

# LAPLACIAN KERNELIZED BANDIT

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

We study multi-user contextual bandits where users are related by a graph and their reward functions exhibit both non-linear behavior and graph homophily. We introduce a principled joint penalty for the collection of user reward functions  $\{f_u\}$ , combining a graph smoothness term based on RKHS distances with an individual roughness penalty. Our central contribution is proving that this penalty is equivalent to the squared norm within a single, unified *multi-user RKHS*. We explicitly derive its reproducing kernel, which elegantly fuses the graph Laplacian with the base arm kernel. This unification allows us to reframe the problem as learning a single "lifted" function, enabling the design of principled algorithms, LK-GP-UCB and LK-GP-TS, that leverage Gaussian Process posteriors over this new kernel for exploration. We provide high-probability regret bounds that scale with an *effective dimension* of the multi-user kernel, replacing dependencies on user count or ambient dimension. Empirically, our methods outperform strong linear and non-graph-aware baselines in non-linear settings and remain competitive even when the true rewards are linear. Our work delivers a unified, theoretically grounded, and practical framework that bridges Laplacian regularization with kernelized bandits for structured exploration.

## 1 INTRODUCTION

Graphs are pervasive in modern sequential decision-making, encoding similarity or interaction among entities like users, items, or sensors. In a multi-user contextual bandit setting, this graph structure is informative since it provides a pathway to share information, allowing an algorithm to learn more efficiently than if it treated each user in isolation. We study the problem where a known user graph promotes *homophily*, meaning connected users tend to have similar reward functions. At each round  $t$ , a learner observes a user  $u_t$  and a set of available arms (contexts)  $\mathcal{D}_t \subset \mathbb{R}^d$ , selects an arm  $x_t \in \mathcal{D}_t$ , and receives a noisy reward  $y_t$ . Naively learning a separate model for each user is inefficient, leading to regret that scales with the number of users. Exploiting the graph structure, however, can yield dramatic improvements in both sample efficiency and performance Szorenyi et al. (2013); Landgren et al. (2016); Gong & Zhang (2025); Wang et al. (2025).

This problem was first formalized as the Gang of Bandits (GOB) Cesa-Bianchi et al. (2013), which models the collection of user reward functions  $\{f_u(\cdot)\}_{u=1}^n$  as a *smooth signal* on the graph. Seminal works like GOB.LIN Cesa-Bianchi et al. (2013) assume linear reward functions,  $f_u(x) = \theta_u^\top x$ , and penalize roughness via the graph Laplacian, leading to the effective linear bandit solution. Subsequent research has extended this approach with improved computational scaling Vaswani et al. (2017); Yang et al. (2020), but has largely remained within the linear paradigm. Yet, in many applications, from recommendation systems to personalized medicine, reward functions exhibit complex, non-linear behavior. While a rich literature on kernelized bandits exists to handle non-linear rewards for a single agent Chowdhury & Gopalan (2017); Du et al. (2021); Bubeck et al. (2021); Li et al. (2022); Zhou & Ji (2022), principled methods for the multi-user graph setting are less developed. Existing approaches construct a multi-user kernel heuristically as a product of user and arm kernels Dubey et al. (2020), leaving a gap between the intuitive modeling goal and the final algorithm.

Our work bridges this gap, starting from a natural first principle for this problem. A desirable collection of reward functions  $\{f_u\}_{u=1}^n$ , where each  $f_u$  lies in a Reproducing Kernel Hilbert Space (RKHS)  $\mathcal{H}_x$ , should be jointly regularized: they should be smooth across the graph (homophily) and individually well-behaved (low complexity). We formalize this via an intuitive, additive penalty that combines a graph smoothness term with a standard ridge penalty. One of our key contributions is

to demonstrate that this penalty is not merely an ad-hoc regularizer; it corresponds **precisely** to the squared RKHS norm of a single, lifted function  $f(\mathbf{x}, u) := f_u(\mathbf{x})$  in a unified *multi-user RKHS*. We explicitly derive the reproducing kernel for this space, which elegantly fuses the graph Laplacian  $\mathbf{L}$  and the base arm kernel  $K_x$ :

$$K((\mathbf{x}, u), (\mathbf{x}', u')) = [\mathbf{L}_\rho^{-1}]_{u, u'} K_x(\mathbf{x}, \mathbf{x}'),$$

where  $\mathbf{L}_\rho = \mathbf{L} + \rho \mathbf{I}$  is the regularized Laplacian.

This unifying perspective transforms the problem of learning  $n$  related functions into the elegant problem of learning a single function in a well-defined kernel space. It allows us to directly apply the powerful machinery of Gaussian Process (GP) bandits Srinivas et al. (2009); Krause & Ong (2011); Vakili et al. (2021). We develop LK-GP-UCB and LK-GP-TS, algorithms whose principled uncertainty estimates are derived from the GP posterior of this unified kernel, enabling them to naturally and jointly leverage non-linear arm structure and Laplacian homophily. We provide regret guarantees for these algorithms in terms of an *effective dimension* that captures the spectral interplay between the graph and the kernel. Our experiments show that our methods are competitive in linear regimes and substantially outperform both linear and non-graph-aware baselines when rewards are non-linear yet graph-smooth.

Our main contributions are:

- We formalize the generalized gang-of-bandits problem with a principled joint penalty combining graph smoothness and RKHS regularity for the collection of reward functions  $\{f_u\}_{u=1}^n$ .
- We prove that this penalty is equivalent to the squared norm in a single multi-user RKHS and explicitly derive its reproducing kernel, unifying the graph and arm structures.
- We develop LK-GP-UCB and LK-GP-TS, GP-based bandit algorithms that leverage this unified kernel for principled and effective exploration.
- We provide novel regret bounds in terms of an effective dimension that depends on the spectral properties of both the kernel and the graph Laplacian.
- We empirically validate our approach, demonstrating significant performance gains over strong baselines in settings with non-linear, graph-smooth reward structures.

**Notations.** Let  $[n]$  be set  $\{1, 2, \dots, n\}$ . For a set or event  $\mathcal{E}$ , we denote its complement as  $\bar{\mathcal{E}}$ . Vectors are assumed to be column vectors.  $\mathbf{e}_i$  is the  $i$ -th canonical basis vector in  $\mathbb{R}^n$ .  $\mathbf{I}$  is the identity matrix.  $\lambda_{\min}(\mathbf{A})$  represents the minimum eigenvalue of matrix  $\mathbf{A}$ .  $\otimes$  is the Kronecker product. Denote the history of randomness up to (but not including) round  $t$  as  $\mathcal{F}_t$  and write  $\mathbb{P}_t(\cdot) := \mathbb{P}(\cdot | \mathcal{F}_t)$  and  $\mathbb{E}_t(\cdot) := \mathbb{E}[\cdot | \mathcal{F}_t]$  for the conditional probability and expectation given  $\mathcal{F}_t$ . We use  $\tilde{O}$  for big- $O$  notation up to logarithmic factor.

## 2 PROBLEM FORMULATION

### 2.1 GANG OF BANDITS WITH NON-LINEAR REWARDS

We consider a multi-user contextual bandit problem, often called Gang of Bandits (GOB) Cesa-Bianchi et al. (2013), with  $n$  users and a potentially infinite set of arms. We denote the set of users as  $\mathcal{U} = \{1, \dots, n\}$  and the arm set as  $\mathcal{D} \subseteq \mathbb{R}^d$ , where each arm is represented by a feature vector  $\mathbf{x} \in \mathcal{D}$ . The users are connected by a known undirected graph  $G = (\mathcal{U}, \mathcal{E})$ , where  $\mathcal{E}$  is the set of edges. Let  $\mathbf{W} \in \mathbb{R}^{n \times n}$  be the matrix of non-negative edge weights  $w_{ij}$ , and  $\mathbf{D}$  be the diagonal degree matrix with entries  $d_i := \sum_j w_{ij}$ . The corresponding graph Laplacian is  $\mathbf{L} := \mathbf{D} - \mathbf{W}$ .

The learning process unfolds over  $T$  rounds. At each round  $t \in \{1, \dots, T\}$ , the environment presents a user  $u_t \in \mathcal{U}$  (for example, randomly/uniformly pick one) and a finite subset of available arms  $\mathcal{D}_t \subseteq \mathcal{D}$ . The learner selects an arm  $\mathbf{x}_t \in \mathcal{D}_t$  following some decision policy  $\pi$  and observes a noisy reward:  $y_t = f_{u_t}(\mathbf{x}_t) + \epsilon_t$  where  $\{f_u : \mathcal{D} \rightarrow \mathbb{R}\}_{u=1}^n$  is a collection of unknown reward functions, one for each user. The noise term  $\epsilon_t$  is assumed to be conditionally zero-mean and sub-Gaussian with variance proxy  $\sigma^2$ , given the history of interactions  $\mathcal{F}_t$ . For the illustrative purpose, we use  $f_{1:n} := \{f_u\}_{u=1}^n$  as the collection of the user-level reward functions.

The learner’s objective is to minimize cumulative regret. The instantaneous regret incurred at time  $t$  is  $\Delta_t = f_{u_t}(\mathbf{x}_t^*) - f_{u_t}(\mathbf{x}_t)$  where  $\mathbf{x}_t^* = \arg \min_{\mathbf{x} \in \mathcal{D}_t} f_{u_t}(\mathbf{x})$  and cumulative regret over  $T$  rounds is defined as  $\mathcal{R}_T = \sum_{t=1}^T \Delta_t$ . A successful algorithm must achieve sub-linear regret,  $\mathcal{R}_T/T \rightarrow 0$  as  $T \rightarrow \infty$ , ensuring that the average per-round regret vanishes.

## 2.2 A PRINCIPLED REGULARITY MODEL FOR GRAPH HOMOPHILY

To make learning tractable, we need to impose regularity on the unknown functions  $f_{1:n}$ . We make two core assumptions. First, we assume that each function  $f_u$  is individually well-behaved, belonging to a common Reproducing Kernel Hilbert Space (RKHS), denoted  $\mathcal{H}_x$ , with a positive semi-definite kernel  $K_x : \mathcal{D} \times \mathcal{D} \rightarrow \mathbb{R}$ . The associated feature map is denoted as  $\varphi$  such that  $K_x(\mathbf{x}, \mathbf{x}') = \langle \varphi(\mathbf{x}), \varphi(\mathbf{x}') \rangle$ . This captures the non-linear structure of rewards with respect to arm features.

Second, we formalize the notion of graph homophily by assuming that users connected by an edge in  $G$  have similar reward functions. This *user similarity* is measured by the squared distance between functions in the RKHS,  $\|f_i - f_j\|_{\mathcal{H}_x}^2$ . Combining these principles, we model the true reward functions as having a small joint penalty that balances graph smoothness with individual function complexity:

$$\text{PEN}(f_{1:n}; \rho) := \underbrace{\frac{1}{2} \sum_{i,j=1}^n w_{ij} \|f_i - f_j\|_{\mathcal{H}_x}^2}_{\text{PEN}_{\text{graph}}(f_{1:n})} + \rho \underbrace{\sum_{i=1}^n \|f_i\|_{\mathcal{H}_x}^2}_{\text{PEN}_{\text{ridge}}(f_{1:n})} = \sum_{i,j=1}^n [\mathbf{L}_\rho]_{ij} \langle f_i, f_j \rangle_{\mathcal{H}_x}, \quad (1)$$

where  $\rho > 0$  is a regularization hyperparameter and  $\mathbf{L}_\rho := \mathbf{L} + \rho \mathbf{I}$  is the regularized graph Laplacian. This penalty is central to our framework, as it provides a clear, interpretable objective for modeling related, non-linear functions.

## 2.3 FROM JOINT PENALTY TO A UNIFIED MULTI-USER KERNEL

Our key theoretical insight is that the intuitive, additive penalty in equation 1 is not merely an ad-hoc regularizer. It is, in fact, the squared norm in a single, unified Hilbert space over the user-arm product domain  $\mathcal{U} \times \mathcal{D}$ . This allows us to reframe the problem from learning  $n$  related functions to learning one “lifted” function,  $f(\mathbf{x}, u) := f_u(\mathbf{x})$ , in this new space. We show that it is the squared RKHS norm for the product space  $\mathcal{H} = \mathcal{H}_u \otimes \mathcal{H}_x$  where  $\mathcal{H}_u$  is the RKHS with kernel  $K_u(u, u') = [\mathbf{L}_\rho^{-1}]_{u,u'}$  in the following theorem.

**Theorem 2.1** (Multi-user Kernel). *Let  $\mathcal{H}_x$  be an RKHS of functions on  $\mathcal{D}$  with kernel  $K_x$ . The vector space of function collections  $\mathcal{H} := \{(f_1, \dots, f_n) : f_u \in \mathcal{H}_x, \forall u \in \mathcal{U}\}$  equipped with the inner product*

$$\langle f, g \rangle_{\mathcal{H}} := \sum_{i,j=1}^n [\mathbf{L}_\rho]_{ij} \langle f_i, g_j \rangle_{\mathcal{H}_x}$$

*is a Reproducing Kernel Hilbert Space of functions on  $\mathcal{U} \times \mathcal{D}$ . The associated squared RKHS norm is precisely the penalty in equation 1, and its reproducing kernel  $K : (\mathcal{D} \times \mathcal{U})^2 \rightarrow \mathbb{R}$  is given by:*

$$K((\mathbf{x}, u), (\mathbf{x}', u')) = [\mathbf{L}_\rho^{-1}]_{u,u'} K_x(\mathbf{x}, \mathbf{x}'). \quad (2)$$

This result is powerful: it provides a direct, canonical construction for a *multi-user kernel* that fuses graph and feature information. The kernel  $K_x$  captures similarity between arms, while the matrix  $\mathbf{L}_\rho^{-1}$  (the graph Green’s function) captures similarity between users, with  $[\mathbf{L}_\rho^{-1}]_{u,u'}$  measuring the strength of connection between users  $u$  and  $u'$  through all paths in the graph.

This unification allows us to represent the lifted reward function  $f(\mathbf{x}, u)$  via a feature map  $\phi(\mathbf{x}, u)$  such that  $f(\mathbf{x}, u) = \langle \boldsymbol{\theta}, \phi(\mathbf{x}, u) \rangle$  for some (potentially infinite-dimensional) parameter  $\boldsymbol{\theta}$ , and  $K((\mathbf{x}, u), (\mathbf{x}', u')) = \langle \phi(\mathbf{x}, u), \phi(\mathbf{x}', u') \rangle$ . Formally, for a context-user pair  $(\mathbf{x}, u) \in \mathcal{D} \times \mathcal{U}$ , the feature map  $\phi$  is defined as  $\phi(\mathbf{x}, u) := \mathbf{L}_\rho^{-1/2} \mathbf{e}_u \otimes \varphi(\mathbf{x})$ . The problem is now cast as learning a single function in the *multi-user RKHS*  $\mathcal{H}$ . This insight paves the way for a principled algorithmic approach based on Gaussian processes, which we detail next.

### 3 LAPLACIAN KERNELIZED BANDIT ALGORITHMS

The identification of the *multi-user RKHS* with its explicit kernel  $K$  provides a powerful, unified framework for the GOB problem. It allows us to model the entire system—across all users and arms—with a single Gaussian Process (GP), sidestepping the complexity of managing  $n$  separate but correlated models.

#### 3.1 A GAUSSIAN PROCESS PERSPECTIVE

We propose algorithms based on the Gaussian process (GP), motivated by the kernelized bandit literature Chowdhury & Gopalan (2017). Our Bayesian modeling is only assumed for derivation of our estimators and it is not necessarily the true model. We place a GP prior over the unknown lifted reward function  $f : \mathcal{D} \times \mathcal{U} \rightarrow \mathbb{R}$ , denoted as

$$[f_1(\cdot), \dots, f_n(\cdot)] \sim \mathcal{GP}(0, K(\cdot, \cdot)).$$

where  $K$  is the multi-user kernel defined in equation 2. For any finite set of user-arm pairs  $\{(\mathbf{x}_i, u_i)\}_{i=1}^t$ , we can also denote the prior distribution in this Gaussian process by  $[f_{u_1}(\mathbf{x}_1), \dots, f_{u_t}(\mathbf{x}_t)]^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_t)$  where  $\mathbf{K}_t \in \mathbb{R}^{t \times t}$  with entries  $[\mathbf{K}_t]_{ij} = K((\mathbf{x}_i, u_i), (\mathbf{x}_j, u_j))$  is the Gram matrix.

