Surrogate Modeling for Bayesian Optimization Beyond a Single Gaussian Process

Qin Lu, *Member, IEEE*, Konstantinos D. Polyzos, *Student Member, IEEE*, Bingcong Li, *Member, IEEE*, and Georgios B. Giannakis, *Fellow, IEEE*

***Abstract*—Bayesian optimization (BO) has well-documented merits for optimizing *black-box* functions with an *expensive* evaluation cost. Such functions emerge in applications as diverse as hyperparameter tuning, drug discovery, and robotics. BO hinges on a Bayesian surrogate model to sequentially select query points so as to balance exploration with exploitation of the search space. Most existing works rely on a single Gaussian process (GP) based surrogate model, where the kernel function form is typically *preselected* using domain knowledge. To bypass such a design process, this paper leverages an ensemble (E) of GPs to adaptively select the surrogate model fit on-the-fly, yielding a GP mixture posterior with enhanced expressiveness for the sought function. Acquisition of the next evaluation input using this EGP-based function posterior is then enabled by Thompson sampling (TS) that requires no additional design parameters. To endow function sampling with scalability, random feature-based kernel approximation is leveraged per GP model. The novel EGP-TS readily accommodates parallel operation. To further establish convergence of the proposed EGP-TS to the global optimum, analysis is conducted based on the notion of *Bayesian regret* for both sequential and parallel settings. Tests on synthetic functions and real-world applications showcase the merits of the proposed method.**

***Index Terms*—Bayesian optimization, Gaussian processes, ensemble learning, Thompson sampling, Bayesian regret analysis.**

I. Introduction

A NUMBER of machine learning and artificial intelligence (AI) applications boil down to optimizing an 'expensive-to-evaluate' black-box function, including hyperparameter tuning [1], drug discovery [2], and policy optimization in robotics [3]. As in hyperparameter tuning, lack of analytic expressions for the objective function and overwhelming evaluation cost discourage grid search, and adoption of gradient-based solvers. To find the global optimum under a limited evaluation budget, Bayesian optimization (BO) offers a principled framework by leveraging a statistical model to guide the acquisition of query points on-the-fly [4], [5].

Manuscript received 3 October 2022; revised 13 February 2023; accepted 26 March 2023. Date of publication 5 April 2023; date of current version 4 August 2023. This work was supported by NSF grants 1901134, 2128593, 2126052, 2212318, and 2220292. K. D. Polyzos was also supported by the Onassis Foundation Scholarship. Recommended for acceptance by M. Zhang. *(Qin Lu and Konstantinos D. Polyzos are equal contribution.) (Corresponding author: Qin Lu.)*

The authors are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: qlu@umn.edu; polyz003@umn.edu; lixx5599@umn.edu; georgios@umn.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TPAMI.2023.3264741, provided by the authors.

Digital Object Identifier 10.1109/TPAMI.2023.3264741

While BO can automate the selection of the best-performing machine learning model along with its optimal hyperparameters, it still necessitates domain-specific expert knowledge to design both the surrogate model and the acquisition function [1]. In the Gaussian process (GP) based surrogate model, one has to select the kernel type and the corresponding hyperparameters. Also, decision has to be made on the selection from the available acquisition functions, and the associated design parameters if there is any. Minimizing such design efforts so as to automate BO is especially appealing for modern AI tasks. Given that in many setups BO is inherently time-consuming, parallelizing function evaluations to reduce convergence time is also of utmost importance. Further, rigorous analysis is desired to establish convergence of BO algorithms to the global optimum. To address the aforementioned desiderata, the goal of the present work is to develop a BO method that entails the least tuning efforts, accommodates parallel operation, and enjoys convergence guarantees.

A. Related Works

Prior art is outlined next to contextualize our contributions.

Ensemble BO: Several choices are available for the surrogate model, acquisition function, and acquisition optimizer for BO [4]. Without prior knowledge of the problem at hand, combining the merits of different options can intuitively robustify performance. As pointed out in the 2020 black-box optimization challenge, ensembling methods can empirically boost BO performance for hyperparameter tuning [6]. In a broader sense, the ensemble rule has been applied to BO in different contexts, including high-dimensional input [7], and meta learning [8]. In the basic BO setup, combining acquisition functions has been explored for a single GP-based surrogate model in a principled way [9], [10]. The complementary setting of an ensemble of (GP) surrogate models with a given acquisition function has *not* been touched upon.

Thompson Sampling (TS) and Regret Analysis for BO: Since its invention by [11], TS has not received much attention in the bandit community until the past decade that its empirical success [12] and theoretical guarantees [13] have been well documented. In the context of BO, TS has been recently explored under different settings, including high-dimensionality [14], inputs with categorical variables [15], [16], as well as distributed learning [17], [18]. Without additional design parameters, TS is very attractive for automated machine learning. Convergence of TS for BO has been recently established using regret analysis

both in the Bayesian [13], [17], and in the frequentist setting [19], [20]. Although TS has been investigated with a mixture prior for linear bandits [21], its counterpart in BO with the associated regret analysis has not been studied so far.

Parallel BO: To reduce convergence time of BO approaches, parallel function evaluations at distributed computing resources is well motivated. Coupled with upper confidence bound [22] and expected improvement [23] based acquisition rules, this parallel operation typically relies on additional hyperparameters or selection rules to ensure the diversity of query points at different locations. On the other hand, TS-based parallel processing necessitates no additional design as in the sequential setting [18], and enjoys rigorous convergence guarantees [17]. Moreover, parallel BO has also been investigated for input spaces with high dimensions [7] as well as categorical variables [15].

Kernel Selection for GPs: Discovery of the form of the kernel function has been considered for conventional GP learning; see, e.g., [24], [25], [26], [27]. These approaches usually operate in the batch mode and rely on a large number of samples, thus rendering them inapplicable for BO where data are not only acquired online, but also scarce due to the expensive evaluation cost. While an online kernel selection scheme has been put forth for prediction-oriented tasks using a candidate of GP models [28], it entails additional design of the acquisition function before being applied to the BO context. How to automatically select the kernel function for the GP model in BO is still unexplored.

B. Contributions

Relative to the aforementioned previous works, the contributions of this work are summarized in the following four aspects.

- c1) Rather than a single GP surrogate model with a preselected kernel function for BO in previous works, an ensemble (E) of GPs is leveraged here to adaptively select the fitted model for the sought function by adjusting the per-GP weight on-the-fly. Capitalizing on the random feature (RF) based approximation per GP, acquisition of the next query input is facilitated by TS with scalability and no additional design parameters.
- c2) The resulting EGP-TS approach readily accommodates parallel function evaluation (a)synchronously.
- c3) Convergence of the novel EGP-TS approach to the global maximum is established by *sublinear* Bayesian regret for both the sequential and parallel settings.
- c4) Tests on synthetic functions and real-world applications, including hyperparameter tuning for three machine learning models and robot pushing tasks, demonstrate the merits of EGP-TS relative to the single GP-based TS, and alternative ensemble approaches.

Relation With [28]: The EGP function model has been considered in our previous work [28] for supervised learning tasks. However, its adaptation to the BO context here is novel and well motivated for the purpose of kernel selection that is important in practice. Coping with limited data in BO, this work differs from [28] in the following directions.

- i) Unlike [28] that relies on a large dataset of *passively* labelled samples, the novel EGP-based BO entails extra design of acquisition functions, which select query points *actively*. Two novel EGP-based acquistion functions are devised and tested, namely, EGP-TS and EGP-EI.
- ii) Although random feature-based approximation has been used also by [28], it serves a different purpose here. In [28], where the number of samples is large, the RF approximation alleviates the computational complexity of updating the GP model; whereas in the current BO context with limited labelled data, RFs are motivated to conduct function sampling with scalability in the TS-based acquisition function.
- iii) An extra weight and model *reinitializaiton* is needed each time the kernel hyperparameters are updated using all data acquired (cf. lines 10–15 in Algorithm 1).
- iv) Building on the novel EGP-TS approach, Bayesian regret analysis has been conducted to guarantee convergence to the global optimum. The analysis is novel and nontrivial to deal with the additional challenge brought by the EGP prior (cf. the proof sketch following Theorem 1).

Notation: Scalars are denoted by lowercase, column vectors by bold lowercase, and matrices by bold uppercase fonts. Superscripts $^\top$ and $^{-1}$ denote transpose, and matrix inverse, respectively; while $\mathbf{0}_N$ stands for the $N \times 1$ all-zero vector; $\mathbf{I}_N$ for the $N \times N$ identity matrix, and $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \mathbf{K})$ for the probability density function (pdf) of a Gaussian random vector $\mathbf{x}$ with mean $\boldsymbol{\mu}$, and covariance $\mathbf{K}$.

II. Preliminaries

Consider the following optimization problem

$$\mathbf{x}_* = \arg\max_{\mathbf{x}\in\mathcal{X}} \; f(\mathbf{x}), \tag{1}$$

where $\mathcal{X}$ is the feasible set for the $d \times 1$ optimization variable $\mathbf{x}$, and the objective $f(\mathbf{x})$ is *black-box* with analytic expression unavailable and is often *expensive* to evaluate. This mathematical abstraction characterizes a variety of application domains. When tuning hyperparameters of machine learning models with $\mathbf{x}$ collecting the hyperparameters, the mapping to the validation accuracy $f(\mathbf{x})$ is not available in closed form, and each evaluation is computationally demanding especially for deep neural networks and large data sizes [1]. For example, it takes 4 days to train BERT-large on 64 TPUs [29]. The lack of analytic expression discourages one from leveraging conventional gradient-based solvers to find $\mathbf{x}_*$. Exhaustive enumeration is also inapplicable given the expensive evaluation cost. Fortunately, BO offers a theoretically elegant solution by judiciously selecting query pairs for a given evaluation budget [4], [5].

In short, BO relies on a statistical surrogate model to extract information from the evaluated input-output pairs $\mathcal{D}_t := \{(\mathbf{x}_\tau, y_\tau)\}_{\tau=1}^t$ so as to select the next query input $\mathbf{x}_{t+1}$. Specifically, this procedure is implemented iteratively via two steps, that is: *s1)* Obtain $p(f(\mathbf{x})|\mathcal{D}_t)$ based on the surrogate model; and, *s2)* Find $\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}\in\mathcal{X}} \alpha(\mathbf{x}|\mathcal{D}_t)$ based on $p(f(\mathbf{x})|\mathcal{D}_t)$. Here, the so-termed acquisition function α, usually available in closed form, is designed to balance *exploration* with *exploitation* of the search space. There are multiple choices for both the surrogate model and the acquisition function, see, e.g., [4], [5]. Next, we

will outline the GP based surrogate model, which is the most widely used in BO, and TS for the acquisition function.

A. GP-Based Surrogate Model and TS for Acquisition

GPs are established nonparametric Bayesian approaches to learning functions in a sample-efficient manner [30]. This sample efficiency makes it extremely appealing for surrogate modeling in BO when function evaluations are expensive. Specifically, to learn $f(\cdot)$ that links the input $\mathbf{x}_\tau$ with the scalar output y_τ as $\mathbf{x}_\tau \to f(\mathbf{x}_\tau) \to y_\tau$, a GP prior is assumed on the unknown f as $f \sim \mathcal{GP}(0, \kappa(\mathbf{x}, \mathbf{x}'))$, where $\kappa(\cdot,\cdot)$ is a positive-definite kernel (covariance) function measuring pairwise similarity of any two inputs. Then, the joint prior pdf of function evaluations $\mathbf{f}_t := [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_t)]^\top$ at inputs $\mathbf{X}_t := [\mathbf{x}_1, \ldots, \mathbf{x}_t]^\top (\forall t)$ is Gaussian distributed as $p(\mathbf{f}_t|\mathbf{X}_t) = \mathcal{N}(\mathbf{f}_t; \mathbf{0}_t, \mathbf{K}_t)$, where $\mathbf{K}_t$ is a $t \times t$ covariance matrix whose (τ, τ')th entry is $[\mathbf{K}_t]_{\tau,\tau'} = \mathrm{cov}(f(\mathbf{x}_\tau), f(\mathbf{x}_{\tau'})) := \kappa(\mathbf{x}_\tau, \mathbf{x}_{\tau'})$. The value $f(\mathbf{x}_\tau)$ is linked with the noisy output y_τ via the per-datum likelihood $p(y_\tau|f(\mathbf{x}_\tau)) = \mathcal{N}(y_\tau; f(\mathbf{x}_\tau), \sigma_n^2)$, where σ_n^2 is the noise variance. The function posterior pdf after acquiring input-output pairs $\mathcal{D}_t$ is then obtained according to Bayes' rule as [30]

$$p(f(\mathbf{x})|\mathcal{D}_t) = \mathcal{N}(f(\mathbf{x}); \hat{f}_t(\mathbf{x}), \sigma_t^2(\mathbf{x})), \tag{2}$$

where the mean and variance are expressed via $\mathbf{k}_t(\mathbf{x}) := [\kappa(\mathbf{x}_1, \mathbf{x}) \ldots \kappa(\mathbf{x}_t, \mathbf{x})]^\top$ and $\mathbf{y}_t := [y_1 \ldots y_t]^\top$ as

$$\hat{f}_t(\mathbf{x}) = \mathbf{k}_t^\top(\mathbf{x})(\mathbf{K}_t + \sigma_n^2\mathbf{I}_t)^{-1}\mathbf{y}_t \tag{3a}$$

$$\sigma_t^2(\mathbf{x}) = \kappa(\mathbf{x}, \mathbf{x}) - \mathbf{k}_t^\top(\mathbf{x})(\mathbf{K}_t + \sigma_n^2\mathbf{I}_t)^{-1}\mathbf{k}_t(\mathbf{x}). \tag{3b}$$

With the function posterior pdf at hand, one readily selects the next evaluation point $\mathbf{x}_{t+1}$ using TS, where the function maximizer $\mathbf{x}_*$ in (1) is viewed as random. Specifically, TS selects the next query point by sampling from the posterior pdf $p(\mathbf{x}_*|\mathcal{D}_t) = \int p(\mathbf{x}_*|f(\mathbf{x}))p(f(\mathbf{x})|\mathcal{D}_t)df(\mathbf{x})$. Upon approximating this integral using a sample from the function posterior $p(f(\mathbf{x})|\mathcal{D}_t)$, the next query is found as

$$\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}\in\mathcal{X}} \tilde{f}_t(\mathbf{x}), \quad \tilde{f}_t(\mathbf{x}) \sim p(f(\mathbf{x})|\mathcal{D}_t)\,. \tag{4}$$

This random sampling procedure nicely balances exploration and exploitation. Implementation of sampling a function from the GP posterior $p(f(\mathbf{x})|\mathcal{D}_t)$ can be realized by discretizing the input space $\mathcal{X}$ [17], leveraging the RF based parametric approximant [10], [31], or more recently relying on sparse GP decomposition for efficiency [32].

