Smooth Bandit Optimization: Generalization to Hölder Space

Yusha Liu
Carnegie Mellon University
yushal@cs.cmu.edu

Yining Wang
University of Florida
yining.wang@warrington.ufl.edu

Aarti Singh
Carnegie Mellon University
aarti@cs.cmu.edu

Abstract

We consider bandit optimization of a smooth reward function, where the goal is cumulative regret minimization. This problem has been studied for α-Hölder continuous (including Lipschitz) functions with $0 < \alpha \leq 1$. Our main result is in generalization of the reward function to Hölder space with exponent $\alpha > 1$ to bridge the gap between Lipschitz bandits and infinitely-differentiable models such as linear bandits. For Hölder continuous functions, approaches based on random sampling in bins of a discretized domain suffices as optimal. In contrast, we propose a class of two-layer algorithms that deploy misspecified linear/polynomial bandit algorithms in bins. We demonstrate that the proposed algorithm can exploit higher-order smoothness of the function by deriving a regret upper bound of $\tilde{\mathcal{O}}(T^{\frac{d+\alpha}{d+2\alpha}})$ for when $\alpha > 1$, which matches existing lower bound. We also study adaptation to unknown function smoothness over a continuous scale of Hölder spaces indexed by α, with a bandit model selection approach applied with our proposed two-layer algorithms. We show that it achieves regret rate that matches the existing lower bound for adaptation within the $\alpha \leq 1$ subset.

1 Introduction

This paper considers the problem of black-box optimization of a reward function $f : \mathcal{X} \to \mathcal{R}$, that is bounded and defined on a compact d-dimensional domain $\mathcal{X}$, using active queries. At each round, the learner chooses an action x_t by leveraging the previously collected data and observes a noisy and zeroth

order feedback of the function value $f(x_t)$. In the bandit setting, the goal is to minimize the cumulative regret with respect to global maxima. This is also known as the continuum-armed bandit problem. The bandit framework is different from standard global zeroth order optimization because of its unique exploration-exploitation dilemma. While in zeroth order optimization problems, pure exploration will often suffice since the performance is measured by simple regret (i.e. difference between the optimized function value and true function maxima), in bandit optimization, the queried function values need to be controlled through the entire optimization process to minimize the cumulative regret. Therefore, the algorithms require different and often more careful design.

Most existing works on continuum-armed bandit optimization either assume parametric models such as linear bandits (Dani et al. (2008); Abbasi-Yadkori et al. (2011); Rusmevichientong and Tsitsiklis (2010)) for the reward function, or a black-box model where the reward function is assumed to be α-Hölder continuous (including Lipschitz) with $0 < \alpha \leq 1$ with respect to some known metric (Kleinberg (2005); Auer et al. (2007); Kleinberg et al. (2008); Bubeck et al. (2010, 2011); Locatelli and Carpentier (2018)). The main purpose of this paper is to extend this assumption to the more general Hölder function space (definition 1) with exponent $\alpha > 1$ and exploit the higher order of function smoothness. Such generalization is a parallel to Hölder smoothness assumpions appearing in fundamental results in nonparametric regression (Stone (1982)), which has been used in a variety of applications such as economics (Yatchew (1998)). Approaches based on fitting an appropriate function using random samples in bins of a discretization of the domain (i.e., exploration) suffice as optimal for non-parametric regression as well as controlling simple regret of any Hölder smooth reward functions with $\alpha > 0$. They are also worst-case optimal for controlling cumulative regret for Hölder continuous reward functions with $\alpha \leq 1$. In contrast, controlling cumulative regret for Hölder smooth reward functions with $\alpha > 1$ requires finer control in bins over the queried values

via a local exploration-exploitation tradeoff. Thus, instead of using a single layer algorithm that randomly samples from selected bins, we propose a class of algorithms that use two layers of bandit algorithms - one multi-armed bandit algorithm operating over the bins, and another set of misspecified linear/polynomial bandit algorithms operating in each bin to govern the local exploration-exploitation tradeoff. We derive regret bounds for this class of two-layer bandit algorithms and show that they match the existing lower bounds apart from log factors.

Additionally, we study the problem of adaptation to smoothness exponent α for a continuous scale of Hölder spaces. Unlike the simple regret minimization setting where this adaptation comes at no cost in terms of the minimax rates, it was shown by Locatelli and Carpentier (2018) that it is generally impossible to achieve minimax adaptation under cumulative regret. We propose a procedure with regret bound that matches the existing adaptive lower bound with only access to the range of the unknown parameter α. We start by describing related works, followed by a summary of our contributions.

1.1   Related Works

Continuum-armed Bandit.   In continuum-armed bandit problems, the domain $\mathcal{X}$ is allowed to be a measurable space and the set of arms is therefore infinite. Previous works in continuum-armed bandit usually assumes global smoothness (Kleinberg (2005)) of the reward function or local smoothness (e.g. Auer et al. (2007)) around the global maxima. The smoothness condition, in particular, is defined as Lipschitz continuity with respect to some metrics (Kleinberg (2005); Kleinberg et al. (2008)) or dissimilarity functions (Kleinberg et al. (2008); Bubeck et al. (2010)), or α-Hölder continuity with $0 < \alpha \leq 1$ (Kleinberg (2005); Auer et al. (2007)). Worst-case lower bound under the Lipschitz assumption is presented in Kleinberg et al. (2008) and that under the Hölder continuity assumption in Locatelli and Carpentier (2018).

Existing works rarely consider generalization to Hölder spaces. Recently, Hu et al. (2020) studied contextual bandit with reward functions in Hölder spaces, however, the reward function is assumed to be smooth with respect to the observed contexts and the action set is finite. For non-contextual continuum-armed bandits, Akhavan et al. (2020) focus on the strongly convex subset of functions in Hölder spaces with $\alpha \geq 2$. They use projected gradient-like algorithms. Grant and Leslie (2020) analyze Thompson sampling (TS), a Bayesian method, on Hölder spaces with integer-valued exponents and derive a suboptimal upper bound based on complexity of the function

space[1]. They also derive lower bound on regret in one-dimension setting, but as we later remark in this paper, the lower bounds can be implied by Wang et al. (2018) in a more general setting.

