Distributed Thompson Sampling Under Constrained Communication

Saba Zerefa, *Graduate Student Member, IEEE*, Zhaolin Ren, *Graduate Student Member, IEEE*, Haitong Ma, *Graduate Student Member, IEEE*, and Na Li, *Senior Member, IEEE*

***Abstract*—In Bayesian optimization, a black-box function is maximized via the use of a surrogate model. We apply distributed Thompson sampling, using a Gaussian process as a surrogate model, to approach the multi-agent Bayesian optimization problem. In our distributed Thompson sampling implementation, each agent receives sampled points from neighbors, where the communication network is encoded in a graph; each agent utilizes their own Gaussian process to model the objective function. We demonstrate theoretical bounds on Bayesian average regret and Bayesian simple regret, where the bound depends on the structure of the communication graph. Unlike in batch Bayesian optimization, this bound is applicable in cases where the communication graph amongst agents is constrained. When compared to sequential single-agent Thompson sampling, our bound guarantees faster convergence with respect to time as long as the communication graph is connected. We confirm the efficacy of our algorithm with numerical simulations on traditional optimization test functions, demonstrating the significance of graph connectivity on improving regret convergence.**

***Index Terms*—Distributed control, optimization, machine learning.**

I. Introduction

BLACK-BOX stochastic optimization involves solving problems where the objective function is not explicitly known and can only be accessed through noisy evaluations [1]. These challenges frequently arise in domains where the evaluation process is costly and uncertain, such as hyperparameter tuning in machine learning [2], [3], simulation-based optimization [4], and experimental design [5]. A variety of methods have been developed to tackle these problems, including evolutionary algorithms [6], particle swarm optimization [7], and finite-difference methods [8]. Among these, Bayesian optimization (BO) [9], [10], has emerged as a particularly powerful framework. In contrast to the aforementioned black-box stochastic optimization algorithms which tend to be model-free, by leveraging a probabilistic surrogate model, often a Gaussian process (GP) [11], BO not only handles the stochastic nature of the evaluations but also balances exploration and exploitation given an appropriately chosen surrogate-based sampling strategy. This data-efficient approach makes BO especially well-suited for optimizing expensive, noisy black-box functions [12]. Moreover, theoretically, BO is also known to satisfy finite-time convergence guarantees to global optima (which we note comes at the cost of a dependence on a term that depends on the complexity of the kernel used to model the underlying function) [12]. To the best of our knowledge, apart from BO, finite-time convergence rates in stochastic optimization are only available for finite-difference type methods. However, due to the local nature of finite-difference type methods, the corresponding finite-time convergence rates for such methods only guarantee convergence to stationary points [8], [13], in contrast to the convergence to global optima achieved by BO algorithms.

In BO, the generation of new sampling points is based on the current surrogate model. A good sampling strategy should balance exploration and exploitation of the current surrogate, which is key for efficient optimization. Common sampling strategies include acquisition function-based approaches such as expected improvement (EI) [10] and BO-upper confidence bound (UCB) [14]. Another popular sampling strategy is Thompson sampling, where the next query point is selected as the optimizer of a random function realization sampled from the current posterior [15], [16]. To evaluate algorithm performance, *regret* is studied, which quantifies the gap between the performance of sampled points and the global optimum [17]. Types of regret include simple regret, which measures the gap between the optimal value and the performance of the best queried point [18], and cumulative regret, which measures the sum of the gaps between the optimal value and the performance of each queried point [19], [20].

We are interested in multi-agent BO, where multiple agents can sample the objective function at a single timestep. Much of existing multi-agent BO literature studies batch BO, in which a central coordinator has access to each agent's acquired information [21], [22]. It then computes the sampling decisions for all agents, and communicates these decisions to each agent. These decisions are disseminated in batches, allowing multiple agents to simultaneously sample points, parallelizing the optimization process [23], [24].

Received 16 September 2024; revised 22 November 2024; accepted 16 December 2024. Date of publication 2 January 2025; date of current version 17 January 2025. This work was supported in part by NSF AI Institute under Grant 2112085; in part by NSF ASCENT under Grant 2328241; and in part by NSF CNS under Grant 2003111. Recommended by Senior Editor P. Tesi. *(Corresponding author: Zhaolin Ren.)*

The authors are with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 USA (e-mail: szerefa@g.harvard.edu; zhaolinren@g.harvard.edu; haitongma@g.harvard.edu; nali@seas.harvard.edu).

Digital Object Identifier 10.1109/LCSYS.2024.3525096

Centralized approaches are inapplicable in distributed cases, in which there is no centralized coordinator and each agent must possess a local instance of the optimization algorithm [25]. Additionally, they often do not scale well, as they require a central coordinator to manage the processing of all agents' data. Distributed networks are prevalent in real-world applications, such as in multi-robot source seeking and sensor networks [21], [26]. It may not be the case that all agents have access to all prior sampled points as in the batch setting - communication may be constrained, where some agents are only able to communicate with specific other agents [27]. These constraints may be due to limited communication capacity or computational capacity of the agents, or due to physical proximity constraints. Prior literature providing theoretical guarantees for distributed Bayesian optimization require fully connected communication graphs, even in asynchronous cases [22], [28], and thus are inapplicable in settings with constrained communication. In this letter, we study the distributed setting with constrained communication, in which at each round, agents send their sampled points to their neighbors and receive points sampled by their neighbors.