Specifically, RF-based approximation leverages the spectral properties of (commonly used) stationary kernels to convert nonparametric GP learning into a parametric one, yielding [31], [33]

$$\check{f}(\mathbf{x}) = \boldsymbol{\phi}_\mathbf{v}^\top(\mathbf{x})\boldsymbol{\theta}, \quad \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta}; \mathbf{0}_{2D}, \sigma_\theta^2\mathbf{I}_{2D})$$
$$\boldsymbol{\phi}_\mathbf{v}(\mathbf{x}) := \frac{1}{\sqrt{D}}[\sin(\mathbf{v}_1^\top\mathbf{x}), \cos(\mathbf{v}_1^\top\mathbf{x}), \ldots, \sin(\mathbf{v}_D^\top\mathbf{x}), \cos(\mathbf{v}_D^\top\mathbf{x})]^\top, \tag{5}$$

where $\{\mathbf{v}_i\}_{i=1}^D$ are drawn i.i.d. from $\pi_\kappa(\mathbf{v})$ – kernel κ's normalized spectral density, and σ_θ^2 is the magnitude of κ (cf. Appendix A in the supplementary file, available online).

Henceforth, the function posterior pdf will be captured by $p(\boldsymbol{\theta}|\mathcal{D}_t) = \mathcal{N}(\boldsymbol{\theta}; \hat{\boldsymbol{\theta}}_t, \boldsymbol{\Sigma}_t)$, based on which TS will select the next query point as

$$\mathbf{x}_{t+1} = \arg\max_{\mathbf{x}\in\mathcal{X}} \boldsymbol{\phi}_\mathbf{v}^\top(\mathbf{x})\tilde{\boldsymbol{\theta}}_t, \quad \tilde{\boldsymbol{\theta}}_t \sim p(\boldsymbol{\theta}|\mathcal{D}_t). \tag{6}$$

It is worth mentioning that the mean $\hat{\boldsymbol{\theta}}_t$ and covariance matrix $\boldsymbol{\Sigma}_t$ can be updated efficiently in a recursive Bayes manner with the inclusion of each new (input, evaluation) pair.

III. Ensemble GPs With TS for BO

The performance of BO approaches depends critically on the chosen surrogate model. While most existing works rely on a single GP with *preselected* kernel form, we here leverage an ensemble (E) of M GPs, each relying on a kernel function selected from a given dictionary $\mathcal{K} := \{\kappa^1, \ldots, \kappa^M\}$. Set $\mathcal{K}$ can be constructed with kernels of different types and different hyperparameters. Specifically, each GP $m \in \mathcal{M} := \{1, \ldots, M\}$ places a unique prior on f as $f|m \sim \mathcal{GP}(0, \kappa^m(\mathbf{x}, \mathbf{x}'))$. Taking a weighted combination of the individual GP priors, yields the EGP prior of $f(\mathbf{x})$ given by

$$f(\mathbf{x}) \sim \sum_{m=1}^M w_0^m \mathcal{G}P(0, \kappa^m(\mathbf{x}, \mathbf{x}')), \quad \sum_{m=1}^M w_0^m = 1, \tag{7}$$

where $w_0^m := \Pr(i = m)$ is the prior probability that assesses the contribution of GP model m. Here, the *latent* variable i is introduced to indicate the contribution from GP m. While this non-Gaussian EGP prior (7) has been advocated for conventional prediction-oriented tasks in [28], the novelty here is its adaptation for BO along with the extra design step needed for query selection. Besides EGP for BO, we will employ TS-based acquisition function, which again, relies on sampling from $p(f(\mathbf{x})|\mathcal{D}_t)$. Coupled with the EGP prior (7), this posterior pdf is expressed via the sum-product rule as

$$p(f(\mathbf{x})|\mathcal{D}_t) = \sum_{m=1}^M \Pr(i = m|\mathcal{D}_t)p(f(\mathbf{x})|i = m, \mathcal{D}_t), \tag{8}$$

which is a mixture of posterior GPs with per-GP weight $w_t^m := \Pr(i = m|\mathcal{D}_t)$ given by

$$w_t^m \propto \Pr(i = m)p(\mathcal{D}_t|i = m) = w_0^m p(\mathcal{D}_t|i = m), \tag{9}$$

where $p(\mathcal{D}_t|i = m)$ is the marginal likelihood of the acquired data $\mathcal{D}_t$ for GP m. As with sampling from a Gaussian mixture (GM) distribution, drawing a sample $\tilde{f}_t(\mathbf{x})$ from (8) is implemented by the following two steps

$$m_t \sim \mathcal{C}AT(\mathcal{M}, \mathbf{w}_t), \quad \tilde{f}_t(\mathbf{x}) \sim p(f(\mathbf{x})|i = m_t, \mathcal{D}_t), \tag{10}$$

where $\mathcal{C}AT(\mathcal{M}, \mathbf{w}_t)$ represents a categorical distribution that assigns one of the values from $\mathcal{M}$ with probabilities $\mathbf{w}_t := [w_t^1, \ldots, w_t^M]^\top$.

There are several choices for the function sampling step (10) in the novel EGP-TS as mentioned in Section II-A. Here, we will

adopt the random feature (RF) based method since it can not only efficiently draw the function path $\tilde{f}_t(\mathrm{x})$ that is differentiable with respect to x, but also accommodate incremental updates of w_t^m (9) and $p(f(\mathrm{x})|i=m,\mathcal{D}_t)$ across iterates, as elaborated next.

A. RF-Based EGP-TS

When the kernels in the dictionary are shift-invariant, the RF vector $\phi_{\mathbf{v}}^m(\mathrm{x})$ per GP m can be formed via (??) by first drawing i.i.d. random vectors $\{\mathrm{v}_j^m\}_{j=1}^D$ from $\pi_{\bar{\kappa}}^m(\mathrm{v})$, which is the spectral density of the standardized kernel $\bar{\kappa}^m$. Let $\sigma_{\theta^m}^2$ be the kernel magnitude so that $\kappa^m=\sigma_{\theta^m}^2\bar{\kappa}^m$. The generative model for the sought function and the noisy output y per GP m can be characterized through the $2D\times 1$ vector $\boldsymbol{\theta}^m$ as

$$\begin{aligned} p(\boldsymbol{\theta}^m)&=\mathcal{N}(\boldsymbol{\theta}^m;\mathbf{0}_{2D},\sigma_{\theta^m}^2\mathbf{I}_{2D})\\ p(f(\mathrm{x}_t)|i=m,\boldsymbol{\theta}^m)&=\delta(f(\mathrm{x}_t)-\boldsymbol{\phi}_{\mathbf{v}}^{m\top}(\mathrm{x}_t)\boldsymbol{\theta}^m)\\ p(y_t|\boldsymbol{\theta}^m,\mathrm{x}_t)&=\mathcal{N}(y_t;\boldsymbol{\phi}_{\mathbf{v}}^{m\top}(\mathrm{x}_t)\boldsymbol{\theta}^m,\sigma_n^2)\,. \end{aligned}\tag{11}$$

This parametric form readily allows one to capture the function posterior pdf per GP m via $p(\boldsymbol{\theta}^m|\mathcal{D}_t)=\mathcal{N}(\boldsymbol{\theta}^m;\hat{\boldsymbol{\theta}}_t^m,\boldsymbol{\Sigma}_t^m)$, which together with the weight w_t^m, approximates the EGP function posterior (8). Next, we will describe how RF-based EGP-TS selects the next evaluation input x_{t+1}, and propagates the EGP function pdf by updating the set $\{w_t^m,\boldsymbol{\theta}_t^m,\boldsymbol{\Sigma}_t^m,m\in\mathcal{M}\}$ from slot to slot.

Given $\mathcal{D}_t$, acquisition of x_{t+1} is obtained as the maximizer of the RF-based function sample $\tilde{\tilde{f}}_t(\mathrm{x})$ based on (10), whose detailed implementation is given by

$$\begin{aligned} \mathrm{x}_{t+1}&=\arg\max_{\mathrm{x}\in\mathcal{X}}\ \tilde{\tilde{f}}_t(\mathrm{x}),\quad \text{where } \tilde{\tilde{f}}_t(\mathrm{x}):=\boldsymbol{\phi}_{\mathbf{v}}^{m_t\top}(\mathrm{x})\tilde{\boldsymbol{\theta}}_t\\ m_t&\sim\mathcal{CAT}(\mathcal{M},\mathrm{w}_t),\ \tilde{\boldsymbol{\theta}}_t\sim p(\boldsymbol{\theta}^{m_t}|\mathcal{D}_t), \end{aligned}\tag{12}$$

which can be solved using gradient-based solvers because the objective is available in an analytic form. Upon acquiring the evaluation output y_{t+1} for the selected input x_{t+1}, the updated weight $w_{t+1}^m:=\Pr(i=m|\mathcal{D}_t,\mathrm{x}_{t+1},y_{t+1})$ can be obtained per GP m via Bayes' rule as

$$\begin{aligned} w_{t+1}^m&=\frac{\Pr(i=m|\mathcal{D}_t,\mathrm{x}_{t+1})p(y_{t+1}|\mathrm{x}_{t+1},i=m,\mathcal{D}_t)}{p(y_{t+1}|\mathrm{x}_{t+1},\mathcal{D}_t)}\\ &=\frac{w_t^m\mathcal{N}\left(y_{t+1};\hat{y}_{t+1|t}^m,(\sigma_{t+1|t}^m)^2\right)}{\sum_{m'=1}^M w_t^{m'}\mathcal{N}\left(y_{t+1};\hat{y}_{t+1|t}^{m'},(\sigma_{t+1|t}^{m'})^2\right)}, \end{aligned}\tag{13}$$

where the sum-product rule allows one to obtain the per-GP predictive likelihood as $p(y_{t+1}|i=m,\mathcal{D}_t,\mathrm{x}_{t+1})=\int p(y_{t+1}|\boldsymbol{\theta}^m,\mathrm{x}_{t+1})p(\boldsymbol{\theta}^m|\mathcal{D}_t)d\boldsymbol{\theta}^m=\mathcal{N}(y_{t+1};\hat{y}_{t+1|t}^m,(\sigma_{t+1|t}^m)^2)$ with $\hat{y}_{t+1|t}^m=\boldsymbol{\phi}_{\mathbf{v}}^{m\top}(\mathrm{x}_{t+1})\hat{\boldsymbol{\theta}}_t^m$ and $(\sigma_{t+1|t}^m)^2=\boldsymbol{\phi}_{\mathbf{v}}^{m\top}(\mathrm{x}_{t+1})\boldsymbol{\Sigma}_t^m\boldsymbol{\phi}_{\mathbf{v}}^m(\mathrm{x}_{t+1})+\sigma_n^2$.

