Bandit Convex Optimization in Non-stationary Environments

Peng Zhao  
zhaop@lamda.nju.edu.cn  
Guanghui Wang  
wanggh@lamda.nju.edu.cn  
Lijun Zhang  
zhanglj@lamda.nju.edu.cn  
Zhi-Hua Zhou  
zhouzh@lamda.nju.edu.cn  
National Key Laboratory for Novel Software Technology  
Nanjing University, Nanjing 210023, China

Abstract

Bandit Convex Optimization (BCO) is a fundamental framework for modeling sequential decision-making with partial information, where the only feedback available to the player is the one-point or two-point function values. In this paper, we investigate BCO in non-stationary environments and choose the dynamic regret as the performance measure, which is defined as the difference between the cumulative loss incurred by the algorithm and that of any feasible comparator sequence. Let $T$ be the time horizon and $P_T$ be the path-length of the comparator sequence that reflects the non-stationary of environments. We propose a novel algorithm that achieves $O(T^{3/4}(1 + P_T)^{1/2})$ and $O(T^{1/2}(1 + P_T)^{1/2})$ dynamic regret respectively for the one-point and two-point feedback models. The latter result is optimal, matching the $\Omega(T^{1/2}(1 + P_T)^{1/2})$ lower bound established in this paper. Notably, our algorithm is more adaptive to non-stationary environments since it does not require prior knowledge of the path-length $P_T$ ahead of time, which is generally unknown.

Keywords: Bandits Convex Optimization, Dynamic Regret, Non-stationary Environments

1. Introduction

Online Convex Optimization (OCO) is a powerful tool for modeling sequential decision-making problems, which can be regarded as an iterative game between the player and environments (Shalev-Shwartz, 2012). At iteration $t$, the player commits a decision $x_t$ from a convex feasible set $\mathcal{X} \subseteq \mathbb{R}^d$, simultaneously, a convex function $f_t : \mathcal{X} \mapsto \mathbb{R}$ is revealed by environments, and then the player will suffer an instantaneous loss $f_t(x_t)$. The standard performance measure is the regret, defined as

$$\text{S-Regret}_T = \min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x) - \sum_{t=1}^{T} f_t(x_t)$$

which is the difference between the cumulative loss of the player and that of the best fixed decision in hindsight. To emphasize the fact that the comparator in (1) is fixed, it is called static regret.

There are two setups for online convex optimization according to the information that environments reveal. In the full-information setup, the player has all the information of
the function \( f_t \), including the gradient of \( f_t \) over \( \mathcal{X} \). By contrast, in the bandit setup, the instantaneous loss is the only feedback available to the player. In this paper, we focus on the latter case, which is referred to as bandit convex optimization (BCO).

Recently, BCO has attracted considerable attention because it successfully models many real-world scenarios where the feedback available to the decision maker is partial or incomplete (Hazan, 2016). The key challenge lies in the limited feedback, i.e., the player has no access to the gradient of the function. In the standard one-point feedback model, the only feedback is the one-point function value, based on which Flaxman et al. (2005) construct an unbiased estimator of the gradient and then appeal to the online gradient descent algorithm that developed in the full-information setting (Zinkevich, 2003) to establish an \( O(T^{3/4}) \) expected regret. Another common variant is the two-point feedback model, where the player is allowed to query function values of two points in each iteration. Agarwal et al. (2010) demonstrate an optimal \( O(\sqrt{T}) \) regret for convex functions. The algorithms and regret bounds are further extended in later studies (Saha and Tewari, 2011; Hazan and Levy, 2014; Bubeck et al., 2015; Dekel et al., 2015; Yang and Mohri, 2016; Bubeck et al., 2017).

Note that the static regret in (1) compares with a fixed benchmark, so it implicitly assumes that there is a reasonably good decision over all iterations. Unfortunately, this may not be true in non-stationary environments, where the underlying distribution of online functions changes. To address this limitation, the notion of dynamic regret is introduced by Zinkevich (2003) and defined as the difference between the cumulative loss of the player and that of a comparator sequence \( u_1, \ldots, u_T \in \mathcal{X} \),

\[
D\text{-Regret}_T(u_1, \ldots, u_T) = \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(u_t). \tag{2}
\]

In contrast to the fixed benchmark in the static regret, dynamic regret compares with a changing comparator sequence and therefore is more suitable in non-stationary environments. We remark that (2) is also called the universal dynamic regret, since it holds universally for any feasible comparator sequence. In the literature, there is a variant named the worst-case dynamic regret (Besbes et al., 2015; Jadbabaie et al., 2015; Yang et al., 2016), which specifies the comparator sequence to be the minimizers of online functions, namely, \( u_t = x^*_t \in \arg\min_{x \in \mathcal{X}} f_t(x) \). As pointed out by Zhang et al. (2018a), the universal dynamic regret is more desired, because the worst-case dynamic regret is typically too pessimistic while the universal one is more adaptive to the non-stationarity of environments. Meanwhile, the universal dynamic regret is more general because it accommodates the worst-case dynamic regret and static regret as special cases.

Recently, there are some studies on the worst-case dynamic regret of BCO problems (Yang et al., 2016; Chen and Giannakis, 2019). However, there are two disadvantages in existing results. On one hand, they only provide the worst-case dynamic regret. On the other hand, their algorithms require prior information of the path-length ahead of time which is generally unknown. Therefore, it is desired to design algorithms that enjoy universal dynamic regret for BCO problems.

In this paper, we propose the Parameter-Free Bandit Gradient Descent algorithm (ParaFree-BGD) that fulfills the above requirement. ParaFree-BGD algorithm is inspired by the strategy of maintaining multiple learning rates in MetaGrad (van Erven and Koolen, 2016), and
Table 1: Comparisons of dynamic regret for BCO problems. In the table, the column of “Param-Free” indicates whether the algorithm requires to know the path-length in advance. Meanwhile, $T$ is the time horizon, $P_T = P_T(u_1, \ldots, u_T)$ and $P^*_T = P_T(x^*_1, \ldots, x^*_T)$ are the path-length defined in (3), where $x^*_t \in \arg \min_{x \in X} f_t(x)$.

| Feedback model | Dynamic regret | Type       | Param-Free | Reference |
|----------------|----------------|------------|------------|-----------|
| one-point      | $O(T^{3/4}(1 + P^*_T))$ | worst-case | NO         | (Chen and Giannakis, 2019) |
| one-point      | $O(T^{3/4}(1 + P_T))$   | universal  | YES        | This work |
| two-point      | $O(\sqrt{T(1 + P^*_T)})$ | worst-case | NO         | (Yang et al., 2016) |
| two-point      | $O(\sqrt{T(1 + P_T)})$   | worst-case | NO         | (Chen and Giannakis, 2019) |
| two-point      | $O(\sqrt{T(1 + P_T)})$   | universal  | YES        | This work |

also consists of meta-algorithm and expert-algorithm. The basic idea is to maintain a pool of candidate parameters, and then invoke multiple instances of the expert-algorithm simultaneously, where each expert-algorithm is associated with a candidate parameter. Then, the meta-algorithm combines predictions from expert-algorithms by an expert-tracking algorithm (Cesa-Bianchi and Lugosi, 2006). However, it is prohibited to run multiple expert-algorithms with different parameters simultaneously because the player is only allowed to query one/two points in the bandit setup. To overcome this difficulty, we carefully design a surrogate function, as the linearization of the smoothed version of the loss function, and make the strategy suitable for BCO problems. Our algorithm and analysis accommodate one-point and two-point feedback models, and Table 1 summarizes existing dynamic regret for BCO problems and our results. The main contributions of this work are listed as follows.

- We establish the first universal dynamic regret that supports to compare with any feasible comparator sequence for BCO problems, in a unified analysis framework.
- We propose a parameter-free algorithm, which does not need to know the upper bound of the path-length $P_T$ ahead of time, and meanwhile enjoys the state-of-the-art dynamic regret.
- We establish the first lower bound of universal dynamic regret for BCO problems.

2. Related Work

In this section, we briefly introduce related works of bandit convex optimization and dynamic regret.

2.1 Bandit Convex Optimization

As opposed to the full-information setting, the gradients are not accessible in BCO setting. The player is only allowed to query the function values of one point or two points.

For the one-point feedback model, the seminal work of Flaxman et al. (2005) constructs an unbiased gradient estimator and establishes an $O(T^{3/4})$ expected regret for convex and Lipschitz functions, and a similar result is independently obtained by Kleinberg (2004).
Later, an $O(T^{2/3})$ rate is demonstrated to be attainable with either strong convexity (Agarwal et al., 2010) or smoothness condition (Saha and Tewari, 2011). When functions are both strongly convex and smooth, Hazan and Levy (2014) design a novel algorithm that achieves a regret of $O(\sqrt{T \log T})$ based on the follow-the-regularized-leader framework with self-concordant barriers, matching the $\Omega(\sqrt{T})$ lower bound (Shamir, 2013) up to logarithmic factors. Furthermore, recent studies (Bubeck et al., 2015, 2017) show that an $O(poly(\log T) \sqrt{T})$ regret is attainable for convex and Lipschitz functions, though with a high dependence on the dimension $d$.

BCO with two-point feedback is proposed and studied by Agarwal et al. (2010), and is also independently studied in the context of stochastic optimization (Nesterov, 2011). Agarwal et al. (2010) first establish the expected regret of $O(d^2 \sqrt{T})$ and $O(d^2 \log T)$ for convex Lipschitz and strongly convex Lipschitz functions, respectively. These bounds are proved to be minimax optimal in $T$ (Agarwal et al., 2010), and the dependence on $d$ is later improved to be optimal (Shamir, 2017).

Besides, bandit linear optimization is a special case of BCO where the feedback is assumed to be a linear function of the chosen decision, and has been studied extensively (Awerbuch and Kleinberg, 2004; McMahan and Blum, 2004; Dani et al., 2008b; Abernethy et al., 2008a; Bubeck et al., 2012).

### 2.2 Dynamic Regret

There are two types of dynamic regret as aforementioned. The universal dynamic regret holds universally for any feasible comparator sequence, while the worst-case one only compares with the sequence of the minimizers of online functions.

For the universal dynamic regret, existing results are only limited to the full-information setting. Zinkevich (2003) shows that online gradient descent achieves an $O(\sqrt{T}(1 + P_T))$ regret, where $P_T = P_T(u_1, \ldots, u_T)$ is the path-length of the comparator sequence $u_1, \ldots, u_T$,

$$P_T(u_1, \ldots, u_T) = \sum_{t=1}^{T} \|u_{t-1} - u_t\|_2. \hspace{1cm} (3)$$

Recently, Zhang et al. (2018a) demonstrate that this upper bound is not optimal by establishing an $\Omega(\sqrt{T}(1 + P_T))$ lower bound, and further propose an algorithm that attains an optimal $O(\sqrt{T}(1 + P_T))$ dynamic regret. However, there is no universal dynamic regret in the bandit setting.