Adaptivity to Smoothness of the Reward Functions.   An intriguing problem is whether an algorithm that is oblivious to the Hölder exponent α can simultaneously achieve minimax rates for a range of values for α. For non-contextual continuum-armed bandits, this has been discussed only under the Hölder continuous($\alpha \leq 1$) setting. In particular, Locatelli and Carpentier (2018) state that generally, such minimax adaptation to α is impossible by providing a worst-case lower bound for adaptation between two Hölder-continuous function spaces. Additionally, they propose conditions under which it would become possible. (For the contextual finite-armed bandit studied in Hu et al. (2020), Gur et al. (2019) provide lower bounds with similar rates and the extra conditions as well.) However, it remains unclear that, without the extra conditions, whether an algorithm can achieve the lower bound when adapting to a continuous scale of general Hölder spaces.

Model Selection for Bandits.   Another relevant line of work is more broadly model selection in bandit settings, which we will leverage in bandit optimization of Hölder-smooth functions as well as adaptation to the smoothness. In this problem, given a set of base algorithms on possibly different domains, the learner needs to adapt to the best one in an online fashion. The goal is to achieve cumulative regret comparable to the best base algorithm if it were run solely. Bubeck et al. (2011) study the model selection problem for adapting to the unknown Lipschitz constant of functions. Foster et al. (2019) study adapting to the unknown policy dimension in contextual linear bandits by estimating the gap between two policy classes. Agarwal et al. (2016) develop a general algorithm named Corral for bandit model selection under adversarial feedback. It uses online mirror descent to balance between base algorithms. For stochastic feedback particularly, Pacchiano et al. (2020) modify the Corral algorithm to relax requirements on base-algorithms and improve the result on some problem instances (including the one in Foster et al. (2019)). Another relevant issue addressed in Krishnamurthy et al. (2019) which study contextual continuum-armed bandits with Lipschitz continuous reward functions, is their use of the original Corral algorithm applied with EXP4 for adaptation to unknown Lipschitz constant. UCB-type algorithm for corralling base-algorithms is

---

[1]They comment that the reason could be either the analysis being suboptimal or the nature of TS.

used in Arora et al. (2020) under the assumption that the base-algorithms are finite-armed, and only one of them has access to the best arm.

1.2 Our Contributions

We study bandit optimization of functions in general Hölder spaces. This paper furthers the previous works in the following two main aspects:

1. We propose a novel class of two-layer bandit algorithms, where a carefully-chosen Meta-algorithm deploys misspecified bandit algorithms as arms. Our algorithms show explicitly how to exploit higher-order smoothness in achieving optimal exploration-exploitation tradeoff. We derive worst-case regret bound for this algorithm that matches the existing lower bound except for log factors, for functions in Hölder space including when $\alpha > 1$. Our results bridges the gap between Lipschitz smooth bandits where the Hölder exponent is $\alpha = 1$ and infinitely-differentiable problems such as linear bandits where the Hölder exponent is $\alpha = \infty$.

2. We study adaptation to a sequence of Hölder spaces indexed by a continuous but unknown variable of exponent α. We propose a strategy with theoretical guarantee, which uses the bandit model selection algorithm Corral from Pacchiano et al. (2020) applied with versions of our proposed two-layer algorithms. The derived regret bound is to our knowledge the first result on upper bounds when adapting to a continuous scale of Hölder spaces in continuum-armed bandit optimization.

The rest of this paper is organized as follows: In section 2 we introduce the problem formulation and assumptions. We present the two-layer Meta-algorithms and theoretical guarantees in section 3. In section 4 we study the adaptation to unknown smoothness and conclude the paper in section 5 with some open questions.

2 Problem Formulation

In this paper, we consider bandit optimization of smooth functions in Hölder space $\sum(\alpha, L)$ with $\alpha > 1$. The Hölder space is defined formally in definition 1. Some works also study benign problem instances with additional "growth" conditions than the smoothness to characterize the difficulty of finding global maxima, for improvements in regret bounds. For example, Auer et al. (2007) use a parameter to model the growth rate of Lebesgue measure of the near-optimal arms set as a

function of the threshold. The near-optimality dimension in Bubeck et al. (2010) uses packing number but has similar meaning. Their proposed HOO algorithm employs a tree-based adaptive discretization algorithm to exploit when the near-optimality dimension is small. In this paper we will focus solely on worst-case regret to preserve simplicity and leave adaptation to benign cases as a future direction. The performance of the learner is measured by cumulative pseudo-regret as stated below where $x^* \in \arg\max_{x \in \mathcal{X}} f(x)$. Throughout this paper we will simply refer to the pseudo-regret as regret.

$$R(T) = \sum_{t=1}^{T} [f(x^*) - f(x_t)]. \tag{1}$$

To formally define Hölder spaces, we first introduce some notations. Define the following notions for a vector $s = (s_1 \ldots s_d)$: let $|s| = s_1 + \cdots + s_d$, $s! = s_1! \ldots s_d$ and $x^s = x_1^{s_1} \ldots x_d^{s_d}$. And define $D^s = \frac{\partial^{|s|}}{\partial x_1^{s_1} \ldots \partial x_d^{s_d}}$.

Definition 1 (Tsybakov (2008)). The Hölder space $\sum(\alpha, L)$ on domain $\mathcal{X} \in \mathcal{R}^d$ is defined as the set of functions $f : \mathcal{X} \to \mathbf{R}$ that are $l = \lfloor \alpha \rfloor$ times differentiable and have continuous derivatives[2]. l is the largest integer that is strictly smaller than α. A function f in $\sum(\alpha, L)$ satisfies the following inequality[3] for $\forall x, y \in \mathcal{X}$.

$$D^s f(x) - D^s f(y) \le L \|x - y\|_\infty^{\alpha - l}, \quad \forall s \text{ s.t.} |s| = l.$$

In particular, a function in $\sum(\alpha, L)$ is close to its Taylor approximation:

$$|f(x) - T_y^l(x)| \le L \|x - y\|_\infty^\alpha, \forall x, y \in \mathcal{X}.$$

We use T_y^l to denote the l-degree Taylor polynomial around y, $T_y^l(x) = \sum_{|s| \le l} \frac{(x-y)^s}{s!} D^s f(y)$.

