Active Search and Bandits on Graphs Using Sigma-Optimality

Yifei Ma
Machine Learning Department
Carnegie Mellon University
yifeim@cs.cmu.edu

Tzu-Kuo Huang*
Microsoft Research
tkhuang@microsoft.com

Jeff Schneider
Robotics Institute
Carnegie Mellon University
schneide@cs.cmu.edu

Abstract

Many modern information access problems involve highly complex patterns that cannot be handled by traditional keyword based search. Active Search is an emerging paradigm that helps users quickly find relevant information by efficiently collecting and learning from user feedback. We consider active search on graphs, where the nodes represent the set of instances users want to search over and the edges encode pairwise similarity among the instances. Existing active search algorithms are either short of theoretical guarantees or inadequate for graph data. Motivated by recent advances in active learning on graphs, namely the Σ-optimality selection criterion, we propose new active search algorithms suitable for graphs with theoretical guarantees and demonstrate their effectiveness on several real-world datasets.

We relate our active search setting to multi-armed bandits whose rewards are binary values indicating search hits or misses and arms cannot be pulled more than once. We also discussed theoretical guarantees for applying Σ-optimality as the exploration term for bandits on graphs.[1]

1 INTRODUCTION

As the world gets increasingly digitized and electronically recorded, how to quickly identify relevant pieces of information becomes a major issue. Internet search engines are an integral part of modern life, serving as a probe into the diverse, complex and expanding space of human digital traces. Despite being successful in many information retrieval tasks, the keyword-based query mechanism in most search engines may fall short when targets are characterized by complex patterns or signatures beyond keywords. For example, financial transactions associated with illegal activities bear signatures involving multiple factors such as time, location, occupation of the account owner, etc. In the investigation of organizational misconduct, such as the Enron scandal, the important leads or evidences, oftentimes buried in a sea of diverse electronic and paper trails, usually involve information exchange among key individuals and their relationship. To fully understand the users' intent in these cases, keyword-based search may serve as a good starting point, but is certainly far from completing the task.

Such needs of more general search paradigms have recently motivated several efforts Garnett et al. (2012); Wang et al. (2013); Vanchinathan et al. (2013), most of which are related to the Active Search framework proposed by Garnett et al. (2012). It is an interactive search mechanism that begins with a full set of instances without supervision and a given task/keyword-specific similarity measure between these instances. Based on the similarity measure and an optional initial set of suggestions from the user, an algorithm figures out what instances the user should examine next and presents it to the user, who then decides whether the presented instance is relevant or not. Upon receiving this feedback, the algorithm updates its search strategy accordingly and selects the next instance to present. The loop continues until the user quits, and the goal is to maximize the total number of relevant instances found.

As one can see, Active Search has close connections to some well-studied machine learning paradigms. At a first glance, Active Learning (Settles, 2010) seems the most related because they both ask for user feedback incrementally and adaptively. However, Active Learning aims at improving generalization performances with as few label queries as possible, while Active Search is evaluated by how many relevant instances it found along the way, and therefore

* Part of this work was done while the author was with Carnegie Mellon University.

[1] An earlier version of this paper included results on bandit cumulative regrets with improved rates (originally Section 4.2.2). These results depended on proof strategies from Contal et al. (2014) (originally in Appendix C) which were found to be incorrect. Therefore, these results have been removed in the current version of the paper.

must carefully balance exploitation and exploration. This trade-off relates Active Search to stochastic optimization in the Multi-Armed Bandit setting (Robbins, 1985; Dani et al., 2008; Kleinberg et al., 2008; Bubeck et al., 2009), where the goal is to find the maximum of an unknown function using as few function evaluations as possible. However, Active Search deviates from this setting in that it selects instances *without replacement* and is competing with the best *subset* of instances rather than the single best.

We investigate Active Search when the instances are represented by the nodes on a graph whose edges encode pairwise similarity among the instances. For a toy example, please see Figure 1. Many real-world datasets are of this type, such as web pages, citation networks, and e-mail correspondences. For data that are not naturally represented as graphs, a graph representation based on pairwise similarity can still be beneficial because it may reveal useful manifold structures (Tenenbaum et al., 2000; Belkin and Niyogi, 2001). Existing active search approaches (Wang et al., 2013; Garnett et al., 2012; Vanchinathan et al., 2013) either lack theoretical guarantees or ignore certain graph properties, thereby degrading empirical performances. By drawing ideas from recent advances in active learning on graphs (Ma et al., 2013), we proposed new active search algorithms with theoretical guarantees, and empirically demonstrate their advantages over existing methods. In particular, our new exploration criteria, motivated by Σ-optimality criterion (Ma et al., 2013) for active learning on graphs, favor nodes with not only high uncertainty, but also high influence on the other nodes.

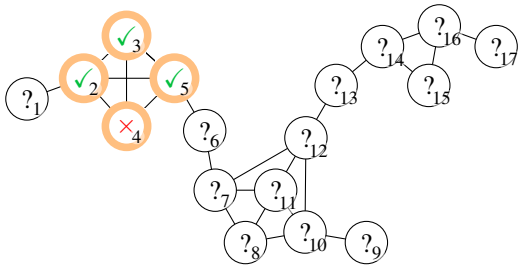


Figure 1: A toy examples for active search where the goals are "✓" nodes. Suppose the yellow nodes are observed in previous rounds, which node should be searched next?

The rest of the paper is organized as follows. We describe related work in Section 2, and introduce the problem setup in Section 3. We then present our new methods in Section 4 along with theoretical guarantees, followed by experimental results in Section 5.

2 RELATED WORK

Wang et al. (2013) proposed an active search algorithm for graphs, building on label propagation and semi-supervised learning using Gaussian random fields (Zhu et al., 2003a,b). Despite decent empirical performances, this approach does not have any theoretical guarantee. Vanchinathan et al. (2013) proposed a Gaussian-Process (GP) based algorithm, GP-SELECT, for sequentially selecting instances with high user scores or ratings (rewards). This algorithm extends the popular GP-UCB algorithm (Cox and John, 1997; Auer, 2003) for stochastic optimization and inherits nice theoretical guarantees (Srinivas et al., 2012). When applied to graphs, however, it tends to select nodes at the periphery of the graph because they have large predictive variances, leading to large exploration factors in the GP-UCB selection rule. Yet the rewards of these nodes reveal little information about the reward distribution over the whole graph.

Similar issues have been observed in active learning on graphs as well. In their experiments, Ma et al. (2013) found that selection rules based on mutual information gain (Krause et al., 2008), which is closely related to per-node predictive variances, usually end up selecting nodes at the periphery of a graph. Ji and Han (2012) proposed a selection criterion based on one-step lookahead decrease of the average variance of all remaining nodes, which effectively considers not only the predictive variance of the search node itself, but also its covariances with all remaining nodes. This criterion corresponds to standard V-optimality in experiment design. Ma et al. (2013) further improved the state of the art by using the Σ-optimality criterion, which demonstrates greater robustness against outliers and better empirical performances than V-optimality. Motivated by these recent advances, we propose new active search algorithms that combine GP-UCB with Σ-optimality.