For the worst-case dynamic regret, there are many studies in the full-information setting (Besbes et al., 2015; Jadababaie et al., 2015; Yang et al., 2016; Mokhtari et al., 2016; Zhang et al., 2017) as well as few works in the bandit setting (Gur et al., 2014; Wei et al., 2016; Karnin and Anava, 2016; Luo et al., 2018; Cheung et al., 2019; Auer et al., 2019; Chen et al., 2019). In the bandit convex optimization, when the upper bound of $P_T^*$ is known, Yang et al. (2016) establish an $O(\sqrt{T}(1 + P_T^*))$ dynamic regret for the two-point feedback model. Here, $P_T^*$ is short for $P_T(x_1^*, \ldots, x_T^*)$, the path-length of the minimizers of online functions, i.e., $x_t^* \in \arg \min_{x \in \mathcal{X}} f_t(x)$. Later, Chen and Giannakis (2019) apply BCO techniques in the dynamic Internet-of-Things management, showing $O(T^{3/4}(1 + P_T^*))$ and $O(T^{1/2}(1 + P_T^*))$ dynamic regrets respectively for one-point and two-point feedback models.
Another closely related performance measure for online convex optimization in non-stationary environments is the *adaptive regret* (Hazan and Seshadhri, 2009), which is defined as the maximum of “local” static regret in every time interval \([q, s] \subseteq [T]\),

\[
A-\text{Regret}_T = \max_{[q, s] \subseteq [T]} \sum_{t=q}^{s} f_t(x_t) - \min_{x \in X} \sum_{t=q}^{s} f_t(x).
\]

Hazan and Seshadhri (2009) propose an efficient algorithm that enjoys \(O(\sqrt{T \log^3 T})\) and \(O(d \log^2 T)\) regrets for convex and exponentially concave functions, respectively. The rate for convex functions is improved later (Daniely et al., 2015; Jun et al., 2017). Moreover, Zhang et al. (2018b) investigate the relation between adaptive regret and the worst-case dynamic regret.

### 3. Our Approach

In this section, we first present assumptions that used in the paper, then provide the universal dynamic regret analysis for the bandit gradient descent algorithm under one-point and two-point feedback models. We finally develop a parameter-free algorithm with corresponding dynamic regret guarantees.

#### 3.1 Assumptions

We make following common assumptions for bandit convex optimization (Flaxman et al., 2005; Agarwal et al., 2010).

**Assumption 1** (Bounded Region). The feasible set \(X\) contains the ball of radius \(r\) centered at the origin and is contained in the ball of radius \(R\),

\[
rB \subseteq X \subseteq RB
\]

where \(B = \{x \in \mathbb{R}^d : \|x\|_2 \leq 1\}\).

**Assumption 2** (Bounded Function Value). The values of all the functions are bounded by \(C\),

\[
\forall t \in [T], \quad \max_{x \in X} |f_t(x)| \leq C.
\]

**Assumption 3** (Bounded Gradient). The gradients of all the functions are bounded by \(G\),

\[
\forall t \in [T], \quad \max_{x \in X} \|\nabla f_t(x)\|_2 \leq G.
\]

**Assumption 4** (Lipschitz Continuity). All the functions are \(L\)-Lipschitz continuous over domain \(X\), that is, for all \(x, y \in X\), we have

\[
\forall t \in [T], \quad |f_t(x) - f_t(y)| \leq L\|x - y\|_2.
\]

Meanwhile, we consider convex loss functions and the comparator sequence are chosen by an oblivious adversary.
Algorithm 1 Bandit Gradient Descent (BGD)

**Input:** Time horizon \(T\); perturbation parameter \(\delta\); shrinkage parameter \(\alpha\); step size \(\eta\).

1: Let \(y_1 = 0\)

2: for \(t = 1\) to \(T\) do

3: Select a unit vector \(s_t\) uniformly at random

   {Case 1. One-Point Feedback Model}

4: Submit \(x_t = y_t + \delta s_t\)

5: Receive \(f_t(x_t)\) as the feedback

6: Calculate the gradient estimator \(\tilde{g}_t = \frac{d}{\delta} f_t(x_t) \cdot s_t\)

7: \(y_{t+1} = \text{Proj}_\left[\left(1 - \alpha\right)\mathcal{X}\right] [y_t - \eta \tilde{g}_t]\)

   {Case 2. Two-Point Feedback Model}

8: Submit \(x^{(1)}_t = y_t + \delta s_t\) and \(x^{(2)}_t = y_t - \delta s_t\)

9: Receive \(f_t(x^{(1)}_t)\) and \(f_t(x^{(2)}_t)\) as the feedback

10: Calculate the gradient estimator \(\tilde{g}_t = \frac{d}{\delta} \left( f_t(x^{(1)}_t) - f_t(x^{(2)}_t) \right) \cdot s_t\)

11: \(y_{t+1} = \text{Proj}_\left[\left(1 - \alpha\right)\mathcal{X}\right] [y_t - \eta \tilde{g}_t]\)

12: end for

3.2 Bandit Gradient Descent: Algorithm and Analysis

In this part, we present the bandit gradient descent algorithm that developed for the BCO problems (Flaxman et al., 2005; Agarwal et al., 2010), and provide the first analysis of their universal dynamic regret.

We start from the online gradient descent (OGD) that developed in the full-information setting (Zinkevich, 2003). OGD begins with any \(x_1 \in \mathcal{X}\), and performs the following updating,

\[
x_{t+1} = \text{Proj}_\mathcal{X} \left[ x_t - \eta \nabla f_t(x_t) \right]
\]

where \(\eta > 0\) is the step size and \(\text{Proj}_\mathcal{X} \left[ \cdot \right]\) denotes the projection onto the nearest point in \(\mathcal{X}\).

The key challenge of BCO problems is the lack of gradients. Therefore, Flaxman et al. (2005) and Agarwal et al. (2010) propose to replace \(\nabla f_t(x_t)\) in (8) with a gradient estimator \(\tilde{g}_t\), obtained by evaluating the function at one (in the one-point feedback model) or two random points (in the two-point feedback model) around \(x_t\). Details of the gradient estimator for each model will be presented later. We unify their algorithms in Algorithm 1, called Bandit Gradient Descent (BGD). Notice that in lines 8 and 14 of the algorithm, the projection of \(y_{t+1}\) is on a slightly smaller set \((1 - \alpha)\mathcal{X}\) instead of \(\mathcal{X}\), to ensure that the final decision \(x_{t+1}\) lies in \(\mathcal{X}\).

In the following, we analyze the universal dynamic regret for each model.

**One-Point Feedback Model.** Flaxman et al. (2005) propose the following gradient estimator,

\[
\tilde{g}_t = \frac{d}{\delta} f_t(y_t + \delta s_t) \cdot s_t
\]

where \(s_t\) is a unit vector selected uniformly at random and \(\delta > 0\) is the perturbation parameter. Then, the following lemma (Flaxman et al., 2005, Lemma 2.1) guarantees that (9) is an unbiased gradient estimator of the smoothed version of the loss function \(f_t\).
Lemma 1. For any convex (but not necessarily differentiable) function \( f : \mathcal{X} \mapsto \mathbb{R} \), define its smoothed version \( \hat{f}(x) = \mathbb{E}_{v \in \mathbb{B}}[f(x + \delta v)] \). Then, for any \( \delta > 0 \), we have
\[
\mathbb{E}_{s \in \mathbb{S}}[f(x + \delta s) \cdot s] = \frac{\delta}{d} \nabla \hat{f}(x)
\]
where \( \mathbb{S} \) is the unit sphere centered around the origin, namely, \( \mathbb{S} = \{ x \in \mathbb{R}^d \mid \|x\|_2 = 1 \} \).

Therefore, we can adopt \( \tilde{g}_t \) to perform the online gradient descent in (8). The main procedures are summarized in the case 1 (line 4-7) of Algorithm 1. We have the following theorem regarding its universal dynamic regret.

Theorem 1. Under Assumptions 1, 2, 3 and 4, for any \( \delta > 0 \), \( \eta > 0 \), and \( \alpha = \frac{\delta}{\eta} \), the expected dynamic regret of BGD(T, \( \delta, \alpha, \eta \)) for the one-point feedback model satisfies
\[
\mathbb{E} \left[ \sum_{t=1}^{T} f_t(x_t) \right] - \sum_{t=1}^{T} f_t(u_t) \leq \frac{7R^2 + 4RT}{4\eta} + \frac{\eta \delta^2 C^2 T}{2\delta^2} + (3L + \frac{LR}{r}) \delta T,
\]
for any feasible comparator sequence \( u_1, \ldots, u_T \in \mathcal{X} \).

Remark 1. By setting \( \eta = (7R^2 + P_T)/T \) and \( \delta = \eta^{1/3} \), we obtain an \( O(T^{3/4}/(1 + P_T)^{1/2}) \) dynamic regret. However, such a configuration requires prior knowledge of \( P_T \), which is generally unavailable. We will develop a parameter-free algorithm to eliminate the undesired dependence later.

Two-Point Feedback Model. In this setup, the player is allowed to query two points, \( x_t^{(1)} = y_t + \delta s_t \) and \( x_t^{(2)} = y_t - \delta s_t \). Then, the function values \( f_t(x_t^{(1)}) \) and \( f_t(x_t^{(2)}) \) are revealed as the feedback. We use the following gradient estimator (Agarwal et al., 2010; Shamir, 2017),
\[
\tilde{g}_t = \frac{d}{2\delta^2} (f_t(y_t + \delta s_t) - f_t(y_t - \delta s_t)) \cdot s_t.
\]

The major limitation of the one-point gradient estimator (9) is that it has a potentially large magnitude, proportional to the \( 1/\delta \) which is usually quite large since the perturbation parameter \( \delta \) is typically small. This is avoided in the two-point gradient estimator (12), whose magnitude is upper bounded by \( Ld \) (shown in (24) in Section 5.2), independent of the perturbation parameter \( \delta \). This advantage leads to the improvement in the dynamic regret (also static regret).

Theorem 2. Under Assumptions 1, 2, 3 and 4, for any \( \delta > 0 \), \( \eta > 0 \), and \( \alpha = \frac{\delta}{\eta} \), the expected dynamic regret of BGD(T, \( \delta, \alpha, \eta \)) for the two-point feedback model satisfies
\[
\mathbb{E} \left[ \sum_{t=1}^{T} \frac{1}{2} (f_t(x_t^{(1)}) + f_t(x_t^{(2)})) \right] - \sum_{t=1}^{T} f_t(u_t) \leq \frac{7R^2 + 4RT}{4\eta} + \frac{\eta \delta^2 L^2 d^2}{2} + (3L + \frac{LR}{r}) \delta T,
\]
for any feasible comparator sequence \( u_1, \ldots, u_T \in \mathcal{X} \).

Remark 2. By setting \( \eta = \sqrt{(7R^2 + 4RT)/(2L^2 d^2 T)} \) and \( \delta = 1/\sqrt{T} \), BGD algorithm achieves an \( O(T^{1/2}/(1 + P_T)^{1/2}) \) dynamic regret. However, the parameter configuration also has an unpleasant dependence on the unknown quantity \( P_T \), which will be removed in the next part.
3.3 Parameter-Free Algorithm

In Theorems 1 and 2, the optimal parameter configuration requires prior knowledge of the path-length $P_T$, which is unfortunately unknown. In this part, we will design a parameter-free algorithm that supports any possible value of $P_T$.