Further, the posterior pdf of $\boldsymbol{\theta}^m$ can be propagated in the recursive Bayes fashion as

$$\begin{aligned} p(\boldsymbol{\theta}^m|\mathcal{D}_{t+1})&=\frac{p(\boldsymbol{\theta}^m|\mathcal{D}_t)p(y_{t+1}|\boldsymbol{\theta}^m,\mathrm{x}_{t+1})}{p(y_{t+1}|\mathrm{x}_{t+1},i=m,\mathcal{D}_t)}\\ &=\mathcal{N}(\boldsymbol{\theta}^m;\hat{\boldsymbol{\theta}}_{t+1}^m,\boldsymbol{\Sigma}_{t+1}^m), \end{aligned}\tag{14}$$

where the updated mean $\hat{\boldsymbol{\theta}}_{t+1}^m$ and covariance matrix $\boldsymbol{\Sigma}_{t+1}^m$ are

$$\hat{\boldsymbol{\theta}}_{t+1}^m=\hat{\boldsymbol{\theta}}_t^m+(\sigma_{t+1|t}^m)^{-2}\boldsymbol{\Sigma}_t^m\boldsymbol{\phi}_{\mathbf{v}}^m(\mathrm{x}_{t+1})(y_{t+1}-\hat{y}_{t+1|t}^m)\tag{15a}$$

$$\boldsymbol{\Sigma}_{t+1}^m=\boldsymbol{\Sigma}_t^m-(\sigma_{t+1|t}^m)^{-2}\boldsymbol{\Sigma}_t^m\boldsymbol{\phi}_{\mathbf{v}}^m(\mathrm{x}_{t+1})\boldsymbol{\phi}_{\mathbf{v}}^{m\top}(\mathrm{x}_{t+1})\boldsymbol{\Sigma}_t^m.\tag{15b}$$

(Re)initialization: In accordance with existing BO implementations, EGP-TS initializes with a small number (t_0) of evaluation pairs $\mathcal{D}_{t_0}$ to obtain kernel hyperparameter estimate $\hat{\boldsymbol{\alpha}}_{t_0}^m$ per GP m by maximizing the marginal likelihood. The weight $w_{t_0}^m$ is then obtained via (9) using $\hat{\boldsymbol{\alpha}}_{t_0}^m$. As proceeding, the kernel hyperparameters per GP are updated every few iterations using all the acquired data, and subsequently the weights are reinitialized via the batch form (9) using the updated hyperparameters. Between updates of hyperparameters, EGP-TS leverages (13) and (14) to incrementally propagate the function posterior pdf. Please refer to Algorithm 1 for the detailed implementation of (sequential) EGP-TS.

B. Parallel EGP-TS

As with the single GP-based TS [17], EGP-TS can readily accommodate parallel implementation for both synchronous and asynchronous settings without extra design. Suppose there are K computing centers/workers that conduct function evaluations in parallel. In the *synchronous* setup, K query points are assigned for the workers to evaluate simultaneously by implementing (10) K times. After all workers obtain the evaluated outputs, the EGP function posterior is then updated using the K input-output pairs. As for the *asynchronous* case, whenever a worker finishes her/his job, the EGP posterior will be updated and the next evaluation point will be acquired. Note that the asynchronous setup is very similar to the sequential one except that multiple function evaluations are performed at the same time; see Algorithm 2 in the supplementary file, available online, for details. Algorithm 1 contains the implementation of synchronous parallel EGP-TS when $K>1$.

The following two remarks are in order.

Remark 1 (EGP With Other Acquisition Functions): Besides TS, the EGP surrogate model can be coupled with other existing single GP-based acquisition functions, including the well-known expected improvement (EI) [34] and upper confidence bound (UCB) [35]. The most direct implementation per iteration is to first draw the model index m_t based on the weights w_t as in (10), and then proceed with the conventional EI/UCB acquisition rule for GP m_t. Results for this preliminary EGP-EI are presented in Appendix E, available online. Instead of sampling one GP model per iteration, one could alternatively build on the GP mixture pdf to devise the EI or UCB based acquisition rule. Further investigation along this direction is deferred to our future agenda.

Remark 2 (Relation With Fully Bayesian GP-Based BO): When the dictionary consists of kernel functions of the same type, the EGP prior amounts to a pseudo Bayesian GP model, where the kernel hyperparameters are chosen from a finite set. This EGP-based pseudo Bayesian model achieves a "sweet spot" between the Bayesian and non-Bayesian treatment of GP hyperparameters, where the former entails specifying a reasonable

Algorithm 1: EGP-TS.

Input: Kernel dictionary $\mathcal{K}$, number D of RFs, number K of workers, and $w_0^m = 1/M \ \forall m$
Initialization:
Randomly evaluate t_0 points to obtain $\mathcal{D}_{t_0}$;
for $m = 1, 2, \ldots, M$ **do**
  Obtain kernel hyperparameters estimates $\hat{\alpha}_{t_0}^m$ by maximizing the marginal likelihood;
  Draw D random vectors $\{\mathrm{v}_i^m\}_{i=1}^D$ from $\pi_{\bar{\kappa}}^m(\mathrm{v})$ using $\hat{\alpha}_{t_0}^m$;
  Obtain $w_{t_0}^m$, $\hat{\theta}_{t_0}^m$, and $\Sigma_{t_0}^m$ based on(9) and(??);
end for
for $t = t_0, t_0 + 1, \ldots$ **do**
  if Reinitialization **then**
    for $m = 1, 2, \ldots, M$ **do**
    Obtain $\hat{\alpha}_t^m$ by marginal likelihood maximization using $\mathcal{D}_t$;
    Draw D random vectors $\{\mathrm{v}_i^m\}_{i=1}^D$ from $\pi_{\bar{\kappa}}^m(\mathrm{v})$ using $\hat{\alpha}_t^m$;
    Obtain w_t^m, $\hat{\theta}_t^m$, and Σ_t^m based on(9) and(??);
  end for
  end if
  for $k = 1, 2, \ldots, K$ **do**
    Sample m_t^k based on pmf w_t;
    Sample $\tilde{\theta}_t^k$ from $\mathcal{N}(\hat{\theta}_t^{m_t^k}, \Sigma_t^{m_t^k})$;
    Obtain $\mathrm{x}_{t+1}^k = \arg\max_{\mathrm{x}\in\mathcal{X}} \ \tilde{\theta}_t^{k\top} \phi^{m_t^k}(\mathrm{x})$;
    Evaluate x_{t+1}^k to obtain y_{t+1}^k;
  end for
  Update $\{w_{t+1}^m, \hat{\theta}_{t+1}^m, \Sigma_{t+1}^m\}_m$ with $\{\mathrm{x}_{t+1}^k, y_{t+1}^k\}_k$ based on(13) and (14);
  $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{\mathrm{x}_{t+1}^k, y_{t+1}^k\}_k$;
end for

prior and also needs demanding MCMC sampling. In addition, the proposed EGP-TS framework not only accommodates different types of kernels, but also enjoys the upcoming convergence guarantees relative to fully Bayesian GP-based BO.

IV. Bayesian Regret Analysis

To establish convergence of the proposed EGP-TS algorithm to the global optimum, analysis will be conducted via the notion of Bayesian regret over T slots, that is defined as

$$\mathcal{BR}(T) := \sum_{t=1}^{T} \mathbb{E}[f(\mathrm{x}_*) - f(\mathrm{x}_t)], \tag{16}$$

where the expectation is over all random quantities, including the function prior, the observations, and the sampling procedure. Unlike previous works that sample the function from a single GP prior [13], [17], here we draw f from the EGP prior (7) as

$$m_* \sim \mathcal{CAT}(\mathcal{M}, \mathrm{w}_0), \qquad f(\mathrm{x}) \sim \mathcal{GP}(0, \kappa^{m_*}(\mathrm{x}, \mathrm{x}')).$$

This EGP prior presents additional challenge to the regret analysis. Towards addressing this challenge, we will adapt the techniques in [21], where TS with a mixture prior is studied for linear bandits, but not in the BO context.

To proceed, we will need the following assumption and intermediate lemmas.

Assumption 1 (Smoothness of a GP Sample Path [36]): If $\mathrm{x} \in \mathcal{X} \subset [0,1]^d$ is compact and convex, there exist constants $a, b, L > 0$ such that for any $f(\mathrm{x}) \sim \mathcal{GP}(0, \kappa^m(\mathrm{x}, \mathrm{x}'))$

$$\Pr\left(\sup_{x_j} \left|\frac{\partial f(\mathrm{x})}{\partial x_j}\right| > L\right) \leq a e^{-(L/b)^2}, \forall j \in \{1, \ldots, d\}.$$

Lemma 1 (Maximum Information Gain (MIG) [35]): Let $I^m(f; \mathrm{y}_{\mathcal{A}})$ represent the Shannon mutual information one can gain about the function $f \sim \mathcal{GP}(0, \kappa^m)$ using observations $\mathrm{y}_{\mathcal{A}}$ evaluated at finite subset $\mathcal{A} := \{\mathrm{x}_1, \ldots, \mathrm{x}_T\} \subset \mathcal{X}$. For any $m \in \mathcal{M}$, the MIG for commonly used kernels can be upper bounded by

$$\gamma_T := \max_{\mathcal{A}\subset\mathcal{X}, |\mathcal{A}|=T} \max_{m\in\mathcal{M}} I^m(f; \mathrm{y}_{\mathcal{A}}) \ \leq \tilde{\mathcal{O}}(T^c), \ 0 \leq c < 1 \,,$$

where $\tilde{O}$ ignores polylog factors.

Lemma 2 (Ratio of Posterior Variances [22]): Let $\mathrm{y}_{\mathcal{A}}$ and $\mathrm{y}_{\mathcal{B}}$ denote the observations when evaluating $f \sim \mathcal{GP}(0, \kappa^m)$ at $\mathcal{A}$ and $\mathcal{B}$, which are finite subsets of $\mathcal{X}$. With $\sigma_{\mathcal{A}}^m(\mathrm{x})$ and $\sigma_{\mathcal{A}\cup\mathcal{B}}^m(\mathrm{x})$ representing the posterior standard deviation of the GP conditioned on $\mathcal{A}$ and $\mathcal{A} \cup \mathcal{B}$, there exists $\rho_K \geq 1$ so that the following holds for $|\mathcal{B}| < K$

$$(\sigma_{\mathcal{A}}^m(\mathrm{x}))^2 \leq \rho_K \, (\sigma_{\mathcal{A}\cup\mathcal{B}}^m(\mathrm{x}))^2 \,, \ \forall \mathrm{x} \in \mathcal{X}, m \in \mathcal{M} \,.$$

As stated in [35], Assumption 1 is satisfied for various commonly used stationary kernels that are four times differentiable, including Gaussian kernels and Matérn ones with parameter $\nu > 2$, which implicitly allows EGP-TS to draw functions with scalability using RFs as in the preceding section. The MIG in Lemma 1 plays an important role in the regret bound. It is an information-theoretic measure quantifying the statistical difficulty of BO [13], [35]. Lemma 2 will be useful in deriving the regret in the parallel setup. After making these comments, we are ready to present a Bayesian regret upper bound pertinent to EGP-TS in the sequential setting.

Theorem 1: Under Assumption 1, the cumulative Bayesian regret (16) of EGP-TS over T slots, is bounded by

$$\mathcal{BR}(T) \leq c_1 \sqrt{M T^{c+1} \log T} + 2\sigma_n \sqrt{M T \log T} + c_2,$$

where the constants $c_1 := (2 + \sqrt{d})(2/\log(1 + \sigma_n^{-2})^{1/2}$ and $c_2 := 6\,\mathrm{MB} + (\pi^2 d)/6 + \sqrt{2\pi} M/12$ (B is a constant given in Lemma 3 in Appendix B, available online) are not dependent on T.

Proof Sketch: The detailed proof of Theorem 1 is deferred to Appendix B, available online. The key step in the proof builds on the connection with UCB based approaches, that is manifested via decomposing the Bayesian regret (16) as

$$\mathcal{BR}(T) = \underbrace{\sum_{t=1}^{T} \mathbb{E}[f(\mathrm{x}_*) - U_t^{m_*}(\mathrm{x}_*)]}_{\mathcal{BR}_1(T)} + \underbrace{\sum_{t=1}^{T} \mathbb{E}[U_t^{m_t}(\mathrm{x}_t) - f(\mathrm{x}_t)]}_{\mathcal{BR}_2(T)},$$

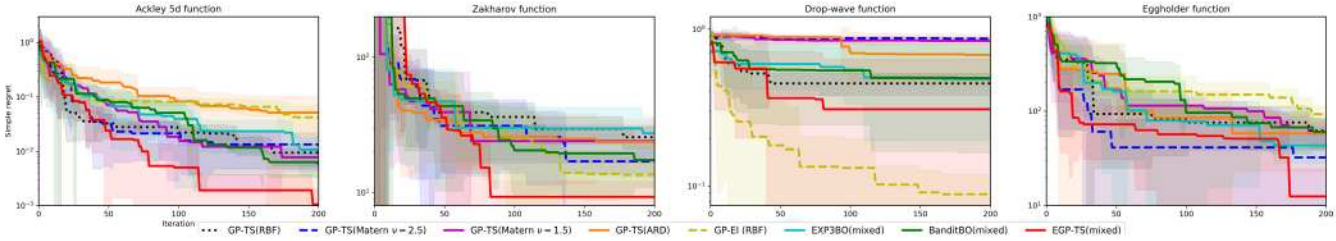

Fig. 1. Simple regret on `Ackley-5 d`, `Zakharov`, `DropWave` and `Eggholder` function (from left to right). Dictionary has 4 kernels with distinct forms: RBF with(out) ARD and Matérn with $\nu = 3/2, 5/2$.

