Title

Multi-Scale Zero-Order Optimization of Smooth Functions in an RKHS

Authors

Lee, Madison
Shekhar, Shubhanshu
Javidi, Tara

Publication Date

2022-07-01

DOI

10.1109/isit50566.2022.9834683

Multi-Scale Zero-Order Optimization of Smooth Functions in an RKHS

Madison Lee
mal017@ucsd.edu

Shubhanshu Shekhar
shubhan2@andrew.cmu.edu

Tara Javidi
tjavidi@eng.ucsd.edu

***Abstract*—Consider the problem of optimizing a black-box function under the assumption that the function is Hölder smooth and has bounded norm in the reproducing kernel Hilbert space associated with a given kernel. We propose the `LP-GP-UCB` algorithm which augments a Gaussian process surrogate model with local polynomial estimators of the function to construct a multi-scale upper confidence bound to guide the search for the optimizer. We provide high probability bounds on the cumulative regret in terms of the maximum information gain and smoothness parameters for the kernel. We then show that the Hölder smoothness assumption is satisfied for several commonly used and practically relevant kernels—the Matérn, rational-quadratic, γ-exponential, and piecewise-polynomial kernels—and obtain explicit regret bounds for them as a result. These regret bounds establish the near-optimality of `LP-GP-UCB` for these kernels and are also the first explicit bounds for many of them. Finally, we demonstrate the practical benefits experimentally.**

I. Introduction

Consider the problem of maximizing a black-box objective function $f : \mathcal{X} \mapsto \mathbb{R}$ which can only be accessed through the noisy observations $y_x = f(x)+\eta_x$ at query points $x \in \mathcal{X}$. Our goal is to design a query point selection strategy which uses a finite query (or evaluation) budget of n to efficiently learn about the maximizer x^* of f and recommend a point once the budget is exhausted. A performance metric we can use to evaluate such a querying strategy is the *cumulative regret* $\mathcal{R}_n$, defined as $\mathcal{R}_n = \sum_{t=1}^n f(x^*) - f(x_t)$, where x_t is the t^{th} query point.

This optimization task is uninteresting without any regularity assumptions on the objective function f. In this paper, we assume that f has a bounded norm in the reproducing kernel Hilbert space (RKHS) associated with a given kernel. This formulation is known as the kernelized continuous-armed bandit problem in the literature [1–3]. We propose an algorithm which exploits the existing smoothness properties of functions in the RKHSs associated with many commonly used kernels in order to obtain upper bounds on $\mathcal{R}_n$. Our algorithm is part of a family of algorithms for bandit optimization based on the GP-UCB algorithm, where a Gaussian process (GP) surrogate for f is used to derive an upper confidence bound (UCB) on the prediction of f in order to guide the search for the optimizer [4, 5].

We first introduce the notations used in this paper in Section I-A. We then provide an overview of our contributions in Section I-B and conclude the section with a discussion of related work in Section I-C.

A. Preliminaries

The objective function f maps $\mathcal{X} = [0,1]^D$ to $\mathcal{Y} = \mathbb{R}$. The function f can be accessed through noisy evaluations $y = f(x) + \eta$, where $x \in \mathcal{X}$ and the additive noise η is assumed to be σ^2-sub-Gaussian.

We use the term *cell* to refer to subsets E of $\mathcal{X}$ of the form $E = \{x \in \mathcal{X} : \|x - x_E\|_\infty \le r_E/2\}$. The terms x_E and r_E shall be referred to as the *center* and *side-length of E*. We let $\mathcal{D} = \{(x_i, y_i) \ : \ 1 \le i \le m\} \subset \mathcal{X} \times \mathcal{Y}$ denote a labelled data set, and introduce $\mathcal{D}_\mathcal{X} := \{x \ : \exists y \in \mathcal{Y}, (x,y) \in \mathcal{D}\}$ and $\mathcal{D}_\mathcal{Y} := \{y \ : \ \exists x \in \mathcal{X}, (x,y) \in \mathcal{D}\}$. For a cell $E \subset \mathcal{X}$, we use $\mathcal{D}^{(E)}$ to denote those (x_i, y_i) pairs in $\mathcal{D}$ such that $x_i \in E$. The sets $\mathcal{D}_\mathcal{X}^{(E)}$ and $\mathcal{D}_\mathcal{Y}^{(E)}$ are also defined in an analogous manner.

For positive integers q and D, we use $\mathcal{P}_D^q$ to denote the set of all polynomials in D variables of degree q. Given $g : \mathcal{X} \mapsto \mathbb{R}$ and $E \subset \mathcal{X}$, the smallest uniform approximation error of g with $p \in \mathcal{P}_D^q$ is denoted as $\Phi_q(g, E) := \inf_{p \in \mathcal{P}_D^q} \sup_{x \in E} |g(x) - p(x)|$.