Our contribution: We propose a distributed Thompson sampling algorithm for the multi-agent Bayesian optimization problem under constrained communication. In the algorithm, each agent uses their own GP model to pick sampling points via Thompson sampling, and shares the queried points with its neighbors. We provide provable guarantees for the proposed distributed BO. In Theorem 1, we establish a Bayesian average regret bound of $\tilde{O}(\frac{\sqrt{\theta(G)}}{\sqrt{Mt}})$, where M is the number of agents, t is the number of optimization rounds, and $\theta(G)$ represents the clique cover number of G, i.e., $\theta(G)$ is the smallest number L such that the graph G can be decomposed into L disjoint complete subgraphs. This implies then that the average regret bound is smaller for graphs with higher connectivity, which can be decomposed into a few large disjoint complete subgraphs. We also characterize Bayesian simple regret, demonstrating a bound of $\tilde{O}(\sqrt{\frac{1}{t|V_{max}|}})$, where $|V_{max}|$ is the size of the largest complete subgraph of the communication network G. We note that this convergence speed is $O(\sqrt{|V_{max}|})$ times better than the best known simple regret rate for sequential single-agent BO, which is $\tilde{O}(\sqrt{\frac{1}{t}})$ [14]. We numerically test our algorithm on two standard optimization test functions [29] with Erdős-Rényi graphs, demonstrating the efficiency of our algorithm. We find that lower regret is achieved with graphs of higher connectivity, supporting our theoretical results.

II. Problem Formulation and Preliminaries

A. Problem Formulation

For a compact set $\mathcal{X} \subset \mathbb{R}^d$, consider an unknown continuous function $f : \mathcal{X} \to \mathbb{R}$, with optimizer x^*. The goal is to find the maximum of this function, where we are only able to sample f through expensive and noisy evaluations. We assume any of M agents can query f at any point and receive a noisy value $y = f(x) + \epsilon$, with $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$. Agents query f throughout a total of T iterations. For agent $i \in \{1, \ldots, M\}$ and iteration $t \in \{1, \ldots, T\}$, $x_{t,i}$ is the query point, and $y_{t,i}$ is the corresponding evaluation. Define $X_{t,i} = \{x_{1,i}, \ldots, x_{t,i}\}$, $Y_{t,i} = \{y_{1,i}, \ldots, y_{t,i}\}$ to be the queries and evaluations made by agent i up to time t. The communication network of M agents is described by graph $G = (V, E)$, where $|V| = M$, and $E \subset \{\{i, j\} : i, j \in V, i \neq j\}$. An unordered pair $\{i, j\} \in E$ if agents i and j are able to communicate with each other. Additionally, we denote the set of neighbors of agent i as $N(i) = \{j : \{i, j\} \in E\}$. The data accessible to agent i at time t is $D_{t,i} = \{(x_{\tau,j}, y_{\tau,j})\}_{j \in N(i) \cup i, \tau < t}$. The set $D_{t,i}$ contains all sampled points up to time t by agent i and its neighbors. We do not make any assumptions regarding the structure of the communication network. The graph may even be unconnected. Our analysis will show how the graph structure affects the algorithm's performance.

B. Gaussian Process

We use a Gaussian process (GP) to model our unknown objective function f in our BO setting. Recall the unknown continuous objective function $f : \mathcal{X} \to \mathbb{R}$. Let $\mathbf{X}_{D_t} = \{x_1, x_2, \ldots, x_t\}$, where x_j is the jth evaluated point, and let $k : \mathcal{X}^2 \to \mathbb{R}$ be a kernel function. Define

$$\mu_{D_t}(x) = \mathbf{k}_t(x)^\intercal \left(\mathbf{K}_{D_t} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{y}_{D_t}$$

$$k_{D_t}(x, x') = k(x, x') - \mathbf{k}_{D_t}(x)^\intercal \left(\mathbf{K}_{D_t} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{k}_{D_t}(x'),$$

where $\mathbf{K}_{D_t} := [k(x', x'')]_{x', x'' \in \mathbf{X}_{D_t}}$, $\mathbf{k}_{D_t}(x) := [k(x', x)]_{x' \in \mathbf{X}_{D_t}}$ and $\mathbf{y}_{D_t} = \{f(x') + \epsilon'\}_{x' \in \mathbf{X}_{D_t}}$, where $\epsilon' \sim \mathcal{N}(0, \sigma_\epsilon^2)$. Thus we can define our GP, in which we denote $f|\mathcal{F}_{D_t} \sim GP(\mu_{D_t}(x), k_{D_t}(x, x'))$. Note that due to the nature of the GP, it is the case that for any $x \in \mathcal{X}$, $f(x)|\mathcal{F}_{D_t} \sim N(\mu_{D_t}(x), \sigma_{D_t}^2(x))$, where $\sigma_{D_t}^2(x) = k_{D_t}(x, x)$ [11]. Furthermore, recall the distributed multi-agent setting, where each of M agents have access to queried points in set $D_{t,i}$, where $D_{t,i}$ and $D_{t,j}$ may not be equal for distinct agents i and j. In our distributed setting, each agent i has a unique GP model of f at time t, $\mathcal{GP}_{t,i}$, since the data $D_{t,i}$ available to each agent i is different. Thus we denote $f \mid \mathcal{F}_{D_{t,i}} \sim \mathcal{GP}_{t,i}(\mu_{D_{t,i}}(x), k_{D_{t,i}}(x, x'))$. In the GP framework, x^*, the optimizer of f, is treated as a random variable. As a result, x^* has a posterior distribution $x^*|D_t$, which is the optimal value of the GP $f|D_t$. This structure is leveraged in our algorithm.