We remark that $P_T$ remains unknown even after all iterations, because the comparator sequence of the universal dynamic regret can be chosen arbitrarily in the feasible set. Therefore, the doubling trick (Cesa-Bianchi et al., 1997) is not applicable even in the full-information setting. To address this issue, we design a parameter-free algorithm by grid searching the optimal parameter inspired by van Erven and Koolen (2016). In the following, we present descriptions of the algorithm for the one-point feedback model, and defer that for the two-point one in the analysis.

From the analysis of Theorem 1, we know that the optimal step size $\eta^*$ depends on the unknown path-length. The basic idea of removing this unpleasant dependence is as follows. We first construct the pool of candidate step sizes $\mathcal{H}$ according to

$$\mathcal{H} = \left\{ \eta_i = \frac{2^{i-1}}{dC} \sqrt{\frac{7R}{2}} \mid i = 1, \ldots, N \right\}$$

(14)

with $N = \left\lceil \frac{1}{2} \log_2 \left( 1 + \frac{2T}{\sqrt{R}} \right) \right\rceil + 1$, to make sure that $\eta^*$ provably lies in the range of $[\eta_1, \eta_N]$. Therefore, there exists an index $k \in [N]$ such that $\eta_k \leq \eta^* \leq \eta_{k+1} = 2\eta_k$. Namely, there is a step size that is not optimal but sufficiently close to $\eta^*$. We then instantiate $N$ expert-algorithms, where the $i$-th expert is a BGD algorithm with $\eta_i$ and $\delta = T^{-1/4}$ and returns its prediction. Finally, we adopt an expert-tracking algorithm as the meta-algorithm to combine predictions as the final decision. Owing to nice theoretical guarantees of the meta-algorithm, the dynamic regret of final decisions is comparable to that of the best expert, i.e., the expert-algorithm with the near-optimal step size.

However, the above grid search strategy requires to run multiple expert-algorithms with different parameters simultaneously, which is prohibited in the bandit setting, because it is only allowed to query the function value once in each iteration. To overcome the difficulty, instead of using the original loss function, we will adopt the following surrogate loss function $\ell_t : (1 - \alpha)\mathcal{X} \mapsto \mathbb{R}$ in expert-algorithms,

$$\ell_t(y) = \langle \tilde{g}_t, y - y_t \rangle.$$  

(15)

Define by $h_t(y) = \tilde{f}_t(y) + y : \xi_t$, where $\xi_t = \tilde{g}_t - \nabla \tilde{f}_t(y_t)$. First, we know that $\mathbb{E} [\tilde{f}_t(y)] = \mathbb{E} [h_t(y)]$ for any fixed $y \in (1 - \alpha)\mathcal{X}$ (see the proof in Appendix A.2, (44)), so the surrogate loss (15) can be regarded as a linearization of smoothed function $\tilde{f}_t$ on the point $y_t$ (in expectation). Besides, we have the following two observations.

**Observation 1.** For any $y \in (1 - \alpha)\mathcal{X}$, we have $\nabla \ell_t(y) = \nabla h_t(y_t) = \tilde{g}_t$.

**Observation 2.** For any $v \in (1 - \alpha)\mathcal{X}$, we have $h_t(y_t) - h_t(v) \leq \langle \nabla h_t(y_t), y_t - v \rangle = -\ell_t(v)$.

The above two observations are simple yet quite useful, and they together make the grid search doable in BCO. Observation 1 enables us to query the function $f_t$ only once and then broadcast the constructed gradient estimator to all the expert-algorithms, in which
**Algorithm 2** PARAFree-BGD: Meta-Algorithm

**Input:** The pool of candidate step sizes $\mathcal{H}$; the step size of the meta-algorithm $\epsilon$.

1. Invoke expert-algorithm (16) with different step sizes simultaneously
2. Initialize the weight of each expert $i \in [N]$, $w_i^1 = \frac{1}{n(1+1)}(1 + \frac{1}{N})$
3. for $t = 1$ to $T$ do
   4. Receive $y_{i}^t$ from each expert $i \in [N]$ and compute $y_t = \sum_{i \in [N]} w_i^t y_{i}^t$
   5. Submit $x_t = y_t + \delta_t$ and incur an instantaneous loss $f_t(x_t)$
   6. Compute the gradient estimator $\tilde{g}_t$ according to (9)
   7. Define the surrogate loss function $\ell_t(\cdot)$ as (15)
   8. Update the weight of expert $i$ by $w_{i+1}^t = w_i^t \exp(-\epsilon \ell_t(y_{i}^t))/\sum_{i \in [N]} w_i^t \exp(-\epsilon \ell_t(y_{i}^t))$
   9. Send the gradient estimator $\tilde{g}_t$ to each expert
4. end for

$\tilde{g}_t$ is the exact gradient of the surrogate loss function. So each expert is able to perform deterministic online gradient descent. Observation 2 guarantees that the expected dynamic regret of smoothed functions $f_t$’s is upper bounded by that of the surrogate loss $\ell_t$’s.

**Remark 3.** The surrogate loss technique is originally proposed by van Erven and Koolen (2016) for the full-information setting to reduce the number of gradient evaluations. To the best of our knowledge, this is the first time to apply the surrogate loss technique in the bandit setup, with the aim to maintain multiple learning rates in bandit convex optimization. Moreover, the linearization is over the original function $f_t$ in their work, whereas ours is over the smoothed function.

We now describe the meta-algorithm and expert-algorithm.

**Meta-Algorithm.** We use the exponentially weighted average forecaster algorithm (Cesa-Bianchi and Lugosi, 2006) as the meta-algorithm, whose input is the pool of step sizes $\mathcal{H}$ in (14) and its own step size $\epsilon$. Notably, it does not require prior knowledge of the unknown path-length. Algorithm 2 summarizes detailed procedures.

**Expert-Algorithm.** The expert-algorithms build upon the surrogate loss defined in (15). For the expert $i \in [N]$, its update procedure is as follows,

$$y_{i+1}^t = \text{Proj}_{(1-\alpha)X}[y_{i}^t - \eta \nabla \ell_t(y_{i}^t)] = \text{Proj}_{(1-\alpha)X}[y_{i}^t - \eta \tilde{g}_t]. \quad (16)$$

From the above update rule, we note that each expert essentially performs deterministic online gradient descent over the surrogate loss.

Additionally, we remark that another possible approach for developing parameter-free bandit algorithms is Bandits over Bandits (Gur et al., 2014; Agarwal et al., 2017; Cheung et al., 2019), however it suffers the limitation that base algorithms have access to a much smaller amount of data. By contrast, our approach is essentially Hedge over full information. More specifically, the surrogate loss has nice properties such that we can query the function only once and then broadcast to all the experts to perform full-information deterministic OGD over the surrogate loss, and they update for all the $T$ rounds.

The expert-algorithm (16) in conjunction with Algorithm 2 gives PARAFree-BGD (short for Parameter-Free Bandit Gradient Descent). The following theorem states the dynamic regret of our approach.
Theorem 3. Under Assumptions 1, 2, 3 and 4, with a proper setting of the pool of candidate step sizes $H$ and the step size $\epsilon$, ParaFree-BGD enjoys the following expected dynamic regret,

- One-Point Feedback Model: $O(T^{\frac{3}{4}}(1 + P_T)^{\frac{1}{2}})$;
- Two-Point Feedback Model: $O(T^{\frac{3}{4}}(1 + P_T)^{\frac{1}{2}})$.

The above results hold universally for any feasible comparator sequence $u_1, \ldots, u_T \in X$.

Remark 4. Theorem 3 demonstrates that the dynamic regret can be improved from $O(T^{\frac{3}{4}}(1 + P_T)^{\frac{1}{2}})$ to $O(T^{\frac{1}{2}}(1 + P_T)^{\frac{1}{2}})$ when it is allowed to query two points in each iteration. The attained dynamic regret (though in expectation) of BCO with two-point feedback, surprisingly, is in the same order with that of the full-information setting (Zhang et al., 2018a). This extends the claim argued by Agarwal et al. (2010) knowing the value of each loss function at two points is almost as useful as knowing the value of each function everywhere to dynamic regret analysis. Furthermore, we will show that the obtained dynamic regret for the two-point feedback model is minimax optimal in the next section.

4. Lower Bound and Extension

In this section, we investigate the attainable dynamic regret for BCO problems, and then extend our analysis to adaptive regret, another measure for online learning in non-stationary environments.

4.1 Lower Bound

We have the following minimax lower bound of universal dynamic regret for BCO problems.

Theorem 4. For any $\tau \in [0, 2RP]$, there exists a comparator sequence $u_1, \ldots, u_T$ satisfying Assumption 1 whose path-length $P_T$ is less than $\tau$, and a sequence of functions satisfying Assumption 3, such that for any algorithm designed for BCO with one-/two-point feedback who returns $x_1, \ldots, x_T$,

$$\sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(u_t) \geq C \cdot dG \sqrt{(R^2 + R\tau)T},$$

where $C$ is a positive constant independent of $T$.

Remark 5. In fact, the lower bound holds even all the functions $f_t$’s are strongly convex and smooth in BCO with one-point feedback. This is to be contrasted with that in the full-information setting. The reason is that Shamir (2013) demonstrates that the minimax static regret of BCO with one-point feedback can neither benefit from the strongly convexity nor smoothness. This implies the inherent difficulty of learning with bandit feedback.

1. Note that (14) gives the configuration for the one-point feedback model, and the setup of $H$ for the two-point feedback model is specified in (33).
4.2 Extension to Adaptive Regret

In this part, we investigate the adaptive regret. Following Hazan and Seshadhri (2009), we define the expected adaptive regret for BCO problems as

\[ E[A\text{-Regret}_T] = \max_{q,s \subseteq [T]} \left( \mathbb{E} \left[ \sum_{t=q}^{s} f_t(x_t) \right] - \min_{x \in \mathcal{X}} \sum_{t=q}^{s} f_t(x) \right). \] (18)

To minimize (18), we propose an algorithm called Minimizing Adaptive regret in Bandit Convex Optimization (MABCO). Our algorithm follows a similar framework used in the Coin Betting for Changing Environment (CBCE) algorithm (Jun et al., 2017), which achieves the state-of-the-art adaptive regret in the full-information setting. However, we note that a direct reduction of CBCE algorithm to the bandit setting requires to query the loss function multiple times in each iteration, which is invalid in the bandit feedback model. To address this difficulty, similar to PARA-FREE-BGD we introduce a new surrogate loss function, which can be constructed by only using the one-point or two-point function values. We provide algorithmic details and proofs of theoretical results in the appendix.

**Theorem 5.** With a proper setting of the surrogate loss functions and parameters, the proposed MABCO algorithm enjoys the following expected adaptive regret,

- **One-Point Feedback Model:** $O(T^{3/4}(\log T)^{1/4})$;
- **Two-Point Feedback Model:** $O(T^{1/2}(\log T)^{3/2})$.

**Remark 6.** Note that we cannot hope for an adaptive regret that is better than the static regret. The adaptive regret in Theorem 5 matches $O(T^{3/4})$ and $O(T^{1/2})$ static regrets for the one-point (Flaxman et al., 2005) and two-point (Agarwal et al., 2010) feedback models, up to logarithmic factors.

5. Analysis

In this section, we present the analysis of our theoretical results.

5.1 Proof of Theorem 1

Before presenting a rigorous proof of the theorem, we first highlight the main idea of the argument as follows.

