_{t=1}^{T} f_t(\mathbf{x}_t)}. \tag{56}$$

To simplify the above inequality, we use the following lemma (Shalev-Shwartz, 2007, Lemma 19).

Lemma 20 *Let $x, b, c \in \mathbb{R}_+$. Then,*

$$x - c \leq b\sqrt{x} \Rightarrow x - c \leq b^2 + b\sqrt{c}.$$

From (56), we have

$$\begin{aligned}
&\left(\frac{G^2}{H} + \sum_{t=1}^{T} f_t(\mathbf{x}_t)\right) - \left(L_T^* + D\sqrt{2\delta + 8HL_T^*} + 4\Gamma GD + 8HD^2 + \frac{G^2}{H}\right) \\
\leq & \frac{\Gamma D\sqrt{4H}}{\sqrt{\ln|\mathcal{E}|}}\sqrt{\frac{G^2}{H} + \sum_{t=1}^{T} f_t(\mathbf{x}_t)}.
\end{aligned}$$

Lemma 20 implies

$$\begin{aligned}
&\left(\frac{G^2}{H} + \sum_{t=1}^{T} f_t(\mathbf{x}_t)\right) - \left(L_T^* + D\sqrt{2\delta + 8HL_T^*} + 4\Gamma GD + 8HD^2 + \frac{G^2}{H}\right) \\
\leq & \frac{4\Gamma^2 D^2 H}{\ln|\mathcal{E}|} + \frac{\Gamma D\sqrt{4H}}{\sqrt{\ln|\mathcal{E}|}}\sqrt{L_T^* + D\sqrt{2\delta + 8HL_T^*} + 4\Gamma GD + 8HD^2 + \frac{G^2}{H}}.
\end{aligned}$$

Thus,

$$\begin{aligned}
&\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T} f_t(\mathbf{x}) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - L_T^* \\
\leq & \frac{\Gamma D\sqrt{4H}}{\sqrt{\ln|\mathcal{E}|}}\sqrt{L_T^* + D\sqrt{2\delta + 8HL_T^*} + 4\Gamma GD + 8HD^2 + \frac{G^2}{H}} + D\sqrt{2\delta + 8HL_T^*} \\
& + 4\Gamma GD + 8HD^2 + \frac{4\Gamma^2 D^2 H}{\ln|\mathcal{E}|} \\
= & O\left(\sqrt{L_T^* \log\log T}\right).
\end{aligned}$$

5.8 Proof of Theorem 12

The proof of Theorem 12 can be derived from the subsequent analysis for Theorems 13, 15 and 17. Specifically, the meta-regret bounds can be obtained from (61) for strongly convex functions, from (66) for exp-concave functions, and from (68) for general convex functions.

5.9 Proof of Theorem 13

Similar to the analysis of Theorem 6, we first bound the meta-regret of Algorithm 2. According to the theoretical guarantee of Optimistic-Adapt-ML-Prod (Wei et al., 2016, Proof of Theorem 3.4), we have

$$\sum_{t=1}^{T} R_t^i \leq \frac{\Psi}{\sqrt{\ln |\mathcal{E}|}} \sqrt{1 + \sum_{t=1}^{T} (R_t^i - m_t^i)^2} + \Xi \tag{57}$$

for all expert $E^i \in \mathcal{E}$, where Ψ and Ξ are given in (39) and (40), respectively. Then, we substitute (34) and (35) into (57), and obtain

$$\begin{aligned}
&\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle + \sum_{t=1}^{T} \sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i) - \sum_{t=1}^{T} r(\mathbf{x}_t^i) \\
\leq & \Xi GD + \frac{\Psi}{\sqrt{\ln |\mathcal{E}|}} \sqrt{G^2 D^2 + \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle^2} \\
\leq & GD \left(\Xi + \frac{\Psi}{\sqrt{\ln |\mathcal{E}|}} \right) + \frac{\Psi}{\sqrt{\ln |\mathcal{E}|}} \sqrt{\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle^2}
\end{aligned} \tag{58}$$

where the last step follows from the basic inequality $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$, $\forall a, b \geq 0$.

To utilize the property of strong convexity in (14), we simplify (58) into the following form:

$$\begin{aligned}
&\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle + \sum_{t=1}^{T} \sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i) - \sum_{t=1}^{T} r(\mathbf{x}_t^i) \\
\leq & GD \left(\Xi + \frac{\Psi}{\sqrt{\ln |\mathcal{E}|}} \right) + \frac{\Psi^2 G^2}{2\lambda \ln |\mathcal{E}|} + \frac{\lambda}{2G^2} \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle^2 \\
\leq & GD \left(\Xi + \frac{\Psi}{\sqrt{\ln |\mathcal{E}|}} \right) + \frac{\Psi^2 G^2}{2\lambda \ln |\mathcal{E}|} + \frac{\lambda}{2G^2} \sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t)\|_2^2 \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2 \\
\overset{(12)}{\leq} & GD \left(\Xi + \frac{\Psi}{\sqrt{\ln |\mathcal{E}|}} \right) + \frac{\Psi^2 G^2}{2\lambda \ln |\mathcal{E}|} + \frac{\lambda}{2} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2
\end{aligned} \tag{59}$$

where the first step follows from the basic inequality $2\sqrt{ab} \leq a + b$, $\forall a, b \geq 0$. According to (14) and Jensen's inequality, i.e., $r(\mathbf{x}_t) = r(\sum_{i=1}^{|\mathcal{E}|} p_t^i \mathbf{x}_t^i) \leq \sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i)$, the meta-regret is upper bounded by

$$\begin{aligned}
&\sum_{t=1}^{T} [f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \sum_{t=1}^{T} \left[f_t(\mathbf{x}_t^i) + r(\mathbf{x}_t^i) \right] \\
\overset{(14)}{\leq} & \sum_{t=1}^{T} \left(\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_t^i \rangle - \frac{\lambda}{2} \|\mathbf{x}_t - \mathbf{x}_t^i\|_2^2 \right) + \sum_{t=1}^{T} \sum_{i=1}^{|\mathcal{E}|} p_t^i r(\mathbf{x}_t^i) - \sum_{t=1}^{T} r(\mathbf{x}_t^i).
\end{aligned} \tag{60}$$

Substituting (59) into (60), we arrive at

$$\sum_{t=1}^{T}[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)]-\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t^i)+r(\mathbf{x}_t^i)\right]\leq GD\left(\Xi+\frac{\Psi}{\sqrt{\ln|\mathcal{E}|}}\right)+\frac{\Psi^2G^2}{2\lambda\ln|\mathcal{E}|}. \tag{61}$$

We complete the proof by combining (61) with the following expert-regret

$$\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t^i)+r(\mathbf{x}_t^i)\right]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}[f_t(\mathbf{x})+r(\mathbf{x})]\leq R(A,\widehat{\lambda}),\ \forall A\in\mathcal{A}_{str},\widehat{\lambda}\in\mathcal{P}_{str}. \tag{62}$$

5.10 Proof of Corollary 14

From the regret bound of SC-COMID for strongly convex and smooth functions in Theorem 25, we have

$$R(\text{SC-COMID},\widehat{\lambda})=O\left(\frac{\log\tilde{L}_T}{\widehat{\lambda}}\right)\overset{\lambda\leq 2\widehat{\lambda}}{=}O\left(\frac{\log\tilde{L}_T}{\lambda}\right). \tag{63}$$

Similarly, from the regret bound of SC-OptCMD (Scroccaro et al., 2023, Theorem 2.9), we have

$$R(\text{SC-OptCMD},\widehat{\lambda})=O\left(\frac{\log V_T}{\widehat{\lambda}}\right)\overset{\lambda\leq 2\widehat{\lambda}}{=}O\left(\frac{\log V_T}{\lambda}\right). \tag{64}$$

We complete the proof by substituting (63) and (64) into Theorem 13.

5.11 Proof of Theorem 15

Similar to (59), we can obtain the following bound for the exp-concave case

$$\begin{aligned}&\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle+\sum_{t=1}^{T}\sum_{i=1}^{|\mathcal{E}|}p_t^ir(\mathbf{x}_t^i)-\sum_{t=1}^{T}r(\mathbf{x}_t^i)\\ \leq& GD\left(\Xi+\frac{\Psi}{\sqrt{\ln|\mathcal{E}|}}\right)+\frac{\Psi^2}{2\beta\ln|\mathcal{E}|}+\frac{\beta}{2}\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle^2.\end{aligned} \tag{65}$$

Then, substituting (65) into (30), we obtain

$$\sum_{t=1}^{T}[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)]-\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t^i)+r(\mathbf{x}_t^i)\right]\leq GD\left(\Xi+\frac{\Psi}{\sqrt{\ln|\mathcal{E}|}}\right)+\frac{\Psi^2}{2\beta\ln|\mathcal{E}|}. \tag{66}$$

The rest of the proof is identical to that of Theorem 13.

5.12 Proof of Corollary 16

From the regret bounds of Exp-OptCMD in Theorems 27 and 28, we have

$$R(\text{Exp-OptCMD},\widehat{\alpha})=O\left(\frac{d}{\widehat{\alpha}}\min\{\log V_T,\log\tilde{L}_T\}\right)\overset{\alpha\leq 2\widehat{\alpha}}{=}O\left(\frac{d}{\alpha}\min\{\log V_T,\log\tilde{L}_T\}\right). \tag{67}$$

We complete the proof by substituting (67) into Theorem 15.