Assumptions We specify the assumptions that are used throughout this paper.

G1. The input domain $\mathcal{X}$ is a hypercube $[0,1]^d$. For simplicity assume the reward function is bounded: $\|f\|_\infty \le 1$.

G2. The function f belongs to Hölder space $\sum(\alpha, L)$ with some constant $L > 0$ [4].

---

[2] Only when referring to the order of Hölder smooth functions' derivatives do we denote $\lfloor \cdot \rfloor$ as the largest integer *strictly* less than input. In other places in this paper it denotes less or equal to input.

[3] We use l_∞ norm as in some works on adaptive confidence bands and optimization (Low et al. (1997); Tsybakov (2008); Hoffmann et al. (2011); Wang et al. (2018)).

[4] In this paper, for simplicity, we assume L is some constant that satisfies assumption G1.

G3. The observations are noisy: $y = f(x) + \eta$ where the noise η is drawn from i.i.d zero mean subgaussian distribution with parameter σ.

3 Meta-algorithm and Analysis

A commonly used method for continuum-armed bandits is fixed discretization, which divides the continuous input domain into finite number of bins, to transform the problem into finite-armed bandit. Previous works mostly consider Hölder-continuous ($\alpha \leq 1$) functions. For example Auer et al. (2007) study the α-Hölder continuous functions with $\alpha \leq 1$ for one-dimension domain, followed by Bubeck et al. (2010) who generalize it to d-dimensional domain and propose the HOO algorithm with adaptive discretization[5]. In these works, it suffices to perform random sampling (Auer et al. (2007); Bubeck et al. (2010)) or midpoint sampling (Kleinberg (2005)) inside each bin. The worst-case regret bound for Lipschitz space of $\tilde{\mathcal{O}}(T^{\frac{d+1}{d+2}})$ are matched by the general lower bound of $\Omega(T^{\frac{d+\alpha}{d+2\alpha}})$(Auer et al. (2007); Bubeck et al. (2010); Locatelli and Carpentier (2018); Bubeck et al. (2011)) apart from log factors. However, if we apply the same methods of random sampling on fixed discretization (Auer et al. (2007)) on functions with Hölder exponent $\alpha > 1$, the regret incurred is $\tilde{\mathcal{O}}(T^{\frac{d+1}{d+2}})$ since the Hölder space with exponent $\alpha > 1$ is a subset of the Lipschitz function space. It prompts us to ask the question of whether a better rate that matches the dependence on α can be achieved for functions that are smoother than Lipschitz. An extreme is when α reaches infinity, where the reward model will be infinitely-differentiable, for example the stochastic linear bandit which enjoys $\tilde{\mathcal{O}}(T^{\frac{1}{2}})$ regret even on continuous domain (Dani et al. (2008); Abbasi-Yadkori et al. (2011)).

3.1 Algorithm Overview

We keep to fixed discretization of the domain since we consider only the worst-case regret. We divide $\mathcal{X} = [0,1]^d$ into n equal-sized hypercubes, leaving n as a parameter of the algorithm. As shown in definition 1, the function is locally well-approximated by Taylor polynomial which reduces to a linear model in some feature representation of x. It is equivalent to observing a misspecified linear model inside each bin, the equivalence formally quantified in Lemma 2. Therefore, local exploration-exploitation tradeoff can be achieved by a base algorithm with sublinear regret

on such misspecified models, with a Meta-algorithm to balance the budgets between the base algorithms in the bins.

Lemma 2. *Let hypercube $\mathcal{B}_\Delta$ be a subset of the input space with volume Δ. If a function satisfies assumption $G1 \sim 2$, there exists a linear parameter[6] $\theta^* \in R^{d(\alpha)}$ and feature map $\phi : x \mapsto \phi(x) \in R^{d(\alpha)}$,such that f can be approximated by the linear function: $\|f - \langle \theta^*, \phi(x) \rangle\|_\infty \leq \epsilon = L\Delta^{\frac{\alpha}{d}}$ for $x \in B_\Delta$. When $\alpha \leq 2$, $d(\alpha) = d$; when $\alpha > 2$, $d(\alpha) = \mathcal{O}(d^l)$ with l (definition 1). The linear parameter may not be unique.*

The proof is in Appendix section A.1. In the following parts of this section, we first present the misspecified bandit algorithm to run inside a bin, and then the Meta-algorithms to control these local algorithms.

3.2 The Misspecified Linear Bandit Algorithm

In this subsection we take a step back briefly to present the misspecified linear bandit algorithm which is modified from the *ConfidenceBall*$_2$ algorithm in Dani et al. (2008) and serves as "arm" of the Meta-algorithms. The algorithm, as shown in its name, is based on construction of confidence ellipsoid of the unobserved linear parameter in dimension d. We prove that the proposed modification can accommodate bias in the function feedback by deriving an upper bound on the cumulative regret[7] of $\tilde{\mathcal{O}}(d\sqrt{T} + dT\epsilon)$. Here ϵ is the upper bound on bias value and is given to the algorithm. We recently discovered that a similar result with proof sketch already appeared in recent work of Lattimore and Szepesvari (2019) (appendix E) who used modification of the algorithm in Abbasi-Yadkori et al. (2011), and hence enjoys the improvement of a multiplicative factor $\sqrt{\log(T)}$. For completeness and to provide necessary intermediate results for Meta-algorithms in later sections, we present our algorithm and full proof as complementary. It is worth mentioning that without the modification, the original algorithm incurs suboptimal regret under misspecification.

Assumptions We make the following assumptions for the misspecified model. Note that they are consistent with the aforementioned global assumptions.