At round  $t$ , given user  $u_t$  and selected arm  $\mathbf{x}_t$ , the noise in the observation model is  $\epsilon_t = y_t - f(\mathbf{x}_t, u_t)$ , which is Gaussian with variance  $\lambda$  in the design of Gaussian process. Therefore, conditioned on the history  $\mathcal{F}_t$ , the posterior distribution for  $f_u(\mathbf{x})$  is  $\mathcal{N}(\mu_{u,t-1}(\mathbf{x}), \sigma_{u,t-1}^2(\mathbf{x}))$ , where the posterior mean and variance for are given by:

$$\begin{aligned} \mu_{u,t}(\mathbf{x}) &= \mathbf{k}_t(\mathbf{x}, u)^\top (\mathbf{K}_t + \lambda \mathbf{I}_t)^{-1} \mathbf{y}_t \\ \sigma_{u,t}^2(\mathbf{x}) &= K((\mathbf{x}, u), (\mathbf{x}, u)) - \mathbf{k}_t(\mathbf{x}, u)^\top (\mathbf{K}_t + \lambda \mathbf{I}_t)^{-1} \mathbf{k}_t(\mathbf{x}, u). \end{aligned} \quad (3)$$

Here  $\mathbf{k}_t(\mathbf{x}, u) := [K((\mathbf{x}_1, u_1), (\mathbf{x}, u)), \dots, K((\mathbf{x}_t, u_t), (\mathbf{x}, u))]^\top \in \mathbb{R}^t$  is the kernel vector between past selected user-action pairs  $\{(\mathbf{x}_s, u_s)\}_{s=1}^t$  and new pair  $(\mathbf{x}, u)$ , and  $\mathbf{y}_t = [y_1, \dots, y_t]^\top \in \mathbb{R}^t$  is the observed reward.

**Connection to Regularized Regression.** It is worth noting that the GP posterior mean estimator in equation 3 is equivalent to the solution of an offline Kernel Laplacian Regularized Regression (KLRR) problem. Specifically, the function  $f \in \mathcal{H}$  that minimizes the regularized least-squares objective

$$\min_{f \in \mathcal{H}} \sum_{s=1}^t (f(\mathbf{x}_s, u_s) - y_s)^2 + \lambda \|f\|_{\mathcal{H}}^2 \quad (4)$$

is precisely the posterior mean function  $\mu_{t-1}(\mathbf{x}, u)$ . This equivalence confirms that our online, GP-based algorithm is deeply connected to the batch learning principle of minimizing prediction error regularized by our proposed multi-user RKHS norm from equation 1.

#### 3.2 DECISION STRATEGIES: UCB AND THOMPSON SAMPLING

With these posterior estimates, we can design bandit algorithms that effectively balance exploration and exploitation. We propose two algorithms based on common and powerful heuristics: Upper Confidence Bound (UCB) and Thompson Sampling (TS). The complete procedures are described in Appendix D.1

**Laplacian Kernelized GP-UCB (LK-GP-UCB).** Following the principle of “optimism in the face of uncertainty,” our UCB algorithm selects the arm with the highest optimistic estimate of the reward. At round  $t$ , upon observing user  $u_t$  and arm set  $\mathcal{D}_t$ , it chooses:

$$\mathbf{x}_t = \arg \max_{\mathbf{x} \in \mathcal{D}_t} \left( \mu_{u_t, t-1}(\mathbf{x}) + \beta_t \sigma_{u_t, t-1}(\mathbf{x}) \right), \quad (5)$$

where  $\beta_t$  is the hyperparameter that ensures the appropriate scale of exploration via confidence width  $\sigma_{u_t, t-1}(\mathbf{x})$ . Our theoretical analysis provides an explicit form for  $\beta_t$  in Theorem 4.2 to guarantee low regret, though in practice it is often treated as a tunable hyperparameter.

**Laplacian Kernelized GP-TS (LK-GP-TS).** Thompson Sampling Thompson (1933); Russo et al. (2018) operates on the principle of "probability matching." At each round, it draws a random function from the posterior distribution and acts greedily with respect to this sample. A practical way to implement this is to select the arm that maximizes a sample from the posterior predictive distribution for the reward:

$$\mathbf{x}_t = \arg \max_{\mathbf{x} \in \mathcal{D}_t} \left( \mu_{u_t, t-1}(\mathbf{x}) + \nu_t z_t(\mathbf{x}) \sigma_{u_t, t-1}(\mathbf{x}) \right), \quad (6)$$

where  $\nu_t$  is the scale hyperparameter for exploration and  $z_t(\mathbf{x}) \sim \mathcal{N}(0, 1)$  is the Gaussian perturbation. Aligned with common Thompson Sampling literature, our decision strategy in equation 6 can be separated into two steps: sampling  $\tilde{\mu}_t(\mathbf{x})$  from  $\mathcal{N}(\mu_{u_t, t-1}(\mathbf{x}), \nu_t^2 \sigma_{u_t, t-1}^2(\mathbf{x}))$  for all  $\mathbf{x} \in \mathcal{D}_t$  and choosing an arm by  $\mathbf{x}_t = \arg \max_{\mathbf{x} \in \mathcal{D}_t} \tilde{\mu}_t(\mathbf{x})$ . Similarly to the UCB algorithm, we also use the explicit theoretical choice for  $\nu_t$  in Theorem 4.3, while it is a tuning hyperparameter in a real application.

### 3.3 PRACTICAL IMPLEMENTATION

A naive implementation of the posterior updates in equation 3 is computationally expensive, requiring an  $\mathcal{O}(t^3)$  matrix inversion at each step. To ensure practical scalability, we can use recursive formulas to update the posterior mean and variance in  $\mathcal{O}(t^2)$  or, for a fixed grid of points, even more efficiently. Specifically, we can maintain and update the inverse matrix  $(\mathbf{K}_t + \lambda \mathbf{I}_t)^{-1}$  or use the following recursive updates for the posterior estimators Chowdhury & Gopalan (2017):

$$\begin{aligned} \mu_{u, t}(\mathbf{x}) &= \mu_{u, t-1}(\mathbf{x}) + \frac{q_{t-1}((\mathbf{x}, u), (\mathbf{x}_t, u_t))}{\lambda + \sigma_{u_t, t-1}^2(\mathbf{x}_t)} (y_t - \mu_{u_t, t-1}(\mathbf{x}_t)) \\ q_t((\mathbf{x}, u), (\mathbf{x}', u')) &= q_{t-1}((\mathbf{x}, u), (\mathbf{x}', u')) - \frac{q_{t-1}((\mathbf{x}, u), (\mathbf{x}_t, u_t)) q_{t-1}((\mathbf{x}_t, u_t), (\mathbf{x}', u'))}{\lambda + \sigma_{u_t, t-1}^2(\mathbf{x}_t)} \\ \sigma_{u, t}^2(\mathbf{x}) &= \sigma_{u, t-1}^2(\mathbf{x}) - \frac{q_{t-1}^2((\mathbf{x}, u), (\mathbf{x}_t, u_t))}{\lambda + \sigma_{u_t, t-1}^2(\mathbf{x}_t)}. \end{aligned} \quad (7)$$

where  $q_t((\mathbf{x}, u), (\mathbf{x}', u'))$  is the estimated posterior covariance at round  $t$ . We explain how to obtain the updates in Appendix D.2. A hybrid approach that uses exact inversion for small  $t$  and switches to recursive updates for larger  $t$  can balance numerical stability and computational efficiency. Further details on our implementation are provided in Appendix E.4.

## 4 REGRET ANALYSIS

We now provide theoretical guarantees for our proposed algorithms. Our analysis is built upon a high-probability confidence bound for our GP posterior estimates, which in turn leads to sub-linear regret bounds for both LK-GP-UCB and LK-GP-TS.

### 4.1 ASSUMPTIONS

Our results rely on the following standard assumptions.

**Assumption 1** (Sub-Gaussian Noise). *The noise process  $\{\epsilon_t\}_{t=1}^T$  is a  $\mathcal{F}_t$ -measurable stochastic process and is conditionally sub-Gaussian with constant  $\sigma^2$ .*

**Assumption 2** (Bounded Base Kernel). *The base arm kernel  $K_x(\cdot, \cdot)$  is positive semi-definite and its diagonal is uniformly bounded:  $\sup_{\mathbf{x} \in \mathcal{D}} K_x(\mathbf{x}, \mathbf{x}) \leq \alpha^2$  for some  $\alpha > 0$ .*

**Assumption 3** (Bounded Multi-User RKHS Norm). *The true lifted reward function  $f$  has a bounded norm in the multi-user RKHS  $\mathcal{H}$ :  $\|f\|_{\mathcal{H}}^2 = \text{PEN}(f_{1:n}; \rho) \leq B_\rho^2$  for some constant  $B_\rho > 0$ .*

Assumption 1 is common assumption in bandit literature. Assumption 2 and 3 indirectly align with the regularity assumptions in kernelized bandit and graph smoothness literatures Belkin et al. (2006); Kocák et al. (2020). These assumptions imply that the rewards and the *multi-user kernel* are bounded. Formally, we have

$$\sup_{(\mathbf{x}, u) \in \mathcal{D} \times \mathcal{U}} K((\mathbf{x}, u), (\mathbf{x}, u)) \leq K_{\max} := \alpha^2 \cdot \max_{u \in \mathcal{U}} [\mathbf{L}_\rho^{-1}]_{u, u}.$$

## 4.2 HIGH-PROBABILITY CONFIDENCE BOUNDS

The core of our regret analysis is the confidence bound that relates the true function  $f$  to our posterior mean estimator  $\mu_t$ . This result quantifies the model’s uncertainty and justifies the exploration strategy of the UCB algorithm.

**Theorem 4.1** (Confidence Bound). *Suppose Assumptions 1, 2, and 3 hold. Let  $\{(\mathbf{x}_t, u_t)\}_{t=1}^\infty$  be the  $\mathcal{F}_{t-1}$ -measurable discrete time stochastic process. Then, using the posterior estimators  $\mu_{u,t}(\mathbf{x})$  and  $\sigma_{u,t}(\mathbf{x})$  in equation 3 yields to a high probability upper bound: for any  $\delta \in (0, 1)$ , with probability at least  $1 - \delta$ , for all  $t \geq 1$  and all  $(\mathbf{x}, u) \in \mathcal{D} \times \mathcal{U}$ :*

$$|\mu_{u,t}(\mathbf{x}) - f(\mathbf{x}, u)| \leq \beta_t \cdot \sigma_{u,t}(\mathbf{x}) \quad (8)$$

where the confidence parameter  $\beta_t$  is given by

$$\beta_t := B_\rho + \sqrt{\frac{\sigma^2}{\lambda} \left( 2 \log \frac{1}{\delta} + \log \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t) \right)}.$$

This confidence bound follows a structure similar to those in the kernelized bandit literature Chowdhury & Gopalan (2017); Valko et al. (2013); Dubey et al. (2020), but our analysis offers two key distinctions. First, our proof does not require the constraint  $\lambda \geq 1$  found in some prior work. More significantly, we retain the term  $\log \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t)$  directly within our confidence width  $\beta_t$ . This contrasts with classical approaches that often proceed by further bounding this term using information-theoretic quantities, which can result in looser bounds. By keeping the exact term, we set the stage for a tighter, data-dependent analysis via the effective dimension.

## 4.3 REGRET BOUNDS VIA EFFECTIVE DIMENSION

To obtain concrete regret rates, we characterize the growth of the  $\log \det$  term using the notion of an *effective dimension*.

**Definition 4.1** (Effective Dimension). *The effective dimension  $\tilde{d}$  of the learning problem, given the sequence of actions up to time  $T$ , is defined as:*

$$\tilde{d} := \frac{\log \det(\mathbf{I}_T + \mathbf{K}_T / \lambda)}{\log(1 + T K_{\max} / \lambda)}. \quad (9)$$

This quantity, inspired by recent work in kernel methods and overparameterized models Wu & Amini (2024); Bietti & Mairal (2019); Yang & Wang (2020), measures the intrinsic complexity of the learning problem. It can be interpreted as the ratio of the sum of log-eigenvalues of the matrix  $\mathbf{I}_T + \mathbf{K}_T / \lambda$  to a bound on the maximum possible log-eigenvalue ( $T K_{\max}$  is an upper bound on the largest eigenvalue of  $\mathbf{K}_T$ ). As such, it serves as a robust, graph-dependent measure of the matrix’s rank, capturing the “dimensionality” of the function space actually explored by the algorithm.

Using the confidence bound in Theorem 4.1 and  $\tilde{d}$  in Definition 4.1, we provide the regret upper bound for LK-GP-UCB and LK-GP-TS as follow.

**Theorem 4.2** (Regret Bound of LK-GP-UCB). *Suppose Assumptions 1, 2 and 3 hold. By setting the exploration parameter  $\beta_t$  in LK-GP-UCB to  $\beta_t$  from Theorem 4.1, the cumulative regret is bounded with high probability as:*

$$\mathcal{R}_T = \mathcal{O}(\tilde{d} \log(T) \sqrt{T}) = \tilde{\mathcal{O}}(\tilde{d} \sqrt{T})$$

**Theorem 4.3** (Regret Bound of LK-GP-TS). *Suppose Assumptions 1, 2 and 3 hold. By setting the exploration parameter  $\nu_t$  in LK-GP-TS to  $\beta_t$  from Theorem 4.1, the cumulative regret is bounded with high probability as:*

$$\mathcal{R}_T = \mathcal{O}(\tilde{d} \log(T)^{3/2} \sqrt{T}) = \tilde{\mathcal{O}}(\tilde{d} \sqrt{T})$$

These bounds demonstrate the efficiency of our approach. The regret scales not with the number of users  $n$  or the ambient feature dimension, but with the effective dimension  $\tilde{d}$ . For problems where the graph and kernel structure lead to a rapid spectral decay,  $\tilde{d}$  can be significantly smaller, resulting in substantial gains in sample efficiency.

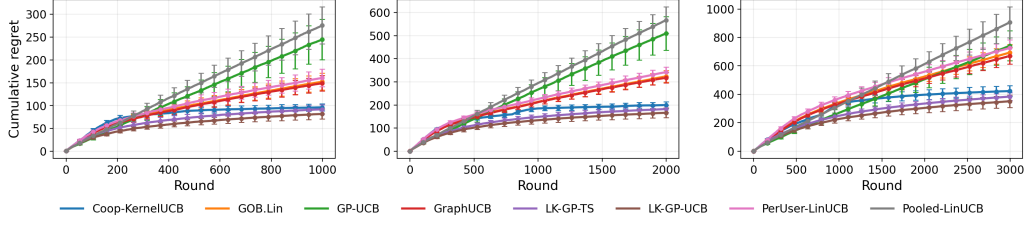


Figure 1: Cumulative Regret under *Linear-GOB* regime. From left to right are tasks of easy level, medium level, to hard level.

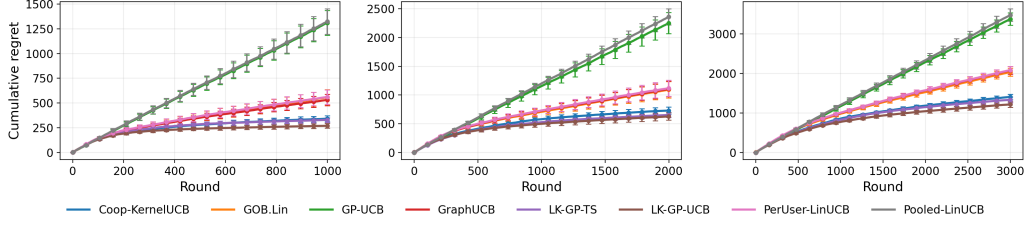


Figure 2: Cumulative Regret under *Laplacian-Kernel* regime using GP draw. From left to right are tasks of easy level, medium level, to hard level.