Valko et al. (2014) considered bandit problems where arms correspond to nodes on a graph and the reward is a smooth function over the graph. Their algorithm can be viewed as a special case of GP-UCB with a kernel defined by the inverse of a graph Laplacian (augmented with an identity matrix). To analyze the performance of their UCB-style algorithm, they propose the notion of *effective dimension* of a graph, which can be viewed as a measure of the spectral decay of the kernel, thereby determining the performance of the algorithm (Srinivas et al., 2012). We also use the effective dimension to analyze our proposed methods.

3 PROBLEM SETUP

The database where active search is performed is given as a graph $\mathcal{G}$ with known structure (edge connections). The edge connections are nonnegative and we use $\mathbf{A}$ to represent the adjacency matrix of $\mathcal{G}$, such that $A_{ij} \geq 0, \forall i, j$. Let $\mathcal{V} = \{v_1, \ldots, v_n\}$ denote the set of all nodes in $\mathcal{G}$. From $\mathbf{A}$ we can derive a graph Laplacian matrix, $\boldsymbol{\mathcal{L}} = \mathbf{D} - \mathbf{A}$, where $\mathbf{D} = \mathrm{diag}(\mathbf{A} \cdot \mathbf{1}) = \mathrm{diag}(\deg(v_1), \ldots, \deg(v_n))$.

Every node v in our graph holds one reward value we denote as $f(v)$, indicating whether the node is the search tar-

get. The reward is unknown at first and can be revealed only when it is queried explicitly. For mathematical benefits, we relax the reward to be a real value and introduce a Gaussian noise to its observation, as

$$y(v) = f(v) + \epsilon, \text{ where } \epsilon \sim \mathcal{N}(0, \sigma_n^2). \tag{1}$$

Similar to bandit problems, querying a node also means collecting the true reward of that node. Our goal is to design a query strategy, which interactively generates a query sequence $\mathbf{v}_t = (v_1, \ldots, v_t)^\top$ without any repeated selections, in order to maximize the cumulative reward

$$F_T = \sum_{t=1}^{T} f(v_t). \tag{2}$$

The cumulative reward is always upper-bounded by the optimal strategy with full knowledge of the true rewards on all the nodes. Let $\mathbf{v}_t^* = (v_1^*, \ldots, v_t^*)$ to be the optimal query sequence (without repeated selections), our analysis in Theorem 2 (Section 4.2) bounds the cumulative regret between our strategy and the optimal strategy,

$$R_T = \sum_{t=1}^{T} f(v_t^*) - f(v_t). \tag{3}$$

The above characterizes an active search problem, provided that the values of $f(v)$ are binary and the sequences $\mathbf{v}_t$ and $\mathbf{v}_t^*$ do not allow repeated selections. Otherwise, the above can also model a multi-armed bandit problem if we relax $f(v)$ to be real and $\mathbf{v}_t$ and $\mathbf{v}_t^*$ to allow repeated selections. In fact, our formulation discusses them together, providing analysis to the slightly more rigorous active search modeling except that $f(v)$ is relaxed to real values.

In our notations, bold letters indicate vectors or matrices, while light letters without subscripts mean functions and light letters with subscripts represent scalars or specific elements. t, τ, and T are time indices, which when applied as subscripts, always mean the selection or model at that time step. Other letters as subscripts, such as i, j, n, always mean the natural indices.

3.1 GAUSSIAN RANDOM FIELD PRIOR

A key assumption in this work is that the reward values, or the target labels, are constrained by the graph structure in a non-trivial way. Otherwise, the input graph provides little information about the reward function, making active search extremely difficult. More specifically, we assume that the reward values of all the nodes in the graph, collectively denoted as a vector $\mathbf{f} \in \mathbb{R}^N$, are random variables distributed jointly as

$$\log p(\mathbf{f}) \simeq -\sum_{i=1}^{N}\sum_{j=1}^{N} \frac{A_{ij}(f_i - f_j)^2}{2} - \sum_{j=1}^{N} \frac{\omega_0 (f_j - \mu_0)^2}{2},$$
$$\text{i.e., } \mathbf{f} \sim \mathcal{N}\Big(\boldsymbol{\mu}_0 = \mu_0 \cdot \mathbf{1},\ \mathbf{C}_0 = (\boldsymbol{\mathcal{L}} + \omega_0 \mathbf{I})^{-1}\Big), \tag{4}$$

where μ_0 is a prior mean, and $\omega_0 > 0$ is a regularization parameter. According to this probabilistic model, it is more likely for connected nodes to share similar values than not. Define the initial covariance matrix as, $\mathbf{C}_0 = (\boldsymbol{\mathcal{L}} + \omega_0 I)^{-1}$, and denote $\widetilde{\boldsymbol{\mathcal{L}}}_0 = \boldsymbol{\mathcal{L}} + \omega_0 \mathbf{I}$. The above prior model is also known as **Gaussian random fields (GRFs)**.

3.2 POSTERIOR INFERENCE

Assume the nature draws one sample from the prior model, (4), and we use query observations, (1), to converge to that particular draw by performing posterior inference conditioned on the history,

$$\mathcal{H}_t = \{(v_\tau, y_\tau)\}_{\tau=1}^t = \{\mathbf{v}_t, \mathbf{y}_t\},$$

which allows us to update the posterior distribution as,

$$\log p(\mathbf{f} \mid \mathcal{H}_t) \simeq -\frac{1}{2}(\mathbf{f} - \boldsymbol{\mu}_0)^\top \widetilde{\boldsymbol{\mathcal{L}}}_0 (\mathbf{f} - \boldsymbol{\mu}_0) - \sum_{\tau=1}^{t} \frac{(y_\tau - f_{v_\tau})^2}{2\sigma_n^2}.$$

Notice that the prior distribution and likelihood model form Gaussian conjugate pairs. Denote the posterior distribution as, $\mathbf{f} \mid \mathcal{H}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{C}_t)$. To some readers, it is easier to express $\boldsymbol{\mu}_t$ and $\mathbf{C}_t$ using the prior *precision* matrix, as

$$\boldsymbol{\mu}_t = \mathbf{C}_t\Big(\widetilde{\boldsymbol{\mathcal{L}}}_0 \boldsymbol{\mu}_0 + \sum_{\tau=1}^{t} \frac{y_\tau \mathbf{e}_{v_\tau}}{\sigma_n^2}\Big), \quad \mathbf{C}_t^{-1} = \widetilde{\boldsymbol{\mathcal{L}}}_0 + \frac{1}{\sigma_n^2}\mathbf{H}_t \tag{5}$$

where $\mathbf{e}_{v_\tau} = (0, \ldots, 0, 1, 0, \ldots, 0)^\top$ is an indicator vector of index v_τ and $\mathbf{H}_t$ is a diagonal matrix of index counts from $\mathbf{v}_t$, whose kth diagonal element is $\sum_{\tau=1}^{t} e_{v_\tau}(v_k)$.