A1. The feedback model is $y = \langle x, \theta^* \rangle + b(x) + \eta$ with $|b(x)| \leq \epsilon, \forall x \in \mathcal{X} \in R^d$.

A2. The mean reward $\mathbb{E}[y]$ is bounded by $[-1, 1]$.

---

[5]The adaptive discretization does not change worst-case regret but has improvements on benign problems, as introduced in section 2.

[6]We slightly abuse the notation and define short-hand notation $\langle \theta, x \rangle := \theta_0 + \sum_{i=1}^{d(\alpha)} \theta_i x_i$.

[7]For clarity this use of $\tilde{\mathcal{O}}$ omits $\ln(T)$ and δ dependence.

A3. The noise η is drawn from zero-mean sub-gaussian with parameter σ[8].

The pseudo-code of the modified algorithm is shown in Algorithm 1. The goal is to minimize the cumulative pseudo-regret of the linear model:

$$R(T) = \sum_{t=1}^{T} r_t = \sum_{t=1}^{T} (\langle x^*, \theta^* \rangle - \langle x_t, \theta^* \rangle). \quad (2)$$

We prove that this regret is $\mathcal{O}\left(d\ln(T)\sqrt{\ln(\frac{T^2}{\delta})T} + \epsilon T d\sqrt{2\ln(T)}\right)$ with probability $1 - \delta$. This is formally stated in Theorem 3.

Algorithm 1 Misspecified linear UCB algorithm ($\mathcal{A}^{local}$)

Require: Misspecification error ϵ, input domain $\mathcal{X}$ and its dimension d, fail probability δ, upper bound on $\|x\|_2^2$: $\kappa^2 = d$.
1: Initialize $A_1 = I_d$ and $x_1 \in \mathcal{X}$ randomly.
2: **for** $t = 1 \ldots$ **do**
3:     Execute action x_t and observe reward y_t
4:     $A_{t+1} = A_t + x_t x_t^T$
5:     $\hat{\theta}_{t+1} = A_{t+1}^{-1}(\sum_{\tau=1}^{t} y_\tau x_\tau)$
6:     $\beta_{t+1} = 128\sigma^2 d \ln(1+t)\ln(\frac{4(t+1)^2}{\delta})$
7:     Define function $UCB_{t+1}(x) = \left(\langle x, \hat{\theta}_{t+1}\rangle + \sqrt{\beta_{t+1}}\|A_{t+1}^{-1/2}x\| + \epsilon\sum_{s=1}^{t}|x^T A_{t+1}^{-1}x_s|\right)$
8:     Compute action $x_{t+1} = \arg\max_{x \in \mathcal{X}} UCB_{t+1}(x)$
9:     Return x_{t+1} and $UCB_{t+1}(x_{t+1})$
10: **end for**

Theorem 3. *If assumptions A1~A3 hold, then with probability $1 - \delta$, the cumulative regret of Algorithm 1 is upper bounded by:*

$$R(T) \leq \sqrt{8d\beta_T T \ln(1+T)} + 2\epsilon T d\sqrt{2\ln(1+T)} + 2\epsilon T. \quad (3)$$

The first term is the standard stochastic linear bandit regret rate same as in Dani et al. (2008). We defer the proof to Appendix section A.2. The increment of a multiplicative factor $\sqrt{d}$ in the second term compared to that in Lattimore and Szepesvari (2019) is due to difference in assumption on $\|x\|^2$. Their assumption is $\|x\|^2 \leq 1$ whereas ours is $\|x\|^2 \leq d$.

3.3 The UCB-Meta-algorithm

We now present the first structure of our Meta-algorithms. We consider the most straightforward structure: UCB-Meta, the pseudo-code is shown in

---

[8]Different from Dani et al. (2008) who assumes bounded noise. This reflects in the difference in β_t.

Algorithm 2 (define $\lfloor\rceil$ as the action of rounding to nearest integer). We keep a version of the base misspecified linear bandit algorithm in each bin. The confidence estimates of the local linear models are passed to the Meta-algorithm as UCB of arms, with adjustment of ϵ, the bias quantity. At every round we choose the bin with the highest UCB and run one step of the local bandit algorithm to update its estimation. For adjusting to different values of $l = \lfloor\alpha\rfloor$, we need only change the linear model, specifically the feature mapping $\phi : x \mapsto \phi(x) \in R^{d(\alpha)}$ as defined in proof of Lemma 2. For example, when $\alpha \leq 2$, the sub-algorithms are misspecified linear bandits whose actions spaces are simply bins $B \in \mathcal{X}$.

Algorithm 2 UCB-Meta-algorithm ($\mathcal{A}^{global}$)

Require: smoothness parameter α, Hölder constant L, dimension of domain d, time horizon T and fail probability δ, action space $\mathcal{X}$.
1: Initialize $n = \lfloor T^{\frac{d}{d+2\alpha}}/\ln(T)^{\frac{2d}{d+2\alpha}}\rceil$ and divide the action space $\mathcal{X}$ into same-sized bins $B_{1\ldots n}$ with volume $\Delta = 1/n$.
2: **for** $k = 1, \ldots, n$ **do**
3:     On bin B_k, start a version of local misspecified base-algorithm $\mathcal{A}_k$ using misspecification error $\epsilon = Ln^{\frac{-\alpha}{d}}$, input domain $\mathcal{X}^* = \{\phi(x), x \in \mathcal{X}\}$ and its dimension $d(\alpha)$, fail probability δ/n.
4:     Initialize counter $s_k = 1$ to indicate how many times $\mathcal{A}_k$ is queried.
5:     Query $\mathcal{A}_k$ once by running steps 3-9 of Algorithm 1 with $t = s_k$ and obtain upper confidence bound UCB_k.
6:     $s_k \leftarrow s_k + 1$
7: **end for**
8: **for** $\tau = 1 \ldots T$ **do**
9:     Select the bin with index $k(\tau) = \arg\max_k UCB_k$.
10:     Execute the local bandit algorithm $\mathcal{A}_{k(\tau)}$ once by running steps 3-9 (of Algorithm 1) with $t = s_{k(\tau)}$
11:     Receive updated recommendation $\phi_\tau \in \{\phi(x), x \in B_{k(\tau)}\}$ and $UCB_{k(\tau)}$.
12:     Increase counter for $\mathcal{A}_{k(\tau)}$: $s_{k(\tau)} \leftarrow s_{k(\tau)} + 1$.
13: **end for**