Given a positive-definite kernel K, we shall use the term $\mathcal{H}_K$ and $\|\cdot\|_K$ to denote the RKHS associated with K and the corresponding RKHS norm. In particular, $\mathcal{H}_K$ is the completion of the inner product space consisting of functions in the linear span of K and the inner product defined by $\langle f, g\rangle_K = \sum_{i=1}^{m_1}\sum_{j=1}^{m_2} a_i b_j K(x_i, z_j)$ for functions $f = \sum_{i=1}^{m_1} a_i K(\cdot, x_i)$ and $g = \sum_{j=1}^{m_2} b_j K(\cdot, z_j)$. In this paper, we specialize our findings to the case of several commonly used and practically relevant kernels: Matérn with parameter ν, rational-quadratic, γ-Exponential, and piecewise-polynomial kernels. We let K_ν be the Matérn kernel with parameter ν, and subscript K with the abbreviated kernel name for other kernels, e.g. K_{RQ} for rational-quadratic kernel, etc.

For $k \in \mathbb{N}$ and $0 < \alpha \le 1$, we use $\mathcal{C}^{k,\alpha}$ to denote the Hölder space of order k and exponent α. In particular, $\mathcal{C}^{k,\alpha}$ contains functions for which the k^{th} partial derivatives are Hölder continuous with exponent α and the derivatives up to and including order k are continuous.

We use $\tilde{\mathcal{O}}(\cdot)$ to represent asymptotic upper bounds that hide the poly-logarithmic factors.

B. Overview of Results

We first formally state the assumptions on the objective function f and the observation noise.

Assumption 1. *We make the following assumptions:* ***(A1.1)*** *$f \in \mathcal{H}_K$ for some known kernel K and $\|f\|_K \le B$ for some*

TABLE I: Summary of Regret Bounds for Continuous Bandit Algorithms

	General Case	K_ν, $\nu \in (0, \frac{1}{2}]$	K_ν, $\nu > \frac{1}{2}$	K_{RQ}	$K_{\gamma\text{-Exp}}$ K_{PP}
Lower Bound [6]	DNE†	$\Omega\left(n^{\frac{\nu+D}{2\nu+D}}\right)$	$\Omega\left(n^{\frac{\nu+D}{2\nu+D}}\right)$	DNE	DNE
(I)GP-UCB [2, 5]	$\tilde{\mathcal{O}}\left(\gamma_n\sqrt{n}\right)$	DNE	$\tilde{\mathcal{O}}\left(n^{\frac{\nu+D(D+1)}{2\nu+D(D+1)}}\right)$	DNE	DNE
GP-ThreDS et al. [7–9]	$\tilde{\mathcal{O}}\left(\sqrt{n\gamma_n}\right)$	DNE	$\tilde{\mathcal{O}}\left(n^{\frac{\nu+D}{2\nu+D}}\right)$	DNE	DNE
LP-GP-UCB	Theorem 1	$\tilde{\mathcal{O}}\left(n^{\frac{\nu+D}{2\nu+D}}\right)$	$\tilde{\mathcal{O}}\left(\min\left\{n^{\frac{1}{2}+\frac{D(D+3)}{4\nu+D(D+5)}}, n^{\frac{D+1}{D+2}}\right\}\right)$	$\tilde{\mathcal{O}}\left(n^{\frac{D+1}{D+2}}\right)$	$\tilde{\mathcal{O}}\left(n^{\frac{2D+1}{2D+2}}\right)$

† DNE indicates that the explicit regret bounds do not exist for the corresponding method and function space.
†† The specific kernel bounds are assumed to be the minimum of the stated bound and the general one, if available.

known constant $B > 0$. ***(A1.2)*** $f \in \mathcal{C}^{k,\alpha}$ *for* $k \in \mathbb{N} \cup \{0\}$ *and* $\alpha \in (0, 1]$ *with* $\|f\|_{\mathcal{C}^{k,\alpha}} \leq L$ *for some known* $L > 0$. ***(A1.3)*** *the observation noise* $(\eta_t)_{t\geq 0}$ *are i.i.d and* σ^2*-sub-Gaussian for some known constant* $\sigma^2 > 0$.

Next, we list the main contributions of our paper:

- We propose a new algorithm (LP-GP-UCB in Sec. II) for the kernelized bandits problem that combines the global GP surrogate model (as in the GP-UCB algorithm of [1]) with local polynomial (LP) estimators to adaptively partition the search space and guide the search for the optimizer of f.
- We analyze LP-GP-UCB under Assumption 1, and show (Theorem 1 in Sec. III-A) that $\mathcal{R}_n$ can be bounded by the minimum of two terms: one depending on the maximum *information gain* γ_n of the kernel K, and the other depending on the smoothness parameters k and α.
- We then derive an embedding result (Proposition 1 in Sec. III-A) which shows that several common kernels satisfy Assumption 1. We use these results to specialize the regret bounds of LP-GP-UCB in Theorem 1 and obtain bounds that are the first known bounds explicit in n for the rational-quadratic, γ-exponential, and piecewise-polynomial kernels.