However, for convenience in later descriptions and to connect to **Gaussian Process (GP)** literature (Rasmussen and Williams, 2006), we also use the prior *covariance* matrix to express the posterior distribution, as,

$$\begin{aligned} \mu_t(v) &= \mu_0(v) + \mathbf{c}_{\mathbf{v}_t v}^\top (\mathbf{C}_{\mathbf{v}_t\mathbf{v}_t} + \sigma_n^2 \mathbf{I})^{-1}(\mathbf{y}_t - \boldsymbol{\mu}_{\mathbf{v}_t}), \\ C_t(v, v') &= C_0(v, v') - \mathbf{c}_{\mathbf{v}_t v}^\top (\mathbf{C}_{\mathbf{v}_t\mathbf{v}_t} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{c}_{\mathbf{v}_t v'}, \end{aligned} \tag{6}$$

where the matrices can all be defined in terms of the prior:

$$\begin{aligned} \mathbf{c}_{\mathbf{v}_t v} &= (C_0(v_1, v), \cdots, C_0(v_t, v))^\top \\ \mathbf{C}_{\mathbf{v}_t\mathbf{v}_t} &= \big(C_0(v_\tau, v_{\tau'})\big)_{\tau,\tau'=1}^t \\ \boldsymbol{\mu}_{\mathbf{v}_t} &= (\mu_0(v_1), \cdots, \mu_0(v_t))^\top. \end{aligned}$$

The above update rules also applies to any time interval that starts with t_0, by replacing prior models (variables with subscript "0") with the model at time t_0.

Define simple notations for correlation coefficients and standard deviations from the covariance matrix, $C_t(v,v') = \rho_t(v,v')\sigma_t(v)\sigma_t(v')$, which implies that $\sigma_t^2(v) = C_t(v,v)$. Define $\mathbf{c}_t(v)$ to be the column of $\mathbf{C}_t$ corresponding to node v.

4 METHOD

Algorithm 1 GP-SOPT and its variants

input μ_0, $\mathbf{A}$, ω_0, σ_n, α_t, T; if warm start, $\{v_\tau, y(v_\tau)\}_{\tau=1}^{t_0}$

1: Obtain initial $\mathcal{N}(\boldsymbol{\mu}_0, \mathbf{C}_0)$ // (4)
2: **for** $t = t_0, \ldots, T-1$, **do**
3: Update to posterior $\mathcal{N}(\boldsymbol{\mu}_t, \mathbf{C}_t)$ // (6)
4: $v_{t+1} \leftarrow \arg\max_{v \in V \setminus S_t} \mu_t(v) + \alpha_{t+1} s_t(v)$ // (9.a, 9.b, or 9.c)
5: Observe $y(v_{t+1})$; include $S_{t+1} \leftarrow S_t \cup \{v_{t+1}\}$
6: **end for**

output S_T.

Our proposed active search algorithms are described in Algorithm 1. They resemble general exploration-exploitation style algorithms with GPs. Here we focus on binary functions that assign value 1 to relevant or target nodes, and 0 to all other nodes. At iteration $t+1$, Algorithm 1 selects the next node to query based on a deterministic selection rule of the form:

$$\arg\max_{v \in V \setminus S_t} \ \mu_t(v) + \alpha_{t+1} \cdot s_t(v), \tag{7}$$

where $\mu_t(v)$ is the usual exploitation term and $s_t(v)$ encourages exploration, with the two being balanced by a possibly iteration-dependent parameter $a_{t+1} > 0$.

Examples from existing literature like the popular GP-UCB algorithm and its extension to Active Search, GP-SELECT (Vanchinathan et al., 2013), amount to setting $s_t(v)^2 = \sigma_t(v)^2$, the posterior (as well as predictive) variance of the reward value at node v. Although this is a very reasonable choice in many situations, it may lead to undesirable exploration behaviors on graphs. Under our model assumption, low-degree nodes, which usually lie at the periphery of a graph, tend to have high predictive variances. Direct applications of GP-UCB may result in the selection of many such outliers, which fail to reveal much information about the reward values of most other nodes at the core of the graph (Figure 2(a)).

Intuitively, a good exploration criterion should favor nodes that have high influences on other parts of the graph. That is, the knowledge of the function values at these nodes should reveal a lot about the function values at other nodes. Under our model assumption, this principle naturally connects with the predictive covariances of a node with others. Research in active learning on graphs has already made use of predictive covariances to construct better selection rules. Ji and Han (2012) proposed to select nodes based on their sums of squares of predictive covariances with other nodes, which is derived from the minimization of squared prediction error, known as V-optimality in experiment design. Ma et al. (2013) observed that V-optimality can still be undesirably sensitive to outliers and used **Σ-optimality criterion** instead, which by itself selects a set of nodes $\mathbf{v}_t$ to minimize the following Bayes *survey risk* on the posterior model after the selection,

$$\mathcal{R}^{\Sigma}_{t|\mathbf{v}_t} = \mathbb{E}\left(\sum_{v' \in \mathcal{V}} f_{t|\mathbf{v}_t}(v') - \sum_{v' \in \mathcal{V}} \mu_{t|\mathbf{v}_t}(v') \right)^2 = \mathbf{1}^\top \mathbf{C}_{t|\mathbf{v}_t} \mathbf{1}.$$

For active search, we use this criterion in a greedy sequential selection manner for exploration scoring, as

$$\begin{aligned} s_t(v) &= \sqrt{\mathcal{R}^{\Sigma}_{t|S_t} - \mathcal{R}^{\Sigma}_{t+1|S_t \cup \{v\}}} = \frac{\sum_{v'} C_t(v,v')}{\sqrt{C_t(v,v) + \sigma_n^2}} \\ &= \frac{1}{\sqrt{1 + \sigma_n^2 / \sigma_t^2(v)}} \cdot \sum_{v' \in V} \rho_t(v,v') \sigma_t(v'), \end{aligned} \tag{8}$$

where the second equality is easily derived from (6). If we ignore σ_n (set it to 0), the Σ-optimality criterion (8) considers the sum of a node's correlation times standard deviation of all nodes on the graph. High score nodes by this criterion are likely to provide rich information for exploration.

We propose three exploitation-exploration style algorithms with exploration criteria motivated by Σ-optimality, which are *vanilla Σ-optimality* and its two variants with an additional parameter k that we will describe next. All algorithms select the next node to query by the general rule (7), but use different exploration terms:

GP-SOPT (Vanilla Σ-Optimality):

$$s_t(v) = \frac{1}{\sqrt{1 + \sigma_n^2 / \sigma_t^2(v)}} \cdot \sum_{v' \in V} \rho_t(v,v') \sigma_t(v'). \tag{9.a}$$

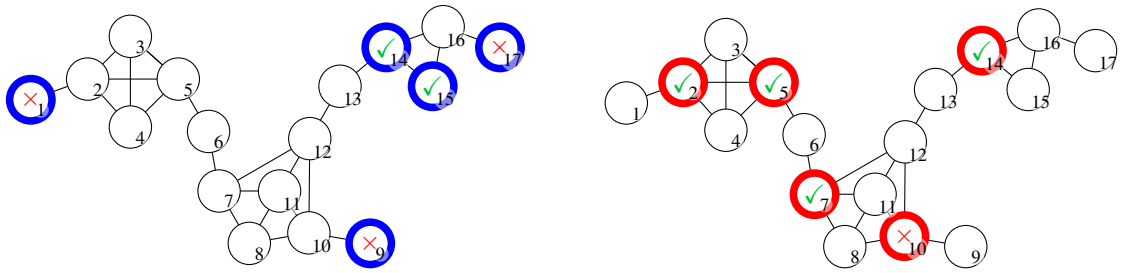
GP-SOPT.TT (Thresholded Total Covariance):

$$s_t(v) = \min\left(k\sigma_t(v), \sum_{v' \in V} \rho_t(v,v') \sigma_t(v') \right). \tag{9.b}$$

GP-SOPT.TOPK (Top-k Covariance):

$$s_t(v) := \max_{B \subset V, |B| = k} \sum_{v' \in B} \rho_t(v,v') \sigma_t(v'). \tag{9.c}$$

As one can see in Figure 2(b), the nodes selected by vanilla GP-SOPT indeed reside in more central parts of the toy graph than the nodes selected by its competitor. In a large graph with many peripheral nodes, we believe that the improved exploration criteria of GP-SOPT and its variants contribute to a better recall rate of search targets in real graphs in Section 5.