3.3.1 Regret Analysis of Algorithm 2

Theorem 4. *Let $d(\alpha)$ be the dimension of (polynomial) feature of x, defined in Lemma 2. If the reward function satisfies G1~G3 in section 2, then with probability $1 - \delta$, the cumulative regret (equation 1) of*

UCB-Meta-Algorithm is upper bounded by[9]

$$R(T) \leq \mathcal{O}\left(d(\alpha)\ln(T)\sqrt{Tn\ln(T^2n/\delta)} + d(\alpha)\epsilon T\sqrt{\ln(T)}\right). \tag{4}$$

The core of the proof is the distribution-independent analysis of UCB, which relies on the honesty of the confidence bands as well as their lengths. In particular, if the function value $f(x)$ at time t is contained in an honest confidence band $[UCB_t(x) - 2l_t(x), UCB_t(x)]$, then we can use the length $l_t(x)$ to bound instantaneous regret incurred by the selected action at this step. The confidence ellipsoids for the piecewise linear parameters $\hat{\theta}_{k,t}$ that are constructed by local misspecified linear bandits offer a convenient confidence estimation of function value, with the additional adjustment factor ϵ, the approximation error. The full proof is deferred to Appendix section A.3. The algorithm defines each bin to be a hypercube with volumn $\Delta = 1/n$, according to Lemma 2 we have $\epsilon = Ln^{\frac{-\alpha}{d}}$. Therefore, setting $n = \mathcal{O}(T^{\frac{d}{d+2\alpha}}/\ln(T)^{\frac{2d}{d+2\alpha}})$ will minimize the upper bound and yield cumulative regret bound of [10]

$$R(T) \leq \tilde{\mathcal{O}}(d(\alpha)T^{\frac{d+\alpha}{d+2\alpha}}). \tag{5}$$

3.3.2 Anytime Regret Guarantee for Algorithm 2

To achieve the rate in bound 5, Algorithm 2 needs to know the time horizon T in advance to set n and ϵ correspondingly. Here we prove that, with the doubling trick (Auer et al. (1995)) , the UCB-Meta-algorithm can get regret that is of the same rate as in bound 5 up to constant factors without knowing T. This result is needed in the adaptation problem studied in section 4.

Theorem 5. *If Algorithm 2 with access to the time horizon T achieves regret of $\tilde{\mathcal{O}}(T^a)$ with probability $1 - \delta$, then the procedure described in Algorithm 3 can achieve regret rate $\tilde{\mathcal{O}}(T^a)$ with probability $1 - \delta$ without the knowledge of T.*

The pseudo-code for Algorithm 3 is in Appendix section B.1 and the proof of Theorem 5 in Appendix A.4.

3.4 The Corral-Meta-algorithm

Another choice for Meta-algorithm is bandit model selection methods. Here we use the Corral algorithm defined in Pacchiano et al. (2020), which will be introduced more formally in section 4. An example of corralling misspecified linear bandit algorithms without corruption to the regret rate apart from log factors

---

[9]The d-dependence of the second term is propagated from Theorem 3

[10]δ-dependence absorbed in $\mathcal{O}$ since they are inside log terms.

has already been given in Pacchiano et al. (2020), but for adaptation to the misspecification error ϵ. Here we demonstrate that it can also be used to corral misspecified bandit base-algorithms on different bins in a discretized domain. We derive the following regret bound that is the same as UCB-Meta-algorithm.

Theorem 6. *First perform the smoothing transformation (Algorithm 3 in Pacchiano et al. (2020)) to our misspecified linear bandits in Algorithm 1, denote the smoothed misspecified linear bandits as $\mathcal{A}_s^{local}$. Then, the Meta-algorithm (Algorithm 5 (Corral-Update) reproduced in Pacchiano et al. (2020)) applied with a set of $\mathcal{A}_s^{local}$ that are initialized in the same way as in Algorithm 2 has expected regret upper bounded by:*

$$\mathbb{E}[R(T)] \leq \tilde{\mathcal{O}}(d(\alpha)T^{\frac{d+\alpha}{d+2\alpha}}). \tag{6}$$

The proof of this theorem is in Appendix section A.5.

3.5 Discussion

Choice of Meta-algorithm. The role of the Meta-algorithm is essentially model selection and adaptation to the base-algorithms. It is not a trivial task since the rewards incurred by the base-algrotihms are not i.i.d as in standard stochastic settings. However, UCB as a stochastic multi-armed bandit algorithm, is applicable as Meta-algorithm because the local parametric (linear) function approximations provide honest upper confidence bounds for each bin even under the misspecifications, thus enabling the distribution-independent analysis for UCB. The advantage of Corral-Meta is that it potentially allows relaxation of the Hölder smoothness to hold only around the global maxima (Auer et al. (2007); Bubeck et al. (2010)), while the same relaxation is not straightforward for UCB-Meta. The advantage of UCB is that under standard stochastic settings where each arm has i.i.d rewards, it achieves the gap-dependent bound of $\mathcal{O}(\log(T)/\Delta)$. Thus an interesting question for the future is whether similar gap-dependent bounds for the UCB-Meta is available. Such bounds would enable exploitation of the growth conditions (section 2) for potential rate improvements.