The kernel function $k(\cdot, \cdot)$ can be selected to reflect prior beliefs about the objective function f, such as function smoothness [11]. Common selections of kernel functions include Linear, Squared Exponential, and Matérn kernels, the latter of which was used in the numerical implementations of our algorithm. Note that the GP problem structure does not make any assumptions regarding function convexity, and that for common kernels, convexity is not reflected by kernel selection.

C. Regret

Our metric for algorithm performance is regret, which is an assessment of the quality of sampled points. We consider average regret, which quantifies the difference between the optimal value of the function and the queried value for each sampled point. In average regret, this difference is accumulated across all agents and timesteps, and then averaged by the

amount of sampled points. To account for randomness of f in our regret expression, we take expectation of average regret to yield the following expression, which we call Bayesian average regret:

$$R_{AB}(t) = \frac{1}{tM}\sum_{\tau=1}^{t}\sum_{i=1}^{M}\mathbb{E}\left[f\left(x^*\right) - f\left(x_{\tau,i}\right)\right] \tag{1}$$

We also consider the simple regret, which is the difference between the optimal value of the function and the best value achieved amongst the previous queried points. This definition of regret is useful because optimization settings focus on locating the extrema of a function, and the simple regret tracks the smallest gap between the value at a sampled point and the optimal value. We take the expectation of simple regret to yield the following expression, which is called Bayesian simple regret:

$$R_{SB}(t) = \min_{i\in\{1,2,\ldots,M\},\tau\in\{1,2,\ldots,t\}}\mathbb{E}\left[f\left(x^*\right) - f\left(x_{\tau,i}\right)\right] \tag{2}$$

In our theoretical analysis, we provide bounds on Bayesian average regret and Bayesian simple regret.

D. Thompson Sampling

Thompson sampling is an algorithm for sequential decision making that can be utilized in this context for determining the next point of the objective function to query [15]. When using Thompson sampling in our Bayesian optimization framework, an acquisition function is sampled from the posterior distribution of the Gaussian process. The maximizer of this function is the next query point at which the black-box objective function is sampled. The Gaussian process is then updated with new information from this sample, and the process repeats for the duration of the experiment.

In sequential single-agent Thompson sampling, each subsequent query point is determined based on a single model updated on all prior sampled points. Alternatively, in batch Thompson sampling, multiple query points are determined as a set at each round, and the objective function is sampled in parallel [22], [24]. Batch Thompson sampling is advantageous in systems capable of parallelizing, e.g., multi-agent systems, because it allows for convergence in fewer number of rounds than sequential single-agent Thompson sampling.

Batch Thompson sampling is centralized, with all agents having access to the same information. However, this may not be realistic in real-world situations, where communication between agents may be constrained due to bandwidth limitations, computational constrictions, or privacy concerns. In these cases, agents may only have access to the sampled points by few other agents, and thus datasets available to distinct agents may differ. We propose a distributed Thompson sampling algorithm for this constrained communication case, and provide theoretical guarantees for the algorithm.

III. Algorithm: Distributed Thompson Sampling

In our implementation of distributed Thompson sampling, each of M agents have distinct Gaussian processes $\mathcal{GP}_i$ for modeling the objective function. At each time step t, all agents update their GPs with the data history available to them. The agent then queries the objective function at $x_{t,i}$, which is the maximizer of the acquisition function sampled from the posterior GP, $\hat{f}_{t,i} \sim \mathcal{GP}_{t,i}$. Each agent then communicates its sampled point to its neighbors, receives the points sampled by their neighbors, and updates their data history accordingly. The collection of data received by neighbors of agent i at time t is denoted as $C_{t,i} = \{(x_{t,j}, y_{t,j})\}_{j\in N(i)}$. Our method is shown in Algorithm 1. We stress that while we do assume a synchronous global clock, there exists no centralized coordinator in our algorithm that coordinates the queries of the different agents.