5.13 Proof of Theorem 17

We follow the similar analysis in (58) to handle the general convex case:

$$\begin{aligned}
&\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle+\sum_{t=1}^{T}\sum_{i=1}^{|\mathcal{E}|}p_t^i r(\mathbf{x}_t^i)-\sum_{t=1}^{T}r(\mathbf{x}_t^i)\\
\leq&\Xi GD+\frac{\Psi}{\sqrt{\ln|\mathcal{E}|}}\sqrt{G^2D^2+\sum_{t=1}^{T}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}_t^i\rangle^2}\\
\overset{(13)}{\leq}&\Xi GD+\frac{\Psi D}{\sqrt{\ln|\mathcal{E}|}}\sqrt{G^2+\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2}.
\end{aligned}\tag{68}$$

Then, we complete the proof by combining (68) with the following expert regret bound

$$\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t^i)+r(\mathbf{x}_t^i)\right]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\left[f_t(\mathbf{x})+r(\mathbf{x})\right]\leq R(A),\ \forall A\in\mathcal{A}_{con}.\tag{69}$$

5.14 Proof of Corollary 18

Combining Lemma 19 and Theorem 17, we have

$$\begin{aligned}
&\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\right]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\left[f_t(\mathbf{x})+r(\mathbf{x})\right]\\
\leq&\min_{A\in\mathcal{A}_{con}}R(A)+\Xi GD+\frac{\Psi D}{\sqrt{\ln|\mathcal{E}|}}\sqrt{G^2+4H\sum_{t=1}^{T}f_t(\mathbf{x}_t)}.
\end{aligned}\tag{70}$$

From the theoretical guarantee of COMID for convex and smooth function $f_t(\cdot)$ in Theorem 24, we have

$$R(\text{COMID})\leq\sqrt{8HD^2}\sqrt{\frac{\delta}{4H}+\sum_{t=1}^{T}f_t(\tilde{\mathbf{x}}_t)}+r(\tilde{\mathbf{x}}_1),\tag{71}$$

where $\tilde{\mathbf{x}}_t$ denotes the decisions made by COMID. We complete the proof by substituting (71) into (70):

$$\begin{aligned}
&\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\right]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\left[f_t(\mathbf{x})+r(\mathbf{x})\right]\\
\leq&\Xi GD+r(\tilde{\mathbf{x}}_1)+\frac{\Psi D}{\sqrt{\ln|\mathcal{E}|}}\sqrt{G^2+4H\tilde{L}_T^{meta}}+\sqrt{8HD^2}\sqrt{\frac{\delta}{4H}+\tilde{L}_T}
\end{aligned}\tag{72}$$

where $\tilde{L}_T^{meta}=\sum_{t=1}^{T}f_t(\mathbf{x}_t)$ denotes the cumulative loss suffered by the meta-algorithm involving $f_t(\cdot)$, and $\tilde{L}_T=\sum_{t=1}^{T}f_t(\tilde{\mathbf{x}}_t)$ denotes that incurred by COMID.

6. Conclusion and Future Work

In this paper, we propose a simple strategy for universal OCO, which can handle three types of loss functions simultaneously in both the standard and composite settings. The fundamental idea is to construct a set of experts by running existing algorithms with different configurations for each type of online functions, and combine them by a meta-algorithm that enjoys a second-order bound with excess losses. The key novelty is to let experts process original functions, and let the meta-algorithm use (partially) linearized losses. In the standard setting, thanks to the second-order bound of the meta-algorithm, our method attains *the best of all worlds* for strongly convex functions and exp-concave functions, up to a double logarithmic factor. For general convex functions, it maintains the minimax optimality and can achieve a small-loss bound. In the composite setting, we employ a different meta-algorithm which is able to achieve a second-order bound that solely depends on the time-varying functions. In this way, our method can handle multiple types of composite loss functions simultaneously.

There are several directions for future research. First, our strategy is designed for the purpose of static regret minimization, but static regret itself may not be suitable for changing environments (Zhang, 2020; Cesa-Bianchi and Orabona, 2021; Wang et al., 2024b). To address this limitation, recent developments in online learning have proposed new performance metrics including adaptive regret (Hazan and Seshadhri, 2007; Daniely et al., 2015) and dynamic regret (Zinkevich, 2003; Zhang et al., 2018). In the future, we will investigate how to modify our strategy to support those stronger notions of regret. Second, our strategy needs to fix the value of the time horizon T, which is then used to construct $\mathcal{P}_{str}$ and $\mathcal{P}_{exp}$. We will study how to design an *anytime* universal algorithm that does not depend on T. One possible method is to use the doubling trick with our universal strategy to estimate T on the fly, but this approach inevitably introduces an additional $\log T$ factor in the regret bound for strongly convex and exp-concave functions. Therefore, developing an optimal and anytime universal algorithm remains an important open problem.

Acknowledgment

This work was partially supported by NSFC (62361146852, U23A20382), and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

Appendix A. Online Extra-Gradient Descent (OEGD)

In this section, we extend the OEGD of Chiang et al. (2012) to strongly convex functions.

A.1 Algorithm

There are two sequences of solutions $\{\mathbf{x}_t\}_{t=1}^T$ and $\{\mathbf{u}_t\}_{t=1}^T$, where $\mathbf{u}_t$ is an auxiliary solution used to exploit the smoothness of the loss function.

Based on the property of strong convexity in (14), we set

$$\mathcal{R}_t(\mathbf{x}) = \frac{1}{2\eta_t}\|\mathbf{x}\|_2^2$$

in Algorithm 1 of Chiang et al. (2012), where

$$\eta_t = \frac{2}{\lambda t}, \tag{73}$$

and obtain the following updating rules:

$$\begin{aligned} \mathbf{u}_{t+1} =& \Pi_{\mathcal{X}}\left[\mathbf{u}_t - \eta_t \nabla f_t(\mathbf{x}_t)\right], \\ \mathbf{x}_{t+1} =& \Pi_{\mathcal{X}}\left[\mathbf{u}_{t+1} - \eta_{t+1} \nabla f_t(\mathbf{x}_t)\right], \end{aligned} \tag{74}$$

where $\Pi_{\mathcal{X}}[\cdot]$ denotes the projection onto the nearest point in $\mathcal{X}$, and $\mathbf{u}_1$ and $\mathbf{x}_1$ are set to be any points in $\mathcal{X}$.

Theorem 21 *Under Assumptions 1, 2, and 3, if the online functions are λ-strongly convex, we have*

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \le \frac{16G^2}{\lambda} \ln(2V_T+1) + \frac{3\lambda D^2}{8} + \frac{16G^2+4}{\lambda} + \frac{16G^2}{\lambda} \ln\left(\frac{256G^2H^2}{\lambda^2} + 1\right)$$

where V_T is defined in (6).

A.2 Proof of Theorem 21

We introduce a useful lemma for optimistic mirror descent (Nemirovski, 2004, Lemma 3.1).

Lemma 22 *Let $\mathcal{Z}$ be a convex compact set in Euclidean space $\mathcal{E}$ with inner product $\langle\cdot,\cdot\rangle$, let $\|\cdot\|$ be a norm on $\mathcal{E}$ and $\|\cdot\|_*$ be its dual norm, and let $\omega(\mathbf{z}) : \mathcal{Z} \mapsto \mathbb{R}$ be a α-strongly convex function with respect to $\|\cdot\|$, and $\mathcal{B}_\omega(\mathbf{z},\mathbf{w})$ be the Bregman distance associated with ω. Let $\mathcal{U}$ be a convex and closed subset of $\mathcal{Z}$, and let $\mathbf{z}_- \in \mathcal{Z}$, let $\boldsymbol{\xi}, \boldsymbol{\eta} \in \mathcal{E}$, and let $\gamma > 0$. Consider the points*

$$\begin{aligned} \mathbf{w} &= \operatorname*{argmin}_{\mathbf{y}\in\mathcal{U}} \left[\langle\gamma\boldsymbol{\xi}, \mathbf{y}\rangle + \mathcal{B}_\omega(\mathbf{y}, \mathbf{z}_-)\right], \\ \mathbf{z}_+ &= \operatorname*{argmin}_{\mathbf{y}\in\mathcal{U}} \left[\langle\gamma\boldsymbol{\eta}, \mathbf{y}\rangle + \mathcal{B}_\omega(\mathbf{y}, \mathbf{z}_-)\right]. \end{aligned}$$

Then for all $\mathbf{z} \in \mathcal{U}$, one has

$$\langle \mathbf{w} - \mathbf{z}, \gamma\boldsymbol{\eta}\rangle \le \mathcal{B}_\omega(\mathbf{z},\mathbf{z}_-) - \mathcal{B}_\omega(\mathbf{z},\mathbf{z}_+) + \frac{\gamma^2}{\alpha}\|\boldsymbol{\eta} - \boldsymbol{\xi}\|_*^2 - \frac{\alpha}{2}\left[\|\mathbf{w} - \mathbf{z}_-\|_2^2 + \|\mathbf{z}_+ - \mathbf{w}\|_2^2\right]$$

and

$$\|\mathbf{w} - \mathbf{z}_+\| \le \alpha^{-1}\gamma\|\boldsymbol{\xi} - \boldsymbol{\eta}\|_*.$$

Applying Lemma 22 to the updating rules in (74), we have

$$\begin{aligned} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}\rangle \le& \frac{1}{2\eta_t}\|\mathbf{x} - \mathbf{u}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{x} - \mathbf{u}_{t+1}\|_2^2 \\ &+ \eta_t\|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 - \frac{1}{2\eta_t}\left[\|\mathbf{x}_t - \mathbf{u}_t\|_2^2 + \|\mathbf{u}_{t+1} - \mathbf{x}_t\|_2^2\right] \end{aligned} \tag{75}$$

for any $\mathbf{x} \in \mathcal{X}$ and

$$\|\mathbf{u}_{t+1} - \mathbf{x}_t\|_2 \leq \eta_t \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2. \tag{76}$$