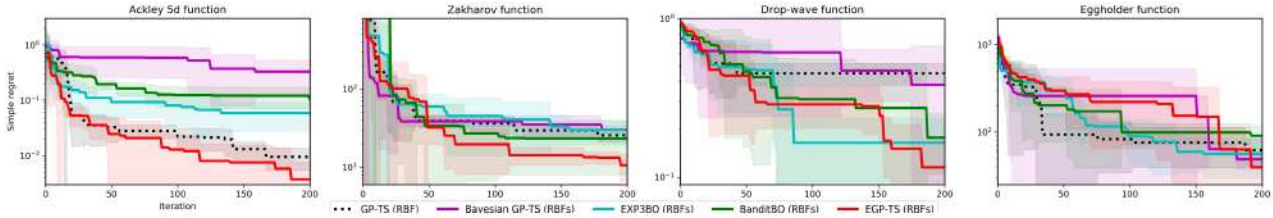

Fig. 2. Simple regret on `Ackley-5 d`, `Zakharov`, `DropWave` and `Eggholder` function (from left to right) using RBF kernels. Dictionary has 11 RBF kernels with lengthscales given by $\{10^c\}_{c=-4}^{6}$.

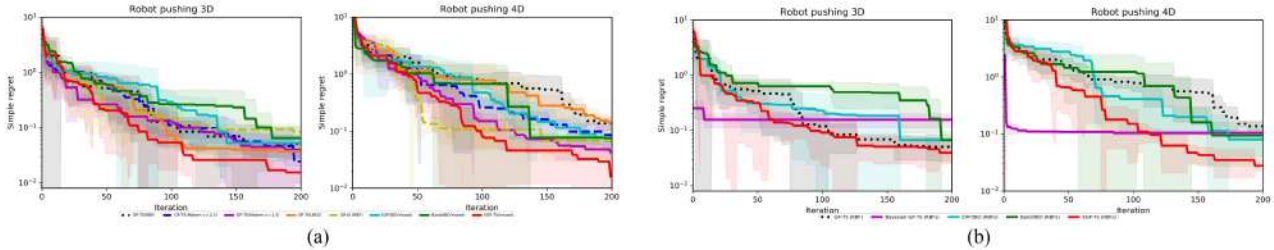

Fig. 3. Simple regret on `Robot pushing 3D` and `Robot pushing 4D` tasks with dictionary (a) that has 4 kernels with distinct forms: RBF with(out) ARD and Matérn with $\nu = 3/2, 5/2$; and (b) that has 11 RBF kernels with characteristic lengthscales given by $\{10^c\}_{c=-4}^{6}$.

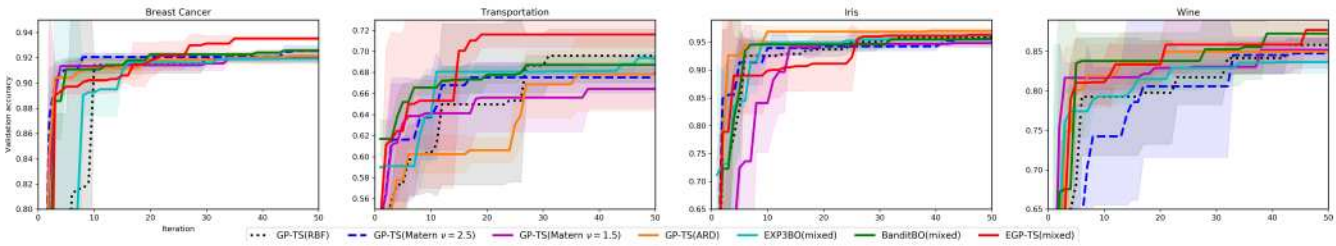

Fig. 4. The best validation accuracy (so far) versus the number of function evaluations on `Breast Cancer`, `Transportation`, `Iris`, and `Wine` datasets (from left to right) for the NN hyperparameter tuning task. Dictionary has 4 kernels with distinct forms: RBF with(out) ARD and Matérn with $\nu = 3/2, 5/2$.

where $U_t^m(\mathrm{x}) := \mu_{t-1}^m(\mathrm{x}) + \beta_t^{1/2}\sigma_{t-1}^m(\mathrm{x})$ with β_t specified by (17) in Appendix B, available online, is a UCB for $f(\mathrm{x})$ under GP m. This decomposition of $\mathcal{BR}(T)$ holds since $\{m_t, \mathrm{x}_t\}$ and $\{m_*, \mathrm{x}_*\}$ are i.i.d. and $U_t^m(\mathrm{x})$ is *deterministic* conditioned on $\mathcal{D}_{t-1}$, yielding [13], [21],

$$\mathbb{E}_{t-1}[U_t^{m_t}(\mathrm{x}_t)] = \mathbb{E}_{t-1}[U_t^{m_*}(\mathrm{x}_*)], \quad \forall t\,.$$

Then, the Bayesian regret bound of EGP-TS can be established by upper bounding $\mathcal{B}R_1(T)$ and $\mathcal{B}R_2(T)$. Since $f \sim \mathcal{GP}(0, \kappa^{m_*})$, the former can be conveniently bounded based on related works that rely on a single GP [13], [17]. Specifically, $\mathcal{B}R_1(T)$ is proved to be upper bounded by a constant, because the probability that $f(\mathrm{x}_*)$ is larger than $U_t^{m_*}(\mathrm{x}_*)$ across all the slots is low [17].

To further bound $\mathcal{B}R_2(T)$ involving the extra latent variable m_t sampled from the EGP posterior (cf. (10)), we adapt the technique in [21] that constructs a confidence set $\mathcal{C}_t$ for the latent variable such that $m_* \in \mathcal{C}_t$ holds with high probability; see Lemma 4 in Appendix B, available online. It turns out that $\mathcal{B}R_2(T)$ can also be bounded by the sum of posterior standard deviations, which further yields the upper bound given by the MIG along the lines of [35].

The proof of Theorem 1 in Appendix B, available online, involves an additional discretization step of $\mathcal{X}$ per step t, in order to cope with the continuous feasible set $\mathcal{X}$.

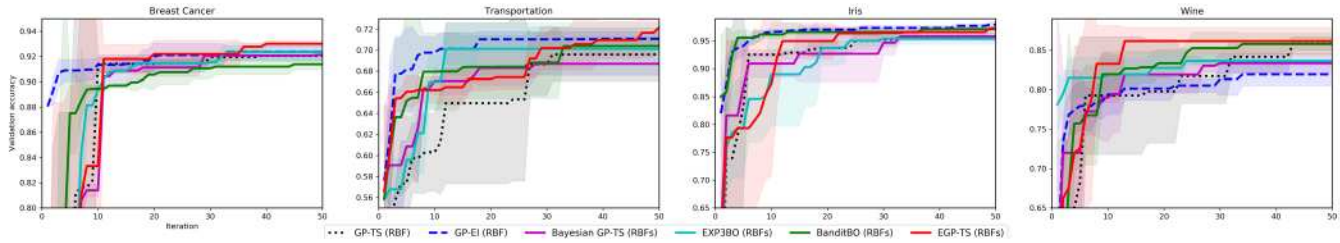

Fig. 5. The best validation accuracy (so far) versus the number of function evaluations on `Breast Cancer`, `Transportation`, `Iris`, and `Wine` datasets (from left to right) for the NN hyperparameter tuning task. Dictionary has 11 RBF kernels with characteristic lengthscales given by $\{10^c\}_{c=-4}^{6}$.

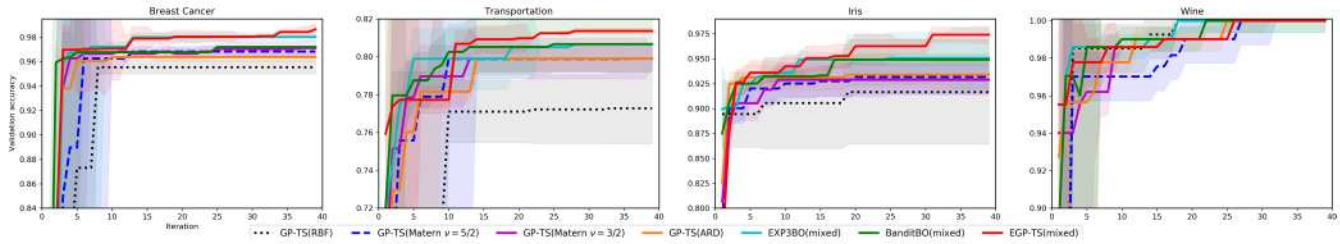

Fig. 6. The best validation accuracy (so far) versus the number of function evaluations on `Breast Cancer`, `Transportation`, `Iris`, and `Wine` datasets (from left to right) for the SVM hyperparameter tuning task. Dictionary has 4 kernels with distinct forms: RBF with(out) ARD and Matérn with $\nu = 3/2, 5/2$.

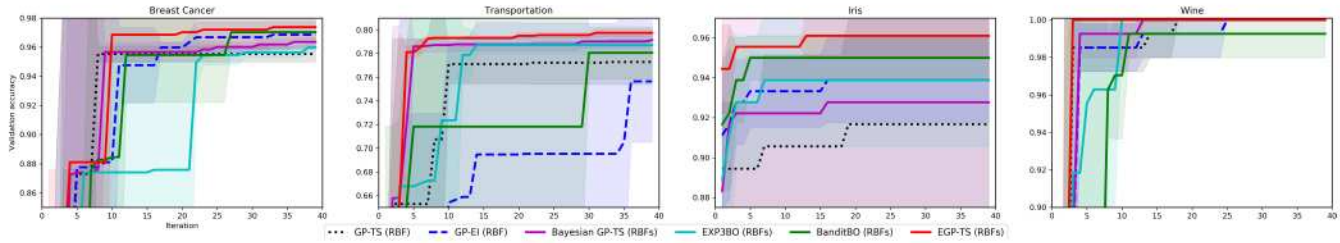

Fig. 7. The best validation accuracy (so far) versus the number of function evaluations on `Breast Cancer`, `Transportation`, `Iris`, and `Wine` datasets (from left to right) for the SVM hyperparameter tuning task. Dictionary has 11 RBF kernels with characteristic lengthscales given by $\{10^c\}_{c=-4}^{6}$.

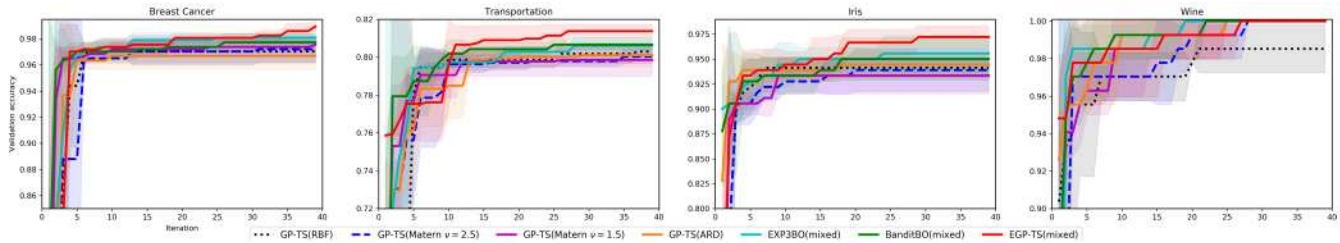

Fig. 8. The best validation accuracy (so far) versus the number of function evaluations on `Breast Cancer`, `Transportation`, `Iris`, and `Wine` datasets (from left to right) for the GradientBoosting hyperparameter tuning task. Dictionary has 4 kernels with distinct forms: RBF with(out) ARD and Matérn with $\nu = 3/2, 5/2$.

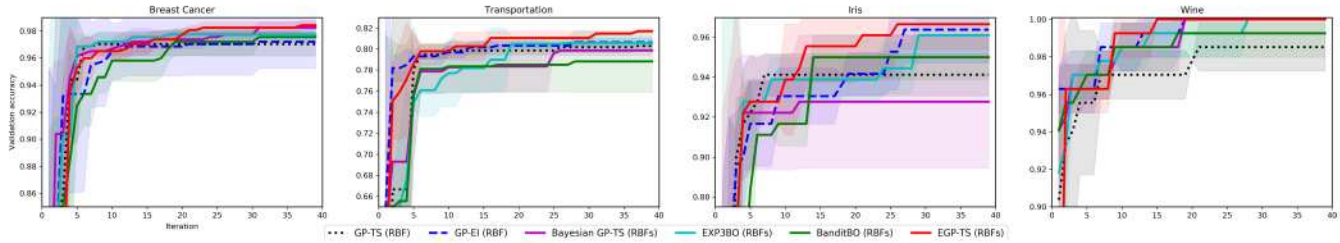

Fig. 9. The best validation accuracy (so far) versus the number of function evaluations on `Breast Cancer`, `Transportation`, `Iris`, and `Wine` datasets (from left to right) for the GradientBoosting hyperparameter tuning task. Dictionary has 11 RBF kernels with characteristic lengthscales given by $\{10^c\}_{c=-4}^{6}$.