**Remark 1.** Classical kernelized contextual bandits control regret via the maximum information gain  $\gamma_T$  and obtain regret on the order of  $\tilde{O}(\gamma_T \sqrt{T})$ . In contrast, our analysis shows that cumulative regret scales as  $\tilde{O}(\tilde{d}\sqrt{T})$  with a effective dimension  $\tilde{d}$  determined by the spectrum of the (regularized) Laplacian  $\mathbf{L}_\rho$ , keeping the log-determinant term explicit. In addition, our result is different from existing GOB algorithms (e.g. *GOB.Lin*), whose regret order rates carry an explicit dependence on the number of users, and in the worst case those bounds scale with  $\sqrt{n}$  (or graph degrees) when the sharing induced by  $\mathbf{L}$  is weak. By contrast, our bound depends on spectral properties of  $\mathbf{L}_\rho$ —in particular through a constant  $K_{\max}$  that reflects graph connectivity. In short, compared with existing methods, our regret upper bound scales with a data-dependent effective dimension governed by the Laplacian spectrum, recovers the standard kernel rates as a special case, and makes the graph effect explicit while often yielding a milder dependence on  $n$ .

## 5 EXPERIMENTS

We evaluate Laplacian Kernelized bandit algorithms, LK-GP-UCB and LK-GP-TS on several synthetic data environments that capture user-user homophily on a known graph while varying reward structure (linear vs. nonlinear) and problem difficulty. Baseline algorithms include GraphUCB Yang et al. (2020), GoB.Lin Cesa-Bianchi et al. (2013), COOP-KernelUCB Dubey et al. (2020), GP-UCB Chowdhury & Gopalan (2017), Pooled LinUCB and Per-User LinUCB. Full implementation details are Provided in Appendix E.

**Environments.** We draw a context pool  $\mathcal{D}$  by sampling from  $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$  first and then normalize the context vectors. At round  $t$  we present  $\mathcal{D}_t$  by sampling  $m_t$  distinct items from  $\mathcal{D}$  without replacement. We generate the user graphs by Erdős-Rényi (ER) random graph model or Radial basis function (RBF) random graph model. After giving the generated graph, we consider one linear regime and two kernelized (nonlinear) regimes for synthetic data simulation. First synthetic data environment is called *Linear-GOB*. We consider simulating the true graph graph-smooth user parameters  $\Theta = (\mathbf{I} + \eta \mathbf{L})^{-1} \Theta_0$ , which enforce graph homophily on the random initial parameters  $\Theta_0 \in \mathbb{R}^{n \times d}$  Yang et al. (2020). The homophily strength is controlled by  $\eta$  in *Linear-GOB* regime. We also generate the true reward functions by simulating *multi-user kernel*, which is called the *Laplacian-Kernel* regime. We first use *Squared Exponential* as our base kernel  $K_x$  over arms  $\mathcal{U}$  and construct the *multi-user kernel* using equation 2. Next, we design two choices to generate  $f$ ,

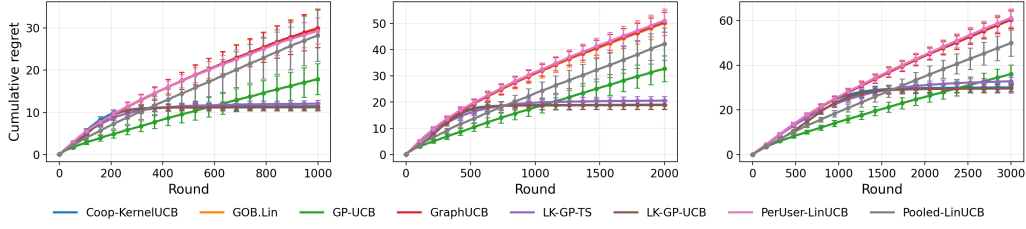


Figure 3: Cumulative Regret under *Laplacian-Kernel* regime using representer draw. From left to right are tasks of easy level, medium level, to hard level.

including a GP draw and a representer draw. We leave all the details for synthetic data simulation in Appendix E.1.

**Task Design.** Our experiment has following design of the bandit tasks for a general comparison. In these tasks, the noise of reward is set as  $\sigma = 0.1$  and the number of users is  $n = 20$ . The simplest level task is a 10-arm bandit problem ( $m = 10$ ) with 50% viewability ( $m_t = 5$ ) at each round for all users, under  $T = 1000$  interaction rounds. Medium level task is a 20-arm bandit problem ( $m = 20$ ) with 25% viewability ( $m_t = 5$ ) at each round for all users, under  $T = 3000$  interaction rounds. The hard task is a 50-arm bandit problem ( $m = 50$ ) with 10% viewability ( $m_t = 5$ ) at each round for all users, under  $T = 5000$  interaction rounds. In our figures (1, 2 and 3), from left to right are tasks of easy level, medium level, to hard level.

**Algorithms Configurations.** Our proposals LK-GP-UCB and LK-GP-TS are given in Algorithm 1 and Algorithm 2 in Appendix D.1. We implement the hybrid updates using practical recursive update in equation 7 and exact update in equation 3 with Cholesky decomposition. Details are in Appendix E.4. Hyperparameters  $\nu$  and  $\beta$  are tuned. For Coop-KernelUCB, we initially set five choices of similarity kernel  $K_z$  and conduct an experiment (Figure in Appendix) to verify that the inverse Laplacian  $L_\rho^{-1}$  is the optimal choice while the empirical maximum mean discrepancy method is close to the best choice. In the experiment,  $K_z$  is set as the empirical MMD method to learn the similarity kernel  $K_z$  unless otherwise stated. The classical baselines for GOB problem, GoB.Lin, GraphUCB, and all the remaining baselines, Pooled LinUCB, Per-User LinUCB and GP-UCB, are all UCB-based algorithms. We also tune their hyperparameter for the confidence bound. The regularization parameter  $\lambda$  is designed as a scheduling  $\lambda_t = \lambda_{\text{base}} \cdot S_{\text{spec}} \cdot \frac{T}{T+t}$  where  $S_{\text{spec}}$  is the ratio of the smallest non-zero eigenvalue to the max eigenvalue and  $\lambda_{\text{base}}$  is tuned. See Appendix E.5 for details of hyperparameters tuning. All methods run in a centralized, no-delay setting.

**Main Findings.** Our proposals LK-GP-UCB and LK-GP-TS have robust performance in all the 9 data environments. In the *Linear-GOB* regime, which is the preferred setting for linear bandit algorithms, our proposals can beat the most baselines with clear gaps. In the *Laplacian-Kernel* regime, our proposals are consistently the best choices. For the GP draw setting, our proposals are always the top algorithms in our experiment. For setting using representer draw, LK-GP-UCB and LK-GP-TS are sublinear while most baselines are hard to achieve sublinear regret. We believe our proposed algorithms can clearly outperform others in a long-term manner due to the achievement of the clear sublinear regret. Lastly, even though we conduct an empirical study on the choice for Coop-KernelUCB and pick a best one in the comparison, leading to the top performances (close to our proposal) of Coop-KernelUCB, our LK-GP-UCB are consistently better than Coop-KernelUCB.

## REFERENCES

- Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.
- Mauricio A Alvarez, Lorenzo Rosasco, Neil D Lawrence, et al. Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266, 2012.



- Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(11), 2006.
- Alberto Bietti and Julien Mairal. On the inductive bias of neural tangent kernels. *Advances in Neural Information Processing Systems*, 32, 2019.
- Sébastien Bubeck, Ronen Eldan, and Yin Tat Lee. Kernel-based methods for bandit convex optimization. *Journal of the ACM (JACM)*, 68(4):1–35, 2021.
- Nicolo Cesa-Bianchi, Claudio Gentile, and Giovanni Zappella. A gang of bandits. *Advances in neural information processing systems*, 26, 2013.
- Sayak Ray Chowdhury and Aditya Gopalan. On kernelized multi-armed bandits. In *International Conference on Machine Learning*, pp. 844–853. PMLR, 2017.
- Yihan Du, Wei Chen, Yuko Kuroki, and Longbo Huang. Collaborative pure exploration in kernel bandit. *arXiv preprint arXiv:2110.15771*, 2021.
- Abhimanyu Dubey et al. Kernel methods for cooperative multi-agent contextual bandits. In *International Conference on Machine Learning*, pp. 2740–2750. PMLR, 2020.
- Xueping Gong and Jiheng Zhang. Efficient graph bandit learning with side-observations and switching constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 16871–16879, 2025.
- Tomáš Kocák, Rémi Munos, Branislav Kveton, Shipra Agrawal, and Michal Valko. Spectral bandits. *Journal of Machine Learning Research*, 21(218):1–44, 2020.
- Andreas Krause and Cheng Ong. Contextual gaussian process bandit optimization. *Advances in neural information processing systems*, 24, 2011.
- Branislav Kveton, Csaba Szepesvari, Mohammad Ghavamzadeh, and Craig Boutilier. Perturbed-history exploration in stochastic linear bandits. *arXiv preprint arXiv:1903.09132*, 2019.
- Peter Landgren, Vaibhav Srivastava, and Naomi Ehrlich Leonard. On distributed cooperative decision-making in multiarmed bandits. In *2016 European Control Conference (ECC)*, pp. 243–248. IEEE, 2016.
- Chuanhao Li, Huazheng Wang, Mengdi Wang, and Hongning Wang. Communication efficient distributed learning for kernelized contextual bandits. *Advances in Neural Information Processing Systems*, 35:19773–19785, 2022.
- Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018.
- Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.
- Balazs Szorenyi, Róbert Busa-Fekete, István Hegedus, Róbert Ormándi, Márk Jelasity, and Balázs Kégl. Gossip-based distributed stochastic bandit algorithms. In *International conference on machine learning*, pp. 19–27. PMLR, 2013.
- William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.
- Sattar Vakili, Kia Khezeli, and Victor Picheny. On information gain and regret bounds in gaussian process bandits. In *International Conference on Artificial Intelligence and Statistics*, pp. 82–90. PMLR, 2021.
- Michal Valko, Nathaniel Korda, Rémi Munos, Ilias Flaounas, and Nelo Cristianini. Finite-time analysis of kernelised contextual bandits. *arXiv preprint arXiv:1309.6869*, 2013.

- Sharan Vaswani, Mark Schmidt, and Laks Lakshmanan. Horde of bandits using gaussian markov random fields. In *Artificial Intelligence and Statistics*, pp. 690–699. PMLR, 2017.
- Yao Wang, Jiannan Li, Yue Kang, Shanxing Gao, and Zhenxin Xiao. Generalized low-rank matrix contextual bandits with graph information. *arXiv preprint arXiv:2507.17528*, 2025.
- Shuang Wu and Arash A Amini. Graph neural thompson sampling. *arXiv preprint arXiv:2406.10686*, 2024.
- Shuang Wu, Chi-Hua Wang, Yuantong Li, and Guang Cheng. Residual bootstrap exploration for stochastic linear bandit. In *Uncertainty in Artificial Intelligence*, pp. 2117–2127. PMLR, 2022.
- Kaige Yang, Laura Toni, and Xiaowen Dong. Laplacian-regularized graph bandits: Algorithms and theoretical analysis. In *International Conference on Artificial Intelligence and Statistics*, pp. 3133–3143. PMLR, 2020.
- Lin Yang and Mengdi Wang. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In *International Conference on Machine Learning*, pp. 10746–10756. PMLR, 2020.
- Yael Yankellevsky and Michael Elad. Dual graph regularized dictionary learning. *IEEE Transactions on Signal and Information Processing over Networks*, 2(4):611–624, 2016.
- Xingyu Zhou and Bo Ji. On kernelized multi-armed bandits with constraints. *Advances in neural information processing systems*, 35:14–26, 2022.

## A PROOF OF THEOREM 2.1

*Proof.* The proof proceeds in three main steps: (1) We construct the Hilbert space for our multi-user problem as the tensor product of the user space and the context space; (2) We define a feature map into this space and show that its inner product yields the kernel  $K$ . This establishes that our constructed space is indeed the RKHS  $\mathcal{H}$ ; (3) We characterize the elements of  $\mathcal{H}$  and derive the expression for their norm.

**Step 1: Constructing the Hilbert Space via Tensor Product.** Let  $\mathcal{H}_u = \mathbb{R}^n$  be the finite-dimensional Hilbert space for the users, equipped with the standard Euclidean inner product  $\langle \mathbf{u}, \mathbf{v} \rangle_{\mathcal{H}_u} = \mathbf{u}^\top \mathbf{v}$ .  $\{\mathbf{e}_i\}_{i=1}^n$  forms the standard orthonormal basis for  $\mathcal{H}_u$ . Our *multi-user RKHS*  $\mathcal{H}$  is the tensor product of  $\mathcal{H}_u$  and  $\mathcal{H}_x$ :

$$\mathcal{H} := \mathcal{H}_u \otimes \mathcal{H}_x = \mathbb{R}^n \otimes \mathcal{H}_x.$$

The elements of  $\mathcal{H}$  are (limits of) finite linear combinations of elementary tensors of the form  $\mathbf{u} \otimes h$ , where  $\mathbf{u} \in \mathcal{H}_u$  and  $h \in \mathcal{H}_x$ . The inner product in  $\mathcal{H}$  is defined on these elementary tensors and extended by linearity:

$$\langle \mathbf{u}_1 \otimes h_1, \mathbf{u}_2 \otimes h_2 \rangle_{\mathcal{H}} := \langle \mathbf{u}_1, \mathbf{u}_2 \rangle_{\mathcal{H}_u} \langle h_1, h_2 \rangle_{\mathcal{H}_x}.$$

**Step 2: Defining the Feature Map and Verifying the Kernel.** Let  $L_\rho^{1/2}$  be the unique symmetric positive definite square root of  $L_\rho$ . We define the feature map  $\phi : (\mathcal{U} \times \mathcal{D}) \rightarrow \mathcal{H}$  as:

$$\phi(\mathbf{x}, u) := (L_\rho^{-1/2} \mathbf{e}_i) \otimes \varphi(\mathbf{x}).$$

This is a valid element of  $\mathcal{H}$  since  $L_\rho^{-1/2} \mathbf{e}_i \in \mathbb{R}^n = \mathcal{H}_u$  and  $\varphi(\mathbf{x}) \in \mathcal{H}_x$ . Let's compute the inner product of two such feature mappings in  $\mathcal{H}$ :

$$\begin{aligned} \langle \phi(\mathbf{x}, i), \phi(\mathbf{y}, j) \rangle_{\mathcal{H}} &= \langle (L_\rho^{-1/2} \mathbf{e}_i) \otimes \varphi(\mathbf{x}), (L_\rho^{-1/2} \mathbf{e}_j) \otimes \varphi(\mathbf{y}) \rangle_{\mathcal{H}} \\ &= \langle L_\rho^{-1/2} \mathbf{e}_i, L_\rho^{-1/2} \mathbf{e}_j \rangle_{\mathcal{H}_u} \cdot \langle \varphi(\mathbf{x}), \varphi(\mathbf{y}) \rangle_{\mathcal{H}_x} \\ &= \mathbf{e}_i^\top L_\rho^{-1} \mathbf{e}_j \cdot K_x(\mathbf{x}, \mathbf{y}) \\ &= [L_\rho^{-1}]_{ij} \cdot K_x(\mathbf{x}, \mathbf{y}) = K((\mathbf{x}, i), (\mathbf{y}, j)). \end{aligned}$$

By the fundamental property of RKHS, since the kernel  $K$  is generated by the inner product of the feature map  $\phi$  in the Hilbert space  $\mathcal{H}$ ,  $\mathcal{H}$  is the unique RKHS associated with  $K$ .