(a) Choices from UCB (b) Choices by our algorithm

Figure 2: For the toy graph example, choices from (a) direct application of UCB (Vanchinathan et al., 2013; Valko et al., 2014) versus (b) our vanilla GP-SOPT. We observe that our method (b) tends to select more from cluster centers, which helps reduce variance of the unobserved values/rewards, whereas previous literature (a) tends to select the graph periphery.

The reason we propose the latter two variants, (9.b) and (9.c), is to both address proof difficulties and increase practical robustness. By Lemma 3 in Appendix A, we have that $s_t(v) \geq \sigma_t(v)$ for both criteria, meaning that $s_t(v)$ maintains the UCB property. Note that the observation noise, σ_n, is also dropped from (9.b) and (9.c). As we will show in our theoretical analysis, we put a threshold in (9.b) against $k\sigma_t(v)$, where k is a tuning parameter, in order to explicitly control the regret of the algorithm. As implied by Lemma 4 in Appendix A, the Top-k Covariance criterion (9.c) is also always upper-bounded by $k\sigma_t(v)$.

In the next two subsections we discuss in more details the properties of various exploration criteria, and present our theoretical analysis.

4.1 DISCUSSIONS

Our approach and two other popular criteria, information gain from Srinivas et al. (2012) and V-optimality of Ji and Han (2012), can also be connected by functions of the eigenvalues of the covariance matrix at each iteration.

To see this connection, assume the updated covariance matrix at iteration $(t+1)$ has eigen-decomposition $\mathbf{C}_t = \sum_{j=1}^{n} \lambda_{t,(j)} \mathbf{q}_{t,(j)} \mathbf{q}_{t,(j)}^\top$, where $\boldsymbol{\lambda}_t = (\lambda_{t,(1)}, \ldots, \lambda_{t,(n)})^\top$ represents the eigenvalues and $\{\mathbf{q}_{t,(j)} : j = 1, \ldots, n\}$ is the set of corresponding eigenvectors. Assume the eigenvalues are sorted by $\lambda_{t,(1)} \geq \ldots \geq \lambda_{t,(n)} \geq 0$. We hope to connect $s_t(v)$ to the following spectral difference,

$$\Delta h_t(v) = h\big(\boldsymbol{\lambda}_t\big) - h\big(\boldsymbol{\lambda}_{t+1|v}\big) \tag{10}$$

where $h(\boldsymbol{\lambda}) : \mathbb{R}^n \to \mathbb{R}$ is a multivariate function defined on the eigenvalues. Further, by the one-step update rule of (6), $\mathbf{C}_t$ has Loewner order as $\mathbf{C}_0 \succ \mathbf{C}_1 \succ \ldots \succ \mathbf{C}_T \succ \mathbf{0}$. It is thus often desirable to require $h(\cdot)$ to be monotone with respect to this ordering, i.e. $\mathbf{C}_t \succ \mathbf{C}_{t'} \Rightarrow h(\boldsymbol{\lambda}_t) \geq h(\boldsymbol{\lambda}_{t'})$.

Case 1. $h(\boldsymbol{\lambda}) = \sum_j \log(\lambda_{(j)})$. Then, $\Delta h_t(v) = 2\mathcal{I}_t(\mathbf{f}; y(v)) = \log(1 + \frac{\sigma_t^2(v)}{\sigma_n^2})$, twice the information gain from $\mathbf{f} \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{C}_t)$ to $\mathcal{N}(\boldsymbol{\mu}_{t+1|v}, \mathbf{C}_{t+1|v})$. This metric is important to **GP-UCB** (Srinivas et al., 2012), which set $s_t(v) = \sigma_t(v)$ and used the inequality, $\log(1+\frac{\sigma_m^2}{\sigma_n^2})\frac{\sigma_t^2(v)}{\sigma_m^2} \leq \log(1 + \frac{\sigma_t^2(v)}{\sigma_n^2})$, where $\sigma_m = \max_{v,t} \sigma_t(v)$, in its proofs.

Case 2. $h(\boldsymbol{\lambda}) = \sum_j \lambda_{(j)}$ gives $\Delta h_t(v) = \mathrm{tr}(\mathbf{C}_t) - \mathrm{tr}(\mathbf{C}_{t+1|v}) = \|\mathbf{c}_t(v)\|_2^2 / (\sigma_t^2(v) + \sigma_n^2)$. For $\sigma_n = 0$, $\Delta h_t(v)$ is used as the greedy **V-optimal** criterion for design of experiments by Ji and Han (2012).

Case 3. $h(\boldsymbol{\lambda}) = \lambda_{(1)}$ connects to the greedy design for **E-optimality** (Pukelsheim, 1993). To some extent, it is also related to greedy **Σ-Optimality**. First, approximate $\Delta h_t(v)$ by $\partial \lambda_{(j)} = \mathbf{q}_{(j)}^\top \partial(\mathbf{C}) \mathbf{q}_{(j)}$ around $\mathbf{C} = \mathbf{C}_t$, as

$$\Delta h_t(v) \approx \mathbf{q}_{t,(1)}^\top (\mathbf{C}_t - \mathbf{C}_{t+1|v}) \mathbf{q}_{t,(1)} = \left(\frac{|\mathbf{c}_t(v)^\top \mathbf{q}_{t,(1)}|}{\sqrt{\sigma_t(v)^2 + \sigma_n^2}} \right)^2$$

The above resembles (8) if $\mathbf{q}_{t,(1)} \propto \mathbf{1}$, which holds true for $t = 0$ and $\omega_0 = 0$ and approximately so for small ts.

In all these cases, exploration is measured by how much the objective, $h(\boldsymbol{\lambda}_T)$, is eventually decreased after T iterations. Each definition of $h(\boldsymbol{\lambda}_t)$ aggregates the eigenvalues of the posterior covariance matrices in a different way, which affects the relative importance of large and small eigenvalues. In **Case 1**, since $\frac{\partial \log(\lambda)}{\partial \lambda} = \frac{1}{\lambda}$, the same change introduced to a smaller λ will have a relatively larger impact on the objective. Such an effect is not evident in the other two cases. Particularly in **Case 3**, changes to small eigenvalues are ignored unless they become the largest eigenvalue.