The following two theorems further establish the cumulative Bayesian regret bounds of parallel EGP-TS in the asynchronous and synchronous settings, whose proofs are deferred to Appendix C– D, available online.

Theorem 2 (Asynchronously Parallel Setting): For K workers conducting parallel function evaluations asynchronously, EGP-TS under Assumption 1 incurs the following cumulative Bayesian regret over T function evaluations

$$\mathcal{BR}^{\text{asy}}(T) \leq c_1\sqrt{\rho_K M T^{c+1}\log T} + 2\sigma_n\sqrt{MT\log T} + c_2.$$

Theorem 3 (Synchronously Parallel Setting): For K workers performing T function evaluations synchronously, the cumulative Bayesian regret of EGP-TS under Assumption 1 is bounded by

$$\begin{aligned}\mathcal{BR}^{\text{syn}}(T) \leq\ & (K-1)\sqrt{d\log(K-1)} + 2\sigma_n\sqrt{MT\log T}\\ & + c_2 + c_3\sqrt{\rho_K M T^{c+1}\log T}\\ & + c_4\sqrt{MT^{c+1}\log(T+K-1)},\end{aligned}$$

where the two constants are given by $c_3 := 2(2/\log(1+\sigma_n^{-2})^{1/2}$, and $c_4 := (2\,d/\log(1+\sigma_n^{-2})^{1/2}$.

The first term of the regret bound in Theorem 2 is $\sqrt{\rho_K}$ times its counterpart in Theorem 1 for the sequential setting. It shall be easily verified that Bayesian regret bounds of parallel EGP-TS become equivalent to that in the sequential setting when $K = 1$ with $\rho_1 = 1$. Note that the regret bounds for parallel EGP-TS here are for the number of evaluations, that will typically exceed the bound in the sequential setup. This can be certainly the other way around if the evaluation time is of interest [17]. In all the three settings, the cumulative Bayesian regret bounds of EGP-TS boil down to $\mathcal{O}(\sqrt{MT^{c+1}\log T})$ after ignoring irrelevant constants, which is *sublinear* in the number of evaluations when $0 \leq c < 1$. Hence, EGP-TS enjoys the diminishing average regret per evaluation as T grows, hereby establishing convergence to the global optimum.

V. Numerical Tests

In this section, the performance of the proposed EGP-TS will be tested on a set of benchmark synthetic functions, two robotic tasks, and the hyperparameter tuning tasks of three machine learning models. The competing baselines are GP-EI [34], the default method for many traditional BO problems, and TS-based methods, including GP-TS with a preselected kernel type, fully Bayesian GP-TS, as well as two *ensemble* approaches, which are BanditBO [15], and EXP3BO [16]. It is worth mentioning that the latter two, combining multi-armed bandits and BO, are originally designed for inputs with categorical variables, but are adapted as ensemble methods here with each "arm" referring to a GP model with the same input variables.

The kernel hyperparameters per GP for all the TS-based methods other than fully Bayesian GP-TS are obtained by maximizing the marginal likelihood using `sklearn`. GP-EI is implemented using `BoTorch` with the ARD kernel, whose hyperparameters are refitted each iteration. The fully Bayesian GP model hinges on a pre-defined kernel type where the kernel hyperparameters are assumed to be random variables. In the present work, the RBF kernel is considered and a uniform prior is assumed for the amplitude σ_θ^2, characteristic lengthscale and noise variance σ_n^2, within intervals $[1, 100]$, $[10^{-3}, 10^3]$ and [0.1,0.3] respectively. The fully Bayesian GP-TS proceeds by first drawing a sample of the kernel hyperparameters using `GPyTorch` and `Pyro` Python packages, based on which function sampling is conducted. Existing kernel selection methods for conventional GP learning operate in batch mode using a large number of samples, hence being not suitable for the low-data BO setting. For initialization in all the methods, the first 10 evaluation pairs are randomly selected and used to obtain the kernel hyperarameters per GP by maximizing the marginal likelihood. In EGP-TS, the per-GP prior weight is set as uniform, i.e., $w_0^m = 1/M\ \forall m$. Unless stated otherwise, the hyperparameters are refitted every 50 iterations for EGP-TS, and every iteration for the rest of the baselines. RF approximation with 50 spectral features is leveraged by all the TS-based approaches for fairness in comparison. All the experiments are repeated 10 times, where the average performance and the standard deviation of all competing approaches are reported.

Additional results concerning ablation studies of the EGP-TS approach, runtime comparison, and the parallel setting are deferred to Appendix B in the supplementary file, available online.

A. Tests on Synthetic Functions

We tested the competing methods on a suite of standard synthethic functions for BO, including `Ackley-5 d`, `Zakharov`, `Drop-wave`, as well as `Eggholder`, where the latter two are challenging functions with many local optima. The performance metric per slot t is given by the simple regret (SR), defined as $\mathcal{S}R(t) := f(\mathbf{x}_*) - \max_{\tau\in\{1,\ldots,t\}} f(\mathbf{x}_\tau)$. First, to explore the effect of the kernel functions in the (E)GP model, we tested GP-TS with the kernel function being RBF with and without auto-relevance determination (ARD), and Matérn kernels with $\nu = 3/2, 5/2$. For all the ensemble methods, the kernel dictionary is comprised of the aforementioned four kernel functions. It is evident from Fig. 1 that the form of kernel function plays an important role in the performance of GP-TS. Combining different kernel functions, EGP-TS not only yields substantially improved performance relative to GP-TS counterparts, but also requires the least design efforts on the choice of the kernel function. In addition, EGP-TS achieves lower simple regret than BanditBO and EXP3BO. Although GP-EI is superior to GP-TS baselines on `Zakharov` function, EGP-TS yields better performance relative to the former, what demonstrates the benefit of ensembling GP models. Upon fixing the kernel type as RBF without ARD and constructing the dictionary as 11 RBF functions with lengthscales given by $\{10^c\}_{c=-4}^{6}$, EGP-TS is also compared with fully Bayesian GP-based TS in addition to the aforementioned baselines. Still, EGP-TS outperforms all competitors as shown in Fig. 2.

B. Robot Pushing Tasks

The second experiment concerns a practical task in robotics, where a robot adjusts its action so as to push an object towards a given goal location. By minimizing the distance between the

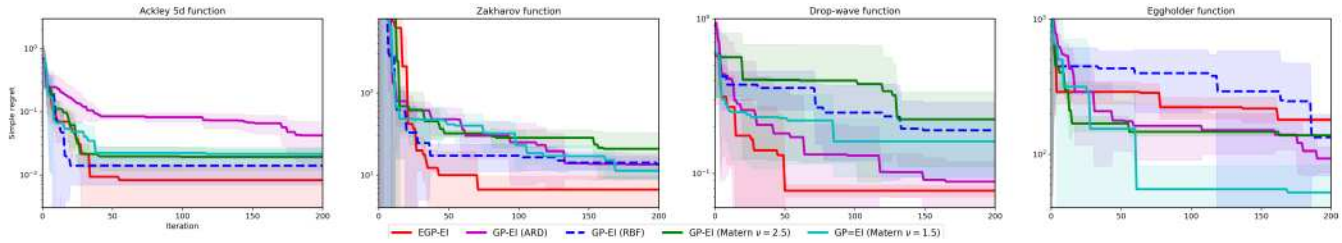

Fig. 10. Simple regret of EGP-EI versus GP-EI with a preselected kernel on `Ackley-5 d`, `Zakharov`, `DropWave` and `Eggholder` function (from left to right). Dictionary has 4 kernels with distinct forms: RBF with(out) ARD and Matérn with $\nu = 3/2, 5/2$.

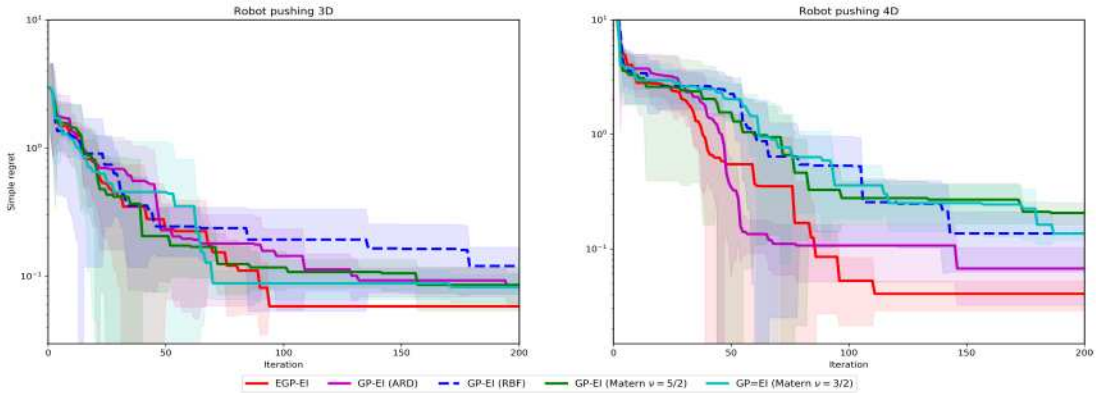

Fig. 11. Simple regret of EGP-EI versus GP-EI with a preselected kernel on `Robot pushing 3D` and `Robot pushing 3D` tasks (from left to right). Dictionary has 4 kernels with distinct forms: RBF with(out) ARD and Matérn with $\nu = 3/2, 5/2$.

target location and the end position of the pushed object, we tested two scenarios with 3 and 4 input variables following [37]. The former optimizes the 2-D position of the robot and the push duration, and the latter entails optimizing an additional push angle. We used the github codes[1] from [37] to generate the movement of the object pushed by the robot. Each scenario was repeated for 10 randomly selected goal locations, and the average performance of the competing methods are depicted in Fig. 3. Adaptively selecting kernel function from the dictionary with 4 distinct forms (that is, RBF with(out) ARD, and Matérn with $\nu = 3/2, 5/2$), the proposed EGP-TS outperforms all the competitors, including GP-EI, GP-TS with a preslected kernel, and the other two ensemble methods, as shown in Fig. 3(a). The superior performance of EGP-TS when the kernel function is fixed as RBF is also shown in Fig. 3(b), what is in accordance with Fig. 2. It is worth highlighting that EGP-TS not only outperforms fully Bayesian GP-TS in simple regret, but also runs much faster.

C. Hyperparameter Tuning Tasks

The last test deals with hyperparameter tuning tasks for three classification models, including a 2-layer FNN with ReLU activation function, support vector machine (SVM), and gradient boosting (GB). Note that although the FNN architecture does not yield the state-of-the-art classification performance, it suffices to be used to evaluate different BO methods. We tested all the competing baselines on `Breast cancer` [38], `Iris` [39], `Transportation` [40], as well as `Wine` [41] datasets. For all the datasets, 70% of the data are used as the training set, and the remaining are used as the validation set based on which the classification accuracy is calculated. The hyperparameters of the FNN consist of the number of neurons per layer, the learning rate, and the batch size. As for SVM, the values of C and γ are to be tuned. For GB, the hyperparameters include the learning rate, subsample ratio, and the ratio of maximum features. Table I summarizes the feasible values of the hyperparameters of the three methods. For each set of hyperparameters, the evaluated validation accuracy is obtained as the average of 10 independent

TABLE I
FEASIBLE VALUES OF THE HYPERPARAMETERS FOR DIFFERENT CLASSIFICATION MODELS

Model	Hyperparameter	Range
FNN	No. of neurons at Layer 1	[2,100]
	No. of neurons at Layer 2	[2,100]
	Learning rate	$[10^{-6}, 10^{-1}]$
	Batch size	$[2^2, 2^6]$
SVM	C	[0.1, 100]
	γ	[0.0001, 10]
GB	Learning rate	[0.1, 10]
	Subsample ratio	[0.1, 0.99]
	Max. features ratio	[0.1, 0.99]

[1]https://github.com/zi-w/Max-value-Entropy-Search

runs on a given dataset. In FNN training, the number of epochs is chosen to be 20 and the optimizer used is Adam.

Figs. 4, 5, 6, 7, 8, and 9 depict the best validation accuracy (so far) of the competing methods versus the number of iterations for these three classification models. Apparently, EGP-TS with different kernel types or RBF kernels of different lengthscales is shown to outperform the competitors in most of the cases, demonstrating the robustness of the EGP model across tasks.