Computational aspects. At each round indexed by t, the algorithm chooses the maximizer of upper confidence bounds over n bins(complexity $\mathcal{O}(n)$). It also needs to choose the maximizer x_t of updated upper confidence bound in selected bin, based on locally fitted polynomial. Finding such maximizer is a non-convex problem if $\alpha > 2$ (akin to finding maximizers of acquisition functions in other complex models such as GP bandits). Hence it is hard to explicitly state the runtime dependence. However when $\alpha \leq 2$, Dani

et al. (2008) suggested a computationally efficient version of the LinUCB algorithm which suffers an extra $\sqrt{d}$ term in regret. Nevertheless, if we treat the LinUCB maximization as a black box, then the runtime of UCB-Meta-algorithm should be polynomial in n and polynomial in T.

3.6 Comparison with Existing Lower Bound

We compare the derived upper bounds of $\tilde{\mathcal{O}}(d(\alpha)T^{\frac{d+\alpha}{d+2\alpha}})$ to the existing lower bound from Wang et al. (2018), which study global optimization. In their work, the performance of optimization algorithms with output $\hat{x}_T$ is measured by simple regret $\mathcal{L}(\hat{x}_T; f) \triangleq f(x^*) - f(\hat{x}_T)$, for f in Hölder spaces including $\alpha \geq 1$. Theorem 2 (coupled with Proposition 3) in Wang et al. (2018) implies that $\sup_{f \in \sum(\alpha)} \mathbb{E}[\mathcal{L}(\hat{x}_T; f)] = \Omega(T^{\frac{-\alpha}{2\alpha+d}})$. We argue that this lower bound can be directly used to lower bound the worst-case cumulative regret, by making the following observation (remark 3 in Bubeck et al. (2010)): If a strategy achieves expected cumulative regret $\mathbb{E}[R_T]$, then by uniformly selecting a past action as the final output $\hat{x}_T$, it can also achieve expected simple regret $\mathbb{E}[\mathcal{L}(\hat{x}_T; f)] = \mathbb{E}[R_T]/T$. Therefore, any strategy with cumulative regret $\tilde{o}(T\mathbb{E}[\mathcal{L}(\hat{x}_T; f)])$ will violate the lower bound. Through proof by contradiction, we take the result from Wang et al. (2018) as an $\Omega(T^{\frac{d+\alpha}{d+2\alpha}})$ lower bound on expected cumulative regret, and argue that our results match this bound up to log factors. Our results show that proposed algorithms are minimax optimal in dependence on T and effectively exploit the function smoothness.

4 Adaptation to Unknown Smoothness

In this section, we study adaptation to the smoothness exponent α of the reward function. Minimax adaptation, which means a learner can simultaneously achieve the minimax optimal rates (Hoffmann et al. (2011); Locatelli and Carpentier (2018)) under a nested set of Hölder spaces, has been proven to be impossible for cumulative regret minimization without additional assumptions. Locatelli and Carpentier (2018) provide a lower bound for adaptation between two Hölder continuous functions spaces. Assume $\alpha < \gamma \leq 1$, for any strategy with a good expected regret $\mathbb{E}[R_\gamma(T)$ in $\sum(\gamma, L)]$, they show that its expected regret in the superset $\sum(\alpha, L)$ will depend inversely on $\mathbb{E}[R_\gamma(T)]$, and therefore be suboptimal for $\sum(\alpha, L)$. They propose a strategy to match that lower bound that requires values of α and γ, thereby also proving that the lower bound is tight.

However, when adapting to a continuous scale of Hölder spaces (possibly $\alpha \geq 1$), it remains unclear what strategy can generalize and achieve this lower bound for some Hölder spaces. We aim to answer that question by proposing a new strategy that uses a recently developed bandit model selection algorithm (Corral with smooth wrapper in Pacchiano et al. (2020)) applied with a set of Meta-algorithms (section 3). We will present this strategy and its theoretical guarantees next. Throughout the following sections, we refer to minimax optimal in terms of dependence of cumulative regret on T only unless otherwise specified.

4.1 Corral Applied with Meta-algorithms

The bandit model selection method Corral is first developed by Agarwal et al. (2016) and based on an instance of online mirror descent with mirror map derived from Foster et al. (2016). Corral with smooth wrapper proposed by Pacchiano et al. (2020) for stochastic feedback problems is different from the original Corral algorithm in the following aspects. The smoothed version no longer needs to send importance-weighted feedback to base-algorithm, therefore no longer requires the base-algorithms themselves to be modified for stability guarantee (definition 3 in Agarwal et al. (2016)). In the following parts, we will use Corral with smooth wrapper to adapt to the smoothness and refer to it as Corral for simplicity[11]. A copy of the pseudo-code of Corral from Pacchiano et al. (2020) can be found in Appendix B.2 for easier reference. We use a set of M Meta-algorithms $\mathcal{A}^{global}(\alpha_i), i \in [M]$ in Algorithm 2 as bases. The input values α_i are from a grid $\mathcal{G}$ defined later. Therefore, we first specify the regret of a Meta-algorithm with input smoothness parameter α' that is ran on functions with actual Hölder smoothness α.

Lemma 7. *For function f that satisfies global assumptions $G1 \sim G3$ with parameter α, the regret of Algorithm 2 with input parameter $\alpha' \leq \alpha$ is bounded with probability $1 - \delta$ by*

$$R(T) \leq \tilde{\mathcal{O}}(d(\alpha')T^{\frac{d+\alpha'}{d+2\alpha'}}). \qquad (7)$$

The bound does not hold for $\alpha' > \alpha$.

The proof is deferred to Appendix section A.6. Having established the performance of base algorithms with misspecified smoothness exponents, we present the adaptation strategy and its regret bound in Theorem 8. Since it is impossible to achieve minimax optimal rates for multiple values of the smoothness parameter simultaneously, we introduce a user-specified

---

[11]Since the core of oneline mirror descent in Corral is not changed.

parameter R that controls the Hölder space over which minimax optimality is desired. We show that conditioned on achieving minimax rate for the space $\sum(R, L)$, our adaptation strategy provides best possible regret bound on all supersets $\sum(\alpha, L)$ where $\alpha \leq R$. The results are stated in Theorem 8.