Establishing biases to penalize larger eigenvalues more has the benefit of improving global robustness because the posterior marginal variance of every node is upper-bounded by $\lambda_{t,(1)}$. Compared with **Cases 2** and **3**, **Case 1** is more sensitive to changes in small eigenvalues, which may be another explanation of **GP-UCB**'s strong tendency to select peripheral nodes, as seen in Figure 4 of Krause et al. (2008) or Figure 1(d) of Gotovos et al. (2013).

Although Algorithm 1 is not built around the concept of functions on eigenvalues, it still establishes strong biases to penalize large eigenvalues in its initial explorations, per

analysis in **Case 3**. Note that Σ**-optimality** achieves a more complex goal than **E-optimality**; exact execution of **E-optimality** may over-simplify the model and select nodes between clusters for separation rather than inside them.

4.2 REGRET ANALYSIS

We present an UCB-style analysis for GP-SOPT.TT and GP-SOPT.TOPK. We combine several results on GP optimization (Srinivas et al., 2012; Vanchinathan et al., 2013) and the spectral bandit analysis (Valko et al., 2014). As in these results, our regret bounds depend on the mutual information between f and the observed values $\mathbf{y}_S$ at a set S of nodes:

$$\mathcal{I}(\mathbf{y}_S; f) := H(\mathbf{y}_S) - H(\mathbf{y}_S \mid f), \tag{11}$$

where $H(\cdot)$ denotes the entropy. If f is drawn from a GP with observation noise distributed independently as $\mathcal{N}(0, \sigma_n)$, the mutual information has the following analytical form:

$$\mathcal{I}(\mathbf{y}_S; f) \;=\; \mathcal{I}(\mathbf{y}_S; f_S) \;=\; \frac{1}{2}\log|I + \sigma_n^{-2}\mathbf{C}_{\mathbf{v}_S\mathbf{v}_S}|. \tag{12}$$

Let

$$\gamma_T := \max_{S\in V, |S|=T} \frac{1}{2}\log|I + \sigma_n^{-2}\mathbf{C}_{\mathbf{v}_S\mathbf{v}_S}|, \tag{13}$$

i.e., the maximum information about f gained by observing T function evaluations. The regrets of our algorithms depend on the growth rate of γ_T, which can be linear in T for arbitrary graphs. However, real-world graphs often possess rich structures, such as clusters or communities, and practical measures of relevance are often highly correlated with these structures, resulting in slowly-growing γ_T. To formalize this intuition, we follow Valko et al. (2014) to consider the *effective dimension*:

$$d_T^* := \max\left\{ i \mid \lambda_i \le \frac{\sigma_n^{-2}T}{(i-1)\log(1+\frac{T}{\sigma_n^2\omega_0})} \right\}, \tag{14}$$

where λ_i is the ith smallest eigenvalue of $\widetilde{\boldsymbol{\mathcal{L}}}_0$ and $\lambda_1 = \omega_0$. The effective dimension is small when the first few λ_i's are small and the rest increase rapidly, as is often the case for graphs with community or cluster structures. On the contrary, if all the eigenvalues are close to ω_0, then d_T^* may be linear in T. The following lemma bounds γ_T in terms of d_T^*:

Lemma 1. *Let T be the total number of rounds. Then*

$$\gamma_T \le 2d_T^* \log\left(1 + \frac{T}{\sigma_n^2\omega_0}\right).$$

Proof. By Lemma 7.6 of Srinivas et al. (2012) and the fact that λ_i^{-1} is the ith largest eigenvalue of the kernel $\mathbf{C}_0 = \widetilde{\boldsymbol{\mathcal{L}}}_0^{-1}$, we have

$$\gamma_T \le \max_{\substack{\{m_i\}_{i=1}^T, m_i\ge 0,\\ \sum_{i=1}^T m_i = T}} \sum_{i=1}^T \log\left(1 + \frac{m_i}{\sigma_n^2\lambda_i}\right). \tag{15}$$

Then by applying the same argument that proves Lemma 6 of Valko et al. (2014), we obtain the desired result. □

We will then derive regret bounds in terms of γ_T.

Recall the cumulative regret of an active search algorithm is defined as $R_T := \sum_{t=1}^T f(v_t^*) - f(v_t)$, where $\{v_t\}_{t=1}^T$ is the sequence of unique nodes selected by the algorithm. For the two proposed UCB-style algorithms, GP-SOPT.TT (9.b) and GP-SOPT.TOPK (9.c), we give the following bounds on their cumulative regrets.

Theorem 2. *Pick $\delta \in (0,1)$. Assume the vector of true node values, $\mathbf{f}$, has bounded quadratic norm, $\|\mathbf{f}\|_{\widetilde{\boldsymbol{\mathcal{L}}}_0} = \sqrt{\mathbf{f}^\top \widetilde{\boldsymbol{\mathcal{L}}}_0 \mathbf{f}} \le B$,[2] and the observation noise ϵ_t is zero-mean conditioned on the past and is bounded by σ_n almost surely. If* GP-SOPT.TT *and* GP-SOPT.TOPK *use* GRF *prior* (4) *with zero-mean and graph Laplacian $\widetilde{\boldsymbol{\mathcal{L}}}_0$, the observation noise model $\mathcal{N}(0, \sigma_n^2)$, and $\alpha_t := \sqrt{2B + 300\gamma_t \log^3(t/\delta)}$, then their cumulative regrets will satisfy*

$$Pr(\{R_T \le k\sqrt{c_1 T \alpha_T \gamma_T} \;\; \forall T \ge 1\}) \ge 1 - \delta,$$

where the randomness is over the observation noise and $c_1 := \frac{8/\omega_0}{\log(1+\sigma_n^{-2})}$. This implies that with high probability,

$$R_T = O(k\sqrt{T}(B\sqrt{d_T^*} + d_T^*)).$$

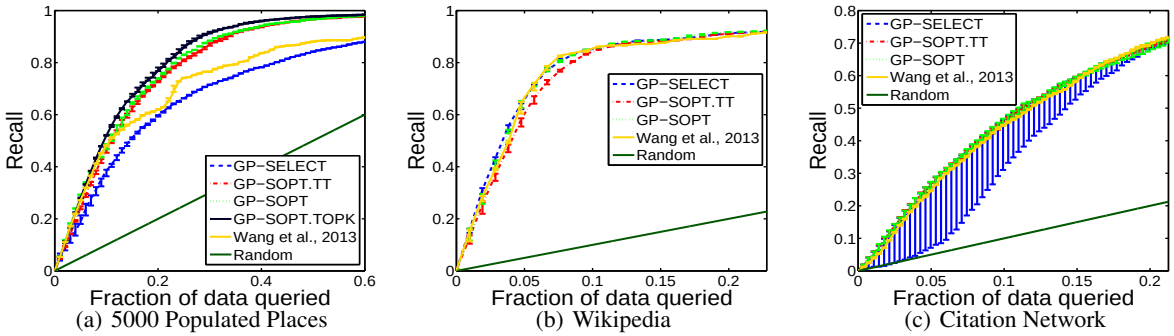
This result is easily derived from the regret analysis of the GP-SELECT algorithm proposed by Vanchinathan et al. (2013) because the exploration terms used by GP-SOPT.TT and GP-SOPT.TOPK both satisfy $\sigma_t(v) \le s_t(v) \le k\sigma_t(v)$, thereby maintaining the UCB property. Although our regret bound is k times worse than the GP-SELECT bound, the actual regret tends to behave more favorably as we observe in our experiments that after a few tens of rounds, $s_t(v)$ becomes smaller than $k\sigma_t(v)$ for almost all unqueried nodes, and the two proposed algorithm usually outperforms GP-SELECT. We give the proof in Appendix B for completeness.[3]

5 EXPERIMENTS

We conduct experiments on three graph data sets that were studied by Wang et al. (2013) and a version of the Enron e-mail data by Priebe et al..