C. Prior Work

The kernelized bandits formulation was first introduced in [1], where the GP-UCB algorithm was proposed based on the UCB strategy for multi-armed bandits [4]. More specifically, given a positive-definite kernel K, the problem can be mapped to a zero-mean GP surrogate with covariance function K. An important quantity of interest associated with K is the *maximum information gain*

$$\gamma_n \coloneqq \max_{S\subset\mathcal{X},|S|=n} I\left(y_S; f\right), \tag{1}$$

where I denotes the mutual information between the zero-mean GP f with covariance K and a noisy observation vector y_S. The GP-UCB algorithm proceeds by selecting query points $(x_t)_{t\geq 1}$ that maximize the UCB index $\mu_t(x)+\beta_t\sigma_t(x)$, where μ_t and σ_t are the mean and standard deviation predictions for f based on evaluations prior to time t. GP-UCB and an improved version introduced in [2], IGP-UCB, admit upper bounds on $\mathcal{R}_n$ of the form $\mathcal{R}_n = \tilde{\mathcal{O}}\left(\sqrt{n}\gamma_n\right)$ under Assumption 1.1. For the Matérn kernel, this bound is always larger than the information-theoretic lower bound of $\Omega(n^{\frac{\nu+D}{2\nu+D}})$ derived in [6]. This is true for other extensions of GP-UCB as well [10–15].

In the special case of finite $|\mathcal{X}|$, KernelUCB [7], RIPS [8], and GP-ThreDS [9] are shown to admit a tighter regret bound of $\tilde{\mathcal{O}}\left(\sqrt{n\gamma_n}\right)$. Furthermore, as shown in [3, 9, 16], this improved bound can also be achieved in the continuous case if the kernel satisfies the additional smoothness constraint of Assumption 1.2. This includes Matérn kernels which we will show in Prop. 1 to satisfy Assumption 1.2. Recently, [17] explicitly characterized information gain of $\gamma_n = \tilde{\mathcal{O}}\left(n^{\frac{D}{2\nu+D}}\right)$ for the Matérn-ν kernel with $\nu > \frac{1}{2}$, indicating that these algorithms achieve the information theoretic lower bound of $\Omega(n^{\frac{\nu+D}{2\nu+D}})$ for $\nu > \frac{1}{2}$.

From an algorithmic point of view, LP-GP-UCB relies on two key ideas: (i) it adaptively constructs and updates a non-uniform partition of the domain $\mathcal{X}$, and (ii) it augments the global GP surrogate for f with local polynomial (LP) estimators on the elements of the partition. The first idea, i.e, adaptive non-uniform discretization of the domain $\mathcal{X}$, has been used for GP bandits [3, 18] and GP level-set estimation [19] and is known to be particularly suitable in high dimensions (the non-adaptive discretization proposed in [3, 16] makes the algorithms impractical when D is large). We note that the choice of LP estimators, which play a key role in both the query point selection strategy as well as in the refinement of the partition of $\mathcal{X}$, is uniquely motivated by our embedding results (Prop. 1) for several kernels of practical interest. The joint design of the local estimators and the partitioning scheme allows the LP-GP-UCB algorithm to zoom into successively smaller neighborhoods of the optimizer x^*, and consequently leads to a cumulative regret that is bounded, under Assumption 1, by the minimum of two terms: one depending on the maximum information gain of the form $\tilde{\mathcal{O}}\left(\sqrt{n}\gamma_n\right)$, and the other depending on the smoothness parameters k and α. This bound is specialized for specific kernels as summarized in Table I, providing the first bounds explicit in n for rational-quadratic, γ-exponential, and piecewise polynomial kernels. Furthermore, for Matérn kernels, the algorithm is shown to have competitive (and in some regimes, near-optimal) regret.

II. LP-GP-UCB Algorithm

We now describe the steps of our proposed algorithm, LP-GP-UCB (Algorithm 1).

Algorithm 1: LP-GP-UCB

Input: n, K, B, (q, α), L, δ, ρ_0.