1. Guarantee that $\forall t \in [T]$, $x_t$ is a feasible point in $\mathcal{X}$, because the projection in Algorithm 1 is over $y_t$ instead of $x_t$.
2. Invoke the dynamic regret guarantee for randomized OGD (Theorem 7) over the smoothed function $\hat{f}_t$ with a certain comparator sequence.
3. Check the gap between the dynamic regret of $\hat{f}_1, \ldots, \hat{f}_T$ and that of $f_1, \ldots, f_T$.

We now present the proof of Theorem 1.

---

2. We note that, in the full-information setting, a stronger version of adaptive regret named *strongly adaptive regret* is introduced by Daniely et al. (2015). However, they prove that it is impossible to achieve meaningful strongly adaptive regret in bandit settings, so we focus on the notion defined by Hazan and Seshadhri (2009).
Proof. Notice that the projection in Algorithm 1 only guarantees that \( y_t \) is in a slightly smaller set \((1 - \alpha) \mathcal{X}\), so we first need to prove that \( \forall t \in [T], x_t \) is a feasible point in \( \mathcal{X} \). This is convinced by Lemma 3, since we know that \( \delta \leq \alpha r \) from the parameter setting \((\alpha = \frac{\hat{\delta}}{r})\).

The expected dynamic regret can be decomposed as follows.

\[
\mathbb{E} \left[ \sum_{t=1}^{T} f_t(x_t) \right] - \sum_{t=1}^{T} f_t(u_t) = \mathbb{E} \left[ \sum_{t=1}^{T} \tilde{f}_t(y_t) \right] - \sum_{t=1}^{T} \tilde{f}_t(v_t) + \mathbb{E} \left[ \sum_{t=1}^{T} \tilde{f}_t(x_t) - \tilde{f}_t(y_t) \right] + \sum_{t=1}^{T} \left( \tilde{f}_t(v_t) - \tilde{f}_t(u_t) \right),
\]

where \( v_1, \ldots, v_T \) is the scaled comparator sequence defined by \( v_t = (1 - \alpha) u_t \), we know that \( v_t \in (1 - \alpha) \mathcal{X} \) for all \( t \in [T] \).

So we proceed to bound the above three terms separately.

First, we bound the term (a), which is essentially the dynamic regret of the smoothed functions. In the one-point feedback model, the gradient estimator is set as \((9)\), from Lemma 1, we know that \( \mathbb{E} [\tilde{g}_t] = \nabla f_t(y_t) \). Therefore, the procedure

\[
y_{t+1} = \text{Proj}_{(1 - \alpha) \mathcal{X}} [y_t - \eta \tilde{g}_t]
\]

is essentially the randomized online gradient descent over the smoothed function \( \tilde{f}_t \). So we first establish the dynamic regret for \( \tilde{f}_t \) by using Theorem 7.

\[
\text{term(a)} = \mathbb{E} \left[ \sum_{t=1}^{T} \tilde{f}_t(y_t) \right] - \sum_{t=1}^{T} \tilde{f}_t(v_t) \leq \frac{7D^2 + D \tilde{P}_T}{4\eta} + \frac{\eta \tilde{G}^2 T}{2} \leq \frac{7R^2 + R \tilde{P}_T}{4\eta} + \frac{\eta d C^2 T}{2\delta^2},
\]

(19)

where \( \tilde{P}_T = \sum_{t=2}^{T} \| v_{t-1} - v_t \|_2 = (1 - \alpha) P_T, \tilde{D} = (1 - \alpha) R \leq R \) and \( \tilde{G} = d C / \delta \) by noticing

\[
\| \tilde{g}_t \|_2 \leq \left\| \frac{d}{\delta} f_t(y_t + \delta s_t) s_t \right\|_2 \leq d C / \delta, \forall t \in [T].
\]

(20)

Now, it suffices to further bound term (b) and term (c). By Assumption 4 and Lemma 4, we have

\[
\text{term(b)} = \sum_{t=1}^{T} f_t(x_t) - \tilde{f}_t(y_t) = \sum_{t=1}^{T} f_t(x_t) - f_t(y_t) + f_t(y_t) - \tilde{f}_t(y_t) \leq 2L \delta T.
\]

(21)

Moreover, term (c) can be bounded by

\[
\text{term(c)} \leq \sum_{t=1}^{T} | \tilde{f}_t(v_t) - f_t(u_t) | \leq \sum_{t=1}^{T} | \tilde{f}_t(v_t) - f_t(v_t) | + | f_t(v_t) - f_t(u_t) | \leq \sum_{t=1}^{T} (L \delta + L \| v_t - u_t \|_2) \leq \sum_{t=1}^{T} (L \delta + LoR) = (L + \frac{LR}{r}) \delta T
\]

(22)
where the second inequality holds due to Lemma 4 and the Lipschitz continuity of functions.

By combining upper bounds of three terms in (19), (21) and (22), we obtain the dynamic regret of original function $f_t$ over the comparator sequence of $\{u_t\}_{t=1}^T$,

$$
\mathbb{E} \left[ \sum_{t=1}^T f_t(x_t) \right] - \sum_{t=1}^T f_t(u_t) \\
\leq \frac{7R^2 + RP_T}{4\eta} + \frac{\eta d^2 C^2 T}{2\delta^2} + 2L\delta T + (L\delta + L\alpha R)T \quad \text{(due to (19), (21) and (22) )}
$$

$$
\leq \frac{7R^2 + RP_T}{4\eta} + \frac{\eta d^2 C^2 T}{2\delta^2} + (3L + \frac{LR}{r})\delta T \\
= O \left( \frac{1}{(1 + P_T)^\frac{1}{4}} T^\frac{3}{4} \right),
$$

where (23) holds due to the setting of $\alpha = \delta/r$, and the last equation is obtained by the AM-GM inequality via optimizing values of $\eta$ and $\delta$. The optimal parameter configuration is

$$
\begin{align*}
\delta^* &= \left( \frac{7R^2 + Pr}{T} \right)^\frac{1}{4} 2^{-\frac{1}{4} \left( dC/(3L + LR/r) \right)} \frac{1}{2}, \\
\eta^* &= \left( \frac{7R^2 + Pr}{T} \right)^\frac{3}{4} 2^{-\frac{3}{4} \left( dC(3L + LR/r) \right)} \frac{1}{2}.
\end{align*}
$$

5.2 Proof of Theorem 2

Proof. In the two-point feedback model, the gradient estimator is

$$
\tilde{g}_t = \frac{d}{2\delta} (f_t(y_t + \delta s_t) - f_t(y_t - \delta s_t)) \cdot s_t.
$$

We can upper bound the norm of the gradient estimator as follows,

$$
\|\tilde{g}_t\|_2 = \frac{d}{2\delta} \| (f_t(y_t + \delta s_t) - f_t(y_t - \delta s_t))s_t \|_2 \\
= \frac{d}{2\delta} |f_t(y_t + \delta s_t) - f_t(y_t - \delta s_t)| \\
\leq \frac{dL}{2\delta} \| 2\delta s_t \|_2 = Ld,
$$

where in the last inequality, we utilize the Lipschitz property due to Assumption 4. Hence, $\tilde{G} = \sup_{t \in [T]} \|\tilde{g}_t\|_2 = Ld$. We remark that by contrast with that in the one-point feedback model as shown in (20), $\tilde{G}$ here is independent of the $1/\delta$, which leads to much improved regret bounds.

Meanwhile, by exploiting the Lipschitz property, we have

$$
f_t(y_t + \delta s_t) \leq f_t(y_t) + L\|\delta s_t\|_2 = f_t(y_t) + \delta L,
$$

$$
\frac{d}{2\delta} \| (f_t(y_t + \delta s_t) - f_t(y_t - \delta s_t))s_t \|_2 \\
\leq \frac{dL}{2\delta} \| 2\delta s_t \|_2 = Ld,
$$

13
and similar result holds for \( f_t(x_t - \delta s_t) \). Therefore, we can bound the expected regret as follows,

\[
\mathbb{E} \left[ \sum_{t=1}^{T} \frac{1}{2} (f_t(y_t + \delta s_t) + f_t(y_t - \delta s_t)) \right] - T \sum_{t=1}^{T} f_t(u_t)
\]

\[\leq \mathbb{E} \left[ \sum_{t=1}^{T} f_t(y_t) \right] + \delta L T - \mathbb{E} \left[ \sum_{t=1}^{T} f_t(u_t) \right]
\]

\[= \mathbb{E} \left[ \sum_{t=1}^{T} \hat{f}_t(y_t) \right] - \sum_{t=1}^{T} \hat{f}_t(v_t) + \delta L T + \mathbb{E} \left[ \sum_{t=1}^{T} f_t(y_t) - \hat{f}_t(y_t) \right] - \sum_{t=1}^{T} (f_t(u_t) - \hat{f}_t(v_t))
\]

\[\leq \frac{7R^2 + RP_T}{4 \eta} + \frac{\eta L^2 d^2}{2} T + \frac{(3L + LR r)\delta T}{r} \tag{26}
\]

\[= O \left( (1 + P_T)^{\frac{1}{2}} T^{\frac{1}{2}} \right) \tag{27}
\]

The core difference between the analysis lies in the second term in (26), which is independent of \( 1/\delta \), and thus is much smaller than that in (23). This owes to the benefit of two-point feedback in each iteration. Notice that (27) is obtained by setting \( \delta = 1/\sqrt{T} \) and \( \eta = \sqrt{\frac{7R^2 + RP_T}{2L^2 d^2}} \).

5.3 Proof of Theorem 3

We start with the analysis of the meta-algorithm, and then present the proof of Theorem 3.

Note that the meta-algorithm is essentially the exponentially weighted average forecaster. Therefore, we have the following theorem, showing that the cumulative surrogate loss of the meta-algorithm is comparable to that of each expert-algorithm.

Lemma 2. For any step size \( \epsilon > 0 \), we have

\[
\sum_{t=1}^{T} \ell_t(y_t) - \min_{i \in [N]} \left( \sum_{t=1}^{T} \ell_t(y^i_t) + \frac{1}{\epsilon} \ln \frac{1}{w^i_t} \right) \leq 2\epsilon T \tilde{G}^2 R^2.
\]

Therefore, by setting \( \epsilon = \sqrt{1/(2T \tilde{G}^2 R^2)} \) to minimize the above upper bound, we obtain

\[
\sum_{t=1}^{T} \ell_t(y_t) - \sum_{t=1}^{T} \ell_t(y^i_t) \leq \tilde{G} R \sqrt{2T} \left( 1 + \ln \frac{1}{w^i_1} \right). \tag{28}
\]

for any index \( i \in [N] \)

Proof. By the definition of the surrogate loss function \( \ell_t \), we have

\[
|\ell_t(y)| = |\langle \tilde{g}_t, y - y_t \rangle| \leq \| \tilde{g}_t \|_2 \| y - y_t \|_2 \leq 2\tilde{G} R, \forall y \in (1 - \alpha)X, t \in [T],
\]

where \( \tilde{G} \) is the upper bound of the norm of the gradient estimator. So we complete the proof by a simple application of the standard result for exponentially weighted average forecaster with a non-uniform initialization (Cesa-Bianchi and Lugosi, 2006, Exercise 2.5).
5.3.1 One-Point Feedback Model

Proof. In the one-point feedback model, the pool of candidate step size $\mathcal{H}$ is set as,

$$\mathcal{H} = \left\{ \eta_i = \frac{2^{i-1}}{dCT^\frac{1}{4}} \sqrt{\frac{7R^2}{2}} \mid i = 1, \ldots, N \right\},$$

where $N = \left\lfloor \frac{1}{2} \log_2 (1 + \frac{2T}{4}) \right\rfloor + 1$. And the optimal step size is $\eta^* = \frac{1}{dCT^\frac{1}{4}} \sqrt{\frac{7R^2 + 2RP_T}{2}}$.