[2] This is similar to a bounded RKHS norm with kernel C_0 in Srinivas et al. (2012).

[3] An earlier version of this paper follows on to discuss bounds on vanilla GP-SOPT. These proofs used strategies from Contal et al. (2014) which were found to be incorrect. Therefore, they have been removed in the current version of the paper.



(a) 5000 Populated Places (b) Wikipedia (c) Citation Network

Figure 3: Recall vs. fraction of data queried

5.1 Three Graph Datasets of Wang et al. (2013)

We briefly summarize the datasets below.

5000 Populated Places. The nodes of this graph are 5000 concepts in the DBpedia[4] ontology marked as populated places. Each place is supported by a Wikipedia page, and an undirected edge is created between two places if either one of their two Wikipedia pages links to the other. There can be multiple edges between two places. The DBpedia ontology divides populated places into five categories: administrative regions, countries, cities, towns and villages. The 725 administrative regions are selected as our target class while all the others are considered to be in null class.

Citation Network. This dataset consists of 14,117 papers in top Computer Science venues available on citeseer. The graph is created by adding an undirected edge between two papers if either one cites the other. The 1844 NIPS papers are chosen as our target class.

Wikipedia Pages on Programming Languages. A total of 5,271 Wikipedia pages related to programming languages are the nodes of this graph, and an undirected edge exists between two pages if they are linked together. Wang et al. (2013) performed topic modeling and chose the 202 pages related to objective oriented programming as our target class.

As demonstrated by Wang et al. (2013), the three graphs and their target label distributions exhibit qualitative differences and thus serve as good benchmarks. The citation network has many small components and target nodes appear in many of them, while the Wikipedia graph has large hubs and most target nodes reside in one of them. The graph of populated places lies in between these two extremes, with components of various sizes containing target nodes.

On all of the three data sets we compare two of the proposed methods: GP-SOPT.TT and GP-SOPT against GP-SELECT (GP-UCB without replacement) and the active search algorithm (AS-on-Graph) by Wang et al. (2013). We only evaluate GP-SOPT.TOPK on the 5000 populated places data due to its heavy computation. For each dataset we perform 5 independent runs, each with a randomly chosen target node as the warm start seed. For the proposed methods and GP-SELECT, the main tuning parameters are the exploration-exploitation trade-off parameter α_t and the observation noise variance σ^2. For GP-SOPT.TT and GP-SOPT.TOPK there is additionally the thresholding parameter k. We consider the following values for them. Populated Places: $\alpha_t \in \{4, 2, 1, 0.1, 0.01, 0.001\}$, $\sigma^2 \in \{1, 0.5, 0.25, 0.1\}$ and $k \in \{200, 400, 800\}$. Wikipedia: $\alpha_t \in \{0.1, 0.01, 0.001\}$, $\sigma^2 \in \{1, 0.5, 0.25, 0.1\}$ and $k \in \{200, 400, 800\}$. Citation Network: $\alpha_t \in \{1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}, \sigma^2 \in \{1, 0.5, 0.25, 0.1\}$ and $k \in \{400, 800, 1600\}$. Although in theory α_t should be iteration-dependent, we find that a fixed value often performs well in practice. On all data sets we set the kernel regularization parameter $\omega_0 = 0.01$. Wang et al. (2013) algorithm has several parameters, and we only tune the exploration-exploitation trade-off parameter α. It is set to 0.1 on Populated Places and Citation Network, and 0.0001 on Wikipedia, which are the best performing values. Other parameters are set based on Wang et al. (2013).

Results are in Figure 3, where we plot the recall, i.e., the fraction of targets found by the algorithms, versus the fraction of the whole data set queried. More specifically, for each algorithm we obtain its mean recall curve over the top 15% (except for Wang et al. (2013)) parameter combinations in each experiment, as judged by the area under the recall curve. We then plot the median, maximum and minimum over the five runs in Figure 3.

The three proposed methods clearly outperform Wang et al. (2013) and GP-SELECT on Populated Places, while all methods perform equally well on Wikipedia. We think this has to do with the underlying graph structure and target distribution. As mentioned before, target nodes in the Populated Places graph are spread over sub-graphs of various

[4] www.dbpedia.org

sizes, and therefore exploration strategies do make a difference. We observe that the proposed methods tend to select high-degree nodes in the first few iterations, thereby gaining much information, while GP-SELECT initially selects low-degree nodes. In contrast, most target nodes in the Wikipedia graph reside in one large component, and therefore less exploration is needed. In fact, the best values for α_t are very small, suggesting that an exploitation-only strategy is good enough for this data. On Citation Network, most methods perform well except that GP-SELECT performs quite poorly in one run. This may again indicate GP-SELECT is less robust against low-degree nodes.

5.2 Enron E-mails

We experimented on the Enron e-mail data set[5] with topics assigned by Priebe et al. based on the annotations by Berry and Browne. We further processed the dataset into a format suitable for active search experiments as detailed below. Each e-mail i is represented by a unique Unix time stamp t_i, a unique sender index and the set of receiver (excluding self-copying) indices, which are collectively denoted as U_i. Between e-mails i and j, we created an edge with the following weight:

$$A_{ij} := \exp\left(-(t_i - t_j)^2/\tau^2\right) \cdot |U_i \cap U_j|/\sqrt{|U_i||U_j|},$$

where $\tau = 12$ weeks in seconds and $|U_i|$ denotes the size of U_i. We thus measure pairwise similarity among e-mails by the product of nearness in time and degree of overlap between users involved. The resulting e-mail graph has 20,112 nodes, and we chose the subset of 803 e-mails that are assigned topic 16 in LDC topics[5], which is related to the downfall of Enron, to be the target class in this experiment.

Due to the size of the dataset, we only compared three methods: GP-SOPT.TT, GP-SELECT and Wang et al. (2013) in three independent runs each initialized with a target node chosen uniformly at random. We also limited the tuning parameters to be the following fixed values across the three runs: $(k, \alpha, \sigma^2, \omega_0) = (800, 0.001, 0.05, 0.01)$ for GP-SOPT.TT, $(\alpha, \sigma^2, \omega_0) = (0.01, 0.05, 0.01)$ for GP-SELECT, and $\alpha = 0.001$ for Wang et al. (2013). These values were chosen based on a coarse parameter search to be indicative of the performance of each method on this data set. Results are in Figure 4, which shows GP-SOPT.TT is more stable across initial seeds than the other methods, and outperforms Wang et al. (2013) significantly at early iterations.