Algorithm 1 Distributed Thompson Sampling

1: Place GP prior on f
2: **for** i= $1, \ldots, M$ **do**
3: Initial data $D_{1,i}$
4: $\mathcal{GP}_{0,i} \leftarrow GP$
5: **end for**
6: **for** $t = 1, \ldots, T$ **do**
7: **for** i= $1, \ldots, M$ **do**
8: Update posterior $\mathcal{GP}_{t,i}$ conditioned on $D_{t,i}$
9: Sample $\hat{f}_{t,i} \sim \mathcal{GP}_{t,i}$
10: Choose next query point
11: $x_{t,i} \leftarrow \arg\max_x \hat{f}_{t,i}(x)$
12: Observe $y_{t,i}$
13: Broadcast $(x_{t,i}, y_{t,i})$ to neighbors $N(i)$;
14: Collect evaluations $C_{t,i}$ from neighbors $N(i)$
15: Update data history $D_{t+1,i} \leftarrow D_{t,i} \cup C_{t,i} \cup \{(x_{t,i}, y_{t,i})\}$
16: **end for**
17: **end for**

In step 11, we select the next sampling point of the objective function by finding the argmax of a function drawn from the posterior distribution of the GP. In our numerical implementation, we did so using gridsearch, but such an approach is computationally expensive for higher dimensional search spaces. Efficient computation of the argmax for Thompson sampling in high dimensional spaces is an active area of research, and a direction for future work.

A. Theoretical Result

We analyze the performance of the distributed Thompson sampling algorithm on the Bayesian average regret and Bayesian simple regret metrics. Our regret bound depends on the number of timesteps T and the structure of the agent communication graph G. As in prior work, we utilize notions from information theory in our regret bound [30].

Our regret bound involves the Maximum Information Gain (MIG), which is a constant that captures the complexity of the objective function. MIG has been shown to be bounded for several kernel functions commonly used with GPs, including Squared Exponential and Matérn kernels, the latter of which was used in our numerical implementation [14].

Let $D = \{x_1, \ldots, x_t\} \subset \mathcal{X}$, and define $y_D = \{(x, f(x) + \epsilon) : x \in D\}$. The MIG is denoted as

$$\Psi_t = \max_{D\subset\mathcal{X},|D|=t} I(f; y_D), \tag{3}$$

where I is the Shannon Mutual Information. The MIG Ψ_t represents the largest mutual information gain from f by sampling t points. Additionally, for any positive integer n, we define the constant ξ_n, which bounds the information gain of the current round of evaluations [22]. Suppose $|D| = t$ points were already sampled, and i points are being queried in the current round of evaluations, with $i < n$; denote these points in set A, where $A \subset \mathcal{X}$, and $y_A = \{(x, f(x)+\epsilon) : x \in A\}$. Then for $i \geq 1$, ξ_n satisfies

$$\max_{A\subset\mathcal{X},|A|<n} I(f; y_A|y_D) \leq \frac{1}{2}\log(\xi_n). \tag{4}$$

We next provide a bound for Bayesian average regret for an M agent system with communication graph G.

Theorem 1: Suppose $k(x, x') \leq 1$ for all x, x'. Let $\{G_k\}_{k\in\{1,\ldots,n\}}$ be a collection of n disjoint complete subgraphs of communication graph $G = (V, E)$, where $G_k = (V_k, E_k)$, and $\cup_{k\in\{1,\ldots,n\}} V_k = V$. Then the Bayesian average regret after t timesteps satisfies $R_{AB}(t) \leq \frac{1}{M}\sum_{k=1}^{n} |V_k|(\frac{C_1}{t|V_k|} + \sqrt{\frac{C_2\xi_{|V_k|}\beta_t\Psi_{t|V_k|}}{t|V_k|}})$, where $\beta_t = 2\log(t^2M|\mathcal{X}|)$, $C_1 = \frac{\sqrt{2}\pi^{3/2}}{12}$, and $C_2 = \frac{2}{\log(1+\sigma_\epsilon^{-2})}$.

Proof: The structure of our proof follows techniques from Kandasamy et al. [22]. We aim to provide a bound on Bayesian average regret. Our proof begins by noting that we can develop an expression for Bayesian average regret by consider the Bayesian average regret of specific subsets of agents. We then decompose this into three sums, each of which utilize a confidence function $U_{t,i}(\cdot)$. We bound each of these sums using notions from information theory, allowing us to use information gain constants introduced in Equations (3) and (4) to analyze the efficacy of the sampling process.

We bound the Bayesian average regret affiliated with agents in communication graph G by bounding the Bayesian average regret within each complete subgraphs of graph G. Let $G = (V, E)$ be the communication graph for the M agents. We can construct a collection of n disjoint complete subgraphs $\{G_k\}_{k\in\{1,\ldots,n\}}$, where each $G_k = (V_k, E_k)$ is a subgraph of G, with $\cup_{k\in\{1,2,\ldots,n\}} V_k = V$.

Recall from Equation (1) that Bayesian average regret $R_{AB}(t) = \frac{1}{tM}\sum_{\tau=1}^{t}\sum_{i=1}^{M}\mathbb{E}[f(x^*) - f(x_{\tau,i})]$. We also introduce $R_{AB,k}(t) = \frac{1}{t|V_k|}\sum_{\tau=1}^{t}\sum_{i\in V_k}\mathbb{E}[f(x^*) - f(x_{\tau,i})]$, which is the Bayesian average regret affiliated with agents in V_k. Recalling the partition of the vertex set V into $\{V_k\}_{k\in\{1,\ldots,n\}}$, we may rewrite Bayesian average regret as follows:

$$\begin{aligned}
R_{AB}(t) &= \frac{1}{tM}\sum_{\tau=1}^{t}\sum_{i=1}^{M}\mathbb{E}\big[f(x^*) - f(x_{\tau,i})\big] \\
&= \frac{1}{tM}\sum_{\tau=1}^{t}\sum_{k=1}^{n}\sum_{i\in V_k}\mathbb{E}\big[f(x^*) - f(x_{\tau,i})\big] \\
&= \frac{1}{tM}\sum_{k=1}^{n}\sum_{\tau=1}^{t}\sum_{i\in V_k}\mathbb{E}\big[f(x^*) - f(x_{\tau,i})\big] \\
&= \frac{1}{M}\sum_{k=1}^{n}|V_k|R_{AB,k}(t)
\end{aligned}$$