Since the path-length $P_T$ is bounded by $2RT$ due to Assumption 1, the optimal step size $\eta^*$ though is unknown but will provably lie in the pool $\mathcal{H}$. Therefore, there exists an index $k \in [N]$ such that $\eta_k \leq \eta^* \leq \eta_{k+1}$, and meanwhile,

$$k \leq \left\lfloor \frac{1}{2} \log_2 \left( R + \frac{P_T}{T} \right) \right\rfloor + 1. \quad (29)$$

We thus apply Lemma 2 with setting the index as $k$, and obtain that

$$\sum_{t=1}^T \ell_t(y_t) - \sum_{t=1}^T \ell_t(v_t) \leq \tilde{G}R\sqrt{2T} \left( 1 + \ln \frac{1}{w_t^k} \right) \leq \frac{dCR}{\delta} \sqrt{2T} (1 + 2\ln(k + 1)), \quad (30)$$

where the last inequality holds due to $w_t^k = \frac{C}{(k+1)^2} \geq \frac{C}{(k+1)^2}$ and the definition of $\tilde{G}$ in (20).

On the other hand, in each iteration, we first collect decisions $y_t^k$ from each expert, and then obtain the final decision $y_t$, based on which the surrogate loss function $\ell_t(\cdot)$ is constructed. So each expert essentially performs deterministic OGD over the surrogate loss, so Theorem 6 applies.

$$\sum_{t=1}^T \ell_t(y_t^k) - \sum_{t=1}^T \ell_t(v_t) \leq \frac{7R^2 + 2RP_T}{4\eta_k} + \frac{\eta_k \tilde{G}^2 T}{2}. \quad (31)$$

Note that Theorem 6 for deterministic OGD holds for any sequence of loss functions, no matter whether they are adaptive or not. Meanwhile, since $\eta_k \leq \eta^* \leq \eta_{k+1} = 2\eta_k$, we further have

$$\sum_{t=1}^T \ell_t(y_t^k) - \sum_{t=1}^T \ell_t(v_t) \leq \frac{7R^2 + 2RP_T}{4\eta_k} + \frac{\eta_k \tilde{G}^2 T}{2} \leq \frac{7R^2 + 2RP_T}{2\eta^*} + \frac{\eta^* d^2 C^2 T}{28} \quad (32)$$

where the last equation holds by plugging the parameter configuration of $\eta^*$ and $\delta = 1/T^{\frac{3}{4}}$.

Therefore, by combining (30) and (32), we have

$$\sum_{t=1}^T h_t(y_t) - \sum_{t=1}^T h_t(v_t) \leq \sum_{t=1}^T \langle \nabla h_t(y_t), y_t - v_t \rangle = \sum_{t=1}^T \ell_t(y_t) - \sum_{t=1}^T \ell_t(v_t) = \sum_{t=1}^T \ell_t(y_t) - \sum_{t=1}^T \ell_t(y_t^k) + \sum_{t=1}^T \ell_t(y_t^k) - \sum_{t=1}^T \ell_t(v_t)$$
By making use of the analysis in Section 5.1, we finally bound the expected dynamic regret of the original loss functions as follows,

\[
E \left[ \sum_{t=1}^{T} f_t(x_t) \right] - \sum_{t=1}^{T} f_t(u_t)
\]

\[
= \mathbb{E} \left[ \sum_{t=1}^{T} \hat{f}_t(y_t) - \sum_{t=1}^{T} \hat{f}_t(v_t) \right] + \mathbb{E} \left[ \sum_{t=1}^{T} f_t(x_t) - \hat{f}_t(y_t) \right] + \mathbb{E} \left[ \sum_{t=1}^{T} (\hat{f}_t(v_t) - f_t(u_t)) \right]
\]

\[
\leq \sqrt{2dCRT^4} \left( 1 + 2 \ln(k + 1) \right) + \frac{3\sqrt{2}}{4} dCT^\frac{3}{2} \sqrt{7R^2 + RP_T} + (3L + \frac{LR}{r})T^\frac{3}{4}
\]

\[
= O \left( T^\frac{3}{4} (1 + P_T)^\frac{3}{4} \right).
\]

This completes the proof. \qed

5.3.2 Two-Point Feedback Model

Proof. In the two-point feedback model, the pool of candidate step size \( \mathcal{H} \) is set as,

\[
\mathcal{H} = \left\{ \eta_i = 2^{i-1} \sqrt{\frac{7R^2}{2L^2d^2T}} \mid i = 1, \ldots, N \right\}, \tag{33}
\]

where \( N = \left\lceil \frac{1}{2} \log_2 \left( 1 + \frac{2T}{R} \right) \right\rceil + 1 \), and the optimal step size is \( \eta^* = \sqrt{\frac{7R^2 + RP_T}{2L^2d^2T}} \).

Since the path-length \( P_T \) is bounded by \( 2RT \) due to Assumption 1, the optimal step size \( \eta^* \) though is unknown but will provably lie in the pool \( \mathcal{H} \). Therefore, there exists an index \( k \in [N] \) such that \( \eta_k \leq \eta^* \leq \eta_{k+1} \), and meanwhile,

\[
k \leq \left\lceil \frac{1}{2} \log_2 \left( 1 + \frac{P_T}{7R} \right) \right\rceil + 1. \tag{34}
\]

Similar to the proof in Section 5.3.1, by substituting \( \tilde{G} = Ld \) as shown in (24), we have

\[
\sum_{t=1}^{T} \ell_t(y_t) - \sum_{t=1}^{T} \ell_t(y_t^k) \leq \tilde{G}\sqrt{2T} \left( 1 + \ln \frac{1}{w_t^k} \right) \leq LdR\sqrt{2T} \left( 1 + 2 \ln(k + 1) \right). \tag{35}
\]

Meanwhile, similar to the argument in the one-point feedback model, namely, (31), we can apply Theorem 6 and obtain

\[
\sum_{t=1}^{T} \ell_t(y_t^k) - \sum_{t=1}^{T} \ell_t(v_t) \leq \frac{7R^2 + RP_T}{4\eta_k} + \frac{\eta_k\tilde{G}^2T}{2} \leq \frac{7R^2 + RP_T}{2\eta^*} + \frac{\eta^*L^2d^2T}{2}
\]

\[
= \frac{7R^2 + RP_T}{2\eta^*} + \frac{\eta^*d^2C^2T^{3/2}}{2} = \frac{3\sqrt{2}}{4} Ld\sqrt{T(7R^2 + RP_T)}.
\]
Therefore, by combining (35) and (36), we have

\[
\sum_{t=1}^{T} h_t(y_t) - \sum_{t=1}^{T} h_t(v_t) \leq \sum_{t=1}^{T} \langle \nabla h_t(y_t), y_t - v_t \rangle = \sum_{t=1}^{T} \ell_t(y_t) - \sum_{t=1}^{T} \ell_t(v_t)
\]

\[
= \sum_{t=1}^{T} \ell_t(y_t) - \sum_{t=1}^{T} \ell_t(y_t^k) + \sum_{t=1}^{T} \ell_t(y_t) - \sum_{t=1}^{T} \ell_t(v_t)
\]

\[
\leq LdR\sqrt{2T}\left(1 + 2\ln(k+1)\right) + \frac{3\sqrt{2}}{4} Ld\sqrt{T(7R^2 + RP_T)}
\]

By making use of the analysis in Section 5.1, we finally bound the expected dynamic regret of the original loss functions as follows,

\[
E\left[\sum_{t=1}^{T} f_t(x_t)\right] - \sum_{t=1}^{T} f_t(u_t)
\]

\[
= E\left[\sum_{t=1}^{T} \hat{f}_t(y_t) - \sum_{t=1}^{T} \hat{f}_t(v_t)\right] + E\left[\sum_{t=1}^{T} f_t(x_t) - \hat{f}_t(y_t)\right] + E\left[\sum_{t=1}^{T} \left(\hat{f}_t(v_t) - f_t(u_t)\right)\right]
\]

\[
\leq E\left[\sum_{t=1}^{T} h_t(y_t) - \sum_{t=1}^{T} h_t(v_t)\right] + 2L\delta T + (L\delta + L\alpha R)T
\]

\[
\leq LdR\sqrt{2T}(1 + 2\ln(k+1)) + \frac{3\sqrt{2}}{4} Ld\sqrt{T(7R^2 + RP_T)} + (3L + \frac{LR}{r})T^{\frac{3}{2}}
\]

\[
= O\left(T^{\frac{1}{2}}(1 + P_T)^{\frac{1}{2}}\right)
\]

which completes the proof. \qed

5.4 Proof of Theorem 4

We present the proof of the minimax lower bound of the universal dynamic regret for bandit convex optimization problems that established in Theorem 4.

Proof. For a given \(\tau \in [2RP]\), we will first construct a specific comparator sequence, which is a piecewise stationary comparator sequence whose path-length is less than \(\tau\). Then, we can split the dynamic regret into several pieces, where in each piece the comparator sequence is fixed. We thus appeal to the lower bound established for static regret of bandit convex optimization (Dani et al., 2008a; Shamir, 2013) in each piece and sum over all to obtain the final result.

Follow the seminal work of Abernethy et al. (2008b) that provides the minimax lower bound for static regret, we adopt the notation of \(R_T(\mathcal{X}, \mathcal{F}, \tau)\) to denote the minimax dynamic regret, defined by

\[
R_T(\mathcal{X}, \mathcal{F}, \tau) = \inf_{x_t \in \mathcal{X}} \sup_{f_t \in \mathcal{F}} \inf_{x_t \in \mathcal{X}} \sup_{f_t \in \mathcal{F}} \left(\sum_{t=1}^{T} f_t(x_t) - \min_{(u_1, \ldots, u_T) \in \mathcal{U}(\tau)} \sum_{t=1}^{T} f_t(u_t)\right)
\]

where \(\mathcal{U}(\tau) = \{(u_1, \ldots, u_T) \mid \forall t \in [T], u_t \in \mathcal{X}, \text{ and } P_T = \sum_{t=1}^{T} \|u_{t-1} - u_t\|_2 \leq \tau\}\) denote the set of feasible comparator sequences with path-length \(P_T\) less than \(\tau\).
We first consider the case when $\tau \leq 2R$. Then, we can utilize the established lower bound of the static regret for BCO problems (Dani et al., 2008a; Shamir, 2013) as a natural lower bound of the dynamic regret,

$$R_T(X, F, \tau) \geq C_1 \cdot dRG\sqrt{T} = \frac{\sqrt{2}}{2}C_1 \cdot dG\sqrt{(R^2 + R^2)T} \geq C \cdot dG\sqrt{(R^2 + R\tau)T},$$

where $C = \frac{\sqrt{2}}{2}C_1$, and $C_1$ is the constant appeared in the lower bound of static regret. The last inequality holds due to the condition $\tau \leq 2R$.