**Step 3: Characterizing Functions in  $\mathcal{H}$  and their Norms.** An element of  $\mathcal{H}$  is a function  $f : (\mathcal{U} \times \mathcal{D}) \rightarrow \mathbb{R}$ . By the Riesz representation theorem, for each  $f \in \mathcal{H}$ , there exists a unique element  $\boldsymbol{\theta} \in \mathcal{H}$  such that  $f(\cdot, \cdot) = \langle \boldsymbol{\theta}, \phi(\cdot, \cdot) \rangle_{\mathcal{H}}$  and  $\|f\|_{\mathcal{H}} = \|\boldsymbol{\theta}\|_{\mathcal{H}}$ . For some component functions  $\{g_k\}_{k=1}^n \subset \mathcal{H}_x$ , we can uniquely express  $\boldsymbol{\theta}$  as

$$\boldsymbol{\theta} = \sum_{k=1}^n \mathbf{e}_k \otimes g_k$$

and the squared norm of  $\boldsymbol{\theta}$  in  $\mathcal{H}$  is then:

$$\|\boldsymbol{\theta}\|_{\mathcal{H}}^2 = \left\langle \sum_k \mathbf{e}_k \otimes g_k, \sum_l \mathbf{e}_l \otimes g_l \right\rangle_{\mathcal{H}} = \sum_{k,l} \langle \mathbf{e}_k, \mathbf{e}_l \rangle_{\mathcal{H}_u} \langle g_k, g_l \rangle_{\mathcal{H}_x} = \sum_{k=1}^n \|g_k\|_{\mathcal{H}_x}^2.$$

Then we can relate our reward functions  $f_{1:n}$  to the component functions  $\{g_k\}_{k=1}^n$ :

$$\begin{aligned}
 f_i(\mathbf{x}) &= \langle \boldsymbol{\theta}, \phi((\mathbf{x}, i)) \rangle_{\mathcal{H}} \\
 &= \left\langle \sum_{k=1}^n \mathbf{e}_k \otimes g_k, (\mathbf{L}_\rho^{-1/2} \mathbf{e}_i) \otimes \varphi(\mathbf{x}) \right\rangle_{\mathcal{H}} \\
 &= \sum_{k=1}^n \langle \mathbf{e}_k, \mathbf{L}_\rho^{-1/2} \mathbf{e}_i \rangle_{\mathcal{H}_u} \cdot \langle g_k, \varphi(\mathbf{x}) \rangle_{\mathcal{H}_x} \\
 &= \sum_{k=1}^n [\mathbf{L}_\rho^{-1/2}]_{ki} \cdot \langle g_k, \varphi(\mathbf{x}) \rangle_{\mathcal{H}_x} \\
 &= \sum_{k=1}^n [\mathbf{L}_\rho^{-1/2}]_{ki} g_k(\mathbf{x}) \quad (\text{since } \langle g, \varphi(\mathbf{x}) \rangle_{\mathcal{H}_x} = g(\mathbf{x}))
 \end{aligned}$$

which leads to

$$g_k(\mathbf{x}) = \sum_{j=1}^n [\mathbf{L}_\rho^{1/2}]_{kj} f_j(\mathbf{x}).$$

This equality holds for the functions as elements of  $\mathcal{H}_\kappa$ :  $g_k = \sum_{j=1}^n [\mathbf{L}_\rho^{1/2}]_{kj} f_j$ .

Finally, we compute the norm of  $f$  in  $\mathcal{H}$ :

$$\begin{aligned}
 \|f\|_{\mathcal{H}}^2 &= \|\boldsymbol{\theta}\|_{\mathcal{H}}^2 = \sum_{k=1}^n \|g_k\|_{\mathcal{H}_x}^2 \\
 &= \sum_{k=1}^n \left\| \sum_{j=1}^n [\mathbf{L}_\rho^{1/2}]_{kj} f_j \right\|_{\mathcal{H}_x}^2 \\
 &= \sum_{k=1}^n \left\langle \sum_{j=1}^n [\mathbf{L}_\rho^{1/2}]_{kj} f_j, \sum_{l=1}^n [\mathbf{L}_\rho^{1/2}]_{kl} f_l \right\rangle_{\mathcal{H}_x} \\
 &= \sum_{k=1}^n \sum_{j,l=1}^n [\mathbf{L}_\rho^{1/2}]_{kj} [\mathbf{L}_\rho^{1/2}]_{kl} \langle f_j, f_l \rangle_{\mathcal{H}_x} \\
 &= \sum_{j,l=1}^n \left( \sum_{k=1}^n [\mathbf{L}_\rho^{1/2}]_{kj} [\mathbf{L}_\rho^{1/2}]_{kl} \right) \langle f_j, f_l \rangle_{\mathcal{H}_x} \\
 &= \sum_{j,l=1}^n [\mathbf{L}_\rho]_{jl} \langle f_j, f_l \rangle_{\mathcal{H}_x}
 \end{aligned}$$

where the last step is because the term in parentheses is the  $(j, l)$ -th element of the matrix product  $(\mathbf{L}_\rho^{1/2})^\top \mathbf{L}_\rho^{1/2} = \mathbf{L}_\rho^{1/2} \mathbf{L}_\rho^{1/2} = \mathbf{L}_\rho$ . By polarization identity, the associated inner product in  $\mathcal{H}$  is:

$$\langle f, g \rangle_{\mathcal{H}} := \sum_{i,j=1}^n [\mathbf{L}_\rho]_{ij} \langle f_i, g_j \rangle_{\mathcal{H}_x}.$$

To see that  $\|f\|_{\mathcal{H}}^2$  is the exactly the penalty in equation 1, we expand  $\mathbf{L}_\rho = \rho \mathbf{I}_n + \mathbf{L}$ :

$$\begin{aligned}
 \|f\|_{\mathcal{H}}^2 &= \sum_{j,l=1}^n (\rho \mathbb{I}\{j=l\} + [\mathbf{L}]_{jl}) \langle f_j, f_l \rangle_{\mathcal{H}_x} \\
 &= \rho \sum_{j=1}^n \|f_j\|_{\mathcal{H}_x}^2 + \sum_{j,l=1}^n [\mathbf{L}]_{jl} \langle f_j, f_l \rangle_{\mathcal{H}_x}.
 \end{aligned}$$

Using the standard identity for the Laplacian quadratic form, the second term in the above equation is exactly  $\frac{1}{2} \sum_{i,j} w_{ij} \|f_i - f_j\|_{\mathcal{H}_x}^2$ , we get:

$$\|f\|_{\mathcal{H}}^2 = \rho \sum_{j=1}^n \|f_j\|_{\mathcal{H}_x}^2 + \frac{1}{2} \sum_{i,j} w_{ij} \|f_i - f_j\|_{\mathcal{H}_x}^2.$$

This completes the proof.  $\square$

## B PROOFS IN ANALYSIS

We first define following additional notations

$$\Phi_t := [\phi(\mathbf{x}_1, u_1), \dots, \phi(\mathbf{x}_t, u_t)]^\top \quad (10)$$

$$\mathbf{J}_t := \Phi_t^\top \Phi_t \quad (11)$$

$$\Gamma_t := \mathbf{J}_t + \lambda \mathbf{I}_\infty \quad (12)$$

$$\Sigma_t := \mathbf{K}_t + \lambda \mathbf{I}_t \quad (13)$$

Here we have  $\Phi_t \in \mathbb{R}^{t \times \infty}$  and  $\mathbf{J}_t, \Gamma_t$  are from  $\mathbb{R}^{\infty \times \infty}$ .

Then we define some useful events for concentration:

$$\mathcal{E}_t^{ts} = \{|z_t(\mathbf{x})| \leq \sqrt{2 \log(t^2 |\mathcal{D}_t|)}, \text{ for all } \mathbf{x} \in \mathcal{D}_t\}$$

$$\mathcal{E}_t^a = \{\mu_{u_t, t-1}(\mathbf{x}_t^*) + \beta_t z_t(\mathbf{x}_t^*) \sigma_{u_t, t-1}(\mathbf{x}_t^*) > f(\mathbf{x}_t^*, u_t)\}$$

where  $z_t(\mathbf{x}) \sim \mathcal{N}(0, 1)$  stands for the resampling randomness in Thompson Sampling. We also define the confidence set at round  $t$ :

$$\mathcal{C}_t := \{|\mu_{u_t, t-1}(\mathbf{x}_t) - f(\mathbf{x}_t, u_t)| \leq \beta_t \cdot \sigma_{u_t, t-1}(\mathbf{x}_t)\} \quad (14)$$

where

$$\beta_t := \left( B_\rho + \sqrt{\frac{\sigma^2}{\lambda} \cdot \log \det(\mathbf{I}_{t-1} + \lambda^{-1} \mathbf{K}_{t-1})} + \frac{2\sigma^2}{\lambda} \log \frac{1}{\delta} \right).$$

In addition, recall the following effective dimension

$$\tilde{d} := \frac{\log \det(\mathbf{I}_T + \mathbf{K}_T / \lambda)}{\log(1 + T K_{\max} / \lambda)}$$

and the upper bound of the optimality gap:

$$|\Delta_t| \leq B_\Delta := 2B_\rho K_{\max}^{1/2}.$$

Lastly, we provide the following Lemmas, which are commonly required in regret analysis.

**Lemma B.1** (Concentrations for TS). *For all  $t \in [T]$ , we have  $\mathbb{P}_t(\bar{\mathcal{E}}_t^{ts}) \leq t^{-2}$  and  $\mathbb{P}_t(\mathcal{E}_t^a | \mathcal{C}_t) \geq (4e\sqrt{\pi})^{-1}$ .*

**Lemma B.2** (One Step Regret Bound for TS). *Suppose  $\mathbb{P}_t(\mathcal{E}_t^a) - \mathbb{P}_t(\bar{\mathcal{E}}_t^{ts}) > 0$ . Then for any  $t$ , almost surely,*

$$\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{C}_t}] \leq \mathbb{I}_{\mathcal{C}_t} \cdot \left\{ \left( \frac{2}{\mathbb{P}_t(\mathcal{E}_t^a) - \mathbb{P}_t(\bar{\mathcal{E}}_t^{ts})} + 1 \right) \cdot \mathbb{E}_t[\gamma_t \sigma_{u_t, t-1}(\mathbf{x}_t)] + B_\Delta \cdot \mathbb{P}_t(\bar{\mathcal{E}}_t^{ts}) \right\}$$

where  $\gamma_t := \beta_t + \beta_t \sqrt{2 \log(t^2 |\mathcal{D}_t|)}$  and  $B_\Delta := 2B_\rho K_{\max}^{1/2}$

**Lemma B.3** (Cumulative Uncertainty Bound). *We have the upper bound for the cumulative estimated uncertainty:*

$$\sum_{t=1}^T \sigma_{u_t, t-1}(\mathbf{x}_t) \leq \sqrt{2T \max\{1, K_{\max}\} \cdot \log \det(\mathbf{I}_T + \lambda^{-1} \mathbf{K}_T)}$$

**Lemma B.4** (Dual Identities). *With the defined notations in equation 10, we have two key identities:*

$$\Sigma_t^{-1} \Phi_t = \Phi_t \Gamma_t^{-1}, \text{ and } \sigma_{u, t}^2(\mathbf{x}) = \lambda \|\phi(\mathbf{x}, u)\|_{\Gamma_t^{-1}}^2.$$

## B.1 PROOF OF CONFIDENCE SET

*Proof of Theorem 4.1.* We first decompose

$$\begin{aligned}\mu_{u,t}(\mathbf{x}) - f(\mathbf{x}, u) &= \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} (\Phi_t \boldsymbol{\theta} + \boldsymbol{\epsilon}_t) - \boldsymbol{\theta}^\top \phi(\mathbf{x}, u) \\ &= (\Phi_t^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u))^\top \boldsymbol{\theta} + \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \boldsymbol{\epsilon}_t - \boldsymbol{\theta}^\top \phi(\mathbf{x}, u) \\ &= \underbrace{\langle \boldsymbol{\theta}, \delta_t(\mathbf{x}, u) \rangle}_{\text{bias}_t(\mathbf{x}, u)} + \underbrace{\mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \boldsymbol{\epsilon}_t}_{\text{noise}_t(\mathbf{x}, u)}\end{aligned}$$

where  $\delta_t(\mathbf{x}, u) = \Phi_t^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) - \phi(\mathbf{x}, u) \in \ell^2$ . Our target is to bound the  $\text{bias}_t(\mathbf{x}, u)$  and  $\text{noise}_t(\mathbf{x}, u)$ . We state the following Lemmas:

**Lemma B.5** (Bias Identity). *The squared bias is the degraded variance for noise:*

$$\|\delta_t(\mathbf{x}, u)\|_{\ell^2}^2 = \sigma_{u,t}^2(\mathbf{x}) - \lambda \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-2} \mathbf{k}_t(\mathbf{x}, u) \quad (15)$$

In particular, we have  $\|\delta_t(\mathbf{x}, u)\|_{\ell^2} \leq \sigma_{u,t}(\mathbf{x})$  and  $\lambda \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-2} \mathbf{k}_t(\mathbf{x}, u) < \sigma_{u,t}^2(\mathbf{x})$ .

**Lemma B.6** (Noise Bound). *With high probability, we have the upper bound for the following norm of noise vector  $\boldsymbol{\epsilon}_t$ :*

$$\|\Phi_t \boldsymbol{\epsilon}_t\|_{\Gamma_t^{-1}} \leq \sqrt{\sigma^2 \log \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t) + 2\sigma^2 \log \frac{1}{\delta}}$$

From Lemma B.5, we could bound the bias by

$$\text{bias}_t(\mathbf{x}, u) \leq \|\boldsymbol{\theta}\|_{\ell^2} \|\delta_t(\mathbf{x}, u)\|_{\ell^2} \leq B_\rho \sigma_{u,t}(\mathbf{x}). \quad (16)$$

Using the identities in above Lemma B.4, we note that

$$\begin{aligned}\text{noise}_t(\mathbf{x}, u) &= \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \boldsymbol{\epsilon}_t \\ &= \phi(\mathbf{x}, u)^\top \Gamma_t^{-1} \Phi_t \boldsymbol{\epsilon}_t \\ &= \langle \phi(\mathbf{x}, u), \Phi_t \boldsymbol{\epsilon}_t \rangle_{\Gamma_t^{-1}} \\ &\leq \|\phi(\mathbf{x}, u)\|_{\Gamma_t^{-1}} \cdot \|\Phi_t \boldsymbol{\epsilon}_t\|_{\Gamma_t^{-1}} \\ &= \frac{\sigma_{u,t}(\mathbf{x})}{\sqrt{\lambda}} \cdot \|\Phi_t \boldsymbol{\epsilon}_t\|_{\Gamma_t^{-1}}\end{aligned}$$

where the inequality is from the Cauchy-Schwarz inequality for the inner product  $\langle \cdot, \cdot \rangle_{\Gamma_t^{-1}}$ .