Combining (75) with Definition 1, we have

$$\begin{aligned} f_t(\mathbf{x}_t) - f_t(\mathbf{x}) \overset{(14),(75)}{\leq} & \frac{1}{2\eta_t}\|\mathbf{x} - \mathbf{u}_t\|_2^2 - \frac{1}{2\eta_t}\|\mathbf{x} - \mathbf{u}_{t+1}\|_2^2 - \frac{\lambda}{2}\|\mathbf{x} - \mathbf{x}_t\|_2^2 \\ & + \eta_t \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 - \frac{1}{2\eta_t}\left[\|\mathbf{x}_t - \mathbf{u}_t\|_2^2 + \|\mathbf{u}_{t+1} - \mathbf{x}_t\|_2^2\right]. \end{aligned}$$

Summing the above inequality over $t = 1, \ldots, T$, we have

$$\begin{aligned} & \sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}) \\ \leq & \underbrace{\sum_{t=1}^T \frac{\|\mathbf{x} - \mathbf{u}_t\|_2^2 - \|\mathbf{x} - \mathbf{u}_{t+1}\|_2^2}{2\eta_t} - \frac{\lambda}{2}\|\mathbf{x} - \mathbf{x}_t\|_2^2}_{\texttt{term (a)}} \\ & + \underbrace{\sum_{t=1}^T \eta_t \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2}_{\texttt{term (b)}} - \underbrace{\sum_{t=1}^T \frac{\|\mathbf{x}_t - \mathbf{u}_t\|_2^2 + \|\mathbf{u}_{t+1} - \mathbf{x}_t\|_2^2}{2\eta_t}}_{\texttt{term (c)}} \end{aligned} \tag{77}$$

for any $\mathbf{x} \in \mathcal{X}$. In the following, we upper bound the three terms above respectively. For `term (a)`, we have

$$\begin{aligned} \texttt{term (a)} = & \frac{1}{2\eta_1}\|\mathbf{x} - \mathbf{u}_1\|_2^2 + \frac{1}{2}\sum_{t=2}^T \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right)\|\mathbf{x} - \mathbf{u}_t\|_2^2 - \frac{1}{2\eta_T}\|\mathbf{x} - \mathbf{u}_{T+1}\|_2^2 \\ & - \frac{\lambda}{2}\sum_{t=1}^T \|\mathbf{x} - \mathbf{x}_t\|_2^2 \\ \overset{(13)}{\leq} & \frac{1}{2\eta_1}D^2 + \frac{1}{2}\sum_{t=2}^T \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}}\right)\|\mathbf{x} - \mathbf{u}_t\|_2^2 - \frac{\lambda}{2}\sum_{t=1}^T \|\mathbf{x} - \mathbf{x}_t\|_2^2 \end{aligned}$$

Then, substituting the choice of η in (73) delivers

$$\begin{aligned} \texttt{term (a)} \overset{(73)}{\leq} & \frac{\lambda D^2}{4} + \frac{\lambda}{4}\sum_{t=2}^T \|\mathbf{x} - \mathbf{u}_t\|_2^2 - \frac{\lambda}{2}\sum_{t=1}^T \|\mathbf{x} - \mathbf{x}_t\|_2^2 \\ \leq & \frac{\lambda D^2}{4} + \frac{\lambda}{4}\sum_{t=1}^{T-1}\left(\|\mathbf{x} - \mathbf{u}_{t+1}\|_2^2 - 2\|\mathbf{x} - \mathbf{x}_t\|_2^2\right) \leq \frac{\lambda}{4}D^2 + \frac{\lambda}{2}\sum_{t=1}^{T-1}\|\mathbf{u}_{t+1} - \mathbf{x}_t\|_2^2 \\ \overset{(76)}{\leq} & \frac{\lambda D^2}{4} + \frac{\lambda}{2}\sum_{t=1}^{T-1}\eta_t^2 \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 \\ \leq & \frac{\lambda D^2}{4} + \sum_{t=1}^{T-1}\eta_t \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 \leq \frac{\lambda D^2}{4} + \texttt{term (b)} \end{aligned} \tag{78}$$

where in the penultimate line, we use the fact that $\eta_t \leq \eta_1 = 2/\lambda$.

From (78), we observe that the upper bound of `term` (a) depends on `term` (b). So, we proceed to bound `term` (b). To this end, we define

$$\alpha = \left\lceil \sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 \right\rceil.$$

Then, we have

$$\begin{aligned}
\texttt{term (b)} &= \sum_{t=1}^{T} \frac{2}{\lambda t} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 \\
&= \sum_{t=1}^{\alpha} \frac{2}{\lambda t} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 + \sum_{t=\alpha+1}^{T} \frac{2}{\lambda t} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 \\
&\overset{(12)}{\leq} \frac{8G^2}{\lambda} \sum_{t=1}^{\alpha} \frac{1}{t} + \frac{2}{\lambda(\alpha+1)} \sum_{t=\alpha+1}^{T} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 \\
&\leq \frac{8G^2}{\lambda} \left(1 + \int_{t=1}^{\alpha} \frac{1}{t} dt\right) + \frac{2}{\lambda} \leq \frac{8G^2}{\lambda} (\ln \alpha + 1) + \frac{2}{\lambda} \\
&\leq \frac{8G^2}{\lambda} \ln \left(\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 + 1 \right) + \frac{8G^2+2}{\lambda}.
\end{aligned} \tag{79}$$

From Lemma 12 of Chiang et al. (2012), we have

$$\sum_{t=1}^{T} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 \leq 2V_T + 2H^2 \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2. \tag{80}$$

Substituting (80) into (79), we have

$$\begin{aligned}
\texttt{term (b)} &\overset{(79),(80)}{\leq} \frac{8G^2}{\lambda} \ln \left(2V_T + 2H^2 \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2 + 1 \right) + \frac{8G^2+2}{\lambda} \\
&\leq \frac{8G^2}{\lambda} \ln(2V_T + 1) + \frac{8G^2}{\lambda} \left(2H^2 \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2 + 1 \right) + \frac{8G^2+2}{\lambda}
\end{aligned} \tag{81}$$

where the last step follows from the inequality below

$$\ln(1+u+v) \leq \ln(1+u) + \ln(1+v), \ \forall u, v \geq 0. \tag{82}$$

For `term` (c), based on the proof of Lemma 21 of Chiang et al. (2012), we have

$$\begin{aligned}
\texttt{term (c)} &= \sum_{t=1}^{T} \frac{\|\mathbf{x}_t - \mathbf{u}_t\|_2^2}{2\eta_t} + \sum_{t=2}^{T+1} \frac{\|\mathbf{x}_{t-1} - \mathbf{u}_t\|_2^2}{2\eta_{t-1}} \geq \sum_{t=2}^{T} \frac{\|\mathbf{x}_t - \mathbf{u}_t\|_2^2}{2\eta_{t-1}} + \sum_{t=2}^{T} \frac{\|\mathbf{x}_{t-1} - \mathbf{u}_t\|_2^2}{2\eta_{t-1}} \\
&\geq \sum_{t=2}^{T} \frac{\|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2}{4\eta_{t-1}} \overset{(73)}{\geq} \frac{\lambda}{8} \sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2.
\end{aligned} \tag{83}$$

Substituting (78), (81) and (83) into (77), we get

$$\begin{aligned}\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}) \leq & \frac{\lambda D^2}{4} + \frac{16G^2+4}{\lambda} + \frac{16G^2}{\lambda}\ln(2V_T+1) \\ & + \frac{16G^2}{\lambda}\left(2H^2\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2+1\right) - \frac{\lambda}{8}\sum_{t=2}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2 \\ \overset{(13)}{\leq} & \frac{3\lambda D^2}{8} + \frac{16G^2+4}{\lambda} + \frac{16G^2}{\lambda}\ln(2V_T+1) \\ & + \frac{16G^2}{\lambda}\left(2H^2\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2+1\right) - \frac{\lambda}{8}\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2.\end{aligned} \tag{84}$$

To simplify the above inequality, we make use of the following inequality (Chen et al., 2023, Lemma 7).

Lemma 23 *Let $A \geq 0$, $a \geq 0$, $b \geq 0$ and $c > 0$, we have*

$$a\ln(bA+1) - cA \leq a\ln\left(\frac{ab}{c}+1\right).$$

From Lemma 23, we have

$$\frac{16G^2}{\lambda}\left(2H^2\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2+1\right) - \frac{\lambda}{8}\sum_{t=1}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2 \leq \frac{16G^2}{\lambda}\ln\left(\frac{256G^2H^2}{\lambda^2}+1\right). \tag{85}$$

Combining (84) with (85), we complete the proof.

Appendix B. Composite Objective Mirror Descent (COMID)

In this section, we first revisit the COMID algorithm of Duchi et al. (2010b), and then establish (pseudo) small-loss bounds for general convex $f_t(\cdot)$ and strongly convex $f_t(\cdot)$.