We next deal with the case when $\tau \geq 2R$. The idea is to construct a special comparator sequence in $U(\tau)$, and split the whole time horizon into $K$ pieces such that the comparator sequence is fixed within each piece and only changes in the split point. Meanwhile, notice that the variation of the comparator sequence at each change point is $\tau/(K - 1)$, at most $2R$. Combining these two observations, we have

$$R_T(X, F, \tau) \geq KdRG\sqrt{|T/K|} \geq dRG\sqrt{KT} \geq dRG\sqrt{\left(\frac{\tau}{2R} + 1\right)T} \geq dG\sqrt{(R^2 + R\tau)T/2}.$$ 

This completes the proof.

6. Conclusion and Future Work

In this paper, we study the bandit convex optimization (BCO) problems in non-stationary environments. We propose a novel BCO algorithm that respectively achieves the state-of-the-art $O(T^{3/4}(1 + P_T)^{1/2})$ and $O(T^{1/2}(1 + P_T)^{1/2})$ dynamic regret for one-point and two-point feedback models. The regret bounds hold universally for any feasible comparator sequence. Meanwhile, the algorithm does not need to know prior information of the path length, which is unknown but required in previous studies. Furthermore, we demonstrate the regret bound for the two-point feedback model is minimax optimal by establishing the first lower bound for the universal dynamic regret in the bandit convex optimization setup. We also extend our analysis to the adaptive regret.

Although our dynamic regret for the two-point feedback model is proved to be optimal, many problems related to the one-point feedback model remains open. The $O(T^{3/4}(1 + P_T)^{1/2})$ dynamic regret exhibits a radical dependence on the path-length, and it will become meaningless when $P_T \geq \sqrt{T}$, though the path-length is typically small. The challenge is that the grid search technique cannot support to approximate the optimal perturbation parameter $\delta^*$ which is also dependent on $P_T$. Otherwise, we have to query the function more than once in each iteration. We will investigate a sharper bound for BCO with one-point feedback in the future. Moreover, we will consider incorporating other properties, like strong convexity and smoothness, to further enhance the results.

References

Jacob Abernethy, Elad Hazan, and Alexander Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In Proceedings of the 21st Annual Conference on Learning Theory (COLT), pages 263–274, 2008a.
Jacob D. 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 (COLT), pages 415–424, 2008b.

Alekh Agarwal, Ofer Dekel, and Lin Xiao. Optimal algorithms for online convex optimization with multi-point bandit feedback. In Proceedings of the 23rd Conference on Learning Theory (COLT), pages 28–40, 2010.

Alekh Agarwal, Haipeng Luo, Behnam Neyshabur, and Robert E. Schapire. Corralling a band of bandit algorithms. In Proceedings of the 30th Conference on Learning Theory (COLT), pages 12–38, 2017.

Peter Auer, Pratik Gajane, and Ronald Ortner. Adaptively tracking the best bandit arm with an unknown number of distribution changes. In Proceedings of the 32nd Conference On Learning Theory (COLT), volume 99, pages 138–158, 2019.

Baruch Awerbuch and Robert D Kleinberg. Adaptive routing with end-to-end feedback: Distributed learning and geometric approaches. In Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC), pages 45–53, 2004.

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

Sébastien Bubeck, Nicolò Cesa-Bianchi, and Sham M. Kakade. Towards minimax policies for online linear optimization with bandit feedback. In Proceedings of the 25th Annual Conference on Learning Theory (COLT), 2012.

Sébastien Bubeck, Yin Tat Lee, and Ronen Eldan. Kernel-based methods for bandit convex optimization. In Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC), pages 72–85, 2017.

Sébastien Bubeck, Ofer Dekel, Tomer Koren, and Yuval Peres. Bandit convex optimization: $\sqrt{T}$ regret in one dimension. In Proceedings of the 28th Conference on Learning Theory (COLT), volume 40, pages 266–278, 2015.

Nicolo Cesa-Bianchi and Gábor Lugosi. Prediction, learning, and games. Cambridge University Press, 2006.

Nicolo Cesa-Bianchi, Yoav Freund, David Haussler, David P. Helmbold, Robert E. Schapire, and Manfred K. Warmuth. How to use expert advice. Journal of the ACM, 44(3):427–485, 1997.

Tianyi Chen and Georgios B. Giannakis. Bandit convex optimization for scalable and dynamic iot management. IEEE Internet of Things Journal, 6(1):1276–1286, 2019.

Yifang Chen, Chung-Wei Lee, Haipeng Luo, and Chen-Yu Wei. A new algorithm for non-stationary contextual bandits: Efficient, optimal, and parameter-free. In Proceedings of the 32nd Conference On Learning Theory (COLT), pages 696–726, 2019.
Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Learning to optimize under non-stationarity. In Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS), pages 1079–1087, 2019.

Varsha Dani, Thomas P. Hayes, and Sham M. Kakade. Stochastic linear optimization under bandit feedback. In Proceedings of the 21st Annual Conference on Learning Theory (COLT), pages 355–366, 2008a.

Varsha Dani, Sham M Kakade, and Thomas P Hayes. The price of bandit information for online optimization. In Advances in Neural Information Processing Systems 20 (NIPS), pages 345–352, 2008b.

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

Ofer Dekel, Ronen Eldan, and Tomer Koren. Bandit smooth convex optimization: Improving the bias-variance tradeoff. In Advances in Neural Information Processing Systems 28 (NIPS), pages 2926–2934, 2015.

Abraham Flaxman, Adam Tauman Kalai, and H. Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 385–394, 2005.

Yonatan Gur, Assaf J. Zeevi, and Omar Besbes. Stochastic multi-armed-bandit problem with non-stationary rewards. In Advances in Neural Information Processing Systems 27 (NIPS), pages 199–207, 2014.

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

Elad Hazan and Kfir Y. Levy. Bandit convex optimization: Towards tight bounds. In Advances in Neural Information Processing Systems 27 (NIPS), pages 784–792, 2014.

Elad Hazan and C. Seshadhri. Efficient learning algorithms for changing environments. In Proceedings of the 26th International Conference on Machine Learning (ICML), pages 393–400, 2009.

Ali Jadbabaie, Alexander Rakhlin, Shahin Shahrampour, and Karthik Sridharan. Online optimization : Competing with dynamic comparators. In Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS), 2015.

Kwang-Sung Jun, Francesco Orabona, Stephen Wright, and Rebecca Willett. Improved strongly adaptive online learning using coin betting. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), pages 943–951, 2017.

Zohar S. Karnin and Oren Anava. Multi-armed bandits: Competing with optimal sequences. In Advances in Neural Information Processing Systems 29 (NIPS), pages 199–207, 2016.
Robert D. Kleinberg. Nearly tight bounds for the continuum-armed bandit problem. In Advances in Neural Information Processing Systems 17 (NIPS), pages 697–704, 2004.

Haipeng Luo, Chen-Yu Wei, Alekh Agarwal, and John Langford. Efficient contextual bandits in non-stationary worlds. In Proceedings of the 31st Conference On Learning Theory (COLT), pages 1739–1776, 2018.

H Brendan McMahan and Avrim Blum. Online geometric optimization in the bandit setting against an adaptive adversary. In Proceedings of the 17th Annual Conference on Learning Theory (COLT), pages 109–123, 2004.

Aryan Mokhtari, Shahin Shahrampour, Ali Jadbabaie, and Alejandro Ribeiro. Online optimization in dynamic environments: Improved regret rates for strongly convex problems. In Proceedings of the 55th IEEE Conference on Decision and Control (CDC), pages 7195–7201, 2016.

Yurii Nesterov. Random gradient-free minimization of convex functions. Technical report, Université catholique de Louvain, Center for Operations Research and Econometrics (ECORE), 2011.

Ankan Saha and Ambuj Tewari. Improved regret guarantees for online smooth convex optimization with bandit feedback. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS), pages 636–642, 2011.

Shai Shalev-Shwartz. Online learning and online convex optimization. Foundations and Trends in Machine Learning, 4(2):107–194, 2012.

Ohad Shamir. On the complexity of bandit and derivative-free stochastic convex optimization. In Proceedings of the 26th Annual Conference on Learning Theory (COLT), pages 3–24, 2013.

Ohad Shamir. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. Journal of Machine Learning Research, 18:52:1–52:11, 2017.

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

Guanghui Wang, Dakuan Zhao, and Lijun Zhang. Minimizing adaptive regret with one gradient per iteration. In Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI), pages 2762–2768, 2018.

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 (NIPS), pages 3972–3980, 2016.

Scott Yang and Mehryar Mohri. Optimistic bandit convex optimization. In Advances in Neural Information Processing Systems 29 (NIPS), pages 2289–2297, 2016.

Tianbao Yang, Lijun Zhang, Rong Jin, and Jinfeng Yi. Tracking slowly moving clairvoyant: Optimal dynamic regret of online learning with true and noisy gradient. In Proceedings of the 33rd International Conference on Machine Learning (ICML), pages 449–457, 2016.
Lijun Zhang, Tianbao Yang, Jinfeng Yi, Rong Jin, and Zhi-Hua Zhou. Improved dynamic regret for non-degeneracy functions. In *Advances in Neural Information Processing Systems 30 (NIPS)*, 2017.

Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pages 1330–1340, 2018a.

Lijun Zhang, Tianbao Yang, Rong Jin, and Zhi-Hua Zhou. Dynamic regret of strongly adaptive methods. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 5877–5886, 2018b.

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

Appendix A. Preliminaries

In this section, we present necessary preliminaries for the analysis of dynamic regret of bandit convex optimization problems.

A.1 Projection Issues

Notice that we run the algorithm on a slightly smaller set $(1-\alpha)\mathcal{X}$ rather than the original feasible set $\mathcal{X}$, where the shrinkage parameter $\alpha > 0$ is sufficiently large so that the decision $y_t + \delta s_t$ (and $y_t - \delta s_t$) must be in $\mathcal{X}$. Consequently, there are some additional terms involved due to the projection over a shrinkage feasible set. In the following, we provide some properties on the relationships between original feasible set and shrinkage set. Most of the following results can be found in the seminal paper (Flaxman et al., 2005), for self-containedness, we provide the proof correspondingly.

**Lemma 3.** For any feasible point $x \in (1-\alpha)\mathcal{X}$, the ball of radius $\alpha r$ centered at $x$ belongs to $\mathcal{X}$.

**Proof.** The result is originally proved in Observation 3.2 of Flaxman et al. (2005). The proof is based on the simple observation that

$$(1-\alpha)\mathcal{X} + \alpha r \mathbb{B} \subseteq (1-\alpha)\mathcal{X} + \alpha \mathcal{X} = \mathcal{X}$$

holds since $r \mathbb{B} \subseteq \mathcal{X}$ and $\mathcal{X}$ is convex.

The following lemma, originally raised in Observation 3.3 of Flaxman et al. (2005), establishes a bound on the maximum that the function can change in $(1-\alpha)\mathcal{X}$, an effective Lipschitz condition.