D. Preliminary Results for EGP-EI

Here, we couple the EGP surrogate with the EI acquisition rule [34], yielding the novel EGP-EI approach. In line with the proposed EGP-TS, one could first select a GP model by random sampling based on the weights w_t, and then implement the EI acquisition function based on the chosen GP model. To benchmark the performance of this advocated EGP-EI, comparison has been made relative to GP-EI with a preselected kernel function. We use BoTorch to implement both EGP-EI and GP-EI with kernel hyperparameters updated every iteration and without RF approximation. As shown in Figs. 10 and 11, EGP-EI outperforms GP-EI in three out of the four synthetic functions and both of the robotic tasks – what demonstrates the benefits accompanied with the more expressive EGP model for the EI acquisition function. Rather than sampling a single GP from the EGP, future work includes investigation of the EI rule based on the GP mixture function model (cf. Remark 1).

VI. Conclusion

This work introduced a non-Gaussian EGP prior with adaptive kernel selection for the sought black-box function in BO. Capitalizing on the RF approximation per GP, acquisition of the subsequent query point is effected via TS, which bypasses the need for design parameters and can readily afford parallel implementation. Convergence of the proposed EGP-TS algorithm has been established by sublinear cumulative Bayesian regret in both the sequential and parallel settings. Numerical tests demonstrated the merits of EGP-TS relative to existing alternatives. Future work includes investigation of other acquisition functions based on the novel EGP surrogate model, as well as analysis via the notion of frequentist regret.

VII. Proofs

A. Proof of Theorem 1

Before performing the Bayesian regret analysis for EGP-TS, the following lemma will be first presented.

Lemma 3 (Supremum of a GP Sample Path [42]): If $f \sim \mathcal{GP}(0, \kappa^m)$ is a continuous sample path for any $m \in \mathcal{M}$, then $\mathbb{E}[\|f\|_\infty] = B < \infty$, and further

$$\max_{\mathrm{x}\in\mathcal{X}} |f(\mathrm{x}_*) - f(\mathrm{x})| \leq 2B \; .$$

Lemma 3 holds when kernels are twice differentiable - what is readily satisfied under Assumption 1.

To bound the cumulative Bayesian regret of EGP-TS, we will rely on its link with the corresponding upper confidence bound algorithm in [13]. Conditioned on GP model m and past data $\mathcal{D}_{t-1}$, the high probability upper confidence bound for $f(\mathrm{x})$ is given by $U_t^m(\mathrm{x}) := \mu_{t-1}^m(\mathrm{x}) + \beta_t^{1/2}\sigma_{t-1}^m(\mathrm{x})$, where $\beta_t := 2\log(t^2|\mathcal{X}_t|)$. Here, $\mathcal{X}_t$ is obtained by discretizing each dimension of $\mathcal{X}$ using $n_t = t^2 dab\sqrt{\pi}$ equally spaced grids. Thus, $|\mathcal{X}_t| = (n_t)^d$, and

$$\beta_t = 4(d+1)\log\, t + 2d\log(dab\sqrt{\pi}) \approx d\log t \; . \tag{17}$$

With $[\mathrm{x}]_t$ representing the closest point to x in $\mathcal{X}_t$, it can be easily verified that

$$\|\mathrm{x} - [\mathrm{x}]_t\|_1 \leq d/n_t, \;\; \forall \mathrm{x} \in \mathcal{X}. \tag{18}$$

Consider next the following decomposition

$$\begin{aligned}
\mathcal{BR}(T) &:= \sum_{t=1}^{T} \mathbb{E}[f(\mathrm{x}_*) - f(\mathrm{x}_t)] \\
&\overset{(a)}{=} \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[f(\mathrm{x}_*) - f([\mathrm{x}_*]_t)\right]}_{:=A_1} + \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[f([\mathrm{x}_*]_t - U_t^{m_*}([\mathrm{x}_*]_t)\right]}_{:=A_2} \\
&\quad + \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[U_t^{m_*}([\mathrm{x}_*]_t) - U_t^{m_t}([\mathrm{x}_t]_t)\right]}_{:=A_3} \\
&\quad + \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[U_t^{m_t}([\mathrm{x}_t]_t) - f([\mathrm{x}_t]_t)\right]}_{:=A_4} \\
&\quad + \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[f([\mathrm{x}_t]_t) - f(\mathrm{x}_t)\right]}_{:=A_5} \; .
\end{aligned} \tag{19}$$

Since $\{\mathrm{x}_t, m_t\}$ and $\{\mathrm{x}_*, m_*\}$ are identically distributed given $\mathcal{D}_{t-1}$, the fact that $U_t^m(\mathrm{x})$ is a *deterministic* function of $\mathcal{D}_{t-1}$ yields $A_3 = 0$ [13], [21].

Next, we will provide an upper bound for A_1 and A_5 following the proof in [17]. Letting $L_{\max} = \sup_{j=\{1,\ldots,d\}} \sup_{\mathrm{x}\in\mathcal{X}} |\frac{\partial f(\mathrm{x})}{\partial x_j}|$, the union bound under Assumption 1 implies that

$$\Pr(L_{\max} \geq c) \leq dae^{-(c/b)^2}$$

which allows us to obtain

$$\begin{aligned}
\mathbb{E}[|f(\mathrm{x}) - f([\mathrm{x}]_t)|] &\leq \mathbb{E}[L\|\mathrm{x} - [\mathrm{x}]_t\|_1] \overset{(a)}{\leq} \frac{d}{n_t}\mathbb{E}[L_{\max}] \\
&\overset{(b)}{=} \frac{d}{n_t}\int_{c=0}^{\infty} \Pr(L_{\max} \geq c)dc \leq \frac{d}{n_t}\int_{c=0}^{\infty} dae^{-(c/b)^2}dc \\
&= \frac{\sqrt{\pi}d^2ab}{2n_t} = \frac{d}{2t^2}
\end{aligned}$$

where (a) results from (18), and (b) utilizes for $L_{\max} \geq 0$ the equality $\mathbb{E}[L_{\max}] = \int_{c=0}^{\infty} \Pr(L_{\max} \geq c)dc$. Hence, A_1 and A_5 are bounded by

$$A_1 = A_5 \leq \sum_{t=1}^{T} \frac{d}{2t^2} \leq \frac{\pi^2 d}{12} \; . \tag{20}$$

Further, A_2 can be upper bounded as

$$\begin{aligned}A_2 &\leq \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{I}(f([\mathrm{x}_*]_t) > U_t^{m_*}([\mathrm{x}_*]_t))\left[f([\mathrm{x}_*]_t - U_t^{m_*}([\mathrm{x}_*]_t)\right]\right] \\ &\leq \sum_{t=1}^{T}\sum_{m\in\mathcal{M}}\sum_{\mathrm{x}\in\mathcal{X}_t} \mathbb{E}\left[\mathbb{I}(f(\mathrm{x}) > U_t^{m}(\mathrm{x}))\left[f(\mathrm{x}) - U_t^{m}(\mathrm{x})\right]\right] \\ &\overset{(a)}{=} \sum_{t=1}^{T}\sum_{m\in\mathcal{M}}\sum_{\mathrm{x}\in\mathcal{X}_t} \frac{\sigma_{t-1}^{m}(\mathrm{x})}{\sqrt{2\pi t^2|\mathcal{X}_t|}} \\ &\overset{(b)}{\leq} \sum_{t=1}^{T}\sum_{m\in\mathcal{M}}\sum_{\mathrm{x}\in\mathcal{X}_t} \frac{1}{\sqrt{2\pi t^2|\mathcal{X}_t|}} = \frac{\sqrt{2\pi}M}{12} \end{aligned} \tag{21}$$

where, since $f(\mathrm{x}) - U_t^m(\mathrm{x})|\mathcal{D}_{t-1} \sim \mathcal{N}(-\beta_t^{1/2}\sigma_{t-1}^m(\mathrm{x}), (\sigma_{t-1}^m(\mathrm{x}))^2)$, (a) holds using the identity $\mathbb{E}[r\mathbb{I}(r>0)] = \frac{\sigma}{\sqrt{2\pi}}\exp(-\frac{\mu^2}{2\sigma^2})$ if $r \sim \mathcal{N}(\mu,\sigma^2)$ and $\mu < 0$. Inequality (b) is simply due to $\sigma_{t-1}^m(\mathrm{x}) \leq 1$.

The last step is to upper bound A_4, by constructing a confidence set $\mathcal{C}_t$ for the latent state per slot t so that $m_* \in \mathcal{C}_t$ holds with high probability [21]. We will replace $[\mathrm{x}_t]_t$ by x_t for notational brevity, given that the following result holds for both cases. Consider $\mathcal{C}_t := \{m \in \mathcal{M} : G_t^m \leq 2\sigma_n\sqrt{N_{t-1}^m \log T}\}$, where $N_{t-1}^m = \sum_{\tau=1}^{t-1}\mathbb{I}(m_\tau = m)$, and

$$G_t^m := \sum_{\tau=1}^{t-1}\mathbb{I}(m_\tau = m)\left(L_\tau^m(\mathrm{x}_\tau) - y_\tau\right) . \tag{22}$$

Here, $L_t^m(\mathrm{x}) = \mu_{t-1}^m(\mathrm{x}) - \eta\sigma_{t-1}^m(\mathrm{x})$ with $\eta = 2\sqrt{\log T}$ is a lower confidence bound for $f(\mathrm{x})$ conditioned on model m. For later use, we will first present the following two lemmas, whose proofs are deferred to Sections VII-A1 and VII-A2.

Lemma 4: It holds that $\Pr(m_* \notin \mathcal{C}_t|\mathcal{D}_{t-1}) \leq 2MT^{-1}, \forall t \in \mathcal{T} := \{1,\ldots,T\}$.

Lemma 5: It holds that $\mathbb{E}[\mu_{t-1}^{m_t}(\mathrm{x}_t) - f(\mathrm{x}_t)] < 2B, \forall m_t \in \mathcal{M}, \mathrm{x}_t \in \mathcal{X}, t \in \mathcal{T}$.

The following decomposition will be applied towards bounding A_4

$$\begin{aligned}A_4 &= \mathbb{E}\left[\sum_{t=1}^{T}\left(U_t^{m_t}(\mathrm{x}_t) - \mu_{t-1}^{m_t}(\mathrm{x}_t)\right)\right] \\ &\quad + \mathbb{E}\left[\sum_{t=1}^{T}\left(\mu_{t-1}^{m_t}(\mathrm{x}_t) - f(\mathrm{x}_t)\right)\right] \\ &\leq \underbrace{\sum_{t=1}^{T}\mathbb{E}[\beta_t^{1/2}\sigma_{t-1}^{m_t}(\mathrm{x}_t)]}_{:=A_{4,1}} + \underbrace{\sum_{t=1}^{T}\mathbb{E}\left[2B\mathbb{I}(m_t \notin \mathcal{C}_t)\right]}_{:=A_{4,3}} \\ &\quad + \underbrace{\sum_{t=1}^{T}\mathbb{E}\left[\left(\mu_{t-1}^{m_t}(\mathrm{x}_t) - f(\mathrm{x}_t)\right)\mathbb{I}(m_t \in \mathcal{C}_t)\right]}_{:=A_{4,2}} \end{aligned} \tag{23}$$

where the last inequality holds based on Lemma 5.

As m_* and m_t are identically distributed given $\mathcal{D}_{t-1}$ [13], [21], Lemma 4 allows $A_{4,3}$ to be readily bounded by

$$A_{4,3} = 2B\sum_{t=1}^{T}\mathbb{E}[\mathbb{I}(m_* \notin \mathcal{C}_t)] \leq 4MB . \tag{24}$$

Meanwhile, we have that

$$\begin{aligned}A_{4,2} &= \sum_{t=1}^{T}\mathbb{E}\left[\left(\mu_{t-1}^{m_t}(\mathrm{x}_t) - L_t^{m_t}(\mathrm{x}_t)\right)\mathbb{I}(m_t \in \mathcal{C}_t)\right] \\ &\quad + \sum_{t=1}^{T}\mathbb{E}\left[\left(L_t^{m_t}(\mathrm{x}_t) - y_t\right)\mathbb{I}(m_t \in \mathcal{C}_t)\right] \\ &\overset{(a)}{\leq} \sum_{t=1}^{T}\mathbb{E}[\eta\sigma_{t-1}^{m_t}(\mathrm{x}_t)] + \sum_{m\in\mathcal{M}}\mathbb{E}[G_{t_{\max}^m}^m] + 2MB \\ &\overset{(b)}{\leq} \sum_{t=1}^{T}\mathbb{E}[\eta\sigma_{t-1}^{m_t}(\mathrm{x}_t)] + \sum_{m\in\mathcal{M}}\mathbb{E}[2\sigma_n\sqrt{N_T^m\log T}] + 2MB \\ &\overset{(c)}{\leq} \sum_{t=1}^{T}\mathbb{E}[\eta\sigma_{t-1}^{m_t}(\mathrm{x}_t)] + 2\sigma_n\sqrt{MT\log T} + 2MB . \end{aligned} \tag{25}$$

where, with $t_{\max}^m$ being the last slot that m is selected, (a) holds by leveraging the definition of $G_{t_{\max}^m}^m$ and bounding the $L_{t_{\max}^m}^m(\mathrm{x}) - y_{t_{\max}^m}$ by $2B$; (b) comes from the definition of $\mathcal{C}_t$; and, (c) leverages Cauchy-Schwarz inequality to yield

$$\sum_{m=1}^{M}\sqrt{N_T^m} \leq \sqrt{M\sum_{m=1}^{M}N_T^m} = \sqrt{MT} . \tag{26}$$