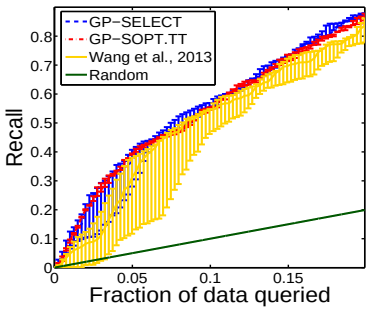


Figure 4: Enron: recall vs. fraction of data queried

6 CONCLUSION AND DISCUSSIONS

In this paper, we discuss active search on a graph with known structure. Each node bears a reward, which is unknown at first but can be noisily observed upon query. An active search algorithm aims to accumulate as large a sum of rewards from the queried nodes as possible under limited budgets. We assume that the node rewards vary smoothly along the graph.

Popular Bayesian UCB-style algorithms (Srinivas et al., 2012; Vanchinathan et al., 2013; Valko et al., 2014) use the marginal standard deviation as their exploration criterion, leading to the undesirable tendency of selecting peripheral nodes on a graph. Instead, we consider Σ-optimality on graphs, which can more efficiently reduce the variance of the reward function estimate by sampling cluster centers. We show the advantage of our method in experiments with real graphs and provide a theoretical guarantee on the cumulative regret.

One interesting future direction is deriving tighter regret bounds for the proposed methods that match their empirical performances. We imagine it may be possible to bound the regret directly by the difference in Σ-optimality (Bayes survey risks, $\mathcal{R}^{\Sigma}$), which may have better properties than differential information gain, γ_T on graphs.

An equally interesting question is the selection of graph kernels. Our discussions and experiments mainly consider Gaussian random fields with unnormalized Laplacian, which is a very popular kernel choice. It is worthwhile to explore active search with other graph kernels, such as the ones discussed in Smola and Kondor (2003).

Acknowledgement

This work was funded in part by DARPA grant FA87501220324.

A Predictive Covariance Matrix

Lemma 3. *For augmented graph Laplacian, the posterior covariance matrix, $C_t(v, v') \geq 0, \forall v, v'$.*

Proof. Let $h_k = \sum_{\tau=1}^{t} e_{v_\tau}(v_k)$ to be the count of queries on node k; further define its diagonal matrix, $\mathbf{H} =$

[5]Available at `http://cis.jhu.edu/~parky/Enron/execs.email.linesnum.ldctopic`

$\mathrm{diag}(h_1, \ldots, h_n)$. We rewrite (5) as,

$$(\mathbf{C}_t)^{-1} = (\mathbf{C}_0)^{-1} + \sigma_n^{-2}\mathbf{H} = \mathbf{D} - \mathbf{A} + \omega_0\mathbf{I} + \sigma_n^{-2}\mathbf{H}$$

Define $\mathbf{D}_t = \mathbf{D} + \omega_0\mathbf{I} + \sigma_n^{-2}\mathbf{H}$, we have

$$\mathbf{C}_t = (\mathbf{D}_t - \mathbf{A})^{-1} = \mathbf{D}_t^{-\frac{1}{2}} \left(\sum_{k=0}^{\infty} \left(\mathbf{D}_t^{-\frac{1}{2}} \mathbf{A} \mathbf{D}_t^{-\frac{1}{2}} \right)^k \right) \mathbf{D}_t^{-\frac{1}{2}},$$

where the right hand side is always nonnegative.

The convergence of $\|\mathbf{D}_t^{-\frac{1}{2}} \mathbf{A} \mathbf{D}_t^{-\frac{1}{2}}\|_2 < 1$ is as follows.

Define the components for the posterior as $\mathbf{D}_t = \mathrm{diag}(d_1^{(t)}, \ldots, d_n^{(t)}$ with $d^{(t)} = \sum_{i=1}^n d_i^{(t)}$. Also, define for the prior model $\mathbf{D} = \mathrm{diag}(d_1^{(0)}, \ldots, d_n^{(0)}$ with $d^{(0)} = \sum_{i=1}^n d_i^{(0)}$.

The following holds for any $\mathbf{v} \in \mathbb{R}^n$,

$$\mathbf{v}^\top \mathbf{D}_t^{-\frac{1}{2}} \mathbf{A} \mathbf{D}_t^{-\frac{1}{2}} \mathbf{v} = \sum_{ij} \frac{v_i v_j a_{ij}}{\sqrt{d_i^{(t)}}\sqrt{d_j^{(t)}}}$$

$$\leq \sqrt{\left(\sum_{ij} \frac{v_i^2 a_{ij}}{d_i^{(t)}} \right) \left(\sum_{ij} \frac{v_j^2 a_{ij}}{d_j^{(t)}} \right)} = \sum_i v_i^2 \frac{d_i}{d_i^{(t)}} \leq \|\mathbf{v}\|_2^2.$$

Further, both equalities cannot hold simultaneously, because for the first equality to hold, it is required that $\frac{v_i^2 a_{ij}}{d_i^{(t)}} \propto \frac{v_j^2 a_{ij}}{d_j^{(t)}}$, i.e., $v_j^2 \propto d_j^{(t)}, \forall j$ in the same connected component, which then dictates that,

$$\sum_i v_i^2 \frac{d_i}{d_i^{(t)}} = \sum_i \left(\frac{d_i^{(t)}}{d^{(t)}} \|\mathbf{v}\|_2^2 \right) \frac{d_i}{d_i^{(t)}} = \frac{d^{(0)}}{d^{(t)}} \|\mathbf{v}\|_2^2 < \|\mathbf{v}\|_2^2.$$

□

Lemma 4. *The diagonal elements in $\mathbf{C}_t$ is always no smaller than the off-diagonal elements, i.e., $\sigma_t(v)^2 = C_t(v, v) \geq C_t(v, v'), \forall v, v'$.*

Proof. Without loss of generality, let v be the last index of $\mathbf{C}_t = (\widetilde{\boldsymbol{\mathcal{L}}}_0 + \sigma_n^{-2}\mathbf{H})^{-1}$. For simplicity, let $\widetilde{\boldsymbol{\mathcal{L}}}_t = \widetilde{\boldsymbol{\mathcal{L}}}_0 + \sigma_n^{-2}\mathbf{H}$ and it has the following matrix partition,

$$\widetilde{\boldsymbol{\mathcal{L}}}_t = \begin{pmatrix} \widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}} & \tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v} \\ \tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v}^\top & \tilde{\ell}_{vv} \end{pmatrix},$$

where $\bar{\mathbf{v}}$ is the complement of v. From Woodbury matrix inversion lemma, we have

$$\mathbf{C}_t = \widetilde{\boldsymbol{\mathcal{L}}}_t^{-1} = \begin{pmatrix} \mathbf{M} & -\frac{1}{m}\widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1}\tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v} \\ -\frac{1}{m}\tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v}^\top \widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1} & \frac{1}{m} \end{pmatrix}, \quad (16)$$

where $m = \tilde{\ell}_{vv} - \tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v}^\top \widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1} \tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v}$ and $\mathbf{M} = \widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1} + \frac{1}{m}\widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1}\tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v}\tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v}^\top\widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1}$. To show that $C_t(v, v) \geq C_t(v, v')$, we need to verify that $(-\widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1}\tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v})_{v'} \leq 1$.