**Lemma 4.** For any $x \in (1-\alpha)\mathcal{X}$, under Assumption 4, we have

$$|\hat{f}_t(x) - f_t(x)| \leq L\delta.$$  \hspace{1cm} (37)

**Proof.** Since $\hat{f}_t$ is an average over inputs within $\delta$ of $x$, the Lipschitz continuity of function $f_t$ gives the result.
A.2 Dynamic Regret

We first present the dynamic regret of online gradient descent in the full-information setting (Zinkevich, 2003).

Suppose the feasible domain $\mathcal{X}$ has an upper bound on its diameter, denoted as $D$, that is, $D = \sup_{x, y \in \mathcal{X}} \|x - y\|_2$. Meanwhile, denote by $G = \sup_{x \in \mathcal{X}, t \in [T]} \|\nabla f_t(x)\|_2$ an upper bound on the norm of the gradients of online functions over $\mathcal{X}$.

**Theorem 6** (Dynamic Regret of OGD). Consider the online gradient descent (OGD), which starts with $x_1 \in \mathcal{X}$ and performs

$$x_{t+1} = \text{Proj}_{\mathcal{X}}[x_t - \eta \nabla f_t(x_t)].$$

(38)

The dynamic regret of OGD is upper bounded by

$$\sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(u_t) \leq \frac{7D^2 + DP_T}{4\eta} + \frac{\eta G^2 T}{2},$$

(39)

for any comparator sequence $u_1, \ldots, u_T \in \mathcal{X}$. In above, $P_T$ is its path-length defined as $P_T = \sum_{t=2}^{T} \|u_t - u_{t-1}\|_2$.

In the bandit convex optimization setting, we cannot access the true gradient but the unbiased gradient estimation instead. Therefore, we extend Theorem 6 to the randomized version for the loss function chosen from adaptive environments as follows.

**Theorem 7** (Expected Dynamic Regret of Randomized OGD). Consider the randomized version online gradient descent (OGD) as follows, with $x_1 \in \mathcal{X}$ and

$$x_{t+1} = \text{Proj}_{\mathcal{X}}[x_t - \eta g_t],$$

(40)

where $E[g_t|x_1, f_1, \ldots, x_t, f_t] = \nabla f_t(x_t)$ and $\|g_t\|_2 \leq \bar{G}$ for some $\bar{G} > 0$. Then, the expected dynamic regret of OGD is upper bounded by

$$E\left[\sum_{t=1}^{T} f_t(x_t)\right] - \sum_{t=1}^{T} f_t(u_t) \leq \frac{7D^2 + DP_T}{4\eta} + \frac{\eta \bar{G}^2 T}{2},$$

(41)

for any fixed comparator sequence $u_1, \ldots, u_T \in \mathcal{X}$.

**Proof.** Define the function $h_t : \mathcal{X} \rightarrow \mathbb{R}$ by

$$h_t(x) = f_t(x) + x^T \xi_t, \quad \text{where} \quad \xi_t = g_t - \nabla f_t(x_t).$$

(42)

Clearly, $\nabla h_t(x_t) = \nabla f_t(x_t) + \xi_t = g_t$. So we can leverage the result of deterministic version OGD in Theorem 6 on the function $h_t$ and obtain that

$$\sum_{t=1}^{T} h_t(x_t) - \sum_{t=1}^{T} h_t(u_t) \leq \frac{7D^2 + DP_T}{4\eta} + \frac{\eta \bar{G}^2 T}{2}.$$

(43)
Note that for any fixed $x \in \mathcal{X}$, we have
\[
E[h_t(x)] = E[f_t(x)] + E[\xi_t^T x] \\
= E[f_t(x)] + E[E[\xi_t^T x | x_1, f_1, \ldots, x_t, f_t]] \\
= E[f_t(x)] + E[E[\xi_t | x_1, f_1, \ldots, x_t, f_t]^T x] \\
= E[f_t(x)]
\]

Therefore, when both the function sequence and comparator sequence are chosen by an oblivious adversary (as specified in Section 3.1), we can take expectations over both sides of (43) and obtain the desired result.

\[\square\]

### A.3 Adaptive Regret

We first present the adaptive regret of the Coin Betting for Changing Environment (CBCE) algorithm proposed by Jun et al. (2017) in the full-information setting.

**Theorem 8** (Adaptive Regret of CBCE (Jun et al., 2017, Theorem 1)). Consider an OCO problem where at iteration $t$ a learner iteratively select a decision $x_t \in \mathcal{X}$ and observes a loss function $h_t$. Assume the gradient of all the loss functions are bounded by $G$, the diameter of $\mathcal{X}$ is bounded by $D$, and the function value of $h_t$ lies in $[0, 1], \forall t \in [T]$. Then, the CBCE algorithm with the standard OGD algorithm as its expert-algorithm and $h_1, \ldots, h_T$ as the input loss functions achieves the following adaptive regret,

\[
\max_{[q,s] \in [T]} \left( \sum_{t=q}^{s} h_t(x_t) - \min_{x \in \mathcal{X}} \sum_{t=q}^{s} h_t(x) \right) \leq 15DG\sqrt{T} + 8DG\sqrt{7 \log T} + 5\sqrt{T}.
\]

The algorithm above is inefficient in the sense that it requires to query the gradient of the loss function $O(\log t)$ times at iteration $t$. To address this limitation, Wang et al. (2018) introduce a surrogate loss function $\ell_t : \mathcal{X} \mapsto [0, 1],$

\[
\ell_t(x) = \frac{1}{2DG} \nabla h_t(x_t)^T (x - x_t) + \frac{1}{2}
\]

for which we have $\forall x \in \mathcal{X},$

\[
h_t(x_t) - h_t(x) \leq -2DG\ell_t(x) + DG = 2DG(\ell_t(x_t) - \ell_t(x)).
\]

Notice that the inequality (46) implies that, to solve the original problem where the loss functions are $h_1(\cdot), \ldots, h_T(\cdot)$, we can deploy CBCE on a new problem where the loss functions are $\ell_1(\cdot), \ldots, \ell_T(\cdot)$. The benefits here is that in this way we only need to query the gradient of $h_t$ once in each iteration and the order of the regret bound remains the same. To be more specific, we have the following result.

**Theorem 9.** Consider the same learning setting as in Theorem 8. Then, the CBCE algorithm with the standard OGD algorithm as its expert-algorithm and $\ell_1, \ldots, \ell_T$ as the input loss functions achieves the following adaptive regret,

\[
\max_{[q,s] \in [T]} \left( \sum_{t=q}^{s} \ell_t(x_t) - \min_{x \in \mathcal{X}} \sum_{t=q}^{s} \ell_t(x) \right) \leq 15DG\sqrt{T} + 8DG\sqrt{7 \log T} + 5\sqrt{T}.
\]
Appendix B. Proofs of Adaptive Regret

In this section, we present the algorithmic details and proofs of theoretical guarantees in Section 4.2.

B.1 Algorithm and Theoretical Guarantees

Our proposed algorithm Minimizing Adaptive regret in Bandit Convex Optimization (MABCO) follows a similar framework to that of CBCE (Jun et al., 2017), which is a two-level structure, presented in Algorithm 3 (meta-algorithm) and Algorithm 4 (expert-algorithm). However, we note that a direct reduction of CBCE algorithm from the full-information setting to the bandit scenario by making use of the estimated gradients is prohibited, because the CBCE algorithm requires to query the loss function $O(\log t)$ times in each iteration $t$, which is not allowed in the bandit setup.

To address this issue, we adopt the idea in Theorem 9 and introduce the surrogate loss function $\ell_t$ (defined in (50) and (51) for different feedback models), whose function values as well as gradients can be computed by only using $f_t(x_t)$ (or $f_t(x_{1,t})$ and $f_t(x_{2,t})$ for the two-point feedback model), without further queries of the loss function. We then deploy standard CBCE algorithm on surrogate loss functions series $\ell_1, \ldots, \ell_T$ (Algorithm 3). Based on the relationship between $\ell_t$ and $f_t$ as shown in (53) and (58), our proposed algorithm essentially minimizes the adaptive regret on the original loss function sequence $f_1, \ldots, f_T$.

The detailed algorithm is described as follows. At iteration $t$, we maintain a set $S_t$ of experts, each of which is an instantiation of the OGD algorithm (Algorithm 4), performing on surrogate loss function $\ell_t$. At the beginning of each iteration, we past the surrogate loss function to experts and collect the predictions (line 3-8), then combine these predictions by their own weights (line 9). Next, we submit the perturbed decision and observe the feedback (line 11-12 for the one-point feedback model, and line 13-14 for the two-point feedback model). Finally, we adjust the set of experts to get $S_{t+1}$, and update the weights of experts in $S_{t+1}$ according to their performance (line 15-25). Specifically, the (unnormalized) weight of expert $E_i$, i.e., $\hat{p}_{i,t+1}$, is computed by

$$\hat{p}_{i,t+1} = \pi_i \max\{w_{i,t+1}, 0\}$$

where $\pi_i = 1/(q_i^2(1 + \lceil \log q_i \rceil))$ is the prior of expert $E_i$,

$$w_{i,t+1} = \frac{\sum_{j=q_i}^t \tilde{m}_{i,j,t}}{t - q_i + 1} \left(1 + \sum_{j=q_i}^t \tilde{g}_{i,j} w_{i,j}\right)$$

and

$$\tilde{m}_{i,t} = \mathbb{1}_{w_{i,t} > 0}(\ell_t(y_t) - \ell_t(y_{i,t})) + \mathbb{1}_{w_{i,t} \leq 0} \max\{\ell_t(y_t) - \ell_t(y_{i,t})\}.$$ 

We refer to the work of Jun et al. (2017) and Wang et al. (2018) for more details about the standard CBCE algorithm.

Next, we provide an elaboration of the theoretical guarantees in Theorem 5 as follows.

**Theorem 10** (one-point feedback model). Under Assumptions 1, 2, 3 and 4, define the surrogate loss function $\ell_t : (1 - \alpha)\mathcal{X} \mapsto \mathbb{R}$ as

$$\ell_t(y) = \frac{1}{2G_{\text{one}}} \langle \tilde{g}_t, y - y_t \rangle + \frac{1}{2}$$

(50)
Algorithm 3 Minimizing Adaptive in Bandit Convex Optimization (MABCO)

Input: Time horizon $T$; perturbation parameter $\delta$; shrinkage parameter $\alpha$.