B.1 Algorithm

The updating rule of COMID (Duchi et al., 2010b) is shown below

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{X}} \left\{\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}\rangle + \mathcal{B}_{\mathcal{R}_t}(\mathbf{x},\mathbf{x}_t) + r(\mathbf{x})\right\}, \tag{86}$$

where $\mathcal{R}_t$ is a strongly convex function and $\mathcal{B}_{\mathcal{R}_t}$ is the Bregman distance associated with $\mathcal{R}_t$. In our setting, we set

$$\mathcal{R}_t(\mathbf{x}) = \frac{1}{2\eta_t}\|\mathbf{x}\|_2^2.$$

Then, the original updating rule (86) of COMID becomes:

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{X}} \left\{\langle \nabla f_t(\mathbf{x}_t), \mathbf{x}\rangle + \frac{1}{2\eta_t}\|\mathbf{x}-\mathbf{x}_t\|_2^2 + r(\mathbf{x})\right\}. \tag{87}$$

For the general convex function $f_t(\cdot)$, we set the learning rate as below

$$\eta_t = \frac{\alpha}{\sqrt{\delta + \sum_{s=1}^{t} \|\nabla f_s(\mathbf{x}_s)\|_2^2}}, \tag{88}$$

where the parameter $\delta > 0$ is introduced to avoid being divided by 0, and $\alpha > 0$ is used to fine-tune the upper bound. Under the smoothness condition in Assumption 3, COMID ensures the following theoretical guarantee.

Theorem 24 *Under Assumptions 1, 2, 3, 4 and 5, if the time-varying function $f_t(\cdot)$ is general convex, we have*

$$\sum_{t=1}^{T} \left[f_t(\mathbf{x}_t) + r(\mathbf{x}_t)\right] - \sum_{t=1}^{T} \left[f_t(\mathbf{x}) + r(\mathbf{x})\right] \leq \sqrt{8HD^2}\sqrt{\frac{\delta}{4H} + \tilde{L}_T} + C = O\left(\sqrt{\tilde{L}_T}\right),$$

where $\tilde{L}_T$ is defined in (38)*.*

For the strongly convex function $f_t(\cdot)$, we modify the learning rate as:

$$\eta_t = \frac{2}{\lambda t}, \tag{89}$$

and obtain the following regret bound.

Theorem 25 *Under Assumptions 1, 2, 3, 4 and 5, if the time-varying function $f_t(\cdot)$ is λ-strongly convex, we have*

$$\begin{aligned}\sum_{t=1}^{T} \left[f_t(\mathbf{x}_t) + r(\mathbf{x}_t)\right] - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} \left[f_t(\mathbf{x}) + r(\mathbf{x})\right] \leq & \frac{\lambda D^2}{4} + C + \frac{1+G^2}{\lambda} + \frac{G^2}{2\lambda} \ln\left(4H\tilde{L}_T + 1\right) \\ = & O\left(\frac{1}{\lambda}\log \tilde{L}_T\right),\end{aligned}$$

where $\tilde{L}_T$ is defined in (38)*.*

B.2 Proof of Theorem 24

Let $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T}[f_t(\mathbf{x}) + r(\mathbf{x})]$. From the analysis of Duchi et al. (2010b, Theorem 2), we have

$$\begin{aligned}&\sum_{t=1}^{T} \left[f_t(\mathbf{x}_t) + r(\mathbf{x}_t)\right] - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} \left[f_t(\mathbf{x}) + r(\mathbf{x})\right] \\ \leq & \sum_{t=1}^{T} \left\{\frac{1}{2\eta_t}\left[\|\mathbf{x}_t - \mathbf{x}^*\|_2^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^*\|_2^2\right] + \frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2\right\} + r(\mathbf{x}_1).\end{aligned}$$

Under Assumptions 2 and 5, the above result can be simplified as

$$\begin{aligned}
&\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\right]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\left[f_t(\mathbf{x})+r(\mathbf{x})\right]\\
\leq&\frac{1}{2\eta_1}\|\mathbf{x}_1-\mathbf{x}^*\|_2^2+\sum_{t=2}^{T}\left(\frac{1}{2\eta_t}-\frac{1}{2\eta_{t-1}}\right)\|\mathbf{x}_t-\mathbf{x}^*\|_2^2+\sum_{t=1}^{T}\frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2+C\\
\leq&\frac{D^2}{2\eta_1}+D^2\sum_{t=2}^{T}\left(\frac{1}{2\eta_t}-\frac{1}{2\eta_{t-1}}\right)+\sum_{t=1}^{T}\frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2+C\\
=&\frac{D^2}{2\eta_T}+\sum_{t=1}^{T}\frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2+C.
\end{aligned}\tag{90}$$

To upper bound $\sum_{t=1}^{T}\frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2$ in (90), we make use of the following lemma (Auer et al., 2002, Lemma 3.5).

Lemma 26 *Let $l_1,\ldots,l_T$ and δ be non-negative real numbers. Then*

$$\sum_{t=1}^{T}\frac{l_t}{\sqrt{\delta+\sum_{s=1}^{t}l_s}}\leq 2\left(\sqrt{\delta+\sum_{t=1}^{T}l_t}-\sqrt{\delta}\right)$$

where $0/\sqrt{0}=0$.

Applying Lemma 26, we have

$$\sum_{t=1}^{T}\frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2\overset{(88)}{=}\frac{\alpha}{2}\sum_{t=1}^{T}\frac{\|\nabla f_t(\mathbf{x}_t)\|_2^2}{\sqrt{\delta+\sum_{s=1}^{t}\|\nabla f_s(\mathbf{x}_s)\|_2^2}}\leq\alpha\sqrt{\delta+\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2}.\tag{91}$$

Substituting (91) into (90), we have

$$\begin{aligned}
\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\right]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\left[f_t(\mathbf{x})+r(\mathbf{x})\right]\leq&\left(\frac{D^2}{2\alpha}+\alpha\right)\sqrt{\delta+\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2}+C\\
=&\sqrt{2D^2}\sqrt{\delta+\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2}+C,
\end{aligned}\tag{92}$$

where $\alpha=D/\sqrt{2}$.

We finish the proof by making use of Lemma 19:

$$\begin{aligned}
&\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\right]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\left[f_t(\mathbf{x})+r(\mathbf{x})\right]\\
\overset{(53),(92)}{\leq}&\sqrt{2D^2}\sqrt{\delta+4H\sum_{t=1}^{T}f_t(\mathbf{x}_t)}+C=\sqrt{8HD^2}\sqrt{\frac{\delta}{4H}+\sum_{t=1}^{T}f_t(\mathbf{x}_t)}+C.
\end{aligned}\tag{93}$$

B.3 Proof of Theorem 25

From the analysis of Duchi et al. (2010b, Theorem 7), we have

$$\begin{aligned}&\sum_{t=1}^{T}\big[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\big]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\big[f_t(\mathbf{x})+r(\mathbf{x})\big]\\ \leq&\sum_{t=1}^{T}\left\{\frac{1}{2\eta_t}\left[\|\mathbf{x}_t-\mathbf{x}^*\|_2^2-\|\mathbf{x}_{t+1}-\mathbf{x}^*\|_2^2\right]+\frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2-\frac{\lambda}{2}\|\mathbf{x}_t-\mathbf{x}^*\|_2^2\right\}+r(\mathbf{x}_1),\end{aligned}$$

where $\mathbf{x}^*=\operatorname{argmin}_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}[f_t(\mathbf{x})+r(\mathbf{x})]$. Then, we make use of Assumptions 2 and 5 to simplify the above inequality

$$\begin{aligned}&\sum_{t=1}^{T}\big[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\big]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\big[f_t(\mathbf{x})+r(\mathbf{x})\big]\\ \leq&\frac{D^2}{2\eta_1}+\sum_{t=2}^{T}\left(\frac{1}{2\eta_t}-\frac{1}{2\eta_{t-1}}-\frac{\lambda}{2}\right)\|\mathbf{x}_t-\mathbf{x}^*\|_2^2+\sum_{t=1}^{T}\frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2+C.\end{aligned} \tag{94}$$

According to (89), we have

$$\frac{1}{\eta_t}-\frac{1}{\eta_{t-1}}-\lambda=\frac{\lambda}{2}-\lambda\leq 0. \tag{95}$$

Substituting (95) into (94), we get

$$\begin{aligned}\sum_{t=1}^{T}\big[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\big]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\big[f_t(\mathbf{x})+r(\mathbf{x})\big]\leq&\frac{D^2}{2\eta_1}+C+\sum_{t=1}^{T}\frac{\eta_t}{2}\|\nabla f_t(\mathbf{x}_t)\|_2^2\\ =&\frac{\lambda D^2}{4}+C+\sum_{t=1}^{T}\frac{1}{\lambda t}\|\nabla f_t(\mathbf{x}_t)\|_2^2.\end{aligned} \tag{96}$$

Then, we proceed to bound $\sum_{t=1}^{T}\frac{1}{\lambda t}\|\nabla f_t(\mathbf{x}_t)\|_2^2$. To this end, we define

$$\alpha=\left\lceil\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2\right\rceil.$$

Next, we have

$$\begin{aligned}\sum_{t=1}^{T}\frac{1}{\lambda t}\|\nabla f_t(\mathbf{x}_t)\|_2^2=&\sum_{t=1}^{\alpha}\frac{1}{\lambda t}\|\nabla f_t(\mathbf{x}_t)\|_2^2+\sum_{t=\alpha+1}^{T}\frac{1}{\lambda t}\|\nabla f_t(\mathbf{x}_t)\|_2^2\\ \overset{(12)}{\leq}&\frac{G^2}{\lambda}\sum_{t=1}^{\alpha}\frac{1}{t}+\frac{1}{\lambda(\alpha+1)}\sum_{t=\alpha+1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2\\ \leq&\frac{G^2}{\lambda}\left(1+\int_{t=1}^{\alpha}\frac{1}{t}dt\right)+\frac{1}{\lambda}\leq\frac{G^2}{\lambda}(\ln\alpha+1)+\frac{1}{\lambda}\\ \leq&\frac{G^2}{\lambda}\ln\left(\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2+1\right)+\frac{1+G^2}{\lambda}.\end{aligned} \tag{97}$$

Substituting (97) into (96), we obtain

$$\begin{aligned}&\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\right]-\min_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}\left[f_t(\mathbf{x})+r(\mathbf{x})\right]\\ \leq&\frac{\lambda D^2}{4}+C+\frac{1+G^2}{\lambda}+\frac{G^2}{2\lambda}\ln\left(\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2+1\right).\end{aligned} \tag{98}$$

Combining (98) and Lemma 19 completes the proof.