Our Lemma B.6 gives the high probability upper bound for the norm  $\|\Phi_t \boldsymbol{\epsilon}_t\|_{\Gamma_t^{-1}}$ , leading to

$$\text{noise}_t(\mathbf{x}, u) \leq \frac{\sigma_{u,t}(\mathbf{x})}{\sqrt{\lambda}} \cdot \sqrt{\sigma^2 \log \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t) + 2\sigma^2 \log \frac{1}{\delta}} \quad (17)$$

Now combine equation 16 and equation 17 together, we have

$$\begin{aligned}|\mu_{u,t}(\mathbf{x}) - f(\mathbf{x}, u)| &\leq |\text{bias}_t(\mathbf{x}, u)| + |\text{noise}_t(\mathbf{x}, u)| \\ &\leq \sigma_{u,t}(\mathbf{x}) \left( B_\rho + \sqrt{\frac{\sigma^2}{\lambda} \cdot \log \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t) + \frac{2\sigma^2}{\lambda} \log \frac{1}{\delta}} \right).\end{aligned}$$

□

## B.2 PROOF OF REGRET BOUND OF LK-GP-UCB

*Proof of Theorem 4.2.* Recall the instantaneous regret at time  $t$  is  $\Delta_t = f(\mathbf{x}_t^*, u_t) - f(\mathbf{x}_t, u_t)$  and the cumulative regret in a time horizon  $T$  is  $\mathcal{R}_T = \sum_{t=1}^T \Delta_t$ . We note event  $\mathcal{C}_t := \{|\mu_{u_t, t-1}(\mathbf{x}_t) - f(\mathbf{x}_t, u_t)| \leq \beta_t \cdot \sigma_{u_t, t-1}(\mathbf{x}_t)\}$  happens with high probability  $(1 - \delta)$ , according to Theorem 4.1,

$$\beta_t := \left( B_\rho + \sqrt{\frac{\sigma^2}{\lambda} \cdot \log \det(\mathbf{I}_{t-1} + \lambda^{-1} \mathbf{K}_{t-1}) + \frac{2\sigma^2}{\lambda} \log \frac{1}{\delta}} \right) \quad (18)$$

By Theorem 4.1, for all  $t \geq 2$  with probability at least  $1 - \delta$ ,

$$\begin{aligned}\Delta_t &= f(\mathbf{x}_t^*, u_t) - f(\mathbf{x}_t, u_t) \leq \mu_{u_t, t-1}(\mathbf{x}_t^*) + \beta_t \sigma_{u_t, t-1}(\mathbf{x}_t^*) - f(\mathbf{x}_t, u_t) \\ &\leq \mu_{u_t, t-1}(\mathbf{x}_t) + \beta_t \sigma_{u_t, t-1}(\mathbf{x}_t) - f(\mathbf{x}_t, u_t) \\ &\leq 2\beta_t \sigma_{u_t, t-1}(\mathbf{x}_t).\end{aligned}$$

Thus we have high probability bound for the cumulative regret

$$\mathcal{R}_T \leq 2\mathbb{E}\left[\beta_t \sum_{t=2}^T \sigma_{u_t, t-1}(\mathbf{x}_t)\right] + B_\Delta.$$

Then we apply Lemma B.3 and the definition of effective dimension in equation 9

$$\begin{aligned}\sum_{t=1}^T \sigma_{u_t, t-1}(\mathbf{x}_t) &\leq \sqrt{2T \max\{1, K_{\max}\} \cdot \log \det(\mathbf{I}_T + \lambda^{-1} \mathbf{K}_T)} \\ &= \sqrt{2T \max\{1, K_{\max}\} \cdot \tilde{d} \log(1 + T\lambda^{-1} K_{\max})}.\end{aligned}$$

Therefore, we have the final high probability upper bound for regret:

$$\mathcal{R}_T \leq 2\mathbb{E}[\beta_T] \sqrt{2T \max\{1, K_{\max}\} \cdot \tilde{d} \log(1 + T\lambda^{-1} K_{\max})} + B_\Delta.$$

The next step is to analyze the order of the upper bound. By using the effective dimension  $\tilde{d}$  again and dropping constants, we have

$$\begin{aligned}\beta_t &\leq B_\rho + \sqrt{\frac{\sigma^2}{\lambda} \cdot \tilde{d} \log(1 + T\lambda^{-1} K_{\max})} + \frac{2\sigma^2}{\lambda} \log \frac{1}{\delta} = \mathcal{O}(\sqrt{\tilde{d} \log(T)}) \\ \Rightarrow \mathcal{R}_T &= \mathcal{O}(\tilde{d} \log(T) \sqrt{T}) = \tilde{\mathcal{O}}(\tilde{d} \sqrt{T}).\end{aligned}$$

□

### B.3 PROOF OF REGRET BOUND OF LK-GP-TS

*Proof of Theorem 4.3.* We start from the decomposition of the cumulative regret

$$\mathcal{R}_T = \sum_{t=1}^T \mathbb{E}[\Delta_t] = \sum_{t=1}^T \mathbb{E}[\Delta_t \mathbb{I}_{\mathcal{C}_t}] + \sum_{t=1}^T \mathbb{E}[\Delta_t \mathbb{I}_{\bar{\mathcal{C}}_t}].$$

By Theorem 4.1 and the upper bound for the optimality gap, we know the second term is bounded:

$$\sum_{t=1}^T \mathbb{E}[\Delta_t \mathbb{I}_{\bar{\mathcal{C}}_t}] \leq \delta B_\Delta$$

by letting  $\mathbb{P}(\mathcal{C}_t) \leq \delta/T$  for all  $t$  in Theorem 4.1.

For the regret on the event  $\mathcal{C}_t$ , by Lemma B.2, almost surely, we have

$$\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{C}_t}] \leq \mathbb{I}_{\mathcal{C}_t} \cdot \left\{ \left( \frac{2}{\mathbb{P}_t(\mathcal{E}_t^a) - \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}})} + 1 \right) \cdot \mathbb{E}_t[\gamma_t \sigma_{u_t, t-1}(\mathbf{x}_t)] + B_\Delta \cdot \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}}) \right\}$$

where  $\gamma_t := \beta_t + \beta_t \sqrt{2 \log(t^2 |\mathcal{D}_t|)}$ . Note that  $\mathbb{P}_t(\mathcal{E}_t^a) - \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}}) \geq \frac{1}{4e\sqrt{\pi}} - \frac{1}{t^2} \geq \frac{1}{20e\sqrt{\pi}}$  by Lemma B.1 and the fact that  $t^2 \geq 5e\sqrt{\pi}$  for all  $t \geq 5$ . Thus we have

$$\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{C}_t}] \leq \mathbb{I}_{\mathcal{C}_t} \cdot \left\{ 194 \mathbb{E}_t[\gamma_t \sigma_{u_t, t-1}(\mathbf{x}_t)] + B_\Delta t^{-2} \right\}$$

by using  $40e\sqrt{\pi} + 1 \leq 194$ . Taking summation on both side for our target cumulative regret, we get

$$\begin{aligned}
\sum_{t=1}^T \mathbb{E}[\Delta_t \mathbb{I}_{\mathcal{C}_t}] &= \mathbb{E}\left[\sum_{t=1}^T \mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{C}_t}]\right] \\
&\leq \mathbb{E}\left[\sum_{t=5}^T (194\mathbb{E}_t[\gamma_t \sigma_{u_t, t-1}(\mathbf{x}_t)] + B_\Delta t^{-2}) + 4B_\Delta\right] \\
&\leq \mathbb{E}\left[194 \sum_{t=5}^T \mathbb{E}_t[\gamma_t \sigma_{u_t, t-1}(\mathbf{x}_t)] + (4 + \frac{\pi^2}{6})B_\Delta\right] \\
&\leq \mathbb{E}\left[194\gamma_T \mathbb{E}_T\left[\sum_{t=1}^T \sigma_{u_t, t-1}(\mathbf{x}_t)\right] + (4 + \frac{\pi^2}{6})B_\Delta\right]
\end{aligned}$$

where the second equality is using  $\sum_{t=1}^{\infty} t^{-2} = \pi^2/6$  and the last step is from the monotonicity of the  $\gamma_t$  and the nonnegative of  $\sigma_{u_t, t}(\mathbf{x})$ . Our next focus is bounding the summation of uncertainty. As the same approach in the proof of Theorem 4.2, we apply Lemma B.3 and the definition of effective dimension in equation 9

$$\begin{aligned}
\sum_{t=1}^T \sigma_{u_t, t-1}(\mathbf{x}_t) &\leq \sqrt{2T \max\{1, K_{\max}\} \cdot \log \det(\mathbf{I}_T + \lambda^{-1} \mathbf{K}_T)} \\
&= \sqrt{2T \max\{1, K_{\max}\} \cdot \tilde{d} \log(1 + T\lambda^{-1} K_{\max})}.
\end{aligned}$$

Thus we have

$$\sum_{t=1}^T \mathbb{E}[\Delta_t \mathbb{I}_{\mathcal{C}_t}] \leq 194\mathbb{E}[\gamma_T] \sqrt{2T \max\{1, K_{\max}\} \cdot \tilde{d} \log(1 + T\lambda^{-1} K_{\max})} + (4 + \frac{\pi^2}{6})B_\Delta$$

leading to the high probability  $(1 - \delta)$  regret upper bound:

$$\mathcal{R}_T \leq 194\mathbb{E}[\gamma_T] \sqrt{2T \max\{1, K_{\max}\} \cdot \tilde{d} \log(1 + T\lambda^{-1} K_{\max})} + (4 + \frac{\pi^2}{6})B_\Delta + \delta B_\Delta.$$

For the order of the upper bound, we first analyze  $\mathbb{E}[\gamma_T]$ , by using the definition of effective dimension  $\tilde{d}$  again and dropping constants

$$\gamma_T \leq \left(1 + \sqrt{2 \log(T^2 m)}\right) \cdot \left(B_\rho + \sqrt{\frac{\sigma^2}{\lambda} \cdot \tilde{d} \log(1 + T\lambda^{-1} K_{\max})} + \frac{2\sigma^2}{\lambda} \log \frac{1}{\delta}\right) \quad (19)$$

$$= \mathcal{O}(\log(T) \sqrt{\tilde{d}}). \quad (20)$$

Therefore,

$$\mathcal{R}_T = \mathcal{O}(\tilde{d} \log(T)^{3/2} \sqrt{T}) = \tilde{\mathcal{O}}(\tilde{d} \sqrt{T}).$$

□

## C PROOF OF LEMMAS

### C.1 PROOF OF LEMMA B.1

*Proof.* Using the standard Gaussian tail bound and the classical union bound, we have

$$\mathbb{P}_t(|z_t(\mathbf{x})| > u) \leq |\mathcal{D}_t| e^{-u^2/2}.$$

By letting  $u = \sqrt{2 \log(t^2 |\mathcal{D}_t|)}$ , we obtain  $\mathbb{P}_t(\tilde{\mathcal{E}}_t^{ts}) \leq t^{-2}$ .



For the result of event  $\mathcal{E}_t^a$ , we have

$$\begin{aligned}\mathbb{P}_t\left(\mu_{u_t,t-1}(\mathbf{x}_t^*) + \beta_t z_t(\mathbf{x}_t^*) \sigma_{u_t,t-1}(\mathbf{x}_t^*) > f(\mathbf{x}_t^*, u_t) | \mathcal{C}_t\right) &= \mathbb{P}_t\left(z_t(\mathbf{x}_t^*) > \frac{f(\mathbf{x}_t^*, u_t) - \mu_{u_t,t-1}(\mathbf{x}_t^*)}{\beta_t \sigma_{u_t,t-1}(\mathbf{x}_t^*)} | \mathcal{C}_t\right) \\ &\geq \mathbb{P}_t(z_t(\mathbf{x}_t^*) > 1) \\ &\geq (4e\sqrt{\pi})^{-1}\end{aligned}$$

where the first inequality is from the fact that  $\mathcal{C}_t$  holds and the last step is directly obtain by the fact that  $\mathbb{P}(Z \geq 1) \geq (4e\sqrt{\pi})^{-1}$  for  $Z \sim \mathcal{N}(0, 1)$ .

□

## C.2 PROOF OF LEMMA B.2

*Proof.* This proof is following the classical analysis for Thompson Sampling algorithms Kveton et al. (2019); Wu et al. (2022); Wu & Amini (2024).

We first recall  $\mathbb{E}_t[\cdot] = \mathbb{E}[\cdot | \mathcal{F}_t]$ . Given the randomness from the history  $\mathcal{F}_t$ , event  $\mathcal{C}_t$  becomes deterministic and the randomness is only from the resampling step. So we have

$$\begin{aligned}\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{C}_t}] &= \mathbb{I}_{\mathcal{C}_t} \cdot \mathbb{E}_t[\Delta_t] \\ &= \mathbb{I}_{\mathcal{C}_t} \cdot (\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{E}_t^{\text{ts}}}] + \mathbb{E}_t[\Delta_t \mathbb{I}_{\bar{\mathcal{E}}_t^{\text{ts}}}]) \\ &\leq \mathbb{I}_{\mathcal{C}_t} \cdot (\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{E}_t^{\text{ts}}}] + B_\Delta \cdot \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}}))\end{aligned}$$

where the last step is from the boundness of the optimality gap  $\Delta_t \leq B_\Delta$ . Our following focus is bounding  $\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{E}_t^{\text{ts}}}]$ , indicating  $\mathcal{C}_t$  holds in the remaining part of proof.

We then define the concept of "least uncertain undersampled" action, which is called unsaturated actions, defined as

$$\mathcal{U}_t := \{\mathbf{x} \in \mathcal{D}_t : f(\mathbf{x}_t^*, u_t) < f(\mathbf{x}, u_t) + \gamma_t \sigma_{u_t,t-1}(\mathbf{x})\}$$

where

$$\gamma_t := \beta_t + \beta_t \sqrt{2 \log(t^2 |\mathcal{D}_t|)}$$

and let  $\bar{\mathbf{x}}_t$  be the least uncertain unsaturated action at time  $t$ :

$$\bar{\mathbf{x}}_t = \arg \min_{\mathbf{x} \in \mathcal{U}_t} \gamma_t \sigma_{u_t,t-1}(\mathbf{x}).$$

Recall the notation for the resampled index is  $\tilde{\mu}_t(\mathbf{x}) = \mu_{u_t,t-1}(\mathbf{x}) + \beta_t z_t(\mathbf{x}) \sigma_{u_t,t-1}(\mathbf{x})$ . On the good situation  $\mathcal{C}_t \cap \mathcal{E}_t^{\text{ts}}$ , we have

$$|\tilde{\mu}_t(\mathbf{x}) - f(\mathbf{x}, u_t)| \leq |\tilde{\mu}_t(\mathbf{x}) - \mu_{u_t,t-1}(\mathbf{x})| + |\mu_{u_t,t-1}(\mathbf{x}) - f(\mathbf{x}, u_t)| \leq \gamma_t \sigma_{u_t,t-1}(\mathbf{x}).$$

Thus we can provide an initial upper bound for regret

$$\begin{aligned}\Delta_t &= f(\mathbf{x}_t^*, u_t) - f(\mathbf{x}_t, u_t) \\ &= f(\mathbf{x}_t^*, u_t) - f(\bar{\mathbf{x}}_t, u_t) + f(\bar{\mathbf{x}}_t, u_t) - f(\mathbf{x}_t, u_t) \\ &\leq \gamma_t \sigma_{u_t,t-1}(\bar{\mathbf{x}}_t) + f(\bar{\mathbf{x}}_t, u_t) - f(\mathbf{x}_t, u_t) + \tilde{\mu}_t(\mathbf{x}_t) - \tilde{\mu}_t(\mathbf{x}_t) \quad (\text{by } \bar{\mathbf{x}}_t \in \mathcal{U}_t) \\ &\leq 2\gamma_t \sigma_{u_t,t-1}(\bar{\mathbf{x}}_t) + \gamma_t \sigma_{u_t,t-1}(\mathbf{x}_t) + \tilde{\mu}_t(\bar{\mathbf{x}}_t) - \tilde{\mu}_t(\mathbf{x}_t) \quad (\text{since } \mathcal{C}_t \cap \mathcal{E}_t^{\text{ts}}) \\ &\leq 2\gamma_t \sigma_{u_t,t-1}(\bar{\mathbf{x}}_t) + \gamma_t \sigma_{u_t,t-1}(\mathbf{x}_t) \quad (\text{by } \tilde{\mu}_t(\bar{\mathbf{x}}_t) < \tilde{\mu}_t(\mathbf{x}_t)).\end{aligned}\tag{21}$$

Note that

$$\gamma_t \sigma_{u_t,t-1}(\bar{\mathbf{x}}_t) \mathbb{I}\{\mathbf{x}_t \in \mathcal{U}_t\} \leq \gamma_t \sigma_{u_t,t-1}(\mathbf{x}_t)$$

and by taking  $\mathbb{E}_t[\cdot]$  after multiplying both sides by  $\mathbb{I}_{\mathcal{E}_t^{\text{ts}}}$ , we have

$$\sigma_{u_t,t-1}(\bar{\mathbf{x}}_t) \mathbb{P}_t(\{\mathbf{x}_t \in \mathcal{U}_t\} \cap \mathcal{E}_t^{\text{ts}}) \leq \mathbb{E}_t[\sigma_{u_t,t-1}(\mathbf{x}_t) \mathbb{I}_{\mathcal{E}_t^{\text{ts}}}].$$

Thus it remains to bound the probability  $\mathbb{P}_t(\{\mathbf{x}_t \in \mathcal{U}_t\} \cap \mathcal{E}_t^{\text{ts}})$  from below.