1: Let $S_1 = \{E_1\}$, $q_1 = 1$, $y_1 = 0$
2: for $t = 1, \ldots, T$ do
3:    for $E_i \in S_t$ do
4:        if $q_i \neq t$ then
5:            Pass the surrogate loss function $\ell_t(\cdot)$ to expert $E_i$ (Algorithm 4)
6:        end if
7:    Get the decision $y_{i,t}$ of expert $E_i$
8: end for
9: $y_t = \sum_{E_i \in S_t} p_{i,t} y_{i,t}$
10: Select a unit vector $s_t$ uniformly at random
11:   {Case 1. One-Point Feedback Model}
12:    Submit $x_t = y_t + \delta s_t$.
13:   {Case 2. Two-Point Feedback Model}
14:    Submit $x_t^{(1)} = y_t + \delta s_t$ and $x_t^{(2)} = y_t - \delta s_t$
15:    Observe $f_t(x_t^{(1)})$ and $f_t(x_t^{(2)})$
16:    {Adjust the expert set and update the weights}
17:    Remove experts whose $e_i$ are less than $t$
18: for $E_i \in S_t$ do
19:    Compute $\widehat{m}_{i,t}$ by (49)
20: end for
21: Initialize $E_{\widehat{n}}$, set $q_{\widehat{n}} = t$ and compute $e_{\widehat{n}}$
22: $\widehat{n} = |S_t| + 1$
23: $S_{t+1} = S_t \cup \{E_n\}$
24: for $E_i \in S_{t+1}$ do
25:    Compute $w_{i,t+1}$ and $\widehat{p}_{i,t+1}$ by (48) and (47)
26: end for

Algorithm 4 Expert-Algorithm

1: Let $G = \max_{y \in (1-\alpha)X, t \in [T]} \|\nabla \ell_t(y)\|_2$.
2: if $q_t = t$ then
3:    $y_{i,t} = 0$
4: else
5:    $y_{i,t} = \text{Proj}_{(1-\alpha)X} \left[ y_{i,t-1} - \frac{R}{G \sqrt{t-q_i}} \nabla \ell_{t-1}(y_{i,t-1}) \right]$
6: end if

where $G^{\text{one}} = \frac{dC}{\delta}$ and $\tilde{g}_t$ is the gradient estimator defined in (9). Let Algorithm 3 be the meta-algorithm, which is fed with $\ell_1, \ldots, \ell_T$ as loss functions, and Algorithm 4 be the expert-
algorithm. Set $\delta$ as in (57) and $\alpha = \frac{\delta}{T}$. Then the expected adaptive regret satisfies

$$
\mathbb{E}[A\text{-Regret}_T] \leq \sqrt{C d \left(15 R \sqrt{T} + 8 R \sqrt{7 \log T + 5 \sqrt{T}}\right)} \left(3LT + \frac{L R}{r} \sqrt{T}\right) = O(T^{\frac{3}{4}}(\log T)^{\frac{1}{2}}).
$$

**Theorem 11** (two-point feedback model). Under Assumptions 1, 2, 3 and 4, define the surrogate loss function $\ell_t : (1 - \alpha)\mathcal{X} \mapsto \mathbb{R}$ as

$$
\ell_t(y) = \frac{1}{2G^\text{two}R} \langle \bar{g}_t, y - y_t \rangle + \frac{1}{2}
$$

where $G^\text{two} = Ld$ and $\bar{g}_t$ is the gradient estimator defined in (12). Let Algorithm 3 be the meta-algorithm, which is fed with $\ell_1, \ldots, \ell_T$ as loss functions, and Algorithm 4 be the expert-algorithm. Set $\alpha = \frac{\delta}{T}$ and $\delta = \frac{1}{T}$. Then the expected adaptive regret satisfies

$$
\mathbb{E}[A\text{-Regret}_T] \leq Ld \left(15 R \sqrt{T} + 8 R \sqrt{7 \log T + 5 \sqrt{T}}\right) + 3L \sqrt{T} + \frac{L R}{r} \sqrt{T} = O(T^{\frac{3}{4}}(\log T)^{\frac{1}{2}}).
$$

**B.2 Proof of Theorem 10**

*Proof.* For any time interval $I = [q, s] \in [T]$, we have

$$
\mathbb{E} \left[ \sum_{t=q}^{s} f_t(x_t) \right] - \min_{x \in \mathcal{X}} \sum_{t=q}^{s} f_t(x)
= \mathbb{E} \left[ \sum_{t=q}^{s} \hat{f}_t(y_t) \right] - \min_{y \in (1 - \alpha)\mathcal{X}} \sum_{t=q}^{s} \hat{f}_t(y) + \mathbb{E} \left[ \sum_{t=q}^{s} f_t(x_t) - \hat{f}_t(y_t) \right] + \min_{x \in (1 - \alpha)\mathcal{X}} \sum_{t=q}^{s} \hat{f}_t(x) - \min_{x \in \mathcal{X}} f_t(x)
\leq \text{term(a)} + 3L \delta T + \frac{LR}{r} \delta T
$$

(52)

where (52) follows from the analysis in dynamic regret (see (21) and (22)). Note that since $y_t$ is the weighted combination of $y_{i,t}$, it still satisfies $y_t \in (1 - \alpha)\mathcal{X}$.

Now, it remains to bound term (a). Define the function $h_t : (1 - \alpha)\mathcal{X} \mapsto \mathbb{R}$ by $h_t(y) = \hat{f}_t(y) + y^T \xi_t$, where $\xi_t = \bar{g}_t - \nabla \hat{f}_t(y_t)$ with $\bar{g}_t = \frac{1}{s} \sum_{i=1}^{s} f_i(y_t + \delta s_i) \cdot s_i$. By the analysis of dynamic regret (see (44)), we know that $\mathbb{E}[h_t(y)] = \mathbb{E}[\hat{f}_t(y)]$ for any fixed $y \in (1 - \alpha)\mathcal{X}$. Besides, since $\nabla h_t(y_t) = \nabla \hat{f}_t(y_t) + \xi_t = \bar{g}_t$, we have $\forall y \in (1 - \alpha)\mathcal{X}$,

$$
h_t(y_t) - h_t(y) \leq \nabla h_t(y_t)^T(y_t - y) \leq -2G^\text{one} R \ell_t(y) + G^\text{one} R.
$$

(50)

Note that since $\ell_t(y_t) = \frac{1}{2}$, we have $\forall y \in (1 - \alpha)\mathcal{X}$,

$$
\mathbb{E} \left[ \hat{f}_t(y_t) - \hat{f}_t(y) \right] = \mathbb{E} \left[ h_t(y_t) - h_t(y) \right] \leq 2G^\text{one} R \cdot \mathbb{E} \left[ (\ell_t(y_t) - \ell_t(y)) \right].
$$

(53)

On the other hand, essentially, Algorithm 3 can be regarded as a standard CBCE algorithm deploying on a full-information problem where the loss function sequence is $\ell_1, \ldots, \ell_T$. Hence, based on Theorem 8, we have

$$
\max_{[q, s] \in [T]} \left( \sum_{t=q}^{s} \ell_t(y_t) - \min_{y \in (1 - \alpha)\mathcal{X}} \sum_{t=q}^{s} \ell_t(y) \right) \leq 15RG \sqrt{T} + 8\sqrt{7 \log T + 5 \sqrt{T}}
$$

(54)
where $\hat{G} = \sup_{y \in (1-\alpha)\mathcal{X} : t \in [T]} \| \ell_t(y) \|_2 \leq \frac{1}{\sqrt{T}}$. Combining with (53), we can upper bound term(a) by

$$\max_{[q,s] \in [T]} \left( \mathbb{E} \left[ \sum_{t=q}^s \hat{f}_t(y_t) \right] - \min_{y \in (1-\alpha)\mathcal{X}} \sum_{t=q}^s \hat{f}_t(y) \right) \leq 15G^{one} R \sqrt{T} + 8G^{one} R \sqrt{7 \log T} + 5 \sqrt{T}. \quad (55)$$

Plugging (55) into (52), we get

$$\mathbb{E} \left[ \sum_{t=q}^s f_t(x_t) \right] - \min_{x \in \mathcal{X}} \sum_{t=q}^s f_t(x) \leq 15G^{one} R \sqrt{T} + 8G^{one} R \sqrt{7 \log T} + 5 \sqrt{T} + \frac{LR}{r} \delta T$$

$$\leq \frac{C d}{\delta} \left( 15R \sqrt{T} + 8R \sqrt{7 \log T} + 5 \sqrt{T} \right) + \delta \left( 3LT + \frac{LR}{r} T \right)$$

$$= \sqrt{C d \left( 15R \sqrt{T} + 8R \sqrt{7 \log T} + 5 \sqrt{T} \right)} \left( 3LT + \frac{LR}{r} T \right)$$

$$= O(T^{\frac{3}{4}} (\log T)^{\frac{1}{4}})$$

where (56) is derived by optimally configuring

$$\delta = \sqrt{\frac{C d (15R \sqrt{T} + 8R \sqrt{7 \log T} + 5 \sqrt{T})}{3LT + LRT/r}} \quad (57)$$

which finishes the proof. \qed

**B.3 Proof of Theorem 11**

**Proof.** The proof is similar to that in Section B.2. Define the function $h_t : (1-\alpha)\mathcal{X} \mapsto \mathbb{R}$ by $h_t(y) = \hat{f}_t^*(y) + y^T \xi_t$, where $\xi_t = \hat{y}_t - \nabla \hat{f}_t(y_t)$ with $\hat{y}_t = \frac{d}{28} (f_t(y_t + \delta s_t) - f_t(y_t - \delta s_t)) \cdot s_t$. Similarly, $\mathbb{E}[h_t(y)] = \mathbb{E}[\hat{f}_t(y)]$ holds for any fixed $y \in (1-\alpha)\mathcal{X}$. Besides, since $\nabla h_t(y_t) = \nabla \hat{f}_t(y_t) + \xi_t = \hat{y}_t$, we have $\forall y \in (1-\alpha)\mathcal{X}$,

$$h_t(y_t) - h_t(y) \leq -2G^{two} R \ell_t(y) + G^{two} R.$$

Note that since $\ell_t(y_t) = \frac{1}{2}$, we have $\forall y \in (1-\alpha)\mathcal{X}$,

$$\mathbb{E} \left[ \hat{f}_t(y_t) - \hat{f}_t(y) \right] = \mathbb{E} [h_t(y_t) - h_t(y)] \leq 2G^{two} R \mathbb{E} [\ell_t(y_t) - \ell_t(y)] \quad (58)$$

Hence, by deploying the standard CBCE algorithm on the loss function series $\ell_1, \ldots, \ell_T$ (Algorithm 3), and based on Theorem 8, we have

$$\max_{[q,s] \in [T]} \left( \sum_{t=q}^s \ell_t(y_t) - \min_{y \in (1-\alpha)\mathcal{X}} \sum_{t=q}^s \ell_t(y) \right) \leq 15 \hat{G} \sqrt{T} + 8 \sqrt{7 \log T} + 5 \sqrt{T} \quad (59)$$

28
where $\hat{G} = \max_{y \in (1-\alpha)X, t \in [T]} \|\nabla \ell_t(y)\|_2 \leq \frac{1}{2R}$. Thus, we have

$$
\mathbb{E} \left[ \sum_{t=q}^{s} f_t(x_t) \right] - \min_{x \in X} \sum_{t=q}^{s} f_t(x) \\
\leq 15G^{two} R \sqrt{T} + 8G^{two} R \sqrt{7 \log T} + 5\sqrt{T} + 3L\delta T + \frac{LR}{r} \delta T \\
\leq LD \left( 15R \sqrt{T} + 8R \sqrt{7 \log T} + 5\sqrt{T} \right) + \delta \left( 3LT + \frac{LR}{r} T \right) \\
= LD \left( 15R \sqrt{T} + 8R \sqrt{7 \log T} + 5\sqrt{T} \right) + 3L\sqrt{T} + \frac{LR}{r} \sqrt{T} \\
= O(\sqrt{T \log T})
$$

Therefore, we complete the proof. \qed