1 **Initialize:** $t = 1$, $n_e = 0$, $\mathcal{P}_t = \{\mathcal{X}\}$, $u_{\mathcal{X}}^{(0)} = +\infty$, $\mathcal{D}_t = \emptyset$;
2 **while** $n_e < n$ **do**
3 **for** $E \in \mathcal{P}_t$ **do**
4 Draw $x_{t,E} \sim \texttt{Unif}(E)$
5 $U_{t,E} = \min\{u_E^{(0)},\ u_{t,E}^{(1)},\ u_{t,E}^{(2)}\}$
6 **end**
7 $E_t \in \arg\max_{E \in \mathcal{P}_t} U_{t,E}, \quad x_t = x_{t,E_t}$
8 $\mathcal{Q}_1 = \mathcal{Q}_2 = \{E_t\}$
9 **if** $\beta_n \sigma_t(x_t) < L(\sqrt{D} r_E)^{\alpha_1}$ *AND* $r_{E_t} \geq \rho_0$ **then**
10 $\mathcal{Q}_2 = \texttt{Partition}(E_t, r_{E_t}/2)$
11 **for** $F \in \mathcal{Q}_2$ **do** $u_F^{(0)} = u_{t,E_t}^{(1)}$;
12 **else if** $b_t(E_t) \leq L(\sqrt{D} r_E)^{\alpha_1}$ *AND* $r_{E_t} \geq \rho_0$ **then**
13 $\mathcal{Q}_2 = \texttt{Partition}(E_t, r_{E_t}/2)$
14 **for** $F \in \mathcal{Q}_2$ **do** $u_F^{(0)} = u_{t,E_t}^{(2)}$;
15 **else if** $b_t(E_t) \leq L(\sqrt{D} r_E)^{q+\alpha}$ *AND* $r_{E_t} \in [\frac{1}{n}, \rho_0)$ **then**
16 $\epsilon = \texttt{MaxErr}(E_t, \mathcal{D}_t, K, B, q, \delta, \sigma)$
17 $\tilde{r} = \min\left\{\frac{r_{E_t}}{2},\ \frac{1}{\sqrt{D}}\left(\frac{\epsilon}{L}\right)^{1/\alpha_1}\right\}$
18 $\mathcal{Q}_2 = \texttt{Partition}(E_t, \tilde{r})$
19 $\hat{f} = \texttt{LocalPoly}(F, \mathcal{D}_t, x_F)$
20 **for** $F \in \mathcal{Q}_2$ **do** $u_F^{(0)} = \hat{f} + 2\epsilon$;
21 **else**
22 Observe $y_t = f(x_t) + \eta_t$, Update μ_t, σ_t,
23 $n_e \leftarrow n_e + 1, \quad \mathcal{D}_t \leftarrow \mathcal{D}_t \cup \{(x_t, y_t)\}$
24 **end**
25 $\mathcal{P}_t \leftarrow (\mathcal{P}_t \setminus \mathcal{Q}_1) \cup \mathcal{Q}_2, \quad t = t + 1$
26 **end**

Output: z_n using Recommend function.

Inputs. LP-GP-UCB takes in as inputs the evaluation budget n, the kernel K, the RKHS norm bound B, the noise parameter σ, an integer q, an $\alpha \in (0, 1]$, the Hölder space norm bound L, a confidence parameter $\delta \in [0, 1]$, and a real-number ρ_0. It defines $\alpha_1 := \max\{\alpha, \min\{1, q\}\}$.

Algorithm Outline. LP-GP-UCB maintains a partition, $\mathcal{P}_t$, of the domain $\mathcal{X}$ at any time t, and to each cell $E \in \mathcal{P}_t$, it assigns a UCB $u_E^{(0)}$ on the maximum value of f in the cell, calculated using local estimates based on prior observations.

At $t = 1$, $\mathcal{P}_t$ is initialized as $\{\mathcal{X}\}$ and $u_{\mathcal{X}}^{(0)}$ is set to $+\infty$. As new cells E are added to $\mathcal{P}_t$, the value of $u_E^{(0)}$ is determined by the cell sizes and confidence interval widths.

For every $t \geq 1$, the LP-GP-UCB algorithm loops through all the cells in $\mathcal{P}_t$, and constructs a UCB denoted by $U_{t,E} = \min\{u_E^{(0)}, u_E^{(1)}, u_E^{(2)}\}$, with $u_{t,E}^{(1)}$ and $u_{t,E}^{(2)}$ defined as

$$u_{t,E}^{(1)} = \mu_t(x_{t,E}) + \beta_n \sigma_t(x_{t,E}) + L(\sqrt{D} r_E)^{\alpha_1}$$
$$u_{t,E}^{(2)} = \hat{\mu}_t(E) + b_t(E) + L(\sqrt{D} r_E)^{\alpha_1},$$

where $x_{t,E}$ is a point drawn uniformly from E, μ_t and σ_t are the posterior mean and variance of the surrogate GP regressor model, β_t is a confidence width multiplier which we set to $\beta_t = B + \sigma\sqrt{2(\gamma_t + 1 + \log(1/\delta))}$, $\hat{\mu}_t(E) = \frac{1}{|\mathcal{D}_{\mathcal{Y}}^{(E)}|} \sum_{y \in \mathcal{D}_{\mathcal{Y}}^{(E)}} y$ is the empirical estimate of $\tilde{f}_E$ (the average value of f in the cell E), $b_t(E) = \sigma\sqrt{2\log((n^D \pi^2 t^2)/2\delta)/n_{t,E}}$ is the length of the confidence interval for $\tilde{f}_E$, and $L(\sqrt{D} r_E)^{\alpha_1}$ is an upper bound on the function variation across the cell.

Then, the algorithm selects a candidate cell E_t and the corresponding point x_t with the largest value of $U_{t,E}$, and decides to either expand the partition or evaluate the function at the point x_t.