In fact, since $\widetilde{\boldsymbol{\mathcal{L}}}_t$ is diagonally dominant, we have $\widetilde{\boldsymbol{\mathcal{L}}}_t \mathbf{1}_n \geq 0$. Take its first $n-1$ rows to get $\widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}} \cdot \mathbf{1}_{n-1} + \tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v} \geq 0$. Notice $\widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}$ is also a valid augmented graph Laplacian. By Lemma 3, we could left multiply the element-wise nonnegative matrix $\widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1}$ to both sides to obtain, $\mathbf{1}_{n-1} + \widetilde{\boldsymbol{\mathcal{L}}}_{\bar{\mathbf{v}}\bar{\mathbf{v}}}^{-1}\tilde{\boldsymbol{\ell}}_{\bar{\mathbf{v}}v} \geq 0$, which completes our proof for any $v' \in \bar{\mathbf{v}}$. □

B Active Search Regret Bound

We start by stating the following result.

Theorem 5 (Theorem 6, Srinivas et al. (2012))**.** *Let $\delta \in (0, 1)$. Assume the observation noises are uniformly bounded by σ_n and f has RKHS norm B with kernel C_0, which is equivalent to $\mathbf{f}^\top \widetilde{\boldsymbol{\mathcal{L}}}_0 \mathbf{f} \leq B^2$. Define $\alpha_t = \sqrt{2B^2 + 300\gamma_t \log(t/\delta)^3}$, then*

$$Pr\left(\forall t, \forall v \in V, \ \ |\mu_t(v) - f(v)| \leq \alpha_{t+1}\sigma_t(v)\right) \geq 1 - \delta.$$

We use this result to bound our instantaneous regrets.

Lemma 6. *Conditioned on the high-probability event in Theorem 5, the following bound holds:*

$$\forall t, \ \ r_t := f(v_t^*) - f(v_t) \leq 2\alpha_t k \sigma_{t-1}(v_t),$$

where v_t^* is the node with the t-th globally largest function value and v_t is node selected at round t.

Proof. At round t there are two possible situations. If v_t^* was picked at some earlier round, the definition of v_t^* implies that there exists some $t' < t$ such that $v_{t'}^*$ has not been picked yet. According to our selection rule, the fact that $s_t(v) \geq \sigma_t(v)$, and Theorem 5, the following holds:

$$\begin{aligned} \mu_{t-1}(v_t) + \alpha_t s_{t-1}(v_t) &\geq \mu_{t-1}(v_{t'}^*) + \alpha_t s_{t-1}(v_{t'}^*) \\ &\geq \mu_{t-1}(v_{t'}^*) + \alpha_t \sigma_{t-1}(v_{t'}^*) \geq f(v_{t'}^*) \geq f(v_t^*). \end{aligned}$$

If v_t^* has not been picked yet, a similar argument gives

$$\mu_{t-1}(v_t) + \alpha_t s_{t-1}(v_t) \geq \mu_{t-1}(v_t^*) + \alpha_t s_{t-1}(v_t^*) \geq f(v_t^*).$$

Thus we always have

$$\begin{aligned} f(v_t^*) &\leq \mu_{t-1}(v_t) + \alpha_t s_{t-1}(v_t) \\ &\leq f(v_t) + \alpha_t \sigma_{t-1}(v - t) + \alpha_t s_{t-1}(v_t) \\ &\leq f(v_t) + 2\alpha_t k \sigma_{t-1}(v_t). \end{aligned}$$

Lemma 7 (Lemma 5.4, Srinivas et al. (2012))**.** *Let α_t be defined as in Theorem 5 and c_1 be defined as in Theorem 2. Conditioned on the high probability event of Theorem 5, the following holds:*

$$\forall T \geq 1, \ \ \sum_{t=1}^T r_t^2 \leq \alpha_T k^2 c_1 \mathcal{I}(\mathbf{y}_{\mathbf{v}_T}; f_{\mathbf{v}_T}) \leq \alpha_T k^2 c_1 \gamma_T.$$

Finally, the Cauchy-Schwarz inequality gives $R_T \leq \sqrt{T \sum_{t=1}^T r_t^2} \leq k\sqrt{T c_1 \alpha_T \gamma_T}$.

References

Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *The Journal of Machine Learning Research*, 3:397–422, 2003.

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, volume 14, pages 585–591, 2001.

Michael W. Berry and Murray Browne. The 2001 annotated (by topic) Enron email data set. URL `http://cis.jhu.edu/~parky/Enron/Anno_Topic_exp_LDC.pdf`.

Sébastien Bubeck, Gilles Stoltz, Csaba Szepesvári, and Rémi Munos. Online optimization in X-armed bandits. In *Advances in Neural Information Processing Systems*, pages 201–208, 2009.

Emile Contal, Vianney Perchet, and Nicolas Vayatis. Gaussian process optimization with mutual information. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 253–261, 2014.

Dennis D Cox and Susan John. Sdo: A statistical method for global optimization. *Multidisciplinary design optimization: state of the art*, pages 315–329, 1997.

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. In *COLT*, pages 355–366, 2008.

Roman Garnett, Yamuna Krishnamurthy, Xuehan Xiong, Jeff Schneider, and Richard Mann. Bayesian optimal active search and surveying. In *ICML*, 2012.

Alkis Gotovos, Nathalie Casati, Gregory Hitz, and Andreas Krause. Active learning for level set estimation. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 1344–1350. AAAI Press, 2013.

Ming Ji and Jiawei Han. A variance minimization criterion to active learning on graphs. In *AISTAT*, 2012.

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

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research (JMLR)*, 9:235–284, February 2008.

Yifei Ma, Roman Garnett, and Jeff Schneider. Σ-optimality in active learning on Gaussian random fields. In *NIPS*, 2013.

Carey E. Priebe, John M. Conroy, David J. Marchette, and Youngser Park. Scan statistics on Enron graphs. URL `http://cis.jhu.edu/~parky/Enron/enron.html`.

Friedrich Pukelsheim. *Optimal design of experiments*, volume 50. siam, 1993.

Carl Edward Rasmussen and Christopher KI Williams. *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, MA, 2006.

Herbert Robbins. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pages 169–177. Springer, 1985.

Burr Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11, 2010.

Alexander J. Smola and Risi Kondor. Kernels and regularization on graphs. In *COLT/Kernel*, pages 144–158, 2003.

Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *Information Theory, IEEE Transactions on*, 58(5):3250–3265, 2012.

Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

Michal Valko, Rémi Munos, Branislav Kveton, and Tomáš Kocák. Spectral bandits for smooth graph functions. In *31th International Conference on Machine Learning*, 2014.

Hastagiri P Vanchinathan, Andreas Marfurt, Charles-Antoine Robelin, Donald Kossmann, and Andreas Krause. Adaptively selecting valuable diverse sets via Gaussian processes and submodularity. In *NIPS Workshop on Discrete and Combinatorial Problems in Machine Learning (DISCML) 2013: Theory and Applications*, 2013.

Xuezhi Wang, Roman Garnett, and Jeff Schneider. Active search on graphs. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 731–738. ACM, 2013.

Xiaojin Zhu, Zoubin Ghahramani, John Lafferty, et al. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, volume 3, pages 912–919, 2003a.

Xiaojin Zhu, John Lafferty, and Zoubin Ghahramani. Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In *ICML 2003 workshop on the continuum from labeled to unlabeled data in machine learning and data mining*, pages 58–65, 2003b.