Appendix C. Optimistic Composite Mirror Descent (OCMD)

In this section, we revisit the OCMD algorithm of Scroccaro et al. (2023), and then extend OCMD to the exp-concave case, i.e., $f_t(\cdot)$ is exp-concave.

C.1 Algorithm

During the online process, OCMD maintains two sequences of solutions $\{\mathbf{x}_t\}_{t=1}^T$ and $\{\mathbf{u}_t\}_{t=1}^T$, where $\mathbf{u}_t$ is an auxiliary solution used to exploit the smoothness of the loss function. The updating rules are shown below:

$$\begin{aligned}\mathbf{u}_{t+1}=&\operatorname*{argmin}_{\mathbf{x}\in\mathcal{X}}\left\{\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}\rangle+r(\mathbf{x})+\mathcal{B}_{\mathcal{R}_t}(\mathbf{x},\mathbf{u}_t)\right\},\\ \mathbf{x}_{t+1}=&\operatorname*{argmin}_{\mathbf{x}\in\mathcal{X}}\left\{\langle M_{t+1},\mathbf{x}\rangle+r(\mathbf{x})+\mathcal{B}_{\mathcal{R}_{t+1}}(\mathbf{x},\mathbf{u}_{t+1})\right\},\end{aligned} \tag{99}$$

where M_{t+1} denotes the optimistic estimation for the gradient of $f_{t+1}(\cdot)$ and is typically set as $M_{t+1}=\nabla f_t(\mathbf{x}_t)$ to exploit the smoothness of $f_t(\cdot)$.

To deal with the α-exp-concave $f_t(\cdot)$, we follow Chiang et al. (2012) and set

$$\mathcal{R}_t(\mathbf{x})=\frac{1}{2}\|\mathbf{x}\|_{H_t}^2 \tag{100}$$

where $H_t = I+(\beta G^2/2)I+(\beta/2)\sum_{s=1}^{t-1}h_s$, I denotes the d-dimension identity matrix, $h_t=\nabla f_t(\mathbf{x}_t)\nabla f_t(\mathbf{x}_t)^\top$ and $\beta=(1/2)\min\{1/(4GD),\alpha\}$.

With the above configurations, we can obtain the following gradient-variation bound of OCMD for exp-concave $f_t(\cdot)$.

Theorem 27 *Under Assumptions 1, 2, 3, 4 and 5, if the time-varying function $f_t(\cdot)$ is α-exp-concave, we have*

$$\sum_{t=1}^{T}\left[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\right]-\sum_{t=1}^{T}\left[f_t(\mathbf{x})+r(\mathbf{x})\right]\leq O\left(\frac{d}{\alpha}\log V_T\right),$$

where V_T is defined in (6).

Furthermore, we establish the following (pseudo) small-loss bound.

Theorem 28 *Under Assumptions 1, 2, 3, 4 and 5, if the time-varying function $f_t(\cdot)$ is α-exp-concave, we have*

$$\sum_{t=1}^{T}\big[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)\big]-\sum_{t=1}^{T}\big[f_t(\mathbf{x})+r(\mathbf{x})\big]\le O\left(\frac{d}{\alpha}\log\tilde{L}_T\right),$$

where $\tilde{L}_T$ is defined in (38).

C.2 Proof of Theorem 27

We present the following lemma regarding OCMD (Scroccaro et al., 2023), which can be treated as an extension of Lemma 22 to composite optimization.

Lemma 29 *Under the conditions of Lemma 22, and assume $r(\cdot)$ is convex. Consider the points*

$$\begin{aligned}\mathbf{w}&=\operatorname*{argmin}_{\mathbf{y}\in\mathcal{U}}\big[\langle\gamma\boldsymbol{\xi},\mathbf{y}\rangle+r(\mathbf{y})+\mathcal{B}_\omega(\mathbf{y},\mathbf{z}_-)\big],\\ \mathbf{z}_+&=\operatorname*{argmin}_{\mathbf{y}\in\mathcal{U}}\big[\langle\gamma\boldsymbol{\eta},\mathbf{y}\rangle+r(\mathbf{y})+\mathcal{B}_\omega(\mathbf{y},\mathbf{z}_-)\big].\end{aligned}$$

Then for all $\mathbf{z}\in\mathcal{U}$, one has

$$\begin{aligned}&\langle\mathbf{w}-\mathbf{z},\gamma\boldsymbol{\eta}\rangle+r(\mathbf{w})-r(\mathbf{z})\\ \le{}&\frac{\gamma^2}{\alpha}\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_*+\mathcal{B}_\omega(\mathbf{z},\mathbf{z}_-)-\mathcal{B}_\omega(\mathbf{z},\mathbf{z}_+)-\frac{\alpha}{2}\big[\|\mathbf{w}-\mathbf{z}_+\|_2^2+\|\mathbf{z}_--\mathbf{w}\|_2^2\big].\end{aligned}$$

and

$$\|\mathbf{w}-\mathbf{z}_+\|\le\alpha^{-1}\gamma\|\boldsymbol{\xi}-\boldsymbol{\eta}\|_*.$$

Let $\mathbf{x}^*=\operatorname{argmin}_{\mathbf{x}\in\mathcal{X}}\sum_{t=1}^{T}[f_t(\mathbf{x})+r(\mathbf{x})]$. According to Lemma 3, we have

$$\begin{aligned}&\sum_{t=1}^{T}[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)]-\sum_{t=1}^{T}[f_t(\mathbf{x}^*)+r(\mathbf{x}^*)]\\ \le&\sum_{t=1}^{T}\left[\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}^*\rangle+r(\mathbf{x}_t)-r(\mathbf{x}^*)-\frac{\beta}{2}\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}^*-\mathbf{x}_t\rangle^2\right].\end{aligned}$$

Then, applying Lemma 29 with $\boldsymbol{\xi}=\nabla f_{t-1}(\mathbf{x}_{t-1}),\boldsymbol{\eta}=\nabla f_t(\mathbf{x}_t)$, $\alpha=\gamma=1$, and $\omega(\cdot)=\mathcal{R}_t(\cdot)$ shown in (100), we obtain

$$\|\mathbf{x}_t-\mathbf{u}_{t+1}\|_{H_t}\le\|\nabla f_{t-1}(\mathbf{x}_{t-1})-\nabla f_t(\mathbf{x}_t)\|_{H_t^{-1}}\tag{101}$$

and

$$\begin{aligned}\sum_{t=1}^{T}\left[\langle\nabla f_t(\mathbf{x}_t),\mathbf{x}_t-\mathbf{x}^*\rangle+r(\mathbf{x}_t)-r(\mathbf{x}^*)-\frac{\beta}{2}\|\mathbf{x}^*-\mathbf{x}_t\|_{h_t}^2\right]\le\underbrace{\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)-\nabla f_{t-1}(\mathbf{x}_{t-1})\|_{H_t^{-1}}^2}_{\texttt{term (a)}}\\ +\underbrace{\frac{1}{2}\sum_{t=1}^{T}\big[\|\mathbf{x}^*-\mathbf{u}_t\|_{H_t}^2-\|\mathbf{x}^*-\mathbf{u}_{t+1}\|_{H_t}^2-\beta\|\mathbf{x}^*-\mathbf{x}_t\|_{h_t}^2\big]}_{\texttt{term (b)}}-\underbrace{\frac{1}{2}\sum_{t=1}^{T}\big[\|\mathbf{u}_{t+1}-\mathbf{x}_t\|_{H_t}^2+\|\mathbf{x}_t-\mathbf{u}_t\|_{H_t}^2\big]}_{\texttt{term (c)}}.\end{aligned}$$

Next, we analyze the above three terms separately. For `term (a)`, we will utilize the following inequality:

$$\begin{aligned}H_t \succeq& I+\frac{\beta}{4}\sum_{\tau=1}^{t}\left(\nabla f_\tau\left(\mathbf{x}_\tau\right)\nabla f_\tau\left(\mathbf{x}_\tau\right)^\top+\nabla f_{\tau-1}\left(\mathbf{x}_{\tau-1}\right)\nabla f_{\tau-1}\left(\mathbf{x}_{\tau-1}\right)^\top\right)\\ \succeq& I+\frac{\beta}{8}\sum_{\tau=1}^{t}\left(\nabla f_\tau\left(\mathbf{x}_\tau\right)-\nabla f_{\tau-1}\left(\mathbf{x}_{\tau-1}\right)\right)\left(\nabla f_\tau\left(\mathbf{x}_\tau\right)-\nabla f_{\tau-1}\left(\mathbf{x}_{\tau-1}\right)\right)^\top=P_t,\end{aligned} \tag{102}$$

where the first step is due to $G^2 I \succeq \nabla f_t\left(\mathbf{x}_t\right)\nabla f_t\left(\mathbf{x}_t\right)^\top$, and the second step is due to

$$\begin{aligned}&\nabla f_\tau\left(\mathbf{x}_\tau\right)\nabla f_\tau\left(\mathbf{x}_\tau\right)^\top+\nabla f_{\tau-1}\left(\mathbf{x}_{\tau-1}\right)\nabla f_{\tau-1}\left(\mathbf{x}_{\tau-1}\right)^\top\\ &\qquad\succeq\frac{1}{2}\left(\nabla f_\tau\left(\mathbf{x}_\tau\right)-\nabla f_{\tau-1}\left(\mathbf{x}_{\tau-1}\right)\right)\left(\nabla f_\tau\left(\mathbf{x}_\tau\right)-\nabla f_{\tau-1}\left(\mathbf{x}_{\tau-1}\right)\right)^\top.\end{aligned}$$