Theorem 8. *Consider adapting to a continuous scale of nested Hölder spaces indexed by α whose value is bounded in a given interval. For simplicity, we assume $0 < \alpha \leq 2$, where $d(\alpha) = d$. Define $R \leq 2$ as a parameter set by the decision-maker that specifies the index of Hölder space for which minimax optimal regret is achieved. Define linear grid $\mathcal{G} = \{\alpha_i = \frac{R}{\lfloor \log(T) \rfloor} i, i = 0, 1 \ldots \lfloor \log(T) \rfloor\}$ so that the total number of base algorithms is $M = |\mathcal{G}| = \lceil \log(T) \rceil$. Consider using Corral with bases that are Meta-algorithms (Algorithm 3 in Appendix section B.1) with input $\alpha_i \in \mathcal{G}, i \in [M]$. Then by setting the learning rate of Corral to be $\eta = (dT^{\frac{d+R}{d+2R}})^{-1}$, the regret rates achieved for any Hölder exponent $\alpha \in (0, 2]$ are:*

$$\sup_{f \in \sum(\alpha, L)} \mathbb{E}[R(T)] \leq \tilde{\mathcal{O}}(dT^{\frac{d^2+2Rd+R\alpha}{(d+2R)(d+\alpha)}}) \text{ for } \alpha \in (0, R],$$
(8)

$$\sup_{f \in \sum(\alpha, L)} \mathbb{E}[R(T)] \leq \tilde{\mathcal{O}}(dT^{\frac{d+R}{d+2R}}) \text{ for } \alpha \in [R, 2].$$
(9)

A straightforward example is shown in Figure 1. Functions with Hölder exponent $\alpha > R$ essentially belong to $\sum(R, L)$ and will therefore have the same regret rate as equation (9) because the algorithm did not fully exploit their smoothness. Similar results can be shown for α in a general given range, if both the linear grid and the sub-algorithms are chosen appropriately. There are two sources of cost of adaptation, first the cost of adapting to M grid points. Since $M = \mathcal{O}(\log(T))$, this has the same difficulty as the adaptation to two values in Locatelli and Carpentier (2018). The second one, however, is a consequence of adapting to a continuous scale of α. The cost is the rate difference between the exponent α and the closest value to it on $\mathcal{G}$, denoted $\hat{\alpha} \in \mathcal{G}$, s.t. $\hat{\alpha} \leq \alpha \leq \hat{\alpha} + \frac{R}{\lfloor \log(T) \rfloor}$. This cost can be alleviated by the design of the linear grid. We defer the full proof to Appendix section A.7.

4.2 Comparison with Existing Lower Bound for Adaptation

In this subsection, we compare the results in Theorem 8 to the existing lower bound in Locatelli and Carpentier (2018). Theorem 3 of Locatelli and Carpentier (2018) state that given two smoothness values $\alpha_1 < \alpha_2 \leq 1$, if a strategy has expected regret $\mathbb{E}[R_{\alpha_2}(T)]$ under exponent α_2 that is $\tilde{\mathcal{O}}(T^{\frac{d+\alpha}{d+2\alpha}})$, then the re-
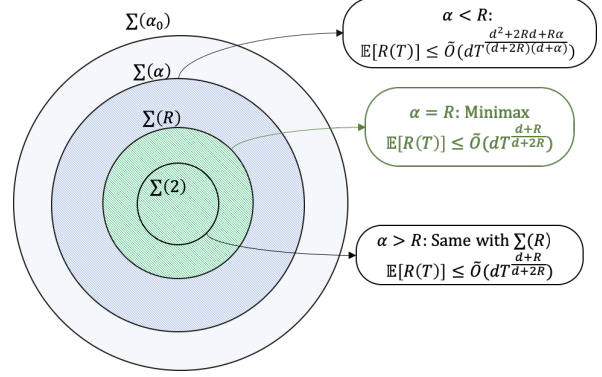

Figure 1: Illustration of adaptation to smoothness for continuous scale of Hölder spaces.

gret of this strategy under the superset characterized by α_1 is lower bounded by $\sup_{f \in \sum(\alpha_1, L)} \mathbb{E}[R(T)] \geq \tilde{\Omega}(T R_{\alpha_2}(T)^{\frac{-\alpha_1}{\alpha_1 + d}})$, even if the strategy has access to both α_1 and α_2.

We make the following remark: for any pair of exponent values (α_1, α_2) where $\alpha_1 < R$ and $R \leq \alpha_2 \leq 1$, the strategy proposed in Theorem 8 matches the lower bound except for log factors. We verify this by plugging in $\mathbb{E}[R_{\alpha_2}(T)] = \tilde{\mathcal{O}}(T^{\frac{d+R}{d+2R}})$, omitting dependence on d, to yield the lower bound on $\sum(\alpha_1, L)$ which is $\tilde{\mathcal{O}}(T^{\frac{d^2+2Rd+R\alpha_1}{(d+2R)(d+\alpha_1)}})$. This is matched by our upper bound in equation (8), apart from log factors and d. An illustration is shown in Figure 2. In other words, the proposed algorithm can perform under unknown smoothness exponent and match the lower bound (available only for exponent values within $(0, 1]$) on a subset of Hölder spaces.
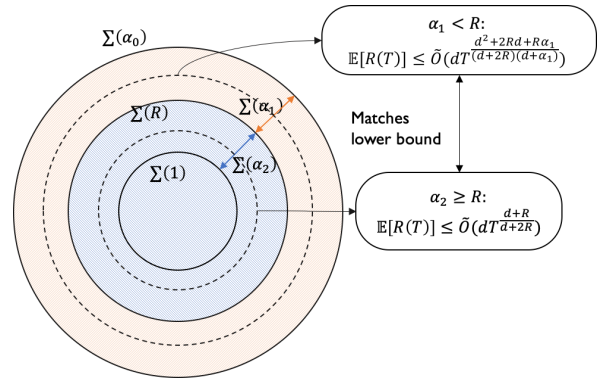

Figure 2: Illustration of values of exponents α_1, α_2 on which our proposed strategy matches the lower bound in Locatelli and Carpentier (2018).