Thus, it suffices to focus on bounding $R_{AB,k}(t)$. Define $U_{t,i}(x) = \mu_{D_{t,i}}(x) + \beta_t^{1/2}\sigma_{D_{t,i}}(x)$. To upper bound $R_{AB,k}(t)$, we can decompose the sum $\sum_{\tau=1}^{t}\sum_{i\in V_k}\mathbb{E}[f(x^*) - f(x_{\tau,i})]$ as follows:

$$\begin{aligned}
&\sum_{\tau=1}^{t}\sum_{i\in V_k}\mathbb{E}\big[f(x^*) - f(x_{\tau,i})\big] \\
&= \sum_{\tau=1}^{t}\sum_{i\in V_k}\mathbb{E}[f(x^*) - U_{\tau,i}(x^*) + U_{\tau,i}(x^*) - U_{\tau,i}(x_{\tau,i}) \\
&\quad + U_{\tau,i}(x_{\tau,i}) - f(x_{\tau,i})] \\
&= \sum_{\tau=1}^{t}\sum_{i\in V_k}\underbrace{\mathbb{E}\big[f(x^*) - U_{\tau,i}(x^*)\big]}_{S1} + \underbrace{\mathbb{E}\big[U_{\tau,i}(x^*) - U_{\tau,i}(x_{\tau,i})\big]}_{S2} \\
&\quad + \underbrace{\mathbb{E}\big[U_{\tau,i}(x_{\tau,i}) - f(x_{\tau,i})\big]}_{S3}
\end{aligned}$$

By leveraging tools from information theory, we can upper bound each of these sums, finding that $S1 \leq \frac{\sqrt{2}\pi^{3/2}}{12}$, $S2 = 0$, and $S3 \leq \sqrt{\frac{2\xi_{|V_k|}t|V_k|\beta_t\Psi_{t|V_k|}}{\log(1+\sigma_\epsilon^{-2})}}$. Therefore, $R_{AB,k}(t) \leq \frac{\sqrt{2}\pi^{3/2}}{12t|V_k|} + \sqrt{\frac{2\xi_{|V_k|}\beta_t\Psi_{t|V_k|}}{t|V_k|\log(1+\sigma_\epsilon^{-2})}}$. A detailed derivation of these bounds is available in our supplement [31].

Equipped with a bound on $R_{AB,k}(t)$, we can revisit our expression for $R_{AB}(t)$.

$$\begin{aligned}
R_{AB}(t) &= \frac{1}{M}\sum_{k=1}^{n}|V_k|R_{AB,k}(t) \\
&\leq \frac{1}{M}\sum_{k=1}^{n}|V_k|\left(\frac{C_1}{t|V_k|} + \sqrt{\frac{C_2\xi_{|V_k|}\beta_t\Psi_{t|V_k|}}{t|V_k|}}\right),
\end{aligned}$$

where $C_1 = \frac{\sqrt{2}\pi^{3/2}}{12}$ and $C_2 = \frac{2}{\log(1+\sigma_\epsilon^{-2})}$. Thus, we have shown that $R_{AB}(t) \leq \frac{1}{M}\sum_{k=1}^{n}|V_k|\left(\frac{C_1}{t|V_k|} + \sqrt{\frac{C_2\xi_{|V_k|}\beta_t\Psi_{t|V_k|}}{t|V_k|}}\right)$, concluding our proof. ∎

The full proof of Theorem 1 is available in our supplement [31]. By picking n to be the clique cover number of the graph G, Theorem 1 yields the following corollary.

Corollary 1: Suppose $k(x, x') \leq 1$ for all x, x'. Let $\theta(G)$ and $\omega(G)$ denote the clique cover number and clique number of the graph G respectively. Then, the Bayesian average regret after t timesteps satisfies $R_{AB}(t) \leq \frac{C_1\theta(G)}{Mt} + \frac{\sqrt{\theta(G)}\sqrt{C_2\xi_{\omega(G)}\beta_t\Psi_{t\omega(G)}}}{\sqrt{Mt}}$, where β_t, C_1 and C_2 are as defined in Theorem 2.