Putting together the bounds for $A_1 - A_5$, the cumulative Bayesian regret of EGP-TS over T evaluations is bounded by

$$\begin{aligned}\mathcal{BR}(T) &\leq (\eta + \beta_T^{1/2})\sum_{t=1}^{T}\mathbb{E}[\sigma_{t-1}^{m_t}(\mathrm{x}_t)] + 2\sigma_n\sqrt{MT\log T} \\ &\quad + 6MB + \frac{\pi^2 d}{6} + \frac{\sqrt{2\pi}M}{12} \end{aligned} \tag{27}$$

where the first term can be bounded with $\mathcal{T}_T^m := \{t \in \mathcal{T} : m_t = m\}$ and $T_m := |\mathcal{T}_T^m|$ as

$$\begin{aligned}\sum_{t=1}^{T}\mathbb{E}[\sigma_{t-1}^{m_t}(\mathrm{x}_t)] &\leq \sum_{m=1}^{M}\mathbb{E}\left[\sum_{t\in\mathcal{T}_T^m}\sigma_{t-1}^m(\mathrm{x}_t)\right] \\ &\overset{(a)}{\leq} \sum_{m=1}^{M}\mathbb{E}\left[\sum_{t=1}^{T_m}\sigma_{t-1}^m(\mathrm{x}_t)\right] \overset{(b)}{\leq} \sum_{m=1}^{M}\mathbb{E}\left(T_m\sum_{t=1}^{T_m}\left(\sigma_{t-1}^m(\mathrm{x}_t)\right)^2\right)^{1/2} \\ &\overset{(c)}{\leq} \sum_{m=1}^{M}\left(\frac{2T_m\gamma_{T_m}}{\log(1+\sigma_n^{-2})}\right)^{1/2} \overset{(d)}{\leq} \sum_{m=1}^{M}\left(\frac{2T_m^{1+c}}{\log(1+\sigma_n^{-2})}\right)^{1/2} \\ &\overset{(e)}{\leq} \left(\frac{2MT^{1+c}}{\log(1+\sigma_n^{-2})}\right)^{1/2} \end{aligned} \tag{28}$$

where (a) holds since $\sigma_t^m(\mathrm{x})$ decreases as t grows; (b) is due to the Cauchy-Schwarz inequality; (c) leverages Lemmas 5.3

and 5.4 of [35] that bound the sum of posterior variances via the MIG; (d) follows upon bounding γ_{T_m} using Lemma 1; and, (e) holds upon utilizing the following inequality based on Cauchy-Schwarz inequality

$$\sum_{m=1}^{M}\left(T_m^{1+c}\right)^{1/2} \leq \left(M\sum_{m=1}^{M}T_m^{1+c}\right)^{1/2} \leq \left(M\left(\sum_{m=1}^{M}T_m\right)^{1+c}\right)^{1/2} = \left(MT^{1+c}\right)^{1/2}.$$

Upon plugging in (28) into (27), Theorem 1 holds with $\eta = 2\sqrt{\log T}$ and $\beta_T^{1/2} \approx \sqrt{d\log T}$.

1) Proof for Lemma 4: For $m_* \in \mathcal{M}$, define the following event at slot t

$$\mathcal{E}_t^{m_*} := \{|f(\mathbf{x}) - \mu_{t-1}^{m_*}(\mathbf{x})| \leq \eta\sigma_{t-1}^{m_*}(\mathbf{x})\} \tag{29}$$

the collection of which over T slots is $\mathcal{E}_{1:T}^{m_*} := \cap_{t=1}^{T}\mathcal{E}_t^{m_*}$. With $\bar{\mathcal{E}}_{1:T}^{m_*}$ representing its complement, it follows that

$$\begin{aligned}\mathbb{E}[I(\bar{\mathcal{E}}_{1:T}^{m_*})] &\leq \sum_{t=1}^{T}\sum_{m\in\mathcal{M}}\mathbb{E}\left[\mathbb{E}_{t-1}[\mathbb{I}(\bar{\mathcal{E}}_t^{m})]\right]\\ &= \sum_{t=1}^{T}\sum_{m\in\mathcal{M}}\mathbb{E}\left[\Pr_{t-1}\left(|\mu_{t-1}^{m}(\mathbf{x}) - f(\mathbf{x})| > \eta\sigma_{t-1}^{m}(\mathbf{x})\right)\right]\\ &\overset{(a)}{\leq} MT^{-1}\end{aligned} \tag{30}$$

where (a) comes from the inequality $\Pr(|r| > \eta) \leq e^{-\eta^2/2}$ with $r = |\mu_{t-1}^{m}(\mathbf{x}) - f(\mathbf{x})|/\sigma_{t-1}^{m}(\mathbf{x}) \sim \mathcal{N}(0,1)$ and $\eta = 2\sqrt{\log T}$.

Since $n_\tau = f(\mathbf{x}_\tau) - y_\tau \sim \mathcal{N}(0,\sigma_n^2)$, $\{n_\tau\}_{\tau\in\mathcal{T}_t^m}$ is then a martingale difference sequence w.r.t. $\{\mathcal{D}_\tau\}_{\tau\in\mathcal{T}_t^m}$, where $\mathcal{T}_t^m := \{\tau | m_\tau = m, \tau \in \{1,\ldots,t\}\}$.

$$\begin{aligned}&G_t^m I(\mathcal{E}_{1:T}^m)\\ &= \sum_{\tau\in\mathcal{T}_t^m}(L_\tau^m(\mathbf{x}_\tau) - y_\tau)\mathbb{I}(|f(\mathbf{x}_\tau) - \mu_{\tau-1}^m(\mathbf{x}_\tau)| \leq \eta\sigma_{\tau-1}^m(\mathbf{x}_\tau))\\ &= \sum_{\tau\in\mathcal{T}_t^m}(L_\tau^m(\mathbf{x}_\tau) - y_\tau)\mathbb{I}(L_\tau^m(\mathbf{x}_\tau) < f(\mathbf{x}_\tau)) \leq \sum_{\tau\in\mathcal{T}_t^m} n_\tau\,.\end{aligned} \tag{31}$$

For any $m \in \mathcal{M}$ and $t \in \mathcal{T}$, $u = |\mathcal{T}_t^m|$ is random and takes value from $\{1,\ldots,t-1\}$. For any u, Azuma's inequality yields

$$\begin{aligned}&\Pr_{t-1}(G_t^m I(\mathcal{E}_{1:T}^m) \geq 2\sigma_n\sqrt{u\log T})\\ &\leq \Pr\left(\sum_{\tau\in\mathcal{T}_t^m} n_\tau \geq 2\sigma_n\sqrt{u\log T}\right) \leq \exp(-2\log T) = T^{-2}\end{aligned}$$

based on which, we arrive at

$$\begin{aligned}\Pr_t(m_* \notin \mathcal{C}_t) &\leq \sum_{m\in\mathcal{M}}\sum_{u=1}^{t-1}\Pr(G_t^m \geq 2\sigma_n\sqrt{u\log T})\\ &\leq \sum_{m\in\mathcal{M}}\sum_{u=1}^{t-1}\mathbb{E}\left[\Pr_{t-1}\left(G_t^m\mathbb{I}(\mathcal{E}_{1:T}^m) \geq 2\sigma_n\sqrt{u\log T}\right)\right]\\ &\quad + \Pr(\bar{\mathcal{E}}_{1:T}^{m_*}) \leq 2MT^{-1}\,.\end{aligned} \tag{32}$$

thus finalizing the proof of Lemma 4.

2) Proof for Lemma 5: Since $\{\mathbf{x}_t, m_t\}$ and $\{\mathbf{x}_*, m_*\}$ are identically distributed conditioned on $\mathcal{D}_{t-1}$, it holds that

$$\begin{aligned}\mathbb{E}[\mu_{t-1}^{m_t}(\mathbf{x}_t)] &= \mathbb{E}[\mathbb{E}_{t-1}[\mu_{t-1}^{m_t}(\mathbf{x}_t)]]\\ &= \mathbb{E}[\mathbb{E}_{t-1}[\mu_{t-1}^{m_*}(\mathbf{x}_*)]] = \mathbb{E}[\mu_{t-1}^{m_*}(\mathbf{x}_*)]\,.\end{aligned}$$

Further, the identity $\mathbb{E}_{t-1}[\mu_{t-1}^{m_*}(\mathbf{x}_*)] = \mathbb{E}_{t-1}[f(\mathbf{x}_*)]$ with $f \sim \mathcal{GP}(0,\kappa^{m_*})$, yields the following result

$$\begin{aligned}&\mathbb{E}\left[\mu_{t-1}^{m_t}(\mathbf{x}_t) - f(\mathbf{x}_t)\right]\\ &= \mathbb{E}\left[\mu_{t-1}^{m_t}(\mathbf{x}_t) - \mu_{t-1}^{m_*}(\mathbf{x}_*) + \mu_{t-1}^{m_*}(\mathbf{x}_*)\right.\\ &\quad \left. - f(\mathbf{x}_*) + f(\mathbf{x}_*) - f(\mathbf{x}_t)\right]\\ &= \mathbb{E}[f(\mathbf{x}_*) - f(\mathbf{x}_t)] \leq 2B\end{aligned}$$

where, thanks to Lemma 3, the last inequality holds.

B. Proof of Theorem 2

For the asynchronous parallel setting, the upper confidence bound for the tth function evaluation is given by

$$\bar{U}_t^m(\mathbf{x}) := \mu_{\mathcal{D}_{t-1}}^m(\mathbf{x}) + \beta_t^{1/2}\sigma_{\mathcal{D}_{t-1}}^m(\mathbf{x}) \tag{33}$$

where $\mathcal{D}_{t-1}$ contains all the acquired data before evaluation index t is assigned. Here, $|\mathcal{D}_{t-1}| = t - K$ for $t > K$, and $|\mathcal{D}_{t-1}| = 0$ for $t \leq K$.

Leveraging a decomposition similar to that in (19), A_1–A_3 and A_5 could be derived as in Section A. Upon replacing the subscript $t-1$ of μ and σ by $\mathcal{D}_{t-1}$, the term A_4 can be bounded as in (23), that is

$$A_4 \leq \sum_{t=1}^{T}\mathbb{E}[(\beta_t^{1/2} + \eta)\sigma_{\mathcal{D}_{t-1}}^{m_t}(\mathbf{x}_t)] + 2\sigma_n\sqrt{MT\log T} + 6MB \tag{34}$$

where the first term can be further bounded based on Lemma 2 and (28) as

$$\begin{aligned}\sum_{t=1}^{T}\mathbb{E}[(\beta_t^{1/2} + \eta)\sigma_{\mathcal{D}_{t-1}}^{m_t}(\mathbf{x}_t)] &\overset{(a)}{\leq} (\beta_T^{1/2} + \eta)\sum_{t=1}^{T}\mathbb{E}[\rho_K^{1/2}\sigma_{t-1}^{m_t}]\\ &\leq (2+\sqrt{d})\left(\frac{2\rho_K MT^{c+1}\log T}{\log(1+\sigma_n^{-2})}\right)^{1/2}.\end{aligned} \tag{35}$$

Thus, the cumulative Bayesian regret for parallel EGP-TS in the asynchronous setup can be established as in Theorem 2.