According to (102), we have

$$\texttt{term (a)}\leq\sum_{t=1}^{T}\|\nabla f_t(\mathbf{x}_t)-\nabla f_{t-1}(\mathbf{x}_{t-1})\|^2_{P_t^{-1}}=\frac{8}{\beta}\sum_{t=1}^{T}\left\|\sqrt{\frac{\beta}{8}}\left(\nabla f_t\left(\mathbf{x}_t\right)-\nabla f_{t-1}\left(\mathbf{x}_{t-1}\right)\right)\right\|^2_{P_t^{-1}}. \tag{103}$$

Then, we introduce the following lemma (Chen et al., 2023, Lemma 10).

Lemma 30 *Let* $u_t\in\mathbb{R}^d$ *(*$t=1,\ldots,T$*, be a sequence of vectors. Define* $S_t=\sum_{i=1}^t u_i u_i^\top+\epsilon I$*, where* $\epsilon>0$*. Then* $\sum_{t=1}^T u_t^\top S_t^{-1}u_t\leq d\ln\left(1+\frac{1}{d\epsilon}\sum_{t=1}^T\|u_t\|_2^2\right)$*.*

Applying Lemma 30 with $\mathbf{u}_t=\sqrt{\frac{\beta}{8}}(\nabla f_t\left(\mathbf{x}_t\right)-\nabla f_{t-1}(\mathbf{x}_{t-1}))$ and $\epsilon=1$, we have

$$\sum_{t=1}^{T}\left\|\sqrt{\frac{\beta}{8}}\left(\nabla f_t\left(\mathbf{x}_t\right)-\nabla f_{t-1}\left(\mathbf{x}_{t-1}\right)\right)\right\|^2_{P_t^{-1}}\leq d\ln\left(\frac{\beta}{8d}\bar{V}_T+1\right), \tag{104}$$

where we define

$$\bar{V}_T=\sum_{t=1}^{T}\|\nabla f_t\left(\mathbf{x}_t\right)-\nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2. \tag{105}$$

Combining (103) and (104), we arrive at

$$\texttt{term (a)}\leq\frac{8d}{\beta}\ln\left(\frac{\beta}{8d}\bar{V}_T+1\right). \tag{106}$$

For `term (b)`, we exploit the fact that $H_{t+1} - H_t = \frac{\beta}{2} h_t$ and obtain

$$\begin{aligned}
\texttt{term (b)} =& \frac{1}{2}\left[\|\mathbf{x}^* - \mathbf{u}_1\|_{H_1}^2 - \|\mathbf{x}^* - \mathbf{u}_{T+1}\|_{H_{T+1}}^2\right] \\
&+ \frac{1}{2}\sum_{t=1}^{T}\left\{\|\mathbf{x}^* - \mathbf{u}_{t+1}\|_{H_{t+1}}^2 - \|\mathbf{x}^* - \mathbf{u}_{t+1}\|_{H_t}^2 - \beta\|\mathbf{x}^* - \mathbf{x}_t\|_{h_t}^2\right\} \\
=& \frac{1}{2}\left[\|\mathbf{x}^* - \mathbf{u}_1\|_{H_1}^2 - \|\mathbf{x}^* - \mathbf{u}_{T+1}\|_{H_{T+1}}^2\right] + \frac{\beta}{4}\sum_{t=1}^{T}\left\{\|\mathbf{x}^* - \mathbf{u}_{t+1}\|_{h_t}^2 - 2\|\mathbf{x}^* - \mathbf{x}_t\|_{h_t}^2\right\} \\
\overset{(12),(13)}{\leq}& \left(\frac{1}{2} + \frac{\beta}{4}G^2\right)D^2 + \frac{\beta}{4}\sum_{t=1}^{T}\left\{\|\mathbf{x}^* - \mathbf{u}_{t+1}\|_{h_t}^2 - 2\|\mathbf{x}^* - \mathbf{x}_t\|_{h_t}^2\right\}.
\end{aligned} \tag{107}$$

According to the fact that $H_t \succeq \frac{\beta}{2}G^2 I \succeq \frac{\beta}{2}h_t$ and (101), we have

$$\begin{aligned}
&\|\mathbf{x}^* - \mathbf{u}_{t+1}\|_{h_t}^2 - 2\|\mathbf{x}^* - \mathbf{x}_t\|_{h_t}^2 \leq 2\|\mathbf{x}_t - \mathbf{u}_{t+1}\|_{h_t}^2 \\
\leq& \frac{4}{\beta}\|\mathbf{x}_t - \mathbf{u}_{t+1}\|_{H_t}^2 \overset{(101)}{\leq} \frac{4}{\beta}\|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_{H_t^{-1}}.
\end{aligned} \tag{108}$$

Substituting (108) into (107), we have

$$\texttt{term (b)} \leq \left(\frac{1}{2} + \frac{\beta}{4}G^2\right)D^2 + \texttt{term (a)}. \tag{109}$$

For `term (c)`, we have

$$\begin{aligned}
\texttt{term (c)} =& \frac{1}{2}\sum_{t=2}^{T+1}\|\mathbf{u}_t - \mathbf{x}_{t-1}\|_{H_{t-1}}^2 + \frac{1}{2}\sum_{t=1}^{T}\|\mathbf{x}_t - \mathbf{u}_t\|_{H_t}^2 \\
\geq& \frac{1}{2}\sum_{t=2}^{T}\left\{\|\mathbf{u}_t - \mathbf{x}_{t-1}\|_2^2 + \|\mathbf{x}_t - \mathbf{u}_t\|_2^2\right\} \geq \frac{1}{4}\sum_{t=2}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2,
\end{aligned} \tag{110}$$

where the first inequality is due to $H_t \succeq H_{t-1} \succeq I$, $\|\mathbf{u}_{T+1} - \mathbf{x}_T\|_{H_{T+1}}^2 \geq 0$, $\|\mathbf{x}_1 - \mathbf{u}_1\|_{H_1}^2 \geq 0$.

Combining (106), (109) and (110), we have

$$\begin{aligned}
&\sum_{t=1}^{T}[f_t(\mathbf{x}_t) + r(\mathbf{x}_t)] - \sum_{t=1}^{T}[f_t(\mathbf{x}^*) + r(\mathbf{x}^*)] \\
\leq& \left(\frac{1}{2} + \frac{\beta}{4}G^2\right)D^2 + \frac{16d}{\beta}\ln\left(\frac{\beta}{8d}\bar{V}_T + 1\right) - \frac{1}{4}\sum_{t=2}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2.
\end{aligned} \tag{111}$$

Notice that $\bar{V}_T$ in (105) can be bounded in the following way:

$$\begin{aligned}
\bar{V}_T \leq& G^2 + 2\sum_{t=2}^{T}\|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_t)\|_2^2 + 2\sum_{t=2}^{T}\|\nabla f_{t-1}(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2 \\
\leq& G^2 + 2V_T + 2H^2\sum_{t=2}^{T}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2.
\end{aligned} \tag{112}$$

Substituting (112) into (111), we have

$$\begin{aligned}
&\sum_{t=1}^{T}[f_t(\mathbf{x}_t)+r(\mathbf{x}_t)]-\sum_{t=1}^{T}[f_t(\mathbf{x}^*)+r(\mathbf{x}^*)]\le\left(\frac{1}{2}+\frac{\beta}{4}G^2\right)D^2-\frac{1}{4}\sum_{t=2}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2\\
&+\frac{16d}{\beta}\ln\left(\frac{\beta}{8d}G^2+\frac{\beta}{4d}V_T+\frac{\beta H^2}{4d}\sum_{t=2}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2+1\right)\\
\overset{(82)}{\le}&\left(\frac{1}{2}+\frac{\beta}{4}G^2\right)D^2+\frac{16d}{\beta}\ln\left(\frac{\beta}{8d}G^2+\frac{\beta}{4d}V_T+1\right)\\
&+\frac{16d}{\beta}\ln\left(\frac{\beta H^2}{4d}\sum_{t=2}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2+1\right)-\frac{1}{4}\sum_{t=2}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2. \qquad (113)
\end{aligned}$$

To simplify (113), we make use of Lemma 23 and obtain

$$\frac{16d}{\beta}\ln\left(\frac{\beta H^2}{4d}\sum_{t=2}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2+1\right)-\frac{1}{4}\sum_{t=2}^{T}\|\mathbf{x}_t-\mathbf{x}_{t-1}\|_2^2\le\frac{16d}{\beta}\ln\left(\frac{\beta H^2}{d}+1\right). \qquad (114)$$

Combining (113) and (114), we complete the proof.