When the budget n is exhausted, the algorithm recommends the point z_n which is either the evaluation x_t with the smallest confidence interval width or the center of the smallest cell depending on whether the function variation in the smallest cell or the confidence interval width for x_t is smaller. This selection is described in the following definition:

Definition 1 (Recommend)**.** Suppose the algorithm stops in round t_n and let T denote the set of times at which function evaluations were performed. Define $E_n \in \arg\min_{E \in \mathcal{P}_{t_n}} r_E$, and $\tau := \arg\min_{t \in \mathrm{T}} \beta_t \sigma_t(x_t)$. If $L(\sqrt{D} r_{E_n})^{\alpha_1} \leq \beta_\tau \sigma_\tau(x_\tau)$, then return $z_n = x_{E_n}$. Else, return $z_n = x_\tau$.

A. Partition and Local Polynomial Construction

Updates to the partition $\mathcal{P}_t$ using the Partition function occur when the cell that maximizes the UCB index has a small confidence interval width.

Definition 2 (Partition)**.** Given a cell $E = \bigtimes_{i=1}^{D} [a_i, b_i] \subset \mathcal{X}$, the function call $\texttt{Partition}(E, r)$ for some $r < r_E$ returns a partition of E of cardinality $\lceil r_E/r \rceil^D$, consisting of sets of the form $F = \bigtimes_{i=1}^{D} [\tilde{a}_i, \min\{\tilde{a}_i + r, b_i\}]$, where $\tilde{a}_i = a_i + lr$ for $l \in \{0, 1, \ldots, \lfloor r_E/r \rfloor\}$.

In particular, when the cell size is sufficiently large and the corresponding confidence interval is small relative to the function variation in the cell (lines 9 and 12), we split the cell boundaries in half, creating 2^D new cells. When the cell size is small enough and the empirical confidence interval width is smaller than the maximum polynomial approximation error (line 15), we compute a local polynomial (LP) approximation of the function and the associated maximum error to guide the partition sizing and UCB computation. We next describe the construction of the LP estimators.

Given a cell $E \subset \mathcal{X}$ and a point $z \in E$, we define the LP estimator at z as $\hat{f}_E(z, \vec{w}) = \sum_{x \in \mathcal{D}^{(E)}} w_x y_x$, where the 'interpolation weights' $\vec{w} = \{w_x : x \in \mathcal{D}_{\mathcal{X}}^{(E)}\}$ are defined as the solution to the following problem [20, Eq. (1.36)]:

$$\vec{w} = \arg\min_{\vec{v} = \{v_x : x \in \mathcal{D}_{\mathcal{X}}^{(E)}\}} \sum_{x \in \mathcal{D}_{\mathcal{X}}^{(E)}} |v_x|^2 \quad \text{s.t.} \quad p(z) = \sum_{x \in \mathcal{D}_{\mathcal{X}}^{(E)}} v_x p(x) \quad \forall p \in \mathcal{P}_D^q. \tag{LP}$$

If the number of data points in the cell E, $|\mathcal{D}_{\mathcal{X}}^{(E)}|$, is larger than $(q+2)^D$, then (LP) is solvable and its optimal solution

is unique [20, Lem. 1.3.1]. Accordingly, we approximate the function using `LocalPoly` and bound the error using `MaxErr` as we describe next.

Definition 3 (`LocalPoly`)**.** Given a cell E and a point $x \in E$, the function `LocalPoly` returns the estimated function value $\hat{f}_E(x, \vec{w})$ at x, calculated according to the formula stated above. If $n_E := |\mathcal{D}^{(E)}| > (q+2)^D$, then $\vec{w}$ is the unique solution to (**LP**), while if $n_E < (q+2)^D$, the weights are set as $w_x = 1/n_E$ for all $x \in \mathcal{D}_{\mathcal{X}}^{(E)}$.

Definition 4 (`MaxErr`)**.** The function `MaxErr` takes E, $\mathcal{D}$, K, B, q, δ and σ as inputs, and returns ϵ defined as

$$\epsilon := \max_{x \in E} (1 + \|\vec{w}_{E,x}\|_1) L(\sqrt{D} r_E)^{q+\alpha} + \sigma \|\vec{w}_{E,x}\|_2 \sqrt{2 \log \frac{2}{\delta}},$$

where $\vec{w}_{E,x}$ denotes the solution to (**LP**) at x.

Remark 1. Recall that the smallest uniform approximation error $\Phi_q(f, E)$ depends on how well polynomials in $\mathcal{P}_D^q$ approximate elements of $\mathcal{H}_K$. For functions f with $\|f\|_{\mathcal{C}^{q,\alpha}} \leq L$, we know $\Phi_k(f, E) \leq L(\sqrt{D} r_E)^{q+\alpha}$. We use this fact and an upper bound on the estimation error between $\hat{f}_E(x, \vec{w})$ and $f(x)$ from [20, Prop. 1.3.1] to define the `MaxErr` function.

This concludes our description of `LP-GP-UCB` .

III. Regret Analysis of `LP-GP-UCB`

We now state the main result of this section which provides high-probability regret bounds for `LP-GP-UCB` and leads to the bounds presented in Table I. Recall first that $\tilde{\mathcal{O}}(\cdot)$ hides the poly-logarithmic factors, and that $\alpha_1 = \max\{\alpha, \min\{1, q\}\}$.