Proof: The proof of Corollary 1 follows from (i) applying Cauchy-Schwarz to bound the term $\sum_{k=1}^{n}\sqrt{|V_k|} \leq \sqrt{n}\sqrt{\sum_{k=1}^{n}|V_k|} = \sqrt{Mn}$, (ii) picking n to be the clique cover number of G, $\theta(G)$, and (iii) the fact that for any clique $G_k = (V_k, E_k)$ in G, $|V_k| \leq \omega(G)$, since $\omega(G)$ denotes the clique number of G (i.e., size of the largest clique in G). ∎

From Corollary 1, the average regret satisfies $R_{AB}(t) = \tilde{O}(\frac{\sqrt{\theta(G)\xi_{\omega(G)}\Psi_{t\omega(G)}}}{\sqrt{Mt}})$ (recall $\omega(G)$ denotes the clique number of G). We note that the term $\Psi_{t\omega(G)}$ corresponds to the maximal mutual information gain from $t\omega(G)$ observations, and that this quantity depends only logarithmically on $t\omega(G)$ for standard

kernels such as the squared exponential kernel. For more details, see Appendix II in our supplement [31]. The term $\xi_{\omega(G)}$ is the price we pay for the absence of coordination within each of the subgraphs G_k in G, and is a standard term that arises in multi-agent Bayesian optimization. By an appropriate initialization phase, this term can be reduced to $\tilde{O}(1)$, (see [24, Appendix B.3]). Thus, compared to the sequential single-agent case with t rounds which has average regret $\tilde{O}\left(\sqrt{\frac{1}{t}}\right)$ [14], our algorithm satisfies a regret of $\tilde{O}\left(\frac{\sqrt{\theta(G)}}{\sqrt{Mt}}\right)$, i.e., an improvement of $\sqrt{\frac{\theta(G)}{M}}$ (note this term is always smaller than 1). Correspondingly, the average regret is smaller for graphs with higher connectivity, whose clique cover number $\theta(G)$ is smaller. We next proceed to bound the Bayesian simple regret.

Theorem 2: Suppose $k(x, x') \leq 1$ for all x, x'. Let $G_s = (V_s, E_s)$ be a complete subgraph of G. Then the Bayesian simple regret after t timesteps satisfies $R_{SB}(t) \leq \frac{C_1}{t|V_s|} + \sqrt{\frac{C_2 \xi_{|V_s|} \beta_t \Psi_{t|V_s|}}{t|V_s|}}$, where $\beta_t = 2\log(t^2|V_s||\mathcal{X}|)$, $C_1 = \frac{\sqrt{2}\pi^{3/2}}{12}$, and $C_2 = \frac{2}{\log(1+\sigma_\epsilon^{-2})}$.

We defer the proof of Theorem 2 to our supplement, and note that it essentially follows from the observation that the Bayesian simple regret is bounded by the Bayesian average regret of any subset of agents. In particular, for any $G_s = (V_s, E_s)$ which is a complete subgraph of G, the simple regret satisfies $R_{SB}(t) \leq \frac{1}{t|V_s|} \sum_{\tau=1}^{t} \sum_{i \in V_s} \mathbb{E}[f(x^*) - f(x_{\tau,i})]$, where the right hand side denotes the Bayesian average regret of the subgraph G_s, which can then be bounded using our analysis in the proof of Theorem 2. Picking G_s to be the largest complete subgraph of the communication network G then yields the following corollary.

Corollary 2: Suppose $k(x, x') \leq 1$ for all x, x'. Let $G_{max} = (V_{max}, E_{max})$ be the largest complete subgraph of G. Then the Bayesian simple regret after t timesteps satisfies $R_{SB}(t) \leq \frac{C_1}{t|V_{max}|} + \sqrt{\frac{C_2 \xi_{|V_{max}|} \beta_t \Psi_{t|V_{max}|}}{t|V_{max}|}}$, where $\beta_t = 2\log(t^2|V_{max}||\mathcal{X}|)$, $C_1 = \frac{\sqrt{2}\pi^{3/2}}{12}$, and $C_2 = \frac{2}{\log(1+\sigma_\epsilon^{-2})}$.

From the above corollary and our discussion following Corollary 1, we see that $R_{SB}(t) = \tilde{O}\left(\sqrt{\frac{1}{t|V_{max}|}}\right)$. Thus, compared to the sequential single-agent case with t rounds which has simple regret $\tilde{O}\left(\sqrt{\frac{1}{t}}\right)$ [14], our algorithm satisfies a regret of $\tilde{O}\left(\sqrt{\frac{1}{t|V_{max}|}}\right)$, i.e., an improvement of $\sqrt{\frac{1}{|V_{max}|}}$, demonstrating the benefit of the network structure for the simple regret case as well.

IV. Numerical Experiments

A. Simulation

In the numerical implementation, performance was assessed utilizing the following regret metrics. We define the *Instant average regret* R_A, and its sum, $\overline{R_A}$, as follows:

$$R_A(t) = \frac{1}{M} \sum_{i=1}^{M} \left(f^* - f\left(x_{t,i}\right)\right), \overline{R_A}(t) = \sum_{\tau=1}^{t} R_A(\tau).$$