C.3 Proof of Theorem 28

Notice that for $\bar{V}_T$ in (105), we have

$$\begin{aligned}
\bar{V}_T&\le\|\nabla f_1(\mathbf{x}_1)\|_2^2+2\sum_{t=2}^{T}\|\nabla f_t(\mathbf{x}_t)\|_2^2+2\sum_{t=2}^{T}\|\nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2\\
&\le 8H\sum_{t=1}^{T}f_t(\mathbf{x}_t)+8H\sum_{t=2}^{T}f_{t-1}(\mathbf{x}_{t-1})\le 16H\sum_{t=1}^{T}f_t(\mathbf{x}_t),
\end{aligned} \qquad (115)$$

where the second step is due to Lemma 19. Substituting (115) into (111) finishes the proof.

C.4 Proof of Lemma 29

Firstly, by using the convexity of $r(\cdot)$, for all $\mathbf{z}\in\mathcal{U}$ we have

$$\begin{aligned}
&\langle\mathbf{w}-\mathbf{z},\gamma\eta\rangle+r(\mathbf{w})-r(\mathbf{z})\\
=&\ \langle\mathbf{w}-\mathbf{z},\gamma\eta\rangle+r(\mathbf{w})-r(\mathbf{z}_+)+r(\mathbf{z}_+)-r(\mathbf{z})\\
\le&\ \langle\mathbf{w}-\mathbf{z},\gamma\eta\rangle+\langle\mathbf{w}-\mathbf{z}_+,\nabla r(\mathbf{w})\rangle+\langle\mathbf{z}_+-\mathbf{z},\nabla r(\mathbf{z}_+)\rangle\\
=&\ \langle\mathbf{w}-\mathbf{z}_+,\gamma\eta-\gamma\xi\rangle+\underbrace{\langle\mathbf{w}-\mathbf{z}_+,\gamma\xi+\nabla r(\mathbf{w})\rangle}_{\texttt{term (a)}}+\underbrace{\langle\mathbf{z}_+-\mathbf{z},\gamma\eta+\nabla r(\mathbf{z}_+)\rangle}_{\texttt{term (b)}}. \qquad (116)
\end{aligned}$$

Next we introduce the following lemma (Scroccaro et al., 2023, Lemma 3.1) to bound `term (a)` and `term (b)`.

Lemma 31 *Suppose that $\mathcal{X}$ is a closed convex set. Let $\varphi : \mathcal{X} \to \mathbb{R}$ be a convex function and $\eta > 0$. Define*

$$u = \operatorname*{argmin}_{x \in \mathcal{X}} \{\eta\varphi(x) + \mathcal{B}_\omega(x, v)\}.$$

It follows that, for all $z \in \mathcal{X}$ and $g(u) \in \partial\varphi(u)$,

$$\eta\langle g(u), u - z\rangle \leq \mathcal{B}_\omega(z, v) - \mathcal{B}_\omega(z, u) - \mathcal{B}_\omega(u, v).$$

Applying Lemma 31 to the update rules, we have

$$\begin{aligned}
\texttt{term (a)} &= \langle \gamma\xi + \nabla r(\mathbf{w}), \mathbf{w} - \mathbf{z}_+\rangle \leq \mathcal{B}_\omega(\mathbf{z}_+, \mathbf{z}_-) - \mathcal{B}_\omega(\mathbf{z}_+, \mathbf{w}) - \mathcal{B}_\omega(\mathbf{w}, \mathbf{z}_-),\\
\texttt{term (b)} &= \langle \gamma\eta + \nabla r(\mathbf{z}_+), \mathbf{z}_+ - \mathbf{z}\rangle \leq \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_-) - \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_+) - \mathcal{B}_\omega(\mathbf{z}_+, \mathbf{z}_-).
\end{aligned}$$

As a result, we can bound (116) as

$$\begin{aligned}
&\langle \mathbf{w} - \mathbf{z}, \gamma\eta\rangle + r(\mathbf{w}) - r(\mathbf{z})\\
\leq\ &\langle \mathbf{w} - \mathbf{z}_+, \gamma\eta - \gamma\xi\rangle + \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_-) - \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_+) - \mathcal{B}_\omega(\mathbf{z}_+, \mathbf{w}) - \mathcal{B}_\omega(\mathbf{w}, \mathbf{z}_-)\\
\leq\ &\gamma \left\|\eta - \xi\right\|_* \left\|\mathbf{w} - \mathbf{z}_+\right\| + \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_-) - \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_+) - \mathcal{B}_\omega(\mathbf{z}_+, \mathbf{w}) - \mathcal{B}_\omega(\mathbf{w}, \mathbf{z}_-)\\
\leq\ &\gamma \left\|\eta - \xi\right\|_* \left\|\mathbf{w} - \mathbf{z}_+\right\| + \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_-) - \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_+) - \frac{\alpha}{2}\left[\|\mathbf{w} - \mathbf{z}_+\|_2^2 + \|\mathbf{z}_- - \mathbf{w}\|_2^2\right],
\end{aligned}$$

where the last inequality is due to that $\omega(\cdot)$ is α-strongly convex. Then, we exploit Lemma 3.2 of Scroccaro et al. (2023) to get

$$\|\mathbf{w} - \mathbf{z}_+\| \leq \alpha^{-1}\gamma\|\xi - \eta\|_*.$$

Hence, we finally arrive at

$$\begin{aligned}
&\langle \mathbf{w} - \mathbf{z}, \gamma\eta\rangle + r(\mathbf{w}) - r(\mathbf{z})\\
\leq\ &\frac{\gamma^2}{\alpha}\|\xi - \eta\|_* + \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_-) - \mathcal{B}_\omega(\mathbf{z}, \mathbf{z}_+) - \frac{\alpha}{2}\left[\|\mathbf{w} - \mathbf{z}_+\|_2^2 + \|\mathbf{z}_- - \mathbf{w}\|_2^2\right].
\end{aligned}$$

References

Jacob Abernethy, Peter L. Bartlett, Alexander Rakhlin, and Ambuj Tewari. Optimal strategies and minimax lower bounds for online convex games. In *Proceedings of the 21st Annual Conference on Learning Theory*, pages 415–423, 2008.

Peter Auer, Nicolò Cesa-Bianchi, and Claudio Gentile. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 64(1):48–75, 2002.

Peter L. Bartlett, Elad Hazan, and Alexander Rakhlin. Adaptive online gradient descent. In *Advances in Neural Information Processing Systems 20*, pages 65–72, 2007.

Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, may 2003.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games.* Cambridge University Press, 2006.

Nicolò Cesa-Bianchi and Francesco Orabona. Online learning algorithms. *Annual Review of Statistics and Its Application*, 8(1):165–190, 2021.

Sijia Chen, Wei-Wei Tu, Peng Zhao, and Lijun Zhang. Optimistic online mirror descent for bridging stochastic and adversarial online convex optimization. In *Proceedings of the 40th International Conference on Machine Learning*, pages 5002–5035, 2023.

Chao-Kai Chiang, Tianbao Yang, Chia-Jung Lee, Mehrdad Mahdavi, Chi-Jen Lu, Rong Jin, and Shenghuo Zhu. Online optimization with gradual variations. In *Proceedings of the 25th Annual Conference on Learning Theory*, pages 6.1–6.20, 2012.

Ashok Cutkosky. Artificial constraints and hints for unbounded online learning. In *Proceedings of the 32nd Conference on Learning Theory*, pages 874–894, 2019.

Ashok Cutkosky and Kwabena Boahen. Online convex optimization with unconstrained domains and losses. In *Advances in Neural Information Processing Systems 29*, pages 748–756, 2016.

Ashok Cutkosky and Kwabena Boahen. Online learning without prior information. In *Proceedings of the 30th Annual Conference on Learning Theory*, pages 643–677, 2017.

Ashok Cutkosky and Francesco Orabona. Black-box reductions for parameter-free online learning in Banach spaces. In *Proceedings of the 31st Conference On Learning Theory*, pages 1493–1529, 2018.

Amit Daniely, Alon Gonen, and Shai Shalev-Shwartz. Strongly adaptive online learning. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1405–1411, 2015.

Chuong Do, Quoc Le, and Chuan-Sheng Foo. Proximal regularization for online and batch learning. In *Proceedings of the 26th International Conference on Machine Learning*, pages 257–264, 2009.

John Duchi and Yoram Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10:2899–2934, 2009.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. In *Proceedings of the 23rd Annual Conference on Learning Theory*, pages 257–269, 2010a.

John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Ambuj Tewari. Composite objective mirror descent. In *Proceedings of the 23rd Annual Conference on Learning Theory*, pages 14–26, 2010b.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

Dylan J Foster, Satyen Kale, Mehryar Mohri, and Karthik Sridharan. Parameter-free online learning via model selection. In *Advances in Neural Information Processing Systems 30*, pages 6020–6030, 2017.

Pierre Gaillard, Gilles Stoltz, and Tim van Erven. A second-order bound with excess losses. In *Proceedings of the 27th Conference on Learning Theory*, pages 176–196, 2014.

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

Elad Hazan and C. Seshadhri. Adaptive algorithms for online decision problems. *Electronic Colloquium on Computational Complexity*, 88, 2007.

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

Pooria Joulani, András György, and Csaba Szepesvári. A modular analysis of adaptive (non-)convex optimization: Optimism, composite objectives, variance reduction, and variational bounds. *Theoretical Computer Science*, 808:108–138, 2020.

Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Wouter M. Koolen and Tim Van Erven. Second-order quantile methods for experts and combinatorial games. In *Proceedings of the 28th Conference on Learning Theory*, pages 1155–1175, 2015.