Theorem 1. *Suppose Assumption 1 holds, and Algorithm 1 is run with a budget n, $q = k$, and inputs as described in Section II. Then with probability at least $1 - \delta$ for a given $\delta \in (0, 1)$:*

$$\mathcal{R}_n = \tilde{\mathcal{O}}\left(\gamma_n \sqrt{n}\right). \tag{2}$$

In addition, the following smoothness-dependent bounds hold for sufficiently large n:

$$\mathcal{R}_n = \tilde{\mathcal{O}}\left(n^{\frac{2(k+\alpha)-\alpha_1+D}{2(k+\alpha)+D}}\right), \text{ if } \gamma_n = \Omega(\sqrt{n}), \tag{3}$$

$$\mathcal{R}_n = \tilde{\mathcal{O}}\left(n^{\frac{\alpha_1+D}{2\alpha_1+D}}\right), \text{ otherwise.} \tag{4}$$

For the full proof of this statement, the reader is referred to the arXiv preprint of this work [21, App. C].

Remark 2. Since the bounds given in (2) and one of (3) or (4) hold simultaneously under the $1 - \delta$ probability event, the bound resulting by taking their minimum is always as good as existing algorithms with regret $\tilde{\mathcal{O}}(\gamma_n \sqrt{n})$.

In the next section, we specialize the results of Theorem 1 to specific kernels of significant theoretical and practical value.

A. Regret Bounds for Specific Kernels

We begin the analysis with a key embedding result which says that we can identify elements of the RKHSs associated with the Matérn, square-exponential (SE), rational-quadratic (RQ), γ-exponential (γ-Exp), and piecewise-polynomial kernels with elements of certain Hölder spaces. Details on these kernels may be found in [22, Ch. 4].

Proposition 1. *If f is in the RKHS $\mathcal{H}_K$ associated to a kernel $K \in \{K_\nu,\ K_{SE},\ K_{RQ},\ K_{\gamma\text{-}Exp},\ K_{PP}\}$, then there exists a constant $C_K \in (0, \infty)$, $k \in \mathbb{N}$, and $\alpha \in (0, 1]$, such that*

$$\|f\|_{\mathcal{C}^{k,\alpha}} \leq C_K \|f\|_K. \tag{5}$$

The proof of this statement relies on the reproducing property of $\mathcal{H}_K$ and the norm-equivalence of RKHSs with certain fractional Sobolev spaces as described in [21, App. B].

Remark 3. For the Matérn kernel, (5) can be shown to hold for $(k, \alpha) = (\lceil \nu \rceil - 1, \nu - k)$. For the RQ kernel, (5) holds for $(k, \alpha) = (0, \frac{1}{2})$, and for the γ-Exp and PP kernels it holds for $(k, \alpha) = (0, 1)$. A bound on C_K can be computed in terms of the parameters for these kernels [21, App. B].

For simplicity, we make the following assumption and then state the regret bounds for the Matérn, SE, RQ, γ-Exp, and PP kernels as a special case of Theorem 1.

Assumption 2. *We assume that for $K = K_\nu$, n is large enough to ensure that $C_K \leq \hat{C}_K \log(n)$, where C_K is the constant introduced in Proposition 1, and $\hat{C}_K$ can be computed explicitly in terms of the kernel parameters.*

Proposition 2. *Suppose Assumptions 1.1 and 2 hold, and that Algorithm 1 is run with a budget n, kernel $K \in \{K_\nu,\ K_{SE},\ K_{RQ},\ K_{\gamma\text{-}Exp},\ K_{PP}\}$, $q = k$ and α according to Prop. 1, $L = BC_K$, and other inputs as described in Sec. II. Then with probability at least $1 - \delta$, the following holds:*

$$\mathcal{R}_n = \tilde{\mathcal{O}}\left(\min\left\{\sqrt{n}\gamma_n,\ n^{\frac{D+\alpha}{D+2\alpha}}\right\}\right).$$

Remark 4. The above result follows from a combination of Theorem 1 and the embedding results of Proposition 1, and employs the bounds on γ_n derived in [1]. The explicit regret bounds for these kernels are displayed in Table I.

Since n-dependent bounds on γ_n are not known for K_{RQ}, $K_{\gamma\text{-Exp}}$, K_{PP}, the result of Proposition 2 provides us with the first regret bounds that are explicit in n. However, as there do not exist any algorithm-independent lower bounds for these kernels, it is not clear how sub-optimal these bounds are.

For the Matérn kernel, when $\nu \in (0, 1]$ we set $\alpha = \nu$, and so our bound matches the lower bound given in [6] up to poly-log factors. Since γ_n-based bounds are known only for $\nu > \frac{1}{2}$, our results provide the first explicit and near-optimal regret bounds for Matérn kernels in the regime $\nu \in (0, \frac{1}{2}]$, which includes, for example, the absolute exponential kernel.