We also define the *Instant simple regret* R_S, and its sum $\overline{R_S}$, as follows:

$$R_S(t) = f^* - \max_{\substack{i \in \{1,2,\ldots M\}, \\ \tau \in \{1,2,\ldots,t\}}} f\left(x_{t,i}\right), \overline{R_S}(t) = \sum_{\tau=1}^{t} R_S(\tau),$$

where $f^* = \max_{x \in \mathcal{X}} f(x)$. Numerical results were constructed in a Python implementation built upon the BOTorch package [32]. The code used to generate the simulation and corresponding Figure 1 is available at https://github.com/sabzer/distributed-bo. The Gaussian processes utilized the Matérn kernel with parameter $\nu = \frac{5}{2}$. The numerical simulations were run over $T = 50$ timesteps. Simulations were run based on two test functions for the objective function: Ackley, which has many local maxima and one global minima in the origin, and Rosenbrock, which contains a large valley in which the global minima is situated. The equations of the aforementioned objective functions and their plots are available in Appendix III in our supplement [31]. Since we were solving a maximization problem, we multiplied the canonical definitions of these functions by -1 for the purpose of our simulation. For the communication networks in our simulations, we used Erdős-Rényi random graphs of 20 agents with connectivities of 0.2, 0.4, and 0.6 [33]. The connection probabilities are the probability that each edge from the complete graph of 20 agents appears in the corresponding random graph.

B. Discussion

Our theoretical result bounds Bayesian average regret, $R_{AB}(t)$, and Bayesian simple regret, $R_{SB}(t)$, with the bound dependent on the structure of the communication network between agents. Our distributed Thompson sampling algorithm was able to achieve the extrema of the Ackley and Rosenbrock objective functions in numerical implementation, and thus is effective at the Bayesian optimization task. Our theoretical results suggests that the distributed Thompson sampling algorithm implementation favors highly connected communication graphs. This is apparent from a lower Bayesian average regret bound when the communication graph can be decomposed into a few large disjoint complete subgraphs, and a lower Bayesian simple regret bound when the largest complete subgraph of the communication graph has a larger number of agents. Our numerical results support this intuition, for in Figure 1, we see better regret convergence for Erdős-Rényi graphs of higher connectivity. This result holds for both Ackley and Rosenbrock objective functions, and for both Instant simple and average regret.

V. Conclusion

In this letter, we proposed a distributed Thompson sampling algorithm to address the multi-agent Bayesian optimization problem under constrained communication. We develop bounds on Bayesian average regret and Bayesian simple regret for this approach, where the bound is dependent on properties of the largest complete subgraph of the graph encoding communication structure between agents. With our bound, we show that in connected multi-agent communication networks, both Bayesian average regret and Bayesian simple regret will converge faster with distributed Thompson sampling than

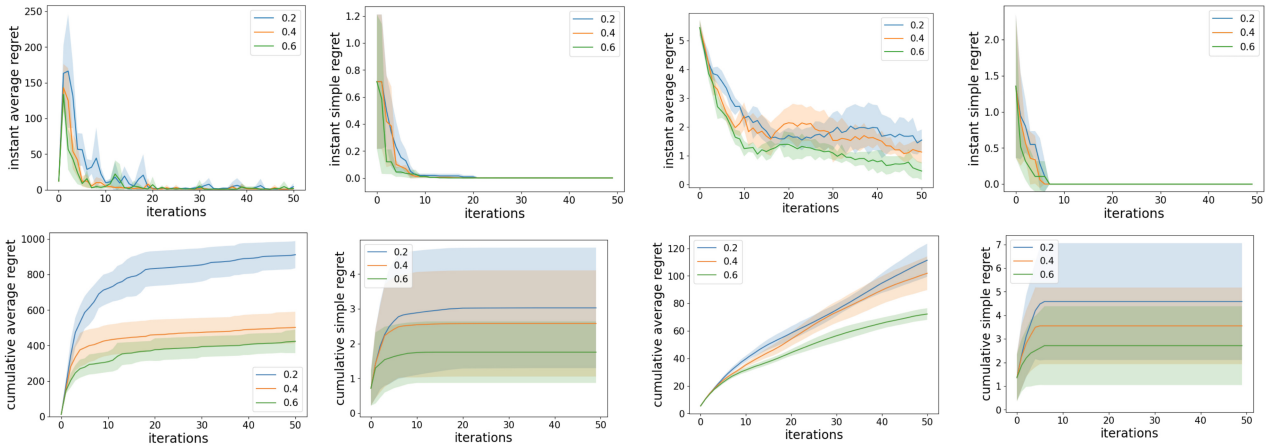

Fig. 1. Regret analysis of numerical simulations with 20 agents on Erdős-Rényi random graphs with connectivity probability 0.2 (blue), 0.4 (orange), and 0.6 (green), on Rosenbrock (left) and Ackley (right) objective functions.

in the sequential single-agent case, with the same number of rounds. Additionally, we demonstrate the efficacy of our algorithm with regret analysis on optimization test functions, illustrating faster convergence with well connected communication graphs. Future work will focus on developing a tighter regret bound, and further tailoring the distributed Thompson sampling algorithm towards the constrained communication case by leveraging the data communicated between agents.

REFERENCES

[1] J. C. Spall, *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. Hoboken, NJ, USA: Wiley, 2005.

[2] J. Bergstra, D. Yamins, and D. Cox, "Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 115–123.

[3] M. Lindauer et al., "SMAC3: A versatile Bayesian optimization package for hyperparameter optimization," *J. Mach. Learn. Res.*, vol. 23, no. 54, pp. 1–9, 2022.