Haochuan Li, Jian Qian, Yi Tian, Alexander Rakhlin, and Ali Jadbabaie. Convex and non-convex optimization under generalized smoothness. In *In Advances in Neural Information Processing Systems 36*, pages 40238–40271, 2023.

Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Haipeng Luo and Robert E. Schapire. Achieving all with no parameters: Adanormalhedge. In *Proceedings of the 28th Conference on Learning Theory*, pages 1286–1304, 2015.

Brendan Mcmahan and Matthew Streeter. No-regret algorithms for unconstrained online convex optimization. In *Advances in Neural Information Processing Systems 25*, pages 2402–2410, 2012.

Brendan Mcmahan and Matthew Streeter. Minimax optimal algorithms for unconstrained linear optimization. In *Advances in Neural Information Processing Systems 26*, pages 2724–2732, 2013.

Zakaria Mhammedi and Wouter M. Koolen. Lipschitz and comparator-norm adaptivity in online learning. In *Proceedings of 33rd Conference on Learning Theory*, pages 2858–2887, 2020.

Zakaria Mhammedi, Wouter M. Koolen, and Tim Van Erven. Lipschitz adaptivity with multiple learning rates in online learning. In *Proceedings of the 32nd Conference on Learning Theory*, pages 2490–2511, 2019.

Mehryar Mohri and Scott Yang. Accelerating online convex optimization via adaptive prediction. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 848–856, 2016.

Mahesh Chandra Mukkamala and Matthias Hein. Variants of RMSProp and Adagrad with logarithmic regret bounds. In *Proceedings of the 34th International Conference on Machine Learning*, pages 2545–2553, 2017.

Arkadi Nemirovski. Prox-method with rate of convergence $O(1/t)$ for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004.

Yurii Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical Programming*, 120(1):261–283, 2009.

Francesco Orabona. Simultaneous model selection and optimization through parameter-free stochastic learning. In *Advances in Neural Information Processing Systems 27*, pages 1116–1124, 2014.

Francesco Orabona. A modern introduction to online learning. *ArXiv e-prints*, arXiv:1912.13213(v6), 2019.

Francesco Orabona and Dávid Pál. Coin betting and parameter-free online learning. In *Advances in Neural Information Processing Systems 29*, pages 577–585, 2016.

Francesco Orabona and Dávid Pál. Scale-free online learning. *Theoretical Computer Science*, 716:50–69, 2018.

Francesco Orabona, Nicolo Cesa-Bianchi, and Claudio Gentile. Beyond logarithmic bounds in online learning. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*, pages 823–831, 2012.

Erik Ordentlich and Thomas M. Cover. The cost of achieving the best portfolio in hindsight. *Mathematics of Operations Research*, 23(4):960–982, 1998.

Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In *Proceedings of the 26th Conference on Learning Theory*, pages 993–1019, 2013.

Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of Adam and beyond. In *International Conference on Learning Representations*, 2018.

Pedro Zattoni Scroccaro, Arman Sharifi Kolarijani, and Peyman Mohajerin Esfahani. Adaptive composite online optimization: Predictions in static and dynamic environments. *IEEE Transactions on Automatic Control*, 68(5):2906–2921, 2023.

Shai Shalev-Shwartz. *Online Learning: Theory, Algorithms, and Applications*. PhD thesis, The Hebrew University of Jerusalem, 2007.

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

Shai Shalev-Shwartz, Yoram Singer, and Nathan Srebro. Pegasos: primal estimated subgradient solver for SVM. In *Proceedings of the 24th International Conference on Machine Learning*, pages 807–814, 2007.

Nathan Srebro, Karthik Sridharan, and Ambuj Tewari. Smoothness, low-noise and fast rates. In *Advances in Neural Information Processing Systems 23*, pages 2199–2207, 2010.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, pages 26–31, 2012.

Kim-Chuan Toh and Sangwoon Yun. An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. *Pacific Journal of Optimization*, 6(3):615–640, 2010.

Tim van Erven and Wouter M. Koolen. MetaGrad: Multiple learning rates in online learning. In *Advances in Neural Information Processing Systems 29*, pages 3666–3674, 2016.

Tim van Erven, Wouter M. Koolen, and Dirk van der Hoeven. MetaGrad: Adaptation using multiple learning rates in online learning. *Journal of Machine Learning Research*, 22(161):1–61, 2021.

Yuanyu Wan and Lijun Zhang. Efficient adaptive online learning via frequent directions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6910–6923, 2022.

Guanghui Wang, Shiyin Lu, and Lijun Zhang. Adaptivity and optimality: A universal algorithm for online convex optimization. In *Proceedings of the 35th Conference on Uncertainty in Artificial Intelligence*, pages 659–668, 2019.

Guanghui Wang, Shiyin Lu, Quan Cheng, Wei-Wei Tu, and Lijun Zhang. SAdam: A variant of Adam for strongly convex functions. In *International Conference on Learning Representations*, 2020a.

Guanghui Wang, Shiyin Lu, Yao Hu, and Lijun Zhang. Adapting to smoothness: A more universal algorithm for online convex optimization. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pages 6162–6169, 2020b.

Yibo Wang, Sijia Chen, Wei Jiang, Wenhao Yang, Yuanyu Wan, and Lijun Zhang. Online composite optimization between stochastic and adversarial environments. In *In Advances in Neural Information Processing Systems 37*, pages 94808–94850, 2024a.

Yibo Wang, Wenhao Yang, Wei Jiang, Shiyin Lu, Bin Wang, Haihong Tang, Yuanyu Wan, and Lijun Zhang. Non-stationary projection-free online learning with dynamic and adaptive regret guarantees. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, pages 15671–15679, 2024b.

Chen-Yu Wei, Yi-Te Hong, and Chi-Jen Lu. Tracking the best expert in non-stationary stochastic environments. In *Advances in Neural Information Processing Systems 29*, pages 3979–3987, 2016.

Lin Xiao. Dual averaging method for regularized stochastic learning and online optimization. In *Advances in Neural Information Processing Systems 22*, pages 2116–2124, 2009.

Yan-Feng Xie, Peng Zhao, and Zhi-Hua Zhou. Gradient-variation online learning under generalized smoothness. In *Advances in Neural Information Processing Systems 37*, pages 37865–37899, 2024.

Yu-Hu Yan, Peng Zhao, and Zhi-Hua Zhou. Universal online learning with gradual variations: A multi-layer online ensemble approach. In *Advances in Neural Information Processing Systems 36*, pages 37682–37715, 2023.

Yu-Hu Yan, Peng Zhao, and Zhi-Hua Zhou. A simple and optimal approach for universal online learning with gradient variations. In *Advances in Neural Information Processing Systems 37*, pages 11132–11163, 2024.

Sifan Yang, Yuanyu Wan, Peijia Li, Yibo Wang, Xiao Zhang, Zhewei Wei, and Lijun Zhang. Dimension-free adaptive subgradient methods with frequent directions. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Tianbao Yang, Mehrdad Mahdavi, Rong Jin, and Shenghuo Zhu. Regret bounded by gradual variation for online convex optimization. *Machine Learning*, 95:183–223, 2014.

Tianbao Yang, Zhe Li, and Lijun Zhang. A simple analysis for exp-concave empirical minimization with arbitrary convex regularizer. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, pages 445–453, 2018.

Wenhao Yang, Wei Jiang, Yibo Wang, Ping Yang, Yao Hu, and Lijun Zhang. Small-loss adaptive regret for online convex optimization. In *Proceedings of the 41st International Conference on Machine Learning*, pages 56156–56195, 2024a.

Wenhao Yang, Yibo Wang, Peng Zhao, and Lijun Zhang. Universal online convex optimization with 1 projection per round. In *Advances in Neural Information Processing Systems 37*, pages 31438–31472, 2024b.

Xu Yang, Peng Tian, Xiao Cheng, Yuanyu Wan, and Mingli Song. Regularized online exponentially concave optimization. *Neurocomputing*, 595:1–8, 2024c.

Matthew D. Zeiler. Adadelta: An adaptive learning rate method. *ArXiv e-prints*, arXiv:1212.5701, 2012.

Lijun Zhang. Online learning in changing environments. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, pages 5178–5182, 2020. Early Career.

Lijun Zhang, Tianbao Yang, Rong Jin, and Zhi-Hua Zhou. A simple homotopy algorithm for compressive sensing. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, pages 1116–1124, 2015.

Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems 31*, pages 1323–1333, 2018.

Lijun Zhang, Tie-Yan Liu, and Zhi-Hua Zhou. Adaptive regret of convex and smooth functions. In *Proceedings of the 36th International Conference on Machine Learning*, pages 7414–7423, 2019a.

Lijun Zhang, Tianbao Yang, Rong Jin, and Zhi-Hua Zhou. Relative error bound analysis for nuclear norm regularized matrix completion. *Journal of Machine Learning Research*, 20(97):1–22, 2019b.

Lijun Zhang, Guanghui Wang, Wei-Wei Tu, Wei Jiang, and Zhi-Hua Zhou. Dual adaptivity: A universal algorithm for minimizing the adaptive regret of convex functions. In *Advances in Neural Information Processing Systems 34*, pages 24968–24980, 2021.

Lijun Zhang, Guanghui Wang, Jinfeng Yi, and Tianbao Yang. A simple yet universal strategy for online convex optimization. In *Proceedings of the 39th International Conference on Machine Learning*, pages 26605–26623, 2022.

Peng Zhao, Yu-Jie Zhang, Lijun Zhang, and Zhi-Hua Zhou. Dynamic regret of convex and smooth functions. In *Advances in Neural Information Processing Systems 33*, pages 12510–12520, 2020.

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