IV. Empirical Results

In this section, we empirically illustrate our finding on benchmark functions as well as a hyperparameter tuning task. In all the experiments, we use the Matérn kernel K_ν with $\nu = 2.5$, and set $B = 1$, $L = \sqrt{2}$, $\sigma = 0.1$ and $\delta = 0.001$.

Algorithms. We consider the following algorithms: **(1)** *LP0*: This is the simplest version of our `LP-GP-UCB` algorithm,

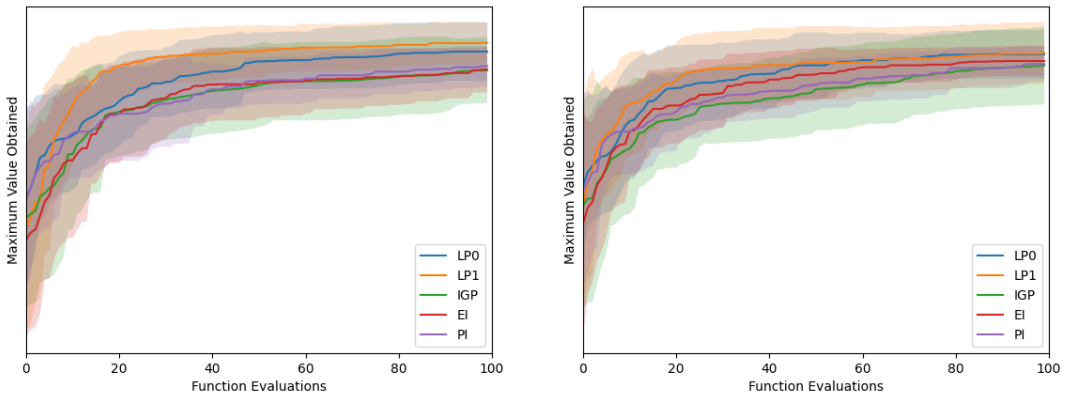

Fig. 1: Average maximum values obtained from 30 trials of optimization of synthetic 8-dimensional Branin (left) and Goldstein (right) functions with underlying low-dimensional structure.

which uses $q = 0$ and $\alpha = 1$, **(2)** LP1, the first-order version of `LP-GP-UCB` with $q = 1$ and $\alpha = 1$, **(3)** *IGP-UCB*, the improved GP-UCB algorithm of [2], **(4)** *EI*, Expected Improvement, and **(5)** *PI*, Probability of Improvement.

Expt. 1: Benchmark functions. The goal of the first experiment is to test if the algorithms can detect underlying structure in the objective functions. In particular, we construct an 8-dimensional function $f : \mathbb{R}^8 \mapsto \mathbb{R}$ by additively extending a 2-dimensional benchmark function g as follows: $f(x) = \sum_{i=1}^{t} c_i g(x[2i-1:2i])$ where $c_1 = 1$ and $c_i = 0.1$ for $i = 2, 3, 4$. For the 2-dimensional benchmark function g, we use the standard Branin and Goldstein functions [23]. Figure 1 plots the maximum value suggested by each algorithm against the number of evaluations or samples used. The figure shows the average maximum values over 30 trials as well as the standard deviation of the values. As we can see, the adaptive partitioning approach allows the `LP-GP-UCB` algorithm to better exploit the simple structure in the objective function and find higher value points. The use of first-order polynomial approximations in LP1 allows the algorithm to sample and partition more effectively than LP0 and converges to the maximum with fewer samples than LP0 as a consequence.

Expt 2: Hyperparameter Tuning. In this experiment, we use LP0 and the other optimization algorithms to select the best hyperparameters of a convolutional neural network (CNN) with two convolutional layers and two fully connected layers. The hyperparameters to be optimized were `batch_size`, the `learning_rate`, the `kernel_size` of the two convolutional layers and the `hidden_nodes` in the first fully-connected layer. The objective to be maximized was the accuracy on the test set. The mean accuracy of the point recommended by the algorithms over 10 trials (each with a budget of $n = 50$ evaluations) is shown in Table II. As indicated by the values, LP0 achieves high classification accuracy with relatively small variability.

TABLE II: Performance on hyperparameter tuning task over 10 trials with 50 iterations each.

Method	Mean Accuracy	Std. Deviation
LP0	90.117	1.479
IGP-UCB	85.027	10.981
EI	88.892	1.679
PI	86.692	2.966

V. Conclusion and Future Work

In this paper, we proposed a new algorithm, `LP-GP-UCB`, for kernelized bandits and obtained high probability bounds on its cumulative regret. For the Matérn family of kernels $(K_\nu)_{\nu>0}$, we derived regret bounds which are novel or near-optimal in certain ranges of ν. We also obtained the first explicit regret bounds, to our knowledge, for some important kernels such as rational-quadratic, γ-exponential, and piecewise-polynomial kernels. Experimental evaluation on some benchmark functions as well as on a hyperparameter tuning task suggest that the proposed multi-scale partitioning approach may be adapted for practical problems.