C. Proof of Theorem 3

The proof of Theorem 3 entails introducing

$$V_t^m(\mathbf{x}) := \mu_{\mathcal{D}_{t-1}}^m(\mathbf{x}) + \beta_{t+K-1}^{1/2}\sigma_{t-1}^m(\mathbf{x}) \tag{36}$$

based on which the cumulative Bayesian regret can be decomposed after using (33) as (cf. (19))

$$\mathcal{BR}^{\text{syn}}(T) := \sum_{t=1}^{T}\mathbb{E}[f(\mathbf{x}_*) - f(\mathbf{x}_t)]$$

$$\overset{(a)}{=} \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[f(\mathbf{x}_*) - f([\mathbf{x}_*]_t)\right]}_{:=C_1} + \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[f([\mathbf{x}_*]_t - \bar{U}_t^{m_*}([\mathbf{x}_*]_t)\right]}_{:=C_2}$$
$$+ \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[\bar{U}_t^{m_*}([\mathbf{x}_*]_t) - V_t^{m_*}([\mathbf{x}_*]_t)\right]}_{:=C_3}$$
$$+ \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[V_t^{m_*}([\mathbf{x}_*]_t) - V_t^{m_t}([\mathbf{x}_t]_t)\right]}_{:=C_4}$$
$$+ \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[V_t^{m_t}([\mathbf{x}_t]_t) - f([\mathbf{x}_t]_t)\right]}_{:=C_5}$$
$$+ \underbrace{\sum_{t=1}^{T} \mathbb{E}\left[f([\mathbf{x}_t]_t) - f(\mathbf{x}_t)\right]}_{:=C_6} . \tag{37}$$

As with the proof of Theorem 1, it follows that

$$C_1 = C_6 \leq \frac{\pi^2 d}{12}, \quad C_4 = 0, \quad C_2 \leq \frac{\sqrt{2\pi}M}{12}. \tag{38}$$

Next, we will further bound C_3 and C_5, starting with

$$C_3 = \sum_{t=1}^{K-1} \beta_t^{1/2} \mathbb{E}[\sigma_{\mathcal{D}_{t-1}}^{m_*}([\mathbf{x}_*]_t)] - \sum_{T-K+1}^{T} \beta_{t+M-1}^{1/2} \mathbb{E}[\sigma_{t-1}^{m_*}([\mathbf{x}_*]_t)]$$
$$+ \sum_{t=K}^{T-K} \beta_t^{1/2} \mathbb{E}[\sigma_{\mathcal{D}_{t-1}}^{m_*}([\mathbf{x}_*]_t) - \sigma_{t-K}^{m_*}([\mathbf{x}_*]_t)]$$
$$\overset{(a)}{\leq} (K-1)\beta_{K-1}^{1/2}$$

where (a) holds since $\sigma_{\mathcal{D}_{t-1}}^{m_*}([\mathbf{x}_*]_t) \leq \sigma_{t-K}^{m_*}([\mathbf{x}_*]_t)$, and $0 < \sigma_t^{m_*}(\mathbf{x}) \leq 1$.

Lastly, C_5 can be bounded as

$$C_5 = \mathbb{E}\left[\sum_{t=1}^{T} \left(V_t^{m_t}(\mathbf{x}_t) - \mu_{\mathcal{D}_{t-1}}^{m_t}(\mathbf{x}_t)\right)\right]$$
$$+ \mathbb{E}\left[\sum_{t=1}^{T} \left(\mu_{\mathcal{D}_{t-1}}^{m_t}(\mathbf{x}_t) - f(\mathbf{x}_t)\right)\right]$$
$$\leq \sum_{t=1}^{T} \mathbb{E}[\beta_{t+K-1}^{1/2} \sigma_{t-1}^{m_t}(\mathbf{x}_t)] + \sum_{t=1}^{T} \mathbb{E}\left[2B\mathbb{I}(m_t \notin \mathcal{C}_t)\right]$$
$$+ \sum_{t=1}^{T} \mathbb{E}\left[\left(\mu_{\mathcal{D}_{t-1}}^{m_t}(\mathbf{x}_t) - f(\mathbf{x}_t)\right)\mathbb{I}(m_t \in \mathcal{C}_t)\right]$$
$$\leq \sum_{t=1}^{T} \mathbb{E}[\beta_{t+K-1}^{1/2} \sigma_{t-1}^{m_t}(\mathbf{x}_t) + \eta\sigma_{\mathcal{D}_{t-1}}^{m_t}(\mathbf{x}_t)]$$
$$+ 2\sigma_n\sqrt{MT\log T} + 6MB$$
$$\leq \sum_{t=1}^{T} \mathbb{E}[\beta_{T+K-1}^{1/2} \sigma_{t-1}^{m_t}(\mathbf{x}_t) + \eta\rho_K^{1/2}\sigma_{t-1}^{m_t}(\mathbf{x}_t)]$$
$$+ 2\sigma_n\sqrt{MT\log T} + 6MB \tag{39}$$

which, based on the derivation of (28), and the bounds of other factors, yields the regret bound in Theorem 3.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their constructive feedback.

REFERENCES

[1] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 2951–2959.
[2] K. Korovina et al., "ChemBO: Bayesian optimization of small organic molecules with synthesizable recommendations," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 3393–3403.
[3] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret, "Robots that can adapt like animals," *Nature*, vol. 521, no. 7553, pp. 503–507, 2015.
[4] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proc. IEEE*, vol. 104, no. 1, pp. 148–175, Jan. 2016.
[5] P. I. Frazier, "A tutorial on Bayesian optimization," 2018, *arXiv: 1807.02811*.
[6] R. Turner et al., "Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020," 2021, *arXiv:2104.10201*.
[7] Z. Wang, C. Gehring, P. Kohli, and S. Jegelka, "Batched large-scale Bayesian optimization in high-dimensional spaces," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2018, pp. 745–754.
[8] M. Feurer, B. Letham, and E. Bakshy, "Scalable meta-learning for Bayesian optimization using ranking-weighted Gaussian process ensembles," in *Proc. AutoML Workshop ICML*, 2018.
[9] M. Hoffman et al., "Portfolio allocation for Bayesian optimization," in *Proc. Conf. Uncerntainty Artif. Intell.*, 2011, pp. 327–336.
[10] B. Shahriari, Z. Wang, M. W. Hoffman, A. Bouchard-Côté, and N. de Freitas, "An entropy search portfolio for Bayesian optimization," 2014, *arXiv:1406.4625*.
[11] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.
[12] O. Chapelle and L. Li, "An empirical evaluation of Thompson sampling," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2011, pp. 2249–2257.
[13] D. Russo and B. Van Roy, "Learning to optimize via posterior sampling," *Math. Operations Res.*, vol. 39, no. 4, pp. 1221–1243, 2014.
[14] M. Mutnỳ and A. Krause, "Efficient high dimensional Bayesian optimization with additivity and quadrature Fourier features," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 9005–9016.
[15] D. Nguyen, S. Gupta, S. Rana, A. Shilton, and S. Venkatesh, "Bayesian optimization for categorical and category-specific continuous inputs," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 5256–5263.
[16] S. Gopakumar, S. Gupta, S. Rana, V. Nguyen, and S. Venkatesh, "Algorithmic assurance: An active approach to algorithmic testing using Bayesian optimisation," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 5470–5478.
[17] K. Kandasamy, A. Krishnamurthy, J. Schneider, and B. Póczos, "Parallelised Bayesian optimisation via Thompson sampling," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2018, pp. 133–142.
[18] 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. Int. Conf. Mach. Learn.*, 2017, pp. 1470–1479.
[19] S. R. Chowdhury and A. Gopalan, "On kernelized multi-armed bandits," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 844–853.
[20] S. Vakili, V. Picheny, and A. Artemev, "Scalable Thompson sampling using sparse Gaussian process models," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021, pp. 5631–5643 .

[21] J. Hong, B. Kveton, M. Zaheer, M. Ghavamzadeh, and C. Boutilier, "Thompson sampling with a mixture prior," 2021, *arXiv:2106.05608*.
[22] T. Desautels, A. Krause, and J. W. Burdick, "Parallelizing exploration-exploitation tradeoffs in Gaussian process bandit optimization," *J. Mach. Learn. Res.*, vol. 15, pp. 3873–3923, 2014.
[23] J. Wang, S. C. Clark, E. Liu, and P. I. Frazier, "Parallel Bayesian global optimization of expensive functions," 2016, *arXiv:1602.05149*.
[24] T. Teng, J. Chen, Y. Zhang, and B. K. H. Low, "Scalable variational Bayesian kernel selection for sparse Gaussian process regression," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 5997–6004.
[25] D. Duvenaud, J. Lloyd, R. Grosse, J. Tenenbaum, and G. Zoubin, "Structure discovery in nonparametric regression through compositional kernel search," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1166–1174.
[26] H. Kim and Y. W. Teh, "Scaling up the automatic statistician: Scalable structure discovery using Gaussian processes," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2018, pp. 575–584.
[27] G. Malkomes, C. Schaff, and R. Garnett, "Bayesian optimization for automated model selection," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 41–47.
[28] Q. Lu, G. Karanikolas, Y. Shen, and G. B. Giannakis, "Ensemble Gaussian processes with spectral features for online interactive learning with scalability," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 1910–1920.
[29] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[30] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.
[31] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2008, pp. 1177–1184.
[32] J. Wilson, V. Borovitskiy, A. Terenin, P. Mostowsky, and M. Deisenroth, "Efficiently sampling functions from Gaussian process posteriors," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 10 292–10 302.
[33] M. Lázaro-Gredilla, J. Quiñonero Candela, C. E. Rasmussen, and A. Figueiras-Vidal, "Sparse spectrum Gaussian process regression," *J. Mach. Learn. Res.*, vol. 11, no. Jun., pp. 1865–1881, 2010.
[34] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Glob. Optim.*, vol. 13, no. 4, pp. 455–492, 1998.
[35] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic regret bounds for Gaussian process optimization in the bandit setting," *IEEE Trans. Inf. Theory*, vol. 58, no. 5, pp. 3250–3265, May 2012.
[36] S. Ghosal and A. Roy, "Posterior consistency of Gaussian process prior for nonparametric binary regression," *Ann. Statist.*, vol. 34, no. 5, pp. 2413–2429, 2006.
[37] Z. Wang and S. Jegelka, "Max-value entropy search for efficient Bayesian optimization," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3627–3635.
[38] Breast Cancer dataset. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/breast+cancer
[39] Iris dataset. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/iris
[40] Transportation dataset. [Online]. Available: https://gisdata.mn.gov
[41] Wine dataset. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/wine
[42] R. J. Adler, *An Introduction to Continuity, Extrema, and Related Topics for General GAussian Processes*. Virginia Beach, VA, USA: IMS, 1990.
[43] W. Rudin, *Principles of Mathematical Analysis*, vol. 3. New York, NY, USA: McGraw-Hill, 1964.

Qin Lu (Member, IEEE) received the PhD degree in electrical engineering from the University of Connecticut (UConn), in 2018. Currently, she is a postdoctoral research associate with the University of Minnesota, Twin Cities. Her research interests span the areas of machine learning, data science, and network science, with special focus on Bayesian inference, Bayesian optimization, and spatio-temporal inference over graphs. In the past, she has worked on statistical signal processing, multiple target tracking, and underwater acoustic communications. She was awarded Summer Fellowship and Doctoral Dissertation Fellowship with UConn. She was also a recipient of the Women of Innovation Award in Collegian Innovation and Leadership by Connecticut Technology Council in March, 2018.

Konstantinos D. Polyzos (Student Member, IEEE) received the diploma degree from the Department of Electrical Engineering and Computer Technology, University of Patras, Greece, in 2018. Currently, he is working toward the PhD degree with the Department of Electrical and Computer Engineering (ECE), University of Minnesota (UMN) – Twin Cities. He is a member of the SPiNCOM Research Group under the supervision of Prof. Georgios B. Giannakis. His research interests span the areas of machine learning, signal processing, network science, and data science. Lately, he focuses on learning over graphs which can model complex networks including financial, social and biological ones to list a few. In the past, he has worked on the development of automatic aerial target recognition systems using passive Radar data. He has been awarded the UMN ECE Department fellowship (2019), Gerondelis Foundation scholarship (2020), Onassis Foundation scholarship (2021), the Best Paper Award at the International CIT & DS 2019 International Conference (2019) and the Outstanding Reviewer Award (top 10 %) at the International Conference on Machine Learning (ICML 2022).

Bingcong Li (Member, IEEE) received the MSc and PhD degrees in electrical and computer engineering (ECE) from the University of Minnesota (UMN), in 2019 and 2022, respectively. He is now with Huawei as a research engineer. His research interests lie in optimization and machine learning systems. He received the National Scholarship twice from China in 2014 and 2015, and UMN ECE Department Fellowship in 2017.

Georgios B. Giannakis (Fellow, IEEE) received the diploma in electrical engineering from the National Technical University of Athens, Greece, in 1981, and the MSc degree in electrical engineering, the MSc degree in mathematics, and the PhD degree in electrical engineering from the University of Southern California (USC), in 1983, 1986, and 1986, respectively. He was a faculty member with the University of Virginia from 1987 to 1998, and since 1999, he has been a professor with the University of Minnesota, where he holds an ADC endowed chair, a University of Minnesota McKnight presidential chair in ECE, and serves as director of the Digital Technology Center. His general interests span the areas of statistical learning, signal processing, communications, and networking - subjects on which he has published more than 480 journal papers, 780 conference papers, 25 book chapters, two edited books, and two research monographs. Current research focuses on Data Science, and Network Science with applications to the Internet of Things, and power networks with renewables. He is the (co-) inventor of 34 issued patents, and the (co-) recipient of 10 best journal paper awards from IEEE Signal Processing (SP) and Communications Societies, including the G. Marconi Prize Paper Award in Wireless Communications. He also received the IEEE-SPS Norbert Wiener Society Award (2019); EURASIP's A. Papoulis Society Award (2020); Technical Achievement Awards from the IEEE-SPS (2000) and from EURASIP (2005); the IEEE ComSoc Education Award (2019); and IEEE Fourier Technical Field Award (2015). He is a member of the Academia Europaea, and fellow of the National Academy of Inventors, the European Academy of Sciences, and EURASIP. He has served the IEEE in a number of posts, including that of a distinguished lecturer for IEEE-SPS.