We notice the following two facts. First, if  $\tilde{\mu}_t(\mathbf{x}_t^*) > \tilde{\mu}_t(\mathbf{x})$  for all  $\mathbf{x} \in \bar{\mathcal{U}}_t$ , then  $\mathbf{x}_t$  must belong to  $\mathcal{U}_t$ , which means  $\{\tilde{\mu}_t(\mathbf{x}_t^*) > \max_{\mathbf{x} \in \bar{\mathcal{U}}_t} \tilde{\mu}_t(\mathbf{x})\} \subseteq \{\mathbf{x}_t \in \mathcal{U}_t\}$ . Second, for any  $\mathbf{x} \in \bar{\mathcal{U}}_t$ , on the good situation  $\mathcal{C}_t \cap \mathcal{E}_t^{\text{ts}} \cap \mathcal{E}_t^a$ , we have

$$\tilde{\mu}_t(\mathbf{x}) \leq f(\mathbf{x}, u_t) + \gamma_t \sigma_{u_t, t-1}(\mathbf{x}) \leq f(\mathbf{x}_t^*, u_t) < \tilde{\mu}_t(\mathbf{x}_t^*)$$

which leads to  $\mathcal{E}_t^a \subseteq \{\tilde{\mu}_t(\mathbf{x}_t^*) > \max_{\mathbf{x} \in \bar{\mathcal{U}}_t} \tilde{\mu}_t(\mathbf{x})\}$

Therefore, on event  $\mathcal{C}_t$ , we have

$$\begin{aligned} \mathbb{P}_t(\{\mathbf{x}_t \in \mathcal{U}_t\} \cap \mathcal{E}_t^{\text{ts}}) &\geq \mathbb{P}_t(\{\tilde{\mu}_t(\mathbf{x}_t^*) > \max_{\mathbf{x} \in \bar{\mathcal{U}}_t} \tilde{\mu}_t(\mathbf{x})\} \cap \mathcal{E}_t^{\text{ts}}) \\ &\geq \mathbb{P}_t(\mathcal{E}_t^a \cap \mathcal{E}_t^{\text{ts}}) \\ &\geq \mathbb{P}_t(\mathcal{E}_t^a) - \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}}) \end{aligned}$$

Now we have a upper bound for  $\sigma_{u_t, t-1}(\bar{\mathbf{x}}_t)$ :

$$\sigma_{u_t, t-1}(\bar{\mathbf{x}}_t) \leq \frac{\mathbb{E}_t[\sigma_{u_t, t-1}(\mathbf{x}_t) \mathbb{I}_{\mathcal{E}_t^{\text{ts}}}] }{\mathbb{P}_t(\{\mathbf{x}_t \in \mathcal{U}_t\} \cap \mathcal{E}_t^{\text{ts}})} \leq \frac{\mathbb{E}_t[\sigma_{u_t, t-1}(\mathbf{x}_t)]}{\mathbb{P}_t(\mathcal{E}_t^a) - \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}})}$$

which gives the upper bound for instantaneous regret by plugging above result in equation 21:

$$\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{E}_t^{\text{ts}}}] \leq \left( \frac{2}{\mathbb{P}_t(\mathcal{E}_t^a) - \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}})} + 1 \right) \cdot \mathbb{E}_t[\gamma_t \sigma_{u_t, t-1}(\mathbf{x}_t)].$$

Therefore,

$$\mathbb{E}_t[\Delta_t \mathbb{I}_{\mathcal{C}_t}] \leq \mathbb{I}_{\mathcal{C}_t} \cdot \left\{ \left( \frac{2}{\mathbb{P}_t(\mathcal{E}_t^a) - \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}})} + 1 \right) \cdot \mathbb{E}_t[\gamma_t \sigma_{u_t, t-1}(\mathbf{x}_t)] + B_\Delta \cdot \mathbb{P}_t(\bar{\mathcal{E}}_t^{\text{ts}}) \right\}$$

□

### C.3 PROOF OF LEMMA B.3

*Proof.* We first apply Cauchy-Schwartz inequality and obtain

$$\sum_{t=1}^T \sigma_{u_t, t-1}(\mathbf{x}_t) \leq \sqrt{T \sum_{t=1}^T \sigma_{u_t, t-1}^2(\mathbf{x}_t)} = \sqrt{\lambda T \sum_{t=1}^T \frac{\sigma_{u_t, t-1}^2(\mathbf{x}_t)}{\lambda}}.$$

If  $\lambda \geq K_{\max}$ , using  $\sigma_{u_t, t-1}^2(\mathbf{x}_t) \leq |K((\mathbf{x}_t, u_t)(\mathbf{x}_t, u_t))| \leq K_{\max}$ , we know  $\frac{\sigma_{u_t, t-1}^2(\mathbf{x}_t)}{\lambda} \leq 1$ , which leads to

$$\sum_{t=1}^T \frac{\sigma_{u_t, t-1}^2(\mathbf{x}_t)}{\lambda} \leq 2 \sum_{t=1}^T \log \left( 1 + \frac{1}{\lambda} \sigma_{u_t, t-1}^2(\mathbf{x}_t) \right) \leq \frac{2K_{\max}}{\lambda} \sum_{t=1}^T \log \left( 1 + \frac{1}{\lambda} \sigma_{u_t, t-1}^2(\mathbf{x}_t) \right)$$

by applying the fact that  $x \leq 2 \log(1+x)$  if  $x \leq 1$ .

If  $\lambda \leq K_{\max}$ , still using  $\sigma_{u_t, t-1}^2(\mathbf{x}_t) \leq |K((\mathbf{x}_t, u_t)(\mathbf{x}_t, u_t))| \leq K_{\max}$ , we know

$$\frac{\sigma_{u_t, t-1}^2(\mathbf{x}_t)}{\lambda} \leq \min \left\{ \frac{K_{\max}}{\lambda}, \frac{\sigma_{u_t, t-1}^2(\mathbf{x}_t)}{\lambda} \right\} \leq \frac{K_{\max}}{\lambda} \min \{1, \frac{\sigma_{u_t, t-1}^2(\mathbf{x}_t)}{\lambda}\}$$

which leads to

$$\sum_{t=1}^T \frac{\sigma_{u_t, t-1}^2(\mathbf{x}_t)}{\lambda} \leq \frac{K_{\max}}{\lambda} \sum_{t=1}^T \min \{1, \frac{1}{\lambda} \sigma_{u_t, t-1}^2(\mathbf{x}_t)\} \leq \frac{2K_{\max}}{\lambda} \sum_{t=1}^T \log \left( 1 + \frac{1}{\lambda} \sigma_{u_t, t-1}^2(\mathbf{x}_t) \right).$$

by applying the fact that  $\min\{1, x\} \leq 2 \log(1+x)$  for  $x \geq 0$ .

We can summarize the above two conditions for  $\lambda$  together and achieve

$$\sum_{t=1}^T \sigma_{u_t, t-1}(\mathbf{x}_t) \leq \sqrt{2T \max\{1, K_{\max}\} \sum_{t=1}^T \log \left( 1 + \frac{1}{\lambda} \sigma_{u_t, t-1}^2(\mathbf{x}_t) \right)}. \quad (22)$$

Now we can use the property of the Shur complement for  $K_t$ :

$$\begin{aligned} \det\left(\mathbf{I}_t + \frac{1}{\lambda}\mathbf{K}_t\right) &= \det\left(\mathbf{I}_{t-1} + \frac{1}{\lambda}\mathbf{K}_{t-1}\right) \\ &\quad \times \left[1 + \frac{1}{\lambda} \underbrace{\left(K((\mathbf{x}_t, u_t), (\mathbf{x}_t, u_t)) - \mathbf{k}_{t-1}(\mathbf{x}_t, u_t)^\top (\mathbf{K}_{t-1} + \lambda\mathbf{I})^{-1} \mathbf{k}_{t-1}(\mathbf{x}_t, u_t)\right)}_{\sigma_{u_t, t-1}^2(\mathbf{x}_t)}\right] \end{aligned}$$

which leads to

$$\sum_{t=1}^T \log\left(1 + \frac{1}{\lambda} \sigma_{u_t, t-1}^2(\mathbf{x}_t)\right) = \sum_{t=1}^T \log \frac{\det(\mathbf{I}_t + \frac{1}{\lambda}\mathbf{K}_t)}{\det(\mathbf{I}_{t-1} + \frac{1}{\lambda}\mathbf{K}_{t-1})} = \log \det(\mathbf{I}_T + \lambda^{-1}\mathbf{K}_T).$$

Therefore, we combine above result with equation 22 and obtain

$$\sum_{t=1}^T \sigma_{u_t, t-1}(\mathbf{x}_t) \leq \sqrt{2T \max\{1, K_{\max}\} \cdot \log \det(\mathbf{I}_T + \lambda^{-1}\mathbf{K}_T)}$$

□

#### C.4 PROOF OF LEMMA B.5

*Proof.* We note that

$$\|\delta_t(\mathbf{x}, u)\|_{\ell^2}^2 = \|\phi((\mathbf{x}, u))\|_{\ell^2}^2 + \|\Phi_t^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u)\|_{\ell^2}^2 - 2\langle \phi((\mathbf{x}, u)), \Phi_t^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) \rangle_{\ell^2}$$

and we have

$$\begin{aligned} \|\Phi_t^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u)\|_{\ell^2}^2 &= \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \Phi_t \Phi_t^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) \\ &= \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \mathbf{K}_t \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) \\ &= \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \Sigma_t \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) - \lambda \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-2} \mathbf{k}_t(\mathbf{x}, u) \\ &= \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) - \lambda \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-2} \mathbf{k}_t(\mathbf{x}, u) \end{aligned}$$

and

$$\langle \phi((\mathbf{x}, u)), \Phi_t^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) \rangle_{\ell^2} = \phi((\mathbf{x}, u))^\top \Phi_t^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) = \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u).$$

Putting above equalities together, we have

$$\begin{aligned} \|\delta_t(\mathbf{x}, u)\|_{\ell^2}^2 &= \|\phi((\mathbf{x}, u))\|_{\ell^2}^2 - \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) - \lambda \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-2} \mathbf{k}_t(\mathbf{x}, u) \\ &= K((\mathbf{x}, u), (\mathbf{x}, u)) - \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) - \lambda \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-2} \mathbf{k}_t(\mathbf{x}, u) \\ &= \sigma_{u, t}^2(\mathbf{x}) - \lambda \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-2} \mathbf{k}_t(\mathbf{x}, u) \\ &\leq \sigma_{u, t}^2(\mathbf{x}) \end{aligned}$$

since  $\Sigma_t^{-1}$  is positive semidefinite.

□

#### C.5 PROOF OF LEMMA B.6

*Proof.* We first define

$$\mathbf{s}_t = \Phi_t \boldsymbol{\epsilon}_t = \sum_{s=1}^t \phi(\mathbf{x}_s, u_s) \epsilon_s.$$

Note that  $\mathbf{s}_t$  is a martingale w.r.t  $\mathcal{F}_t$ .

Also we define a supermartingale

$$M_t(\mathbf{g}) = \exp\left(\sum_{s=1}^t \frac{1}{\sigma} \langle \mathbf{g}, \mathbf{s}_s \rangle - \frac{1}{2} \|\mathbf{g}\|^2\right)$$

which has an alternative form

$$M_t(\mathbf{g}) = \exp\left(\sum_{s=1}^t \frac{1}{\sigma} \langle \mathbf{g}, \phi(\mathbf{x}_s, u_s) \rangle \epsilon_s - \frac{1}{2} \|\mathbf{g}\|^2\right)$$

where  $\mathbf{g}$  is the function vector with elements

We follow the approach from classical linear bandit Abbasi-Yadkori et al. (2011), which is averaging  $M_t(\mathbf{g})$  w.r.t a Gaussian distribution on  $\mathbf{g}$ . The key technical issue is the infinite dimension of the function vector  $\mathbf{g}$ . We will first perform the truncated version which can precisely match the classical result. Let  $d$  be the dimension of the feature map. Our target is the obtain the limiting result when  $d \rightarrow \infty$ . Now assume  $\mathbf{g}^d \sim \mathcal{N}(\mathbf{0}, \frac{1}{\lambda} \mathbf{I}_d)$ , independent of everything else, and define

$$M_t^{(d)} = \mathbb{E}_{\mathbf{g}^d} [M_t(\mathbf{g}^d)] = \int M_t^{(d)}(\mathbf{g}) d\rho^d(\mathbf{g})$$

and by iterated expectation (i.e Fubini's theorem), we have

$$\mathbb{E}[M_t^{(d)} | \mathcal{F}_t] \leq M_{t-1}$$

which shows that  $M_t$  is a supermartingale.

Then we define  $\Psi : \ell^2 \rightarrow \mathbb{R}^d$  as the truncation projection onto the first  $d$  coordinates:  $\Psi_d \boldsymbol{\theta} = [\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_d]^\top$  for any  $\boldsymbol{\theta} \in \ell^2$ . We further denote

$$\Psi_d \Phi_t^\top = [\Psi_d \phi(\mathbf{x}_1, u_1), \dots, \Psi_d \phi(\mathbf{x}_t, u_t)] \in \mathbb{R}^{d \times t}$$

and

$$\Psi_d \mathbf{J}_t \Psi_d^\top = \Psi_d \Phi_t^\top \Phi_t \Psi_d.$$

We notices that

$$\frac{\det(\lambda \mathbf{I}_d)}{\det(\lambda \mathbf{I}_d + \Psi_d \mathbf{J}_t \Psi_d^\top)} = \frac{1}{\det(\mathbf{I}_d + \lambda^{-1} \Psi_d \mathbf{J}_t \Psi_d^\top)}$$

which leads to

$$\begin{aligned} M_t^{(d)} &= \left( \frac{\det(\lambda \mathbf{I}_d)}{\det(\lambda \mathbf{I}_d + \Psi_d \mathbf{J}_t \Psi_d^\top)} \right)^{1/2} \exp\left( \frac{1}{2\sigma^2} \|\Psi_d \Phi_t \boldsymbol{\epsilon}_t\|_{(\lambda \mathbf{I}_d + \Psi_d \mathbf{J}_t \Psi_d^\top)^{-1}}^2 \right) \\ &= \det(\mathbf{I}_d + \lambda^{-1} \Psi_d \mathbf{J}_t \Psi_d^\top)^{-1/2} \exp\left( \frac{1}{2\sigma^2} \|\Psi_d \Phi_t \boldsymbol{\epsilon}_t\|_{(\lambda \mathbf{I}_d + \Psi_d \mathbf{J}_t \Psi_d^\top)^{-1}}^2 \right). \end{aligned}$$

Let  $M_t$  be the limit of  $M_t^{(d)}$  as  $d \rightarrow \infty$ , we have

$$\begin{aligned} M_t &= \det(\mathbf{I}_\infty + \lambda^{-1} \mathbf{J}_t)^{-1/2} \exp\left( \frac{1}{2\sigma^2} \|\Phi_t \boldsymbol{\epsilon}_t\|_{(\lambda \mathbf{I}_\infty + \mathbf{J}_t)^{-1}}^2 \right) \\ &= \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t)^{-1/2} \exp\left( \frac{1}{2\sigma^2} \|\Phi_t \boldsymbol{\epsilon}_t\|_{\Gamma_t^{-1}}^2 \right) \end{aligned}$$

where the second step is from (Sylvestr) or Weinstein–Aronszajn identity.