5 Conclusion

The core of this paper is extending the assumption on function space from Lipschitz to Hölder spaces with higher-order smoothness in bandit optimization of black-box functions. We also study adaptation to the smoothness under this scope. The class of two-layer algorithms that we proposed consists of a Meta-algorithm with the choice of UCB (Auer et al. (2002)) or Corral (Agarwal et al. (2016); Pacchiano et al. (2020)) and a set of misspecified bandit base-algorithms as arms. We derive regret upper bounds for α-Hölder smooth functions with $\alpha > 1$ that matches existing lower bounds in their dependence on T, the number of active queries, with straightforward generalization to larger α. Our framework provides useful insights in exploiting higher-order smoothness of reward functions for cumulative regret minimization, because our two-layer structure allows base-algorithms to perform local exploration-exploitation tradeoff as opposed to the local pure exploration done for bandit optimization of α-Hölder continuous functions. For adaptation to the smoothness exponent, we further previous works by deriving regret upper bound for adaptation to a continuous scale of Hölder spaces with exponent α in a given range. We show that by using bandit model selection algorithms, it can achieve the existing lower bound between two Hölder spaces, even if the algorithm does not know both exponent values.

Our work inspires several directions for the future. An intriguing direction is to study whether there exist gap-dependent bounds for the UCB-Meta algorithm, whose arms have non-i.i.d rewards because they are bandit algorithms themselves. Such bounds could enable better rates for benign problem instances, for example with the growth conditions (mentioned in section 2). Another direction is the relaxation of the Hölder smooth assumption, to hold only around the maxima instead of everywhere on $\mathcal{X}$, which is considered by prior works such as Auer et al. (2007); Kleinberg et al. (2008); Bubeck et al. (2010). Finally, it remains an open problem to establish the lower bound for adaptation when the smoothness exponents are larger than 1.

Acknowledgements

References

Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.

A. Agarwal, H. Luo, B. Neyshabur, and R. E. Schapire. Corralling a band of bandit algorithms. *arXiv preprint arXiv:1612.06246*, 2016.

A. Akhavan, M. Pontil, and A. B. Tsybakov. Exploiting higher order smoothness in derivative-free optimization and continuous bandits. *arXiv preprint arXiv:2006.07862*, 2020.

R. Arora, T. V. Marinov, and M. Mohri. Corralling stochastic bandit algorithms. *arXiv preprint arXiv:2006.09255*, 2020.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 322–331. IEEE, 1995.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

P. Auer, R. Ortner, and C. Szepesvári. Improved rates for the stochastic continuum-armed bandit problem. In *International Conference on Computational Learning Theory*, pages 454–468. Springer, 2007.

S. Bubeck, R. Munos, G. Stoltz, and C. Szepesvari. X-armed bandits. *arXiv preprint arXiv:1001.4475*, 2010.

S. Bubeck, G. Stoltz, and J. Y. Yu. Lipschitz bandits without the lipschitz constant. In *International Conference on Algorithmic Learning Theory*, pages 144–158. Springer, 2011.

V. Dani, T. P. Hayes, and S. M. Kakade. Stochastic linear optimization under bandit feedback. 2008.

D. J. Foster, Z. Li, T. Lykouris, K. Sridharan, and E. Tardos. Learning in games: Robustness of fast convergence. In *Advances in Neural Information Processing Systems*, pages 4734–4742, 2016.

D. J. Foster, A. Krishnamurthy, and H. Luo. Model selection for contextual bandits. In *Advances in Neural Information Processing Systems*, pages 14741–14752, 2019.

J. A. Grant and D. S. Leslie. On thompson sampling for smoother-than-lipschitz bandits. *arXiv preprint arXiv:2001.02323*, 2020.

Y. Gur, A. Momeni, and S. Wager. Smoothness-adaptive stochastic bandits. *arXiv preprint arXiv:1910.09714*, 2019.

M. Hoffmann, R. Nickl, et al. On adaptive inference and confidence bands. *The Annals of Statistics*, 39 (5):2383–2409, 2011.

Y. Hu, N. Kallus, and X. Mao. Smooth contextual bandits: Bridging the parametric and non-differentiable

regret regimes. In *Conference on Learning Theory*, pages 2007–2010, 2020.

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

R. D. Kleinberg. Nearly tight bounds for the continuum-armed bandit problem. In *Advances in Neural Information Processing Systems*, pages 697–704, 2005.

A. Krishnamurthy, J. Langford, A. Slivkins, and C. Zhang. Contextual bandits with continuous actions: Smoothing, zooming, and adapting. *arXiv preprint arXiv:1902.01520*, 2019.

T. Lattimore and C. Szepesvari. Learning with good feature representations in bandits and in rl with a generative model. *arXiv preprint arXiv:1911.07676*, 2019.

A. Locatelli and A. Carpentier. Adaptivity to smoothness in x-armed bandits. In *Conference on Learning Theory*, pages 1463–1492, 2018.

M. G. Low et al. On nonparametric confidence intervals. *The Annals of Statistics*, 25(6):2547–2554, 1997.

A. Pacchiano, M. Phan, Y. Abbasi-Yadkori, A. Rao, J. Zimmert, T. Lattimore, and C. Szepesvari. Model selection in contextual stochastic bandit problems. *arXiv preprint arXiv:2003.01704*, 2020.

P. Rusmevichientong and J. N. Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

C. J. Stone. Optimal global rates of convergence for nonparametric regression. *The annals of statistics*, pages 1040–1053, 1982.

A. B. Tsybakov. *Introduction to nonparametric estimation*. Springer Science & Business Media, 2008.

M. J. Wainwright. *High-dimensional statistics: A nonasymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Y. Wang, S. Balakrishnan, and A. Singh. Optimization of smooth functions with noisy observations: Local minimax rates. In *Advances in Neural Information Processing Systems*, pages 4338–4349, 2018.

A. Yatchew. Nonparametric regression techniques in economics. *Journal of Economic Literature*, 36(2):669–721, 1998.