[4] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo Method*. Hoboken, NJ, USA: Wiley, 2016.

[5] R.-R. Griffiths and J. M. Hernández-Lobato, "Constrained Bayesian optimization for automatic chemical design using variational autoencoders," *Chem. Sci.*, vol. 11, no. 2, pp. 577–586, 2020.

[6] A. Slowik and H. Kwasnicka, "Evolutionary algorithms and their applications to engineering problems," *Neural Comput. Appl.*, vol. 32, pp. 12363–12379, Mar. 2020.

[7] F. Marini and B. Walczak, "Particle swarm optimization (PSO). A tutorial," *Chemom. Intell. Lab. Syst.*, vol. 149, pp. 153–165, Dec. 2015.

[8] A. Wibisono, M. J. Wainwright, M. Jordan, and J. C. Duchi, "Finite sample convergence rates of zero-order stochastic optimization methods," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1–9.

[9] H. J. Kushner, "A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise," *J. Basic Eng.*, vol. 86, no. 1, pp. 97–106, 1964.

[10] J. Močkus, "On Bayesian methods for seeking the extremum," in *Proc. Optim. Techn. IFIP Techn. Conf.*, 1975, pp. 400–404.

[11] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[12] P. I. Frazier, "A tutorial on Bayesian optimization," 2018, *arXiv:1807.02811*.

[13] Y. Ha and S. Shashaani, "Iteration complexity and finite-time efficiency of adaptive sampling trust-region methods for stochastic derivative-free optimization," *IISE Trans.*, pp. 1–26, Mar. 2024.

[14] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 1015–1022.

[15] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, nos. 3–4, pp. 285–294, 1933.

[16] D. Russo, B. V. Roy, A. Kazerouni, I. Osband, and Z. Wen, "A tutorial on Thompson sampling," 2020, *arXiv:1707.02038*.

[17] O. Chapelle and L. Li, "An empirical evaluation of Thompson sampling," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 1–9.

[18] S. Vakili, N. Bouziani, S. Jalali, A. Bernacchia, and D.-s. Shiu, "Optimal order simple regret for Gaussian process bandits," in *Proc. 35th Int. Conf. Neural Inf. Process. Syst.*, 2024, pp. 1–14.

[19] T. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Adv. Appl. Math.*, vol. 6, no. 1, pp. 4–22, 1985.

[20] S. Agrawal and N. Goyal, "Thompson sampling for contextual bandits with linear payoffs," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 127–135.

[21] H. Ma, T. Zhang, Y. Wu, F. P. Calmon, and N. Li, "Gaussian max-value entropy search for multi-agent Bayesian optimization," 2023, *arXiv:2303.05694*.

[22] K. Kandasamy, A. Krishnamurthy, J. Schneider, and B. Poczos, "Parallelised Bayesian optimisation via Thompson sampling," in *Proc. 21st Int. Conf. Artif. Intell. Stat.*, 2018, pp. 133–142.

[23] J. Wu and P. Frazier, "The parallel knowledge gradient method for batch Bayesian optimization," in *Proc. 30th Adv. Neural Inf. Process. Syst.*, 2016, pp. 1–9.

[24] Z. Ren and N. Li, "TS-RSR: A provably efficient approach for batch Bayesian optimization," 2024, *arXiv:2403.04764*.

[25] J. M. Hernández-Lobato, J. Requeima, E. O. Pyzer-Knapp, and A. Aspuru-Guzik, "Parallel and distributed Thompson sampling for large-scale accelerated exploration of chemical space," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 1470–1479.

[26] T. Zhang, V. Qin, Y. Tang, and N. Li, "Distributed information-based source seeking," *IEEE Trans. Robot.*, vol. 39, no. 6, pp. 4749–4767, Dec. 2023.

[27] K. Nakamura, M. Santos, and N. E. Leonard, "Decentralized learning with limited communications for multi-robot coverage of unknown spatial fields," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2022, pp. 9980–9986.

[28] J. Garcia-Barcos and R. Martinez-Cantin, "Fully distributed Bayesian optimization with stochastic policies," in *Proc. 28th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2019, pp. 1–7.

[29] M. Jamil, X.-S. Yang, and H.-J. Zepernick, "8—Test functions for global optimization: A comprehensive survey," in *Swarm Intelligence and Bio-Inspired Computation: Theory and Applications*. Amsterdam, The Netherlands: Elsevier, 2013, pp. 193–222.

[30] D. Russo and B. V. Roy, "An information-theoretic analysis of Thompson sampling," *J. Mach. Learn. Res.*, vol. 17, no. 68, pp. 1–30, 2016.

[31] S. Zerefa, Z. Ren, H. Ma, and N. Li, "Distributed Thompson sampling under constrained communication," 2025, *arXiv:2410.15543*.

[32] M. Balandat et al., "BOTORCH: A framework for efficient monte-carlo Bayesian optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 21524–21538.

[33] P. Erdős and A. Rényi, "On random graphs I," *Publ. Math. Debrecen*, vol. 6, nos. 290–297, p. 18, 1959.