By Ville's inequality,

$$\mathbb{P}\left(\sup_{t=0,1,2,\dots} M_t \geq \frac{1}{\delta}\right) \leq \mathbb{E}[M_0] \cdot \delta$$

and  $M_0 = 1$ . Thus we know that, with probability at least  $1 - \delta$ , for all  $t = 0, 1, 2, \dots$

$$\log(M_t) \leq \log\left(\frac{1}{\delta}\right)$$

which leads to

$$-\frac{1}{2} \log \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t) + \frac{1}{2\sigma^2} \|\Phi_t \boldsymbol{\epsilon}_t\|_{\Gamma_t^{-1}}^2 \leq \log\left(\frac{1}{\delta}\right).$$

After re-arranging, we get

$$\|\Phi_t \boldsymbol{\epsilon}_t\|_{\Gamma_t^{-1}}^2 \leq 2\sigma^2 \log \frac{\sqrt{\det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t)}}{\delta}.$$

which shows our result.  $\square$

## C.6 PROOF OF LEMMA B.4

*Proof.* Let us write  $\Phi_t = U_t \Lambda_t V_t^\top$  as the singular value decomposition(SVD) of  $\Phi_t$ . We have  $\Lambda_t = [\Lambda_{1,t}, \mathbf{0}]$  where  $\Lambda_{1,t}$  is a  $t \times t$  diagonal matrix with singular values of  $\Phi_t$ . We also note that  $\Sigma_t \in \mathbb{R}^{t \times t}$  and  $U_t \in \mathbb{R}^{t \times t}$ . We also have

$$J_t = \Phi_t^\top \Phi_t = V_t \begin{bmatrix} \Lambda_{1,t}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} V_t^\top$$

and similarly

$$K_t = \Phi_t \Phi_t^\top = U_t \Lambda_{1,t}^2 U_t^\top.$$

Then, we have

$$\Gamma_t = V_t \begin{bmatrix} \Lambda_{1,t}^2 + \lambda I_t & \mathbf{0} \\ \mathbf{0} & \lambda I_\infty \end{bmatrix} V_t^\top, \quad \Sigma_t = U_t (\Lambda_{1,t}^2 + \lambda I_t) U_t^\top.$$

It is clear to have the identity:

$$\Sigma_t^{-1} \Phi_t = \Phi_t \Gamma_t^{-1}$$

since both side equal  $U_t [D_t, \mathbf{0}] V_t^\top$  where  $D_t = \Lambda_{1,t} (\Lambda_{1,t}^2 + \lambda I_t)^{-1}$ , which is a diagonal matrix.

Next, we note that

$$\begin{aligned} \sigma_{u,t}^2(\mathbf{x}) &= K((\mathbf{x}, u), (\mathbf{x}, u)) - \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}, u) \\ &= \phi(\mathbf{x}, u)^\top (I_\infty - \Phi_t^\top \Sigma_t^{-1} \Phi_t) \phi(\mathbf{x}, u) \\ &= \phi(\mathbf{x}, u)^\top (I_\infty - \Phi_t^\top \Phi_t \Gamma_t^{-1}) \phi(\mathbf{x}, u) \end{aligned}$$

which is a norm of  $\phi(\mathbf{x}, u)$  induced by matrix

$$\begin{aligned} I_\infty - \Phi_t^\top \Phi_t \Gamma_t^{-1} &= I_\infty - J_t \Gamma_t^{-1} \\ &= V_t \begin{bmatrix} \lambda I_t (\Lambda_{1,t}^2 + \lambda I_t)^{-1} & \mathbf{0} \\ \mathbf{0} & \lambda I_\infty \end{bmatrix} V_t^\top \\ &= \lambda V_t \begin{bmatrix} (\Lambda_{1,t}^2 + \lambda I_t)^{-1} & \mathbf{0} \\ \mathbf{0} & I_\infty \end{bmatrix} V_t^\top \\ &= \lambda \Gamma_t^{-1}. \end{aligned}$$

Therefore, we have the other identity

$$\sigma_{u,t}^2(\mathbf{x}) = \lambda \|\phi(\mathbf{x}, u)\|_{\Gamma_t^{-1}}^2.$$

□

## D MISCELLANEOUS

### D.1 ALGORITHMS

---

#### Algorithm 1 LK-GP-UCB

---

- 1: **Input:**  $T, \lambda, \{\beta_t\}_{t=1}^T$
  - 2: **Initialization:**  $\mu_{u,0}(\mathbf{x}), \sigma_{u,0}(\mathbf{x})$
  - 3: **for**  $t = 1, \dots, T$  **do**
  - 4:   Observe user  $u_t$  and arm set  $\mathcal{D}_t$ .
  - 5:   Select arm  $\mathbf{x}_t = \arg \max_{\mathbf{x} \in \mathcal{D}_t} \mu_{u_t, t-1}(\mathbf{x}) + \beta_t \sigma_{u_t, t-1}(\mathbf{x})$ .
  - 6:   Receive feedback  $y_t = f(\mathbf{x}_t, u_t) + \epsilon_t$ .
  - 7:   Update  $\mu_{u_t, t}(\mathbf{x})$  and  $\sigma_{u_t, t}^2(\mathbf{x})$ .
  - 8: **end for**
-

**Algorithm 2** LK-GP-TS

---

```

1: Input:  $T, \lambda, \{\nu_t\}_{t=1}^T$ 
2: Initialization:  $\mu_{u,0}(\mathbf{x}), \sigma_{u,0}(\mathbf{x})$ 
3: for  $t = 1, \dots, T$  do
4:   Observe user  $u_t$  and arm set  $\mathcal{D}_t$ .
5:   Sample  $\tilde{\mu}_t(\mathbf{x})$  from  $\mathcal{N}(\mu_{u_t,t-1}(\mathbf{x}), \nu_t^2 \sigma_{u_t,t-1}^2(\mathbf{x}))$  for all  $\mathbf{x} \in \mathcal{D}_t$ 
6:   Select arm  $\mathbf{x}_t = \arg \max_{\mathbf{x} \in \mathcal{D}_t} \tilde{\mu}_t(\mathbf{x})$ .
7:   Receive feedback  $y_t = f(\mathbf{x}_t, u_t) + \epsilon_t$ .
8:   Update  $\mu_{u_t,t}(\mathbf{x})$  and  $\sigma_{u_t,t}^2(\mathbf{x})$ .
9: end for

```

---

**D.2 RECURSIVE UPDATE OF POSTERIOR MEAN AND VARIANCE**

This sections refers to the derivation of incremental update of the posterior mean and posterior variance Chowdhury & Gopalan (2017), via the properties of Schur complement. Recall that we need to handle the inversion of  $\Sigma_t = \mathbf{I} + \lambda \mathbf{K}_t \in \mathbb{R}^{t \times t}$  which grows with the number of rounds. To compute the inversion of  $\Sigma_t$  efficiently, we use the recursive formula from  $\Sigma_{t-1}$  by block matrix inverse formula

$$\Sigma_t^{-1} = \begin{bmatrix} M_{11,t} & M_{12,t} \\ M_{12,t}^\top & d_t^{-1} \end{bmatrix} \quad (23)$$

where

$$\begin{aligned} M_{11,t} &= \Sigma_{t-1}^{-1} + d_t^{-1} \mathbf{G}_t \\ M_{12,t} &= -d_t^{-1} \Sigma_{t-1}^{-1} \mathbf{k}_{t-1}(\mathbf{x}_t, u_t) \end{aligned} \quad (24)$$

and

$$\begin{aligned} d_t &= K((\mathbf{x}_t, u_t), (\mathbf{x}_t, u_t)) - \mathbf{k}_{t-1}(\mathbf{x}_t, u_t)^\top \Sigma_{t-1}^{-1} \mathbf{k}_{t-1}(\mathbf{x}_t, u_t) + \lambda = \sigma_{u_t,t-1}^2(\mathbf{x}_t) + \lambda \\ \mathbf{G}_t &= \Sigma_{t-1}^{-1} \mathbf{k}_{t-1}(\mathbf{x}_t, u_t) \mathbf{k}_{t-1}(\mathbf{x}_t, u_t)^\top \Sigma_{t-1}^{-1} \end{aligned}$$

Here  $d_t$  is the Schur complement.

Thus we have the posterior mean using equation 23

$$\begin{aligned} \mu_{u,t}(\mathbf{x}) &= \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \mathbf{y}_t \\ &= [\mathbf{k}_{t-1}(\mathbf{x}, u)^\top \quad K((\mathbf{x}, u), (\mathbf{x}_t, u_t))] \begin{bmatrix} M_{11,t} & M_{12,t} \\ M_{12,t}^\top & d_t^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{y}_{t-1} \\ y_t \end{bmatrix} \\ &= \mathbf{k}_{t-1}(\mathbf{x}, u)^\top M_{11,t} \mathbf{y}_{t-1} + K((\mathbf{x}, u), (\mathbf{x}_t, u_t)) M_{12,t}^\top \mathbf{y}_{t-1} \\ &\quad + \mathbf{k}_{t-1}(\mathbf{x}, u)^\top M_{12,t} y_t + K((\mathbf{x}, u), (\mathbf{x}_t, u_t)) d_t^{-1} y_t \\ &= \underbrace{\mathbf{k}_{t-1}(\mathbf{x}, u)^\top \Sigma_{t-1}^{-1} \mathbf{y}_{t-1}}_{\mu_{u,t-1}(\mathbf{x})} + d_t^{-1} (\beta_1 \mathbf{y}_{t-1} - \beta_2 \mathbf{y}_{t-1} - \beta_3 y_t + \beta_4 y_t) \end{aligned}$$

where

$$\begin{aligned} \beta_1 &= \mathbf{k}_{t-1}(\mathbf{x}, u)^\top \mathbf{G}_t \Rightarrow \beta_1 \mathbf{y}_{t-1} = \left( \mathbf{k}_{t-1}(\mathbf{x}, u)^\top \Sigma_{t-1}^{-1} \mathbf{k}_{t-1}(\mathbf{x}_t, u_t) \right) \mu_{u_t,t-1}(\mathbf{x}_t) \\ \beta_2 &= K((\mathbf{x}, u), (\mathbf{x}_t, u_t)) \mathbf{k}_{t-1}(\mathbf{x}_t, u_t)^\top \Sigma_{t-1}^{-1} \Rightarrow \beta_2 \mathbf{y}_{t-1} = K((\mathbf{x}, u), (\mathbf{x}_t, u_t)) \mu_{u_t,t-1}(\mathbf{x}_t) \\ \beta_3 &= \mathbf{k}_{t-1}(\mathbf{x}, u)^\top \Sigma_{t-1}^{-1} \mathbf{k}_{t-1} \\ \beta_4 &= K((\mathbf{x}, u), (\mathbf{x}_t, u_t)). \end{aligned}$$

Thus we have the recursive update of posterior mean

$$\begin{aligned} \mu_{u,t}(\mathbf{x}) &= \mu_{u,t-1}(\mathbf{x}) + d_t^{-1} \\ &\quad \times \left( \mathbf{k}_{t-1}(\mathbf{x}, u)^\top \Sigma_{t-1}^{-1} \mathbf{k}_{t-1}(\mathbf{x}_t, u_t) (\mu_{u_t,t-1}(\mathbf{x}_t) - y_t) + K((\mathbf{x}, u), (\mathbf{x}_t, u_t)) (y_t - \mu_{u_t,t-1}(\mathbf{x}_t)) \right) \\ &= \mu_{u,t-1}(\mathbf{x}) + d_t^{-1} q_{t-1}((\mathbf{x}, u), (\mathbf{x}_t, u_t)) (y_t - \mu_{u_t,t-1}(\mathbf{x}_t)) \end{aligned}$$

where  $q_{t-1}((\mathbf{x}, u), (\mathbf{x}_t, u_t))$  is defined from

$$q_t((\mathbf{x}, u), (\mathbf{x}', u')) = K((\mathbf{x}, u), (\mathbf{x}', u')) - \mathbf{k}_t(\mathbf{x}, u)^\top \Sigma_t^{-1} \mathbf{k}_t(\mathbf{x}', u')$$

which can be transferred into a recursive form using equation 23

$$\begin{aligned} & q_t((\mathbf{x}, u), (\mathbf{x}', u')) \\ &= K((\mathbf{x}, u), (\mathbf{x}', u')) - \left( \mathbf{k}_{t-1}(\mathbf{x}, u)^\top \Sigma_{t-1}^{-1} \mathbf{k}_{t-1}(\mathbf{x}', u') \right. \\ & \quad \left. + d_t^{-1} (\beta_1 \mathbf{k}_{t-1}(\mathbf{x}', u') - \beta_2 \mathbf{k}_{t-1}(\mathbf{x}', u') - \beta_3 K((\mathbf{x}_t, u_t), (\mathbf{x}', u')) + \beta_4 K((\mathbf{x}_t, u_t), (\mathbf{x}', u'))) \right) \\ &= q_{t-1}((\mathbf{x}, u), (\mathbf{x}', u')) - d_t^{-1} q_{t-1}((\mathbf{x}, u), (\mathbf{x}_t, u_t)) q_{t-1}((\mathbf{x}_t, u_t), (\mathbf{x}', u')). \end{aligned}$$

Now using the incremental update of the posterior covariance, we can easily obtain the recursive update for the posterior variance

$$\sigma_{u,t}^2(\mathbf{x}) = \sigma_{u,t-1}^2(\mathbf{x}) - d_t^{-1} q_{t-1}^2((\mathbf{x}, u), (\mathbf{x}_t, u_t)).$$

Now replace  $d_t$  by  $\sigma_{u,t-1}^2(\mathbf{x}_t) + \lambda$  and we achieve the recursive updates in equation 7.

### D.3 COMPARISON TO RELATED RESEARCH

#### D.3.1 COMPARISON ON REGRET ANALYSIS IN KERNELIZED BANDIT

Our proposal methodology for the multi-user bandit problem is the GP-based bandit algorithms and we adapt the regret analysis kernelized GP-based bandit Chowdhury & Gopalan (2017). In classical kernelized bandit, the confidence bound and regret bound relies on the maximum information gain, defined as

$$\gamma_t = \max_{A \subset \mathcal{D}} I(\mathbf{y}_A; f_A)$$

where  $I(\mathbf{y}_A; f_A)$  is the mutual information between  $f_A = [f(\mathbf{x})]_{\mathbf{x} \in A}$  and  $\mathbf{y}_A = f_A + \epsilon$  and  $\epsilon$  is the Gaussian noise vector. The maximum information gain is mainly applied on the bounding of information in the log-determinant term  $\log \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t)$ . Instead, our regret analysis use effective dimension  $\tilde{d}$  to capture the information of  $\log \det(\mathbf{I}_t + \lambda^{-1} \mathbf{K}_t)$ . Using our method can avoid the restriction on the regularization that  $\lambda > 1$  in the classical regret analysis for kernelized bandit.

#### D.3.2 COMPARISON TO COOPERATIVE MULTI-AGENT KERNELIZED BANDITS

**Comparison on the problem.** Our work consider the Gang-of-Bandits problem, which is exactly the cooperative multi-agent kernelized bandit Dubey et al. (2020). There are 3 key differences between the problems we focus. We explicitly consider the regularized Laplacian  $\mathbf{L}_\rho^{-1/2}$  as the agent similarity kernel. Formally,  $K_z(u, v) = [\mathbf{L}_\rho^{-1}]_{u,v}$ . Additionally, we assume the user-level reward functions  $f_{1:n}$  are from the same RKHS while the previous paper assumes  $f_u$  has its own RKHS  $\mathcal{H}_{x,u}$  for all  $u \in \mathcal{U}$ . Lastly, we consider a centralized agent/learner to observe with on-time feedback while they consider a decentralized version of the problem. There are also other differences in detail, such as the assumption on graph smoothness.

**Comparison on the method.** We propose to use Laplacian Kernelized Gaussian Process (LK-GP) method in bandit algorithm design, leading to LK-GP-UCB and LK-GP-TS. The CoopKernelUCB for cooperative multi-agent kernelized bandit problem is a UCB-based algorithm. In addition, the similarity kernel across the agents are learned in CoopKernelUCB and we apply the regularized Laplacian. Regarding the regret analysis, we use the effective dimension while the analysis for CoopKernelUCB follows the idea of information gain, which is classical but distribution-dependent for kernel-based method.

As a summary, we consider a more concrete and practical setting comparing to the problem in the paper for CoopKernelUCB. Also, we propose TS-based algorithm with kernel method to solve the problem. The advantage of using CoopKernelUCB is to solve more general and hard setting while our LK-GP bandit algorithms provide the explicit form choice of the similarity kernel and show superiority on our preferred regime.