Our work opens several interesting directions for future research: **(1)** improving the cubic computational complexity associated with the exact GP inference in our implementations by using techniques such as adaptive sketching [24], **(2)** extension of the algorithmic techniques of this paper to related topics such as contextual GP bandits, GP level set estimation, and parallel GP bandits. Since the existing theoretical results for these problems also depend on γ_n, the methods of our paper may potentially lead to significant improvements in these problems as well.

VI. Acknowledgments

This work is supported in part by NSF CCF-2112665 (TILOS) and the industrial sponsors of UCSD Center for Machine Intelligence, Computing, and Security.

References

[1] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, 2012.

[2] S. R. Chowdhury and A. Gopalan, "On kernelized multi-armed bandits," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 844–853.

[3] D. Janz, D. R. Burt, and J. González, "Bandit optimisation of functions in the Matérn kernel RKHS," *arXiv preprint arXiv:2001.10396*, 2020.

[4] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time Analysis of the Multiarmed Bandit Problem," *Machine learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[5] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Gaussian process bandits without regret: An experimental design approach," *CoRR*, vol. abs/0912.3995, 2009. [Online]. Available: http://arxiv.org/abs/0912.3995

[6] J. Scarlett, I. Bogunovic, and V. Cevher, "Lower bounds on regret for noisy gaussian process bandit optimization," in *Conference on Learning Theory*, 2017, pp. 1723–1742.

[7] M. Valko, N. Korda, R. Munos, I. Flaounas, and N. Cristianini, "Finite-time analysis of kernelised contextual bandits," *arXiv preprint arXiv:1309.6869*, 2013.

[8] R. Camilleri, K. Jamieson, and J. Katz-Samuels, "High-dimensional experimental design and kernel bandits," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 1227–1237. [Online]. Available: https://proceedings.mlr.press/v139/camilleri21a.html

[9] S. Salgia, S. Vakili, and Q. Zhao, "A domain-shrinking based bayesian optimization algorithm with order-optimal regret performance," in *Advances in Neural Information Processing Systems*, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021. [Online]. Available: https://openreview.net/forum?id=MR4I3CjpeCv

[10] T. Desautels, A. Krause, and J. W. Burdick, "Parallelizing Exploration-Exploitation Trade-offs in Gaussian Process Bandit Optimization," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3873–3923, 2014.

[11] E. Contal, D. Buffoni, A. Robicquet, and N. Vayatis, "Parallel Gaussian Process Optimization with Upper Confidence Bound and Pure Exploration," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2013, pp. 225–240.

[12] A. Krause and C. S. Ong, "Contextual Gaussian Process Bandit Optimization," in *Advances in neural information processing systems*, 2011, pp. 2447–2455.

[13] K. Kandasamy, J. Schneider, and B. Póczos, "High dimensional Bayesian Optimisation and Bandits via Additive Models," in *International Conference on Machine Learning*, 2015, pp. 295–304.

[14] I. Bogunovic, J. Scarlett, S. Jegelka, and V. Cevher, "Adversarially Robust Optimization with Gaussian Processes," in *Advances in Neural Information Processing Systems*, 2018, pp. 5760–5770.

[15] K. Kandasamy, G. Dasarathy, J. B. Oliva, J. Schneider, and B. Póczos, "Gaussian process bandit optimisation with multi-fidelity evaluations," in *Advances in Neural Information Processing Systems*, 2016, pp. 992–1000.

[16] Z. Li and J. Scarlett, "Gaussian process bandit optimization with few batches," 2021. [Online]. Available: https://arxiv.org/abs/2110.07788

[17] S. Vakili, K. Khezeli, and V. Picheny, "On information gain and regret bounds in gaussian process bandits," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 82–90.

[18] S. Shekhar, T. Javidi *et al.*, "Gaussian process bandits with adaptive discretization," *Electronic Journal of Statistics*, vol. 12, no. 2, pp. 3829–3874, 2018.

[19] S. Shekhar and T. Javidi, "Multiscale gaussian process level set estimation," in *The 22nd International Conference on Artificial Intelligence and Statistics*, 2019, pp. 3283–3291.

[20] A. Nemirovski, "Topics in Non-parametric Statistics," *Ecole d'Eté de Probabilités de Saint-Flour*, vol. 28, p. 85, 2000.

[21] S. Shekhar and T. Javidi, "Multi-scale zero-order optimization of smooth functions in an RKHS," *CoRR*, vol. abs/2005.04832, 2020. [Online]. Available: https://arxiv.org/abs/2005.04832

[22] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2.

[23] M. Jamil and X.-S. Yang, "A literature survey of benchmark functions for global optimisation problems," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 4, no. 2, pp. 150–194, 2013.

[24] D. Calandriello, L. Carratino, A. Lazaric, M. Valko, and L. Rosasco, "Gaussian process optimization with adaptive sketching: Scalable and no regret," in *Conference on Learning Theory*, 2019, pp. 533–557.