## E SUPPLEMENT TO EXPERIMENTS

This appendix provides full details of our synthetic environments, algorithm configurations, hyperparameter selection, implementation choices, ablations, and reporting protocol.

### E.1 SYNTHETIC ENVIRONMENTS

Let  $\mathcal{U} = \{1, \dots, n\}$  denote users,  $\mathcal{D} \subset \mathbb{R}^d$  the arm (context) space, and  $m_t := |\mathcal{D}_t|$  the number of candidates shown at round  $t$ . We draw a global normalized context pool  $\mathcal{D} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$  with  $\mathbf{x}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$  and  $\mathbf{x}^{(i)} \leftarrow \mathbf{x}^{(i)} / \|\mathbf{x}^{(i)}\|$ . At round  $t$  we present  $\mathcal{D}_t$  by sampling  $m_t$  distinct items from  $\mathcal{D}$  without replacement. One user  $u_t$  is served per round, drawn uniformly from  $\mathcal{U}$  unless stated otherwise. Rewards are observed with additive noise  $y_t = f(\mathbf{x}_t, u_t) + \epsilon_t$ . We generate graphs, contexts, and ground-truth rewards under one linear regime (*Linear-GOB*) and two kernelized regimes (*Laplacian-Kernel* using GP draw and representer draw).

**User graph.** We consider two graph random generators on  $\mathcal{U}$ . First random graph family is Erdős-Rényi (ER) random graphs: each (undirected) edge is present with probability  $p$  and weights  $w_{ij} = 1$ . We set  $p = 0.2$  in our experiment. Second one is Radial basis function(RBF) random graphs: sample latent  $\mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$ , set  $w_{ij} = \exp(-\rho_L \|\mathbf{z}_i - \mathbf{z}_j\|_2^2)$ , and sparsify by keeping edges with  $w_{ij} \geq s$ . We choose  $s = 0.1$ ,  $\rho_L = 0.1$  and  $q = 4$  in our simulation.

**Task Design.** We design different level of the task. The simplest case is  $(m, m_t, n, d, T) = (10, 5, 20, 5, 1000)$ . This is a 10-arm bandit problem with 50% viewability at each round for all users. The medium level is  $(m, m_t, n, d, T) = (20, 5, 20, 10, 3000)$  which leads to a 20-arm bandit problem with 25% viewability at each round for all users. We also have the toughest case using  $(m, m_t, n, d, T) = (50, 5, 20, 20, 3000)$  which leads to a 50-arm bandit problem with 10% viewability at each round for all users.  $\sigma$  is set as 0.1 unless additional specification.

#### E.1.1 REGIME 1: *Linear-GOB* (GRAPH-SMOOTH LINEAR REWARDS)

Sample initial user parameters  $\Theta_0 \in \mathbb{R}^{n \times d}$  with rows  $\theta_{0,i} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ . Enforce the graph homophily via Tikhonov smoothing Yankelevsky & Elad (2016):

$$\Theta = \arg \min_{\Theta} \|\tilde{\Theta} - \Theta_0\|_F^2 + \eta \text{tr}(\tilde{\Theta}^\top \mathbf{L} \tilde{\Theta}) = (\mathbf{I}_n + \eta \mathbf{L})^{-1} \Theta_0.$$

Thus  $f(\mathbf{x}, u) = \mathbf{x}^\top \theta_u$ , where  $\theta_u$  is row  $u$  of  $\Theta$ . The strength of the graph homophily  $\eta$  is set as 1.0 as default.

#### E.1.2 REGIME 2: *Laplacian-Kernel*

Our choice of the base kernel  $K_x$  over arms is *Squared Exponential* which are defined as

$$K_{SE}(\mathbf{x}, \mathbf{x}') = \exp\left(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\ell^2\right)$$

where length-scale  $\ell > 0$  and is set to be 1.0 in our experiment. Then we construct the *multi-user kernel* by the definition:

$$K((\mathbf{x}, u), (\mathbf{x}', u')) = [\mathbf{L}_\rho^{-1/2}]_{u, u'} K_x(\mathbf{x}, \mathbf{x}')$$

where we set  $\rho = 0.01$  in our experiment.

#### Option A: *Laplacian-Kernel* with GP draw

We draw the joint values  $\{f(\mathbf{x}, u)\}_{u \in \mathcal{U}, \mathbf{x} \in \mathcal{D}}$  from the zero-mean GP with covariance induced by  $K$  and fix  $f$  by interpolation on  $\mathcal{D} \times \mathcal{U}$ . Noise is  $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$  with  $\sigma = 0.01 \cdot \text{range}(f)$ .

**Option B: *Laplacian-Kernel* with representer draw** We consider the representer theorem for RKHS and sample the i.i.d. coefficients via  $\alpha_{\mathbf{x}, u} \sim \mathcal{N}(0, 1)$  on  $\mathcal{D} \times \mathcal{U}$  and set

$$f(\mathbf{x}, u) = \sum_{u' \in \mathcal{U}, \mathbf{x}' \in \mathcal{D}} \alpha_{\mathbf{x}, u} K((\mathbf{x}, u), (\mathbf{x}', u')).$$



## E.2 BASELINES

All methods face the *same* sequence  $\{u_t, \mathcal{D}_t, \epsilon_t\}_{t=1}^T$  in each trial of each synthetic environment to ensure a fair comparison. Our experiment include the following baselines.

**Per-User LinUCB(no graph).** We implement Per-User LinUCB, which ignores the whole graph and perform the linear bandit algorithm independently on each user.

**Pooled LinUCB(no graph).** We implement Pooled LinUCB, which ignores graph and personalization by treating the multi-user problem as a single agent bandit problem. Simply speaking, there is global linear UCB algorithm to solve the problem.

**GP-UCB(no graph).** We implement GP-UCB Chowdhury & Gopalan (2017), which is the IGP-UCB from the previous study on GP and UCB Chowdhury & Gopalan (2017). This is a kernelized baseline using  $K_x$  on arms only, ignoring the similarities across users (the Laplacian).

**GoB.Lin.** We implement GoB.Lin, which is the classical methods in gang-og-bandits problem Cesa-Bianchi et al. (2013). This is a Laplacian-regularized linear UCB algorithm on graph-whitened features (equivalent to GraphUCB with  $\rho = 1$  i.e  $A = I + L$ ). The confidence scale in the algorithm is tuned from the table.

**GraphUCB.** We implement GraphUCB Yang et al. (2020), the Laplacian-regularized LinUCB. Also, the confidence scale in the algorithm is tuned from the table.

**COOP-KernelUCB.** We implement COOP-KernelUCB Dubey et al. (2020), which utilizes the product kernel over agents  $\times$  arms. We consider five choices of  $K_z$  (presented below); the full kernel is  $K = K_z \otimes K_x$  and we apply the same UCB rule in LK-GP-UCB.

The five PSD options for the agent kernel  $K_z$ :

1. **laplacian\_inv:**  $K_z = (L + \rho I)^{-1}$ ,  $\rho > 0$ .
2. **heat:**  $K_z = \exp(-\tau L)$  via the spectral decomposition of  $L$ .
3. **spectral\_rbf:** embed nodes using the  $k$  lowest nontrivial Laplacian eigenvectors  $Z \in \mathbb{R}^{n \times k}$  and set

$$K_z[u, u'] = \exp\left(-\frac{\|Z_u - Z_{u'}\|^2}{2\sigma_z^2}\right).$$

4. **all ones:** full cooperation,  $K_z = \mathbf{1}\mathbf{1}^\top$ .
5. **learned\_mmd** (*network contexts*, faithful to Dubey et al. (2020)): define per-user kernel mean embeddings  $\Psi_u$  of the observed arm-context distribution in  $\mathcal{H}_x$ , and let

$$K_{z,t}(u, u') = \exp\left(-\frac{\|\hat{\Psi}_u(t) - \hat{\Psi}_{u'}(t)\|^2}{2\sigma_z^2}\right),$$

where  $\hat{\Psi}_u(t)$  is the empirical mean embedding of contexts observed for user  $u$  up to time  $t$ . In our implementation we use an efficient random Fourier feature approximation for  $K_x$  and update  $K_{z,t}$  on a fixed schedule; users with fewer than a small threshold of observations cooperate only with themselves (diagonal entries).

For time-varying  $K_z$  (learned\_mmd), the GP state is rebuilt at  $K_z$  refresh points using the algorithm's own history, ensuring consistency of the Gram matrix with the current kernel. Figure shows the comparison of the choice of  $K_z$  for COOP-KernelUCB.

## E.3 CENTRALIZED PROTOCOL

At each  $t$ : sample  $u_t \sim \text{Unif}(\mathcal{U})$ , present  $\mathcal{D}_t$  (size  $m_t$ ), select  $\mathbf{x}_t \in \mathcal{D}_t$  per the algorithm, observe  $y_t$ , update our decision policy(model), and record  $\Delta_t = \max_{\mathbf{x} \in \mathcal{D}_t} f(\mathbf{x}, u_t) - f(\mathbf{x}_t, u_t)$ . Each configuration is repeated for  $R$  trials (final results use  $R = 20$ ; preliminary/pilot tuning uses  $R \in [5, 10]$ ).

## E.4 POSTERIOR UPDATES AND NUMERICAL DETAILS

**Motivation.** For the original update equation 3 at round  $t$ , the inversion takes  $\mathcal{O}(t^3|\mathcal{D}_t|)$  time. The practical updates is efficient for each pair  $(\mathbf{x}, u)$  while it requires the updates for all pairs,

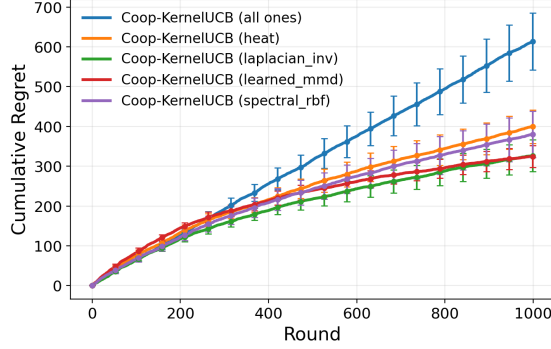


Figure 4: Comparison of the choice of user-similarity kernel for COOP-KernelUCB.

leading to  $\mathcal{O}(|\mathcal{D}_t|n)$  time. Therefore, high-level idea is to perform original updates equation 3 when  $t \leq n^{1/3}$  and perform practical updates equation 7 when  $t > n^{1/3}$ . Therefore, for our GP-based methods we use a hybrid implementation, which is described as below.

**Exact (Cholesky) phase:** maintain  $\Sigma_t = K_t + \lambda I$  and update via rank-one Cholesky for  $t < t_*$  (cost  $\mathcal{O}(t^2)$  per step; initial inversion  $\mathcal{O}(t^3)$ ).

**Recursive phase:** switch to the rank-one recursions in equation 7, with  $q_0 = K$  restricted to  $\mathcal{D} \times \mathcal{U}$ . This costs  $\mathcal{O}(n|\mathcal{D}_t|)$  per update when applied to the whole grid  $\mathcal{D} \times \mathcal{U}$ .

By default we take  $t_* = \min\{1500, \lfloor n^{1/3} \rfloor |\mathcal{D}|\}$  as the phase switch. We use Cholesky jitter  $10^{-8}$ , clip negative variances to zero, and cache  $K_x(\mathcal{D}, \mathcal{D})$ . For large  $n$  we optionally apply graph spectral truncation  $L_\rho \approx U_r \Lambda_r U_r^\top$  (top- $r$  eigenpairs), yielding  $K \approx (U_r \Lambda_r^{-1} U_r^\top) \otimes K_x$ .

## E.5 HYPERPARAMETERS AND TUNING

**What is fixed across algorithms.** For fairness, *base-kernel* hyperparameters are fixed inside each environment: the length-scale  $\ell$  uses the median heuristic on  $\mathcal{D}$ , and the Laplacian ridge  $\rho = 0.1$  is fixed. For  $K_z$ , *laplacian\_inv* uses  $\rho = 0.1$ ; *heat* uses  $\tau = 1.0$ ; *spectral\_rbf* uses  $k = 8$  and median bandwidth; *learned\_mmd* uses random-feature dimension 256, a median bandwidth heuristic, update interval around 200 rounds, and a minimum count of 5 observations before a user participates in cooperation.

**What is design.** To avoid using unknown noise scale as a prior, all GP-style methods use a graph- and time-aware ridge schedule

$$\lambda_t = \lambda_{\text{base}} \cdot S_{\text{spec}} \cdot \frac{T}{T+t}, \quad S_{\text{spec}} = \frac{\lambda_2(L)}{\lambda_{\max}(L)} \in [0, 1],$$

where  $\lambda_{\text{Fiedler}}(L)$  is the Fiedler value (smallest non-zero eigenvalue). We clip  $\lambda_t$  to  $[\lambda_{\min}, \lambda_{\max}]$  with  $\lambda_{\min} = 10^{-6}$  and  $\lambda_{\max} = 10^{-1}$ . To limit refactorizations, we update  $\lambda$  on a doubling epoch schedule (approximately at  $t \approx 200, 400, 800, \dots$ ) and only rebuild if the change exceeds 20%.

**What is tuned.** Only the exploration scales are tuned by grid search on a pilot horizon ( $T_{\text{pilot}} = 1500$  for medium/hard;  $T_{\text{pilot}} = 1000$  for simple) using  $R_{\text{pilot}} \in \{5, 10\}$ :

Algorithm	Grid (pilot)
LK-GP-UCB, GP-UCB, Coop-KernelUCB	$\beta \in \{0.5, 1, 2, 4\}$
LK-GP-TS	$\nu \in \{0.5, 1, 2, 4\}$
GOB.Lin, GraphUCB, LinUCB variants	$\alpha \in \{0.5, 1, 2, 4\}$

The best pilot setting (by mean pilot cumulative regret) is then *frozen* for the full-horizon evaluation. Noise/ridge  $\lambda_{\text{base}}$  in GP updates uses  $\lambda_{\text{base}} \in \{0.001, 0.005, 0.01, 0.05, 0.1\}$  on the pilot.

## E.6 ABLATIONS AND STRESS TESTS

We report an ablation under the *medium, Laplacian-Kernel with GP Draw* environment (ER graph, fixed  $\ell$  and  $\rho$ ) on **Scalability in users ( $n$ )**:  $n \in \{20, 50, 100, 200\}$  with fixed  $(m, m_t, d, T)$  and graph generator. We plot final cumulative regret vs.  $n$  and report the last step cumulative regret in Table 1.

Table 1: Ablation over number of users  $n$  (final cumulative regret; mean $\pm$ SE).

Algorithm	$n = 20$	$n = 50$	$n = 100$	$n = 200$
LK-GP-UCB	627.22 $\pm$ 32.98	892.43 $\pm$ 21.73	1062.69 $\pm$ 18.29	1157.74 $\pm$ 23.02
LK-GP-TS	634.46 $\pm$ 22.78	943.41 $\pm$ 19.56	1176.23 $\pm$ 15.77	1260.35 $\pm$ 16.23
Coop-KernelUCB	730.06 $\pm$ 31.02	1015.35 $\pm$ 22.18	1273.28 $\pm$ 17.36	1358.48 $\pm$ 14.22
GOB.Lin	1092.86 $\pm$ 71.70	1203.32 $\pm$ 18.57	1370.51 $\pm$ 16.78	1432.48 $\pm$ 18.72
GraphUCB	1105.20 $\pm$ 68.54	1192.30 $\pm$ 22.12	1360.02 $\pm$ 15.32	1453.21 $\pm$ 17.81
GP-UCB	2222.20 $\pm$ 90.26	1964.65 $\pm$ 61.40	1641.43 $\pm$ 37.43	1444.83 $\pm$ 36.33
Pooled-LinUCB	2360.95 $\pm$ 70.55	1909.81 $\pm$ 49.49	1723.27 $\pm$ 40.23	1438.74 $\pm$ 26.44
PerUser-LinUCB	1117.87 $\pm$ 72.04	1221.99 $\pm$ 22.03	1432.89 $\pm$ 18.81	1527.04 $\pm$ 17.61