Received: 2 February 2024 | Revised: 28 April 2024 | Accepted: 7 May 2024

DOI: 10.1002/sam.11697

WILEY

RESEARCH ARTICLE

Sequential metamodel-based approaches to level-set estimation under heteroscedasticity

Yutong Zhang | **Xi Chen**

Grado Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, Virginia, USA

Correspondence
Xi Chen, Virginia Tech, Grado Department of Industrial and Systems Engineering, Blacksburg, VA, USA.
Email: xchen6@vt.edu

Funding information
National Science Foundation, Grant/Award Number: IIS-1849300; National Science Foundation, CMMI, 1846663

Abstract

This paper proposes two sequential metamodel-based methods for level-set estimation (LSE) that leverage the uniform bound built on stochastic kriging: predictive variance reduction (PVR) and expected classification improvement (ECI). We show that PVR and ECI possess desirable theoretical performance guarantees and provide closed-form expressions for their respective sequential sampling criteria to seek the next design point for performing simulation runs, allowing computationally efficient one-iteration look-ahead updates. To enhance understanding, we reveal the connection between PVR and ECI's sequential sampling criteria. Additionally, we propose integrating a budget allocation feature with PVR and ECI, which improves computational efficiency and potentially enhances robustness to the impacts of heteroscedasticity. Numerical studies demonstrate the superior performance of the proposed methods compared to state-of-the-art benchmarking approaches when given a fixed simulation budget, highlighting their effectiveness in addressing LSE problems.

KEYWORDS

heteroscedastic Gaussian process metamodeling, level-set estimation, sequential sampling, uniform bounds

1 | INTRODUCTION

Many scientific and engineering applications require determining, through simulation, which system designs have superior (or inferior) performance to a predefined threshold. Such problems range from environmental science [9], biology [20], financial engineering [26] to operations management [35]. Consider the following two concrete examples.

- Service center design: Various agent allocation configurations have been proposed for a single queueing network. However, only alternatives with a mean waiting time below a certain threshold are considered acceptable.
- Supply chain management: A company is uncertain about the market and operations conditions that their inventory system will face in the upcoming months. Managers want to determine which ordering policies can keep the average total cost incurred per period below a specified level. They intend to utilize a simulation model of the inventory system to address this inquiry.

Stochastic simulation is widely used to evaluate complex system performance. However, running a stochastic simulation model can be expensive, especially when comparing numerous alternative system designs. Additionally, simulation outputs are inherently subject to heteroscedasticity, where the noise variance varies across the input space. In such cases, a reasonable algorithm may need to resample at the same input point multiple times (i.e., run the simulation model for multiple independent simulation replications at the same alternative) to obtain a more

accurate estimate of the system performance. Therefore, there is a need for intelligent allocation of the simulation budget.

1.1 | Relation to existing methods

In the literature, the problems of interest in this work are often referred to as level-set estimation (LSE), multiple comparisons with a known standard (MCS), or feasibility detection (FD) problems. For simplicity, we will refer to them as LSE hereinafter.

Existing work on LSE in the stochastic simulation literature primarily focuses on developing model-free, efficient sampling procedures. These methods aim to allocate the simulation budget among the alternatives to support an accurate estimation of the corresponding system performance, thereby facilitating accurate determination relative to the given threshold. From a frequentist perspective, two-stage [4, 28] and fully sequential procedures [22, 29] have been developed; these methods do not assume a fixed sampling budget and intend to achieve a given LSE accuracy with a prescribed statistical guarantee. Moreover, asymptotically optimal sampling procedures are investigated which maximize the large deviation rate for LSE with no finite-time performance guarantee [37]. LSE has also been explored from a Bayesian perspective. For instance, Bayes-optimal sequential sampling policies are derived using methods from multi-armed bandits and optimal stopping [44]. A myopic allocation procedure is proposed for sequentially allocating the sampling budget to the input point that maximizes the posterior performance gain for LSE [38]. The dynamic finite-budget allocation rule (FAR) is a state-of-the-art LSE method that achieves not only desirable finite budget properties but also asymptotic optimality for maximizing the posterior performance [35]. *However, all the aforementioned model-free sampling methods can only efficiently handle LSE problems where the number of alternatives is not large.*

When a stochastic model is computationally expensive to run, and there are numerous input points (alternatives) to assess, a metamodel (or a surrogate model) can be an effective substitute for the simulation model, allowing the simulation to be virtually run "on demand" to support real-time decision-making processes [34]. Metamodels are simplified models that approximate the underlying input–output relationship implied by the simulation model of interest. Among various metamodeling techniques, Gaussian process regression (GPR) is arguably the most important one. One primary reason for GPR models' popularity is that they unite sophisticated and consistent theoretical investigations with computational tractability [33].

There has been a substantial body of research focused on using metamodels to tackle LSE problems. GPR is a notable approach that offers a probabilistic model of the target system response function using a GP prior. Numerous studies have investigated GPR-based sequential sampling strategies. These methods utilize the posterior distribution from GP to select the next input point or alternative for sampling. Some work focuses on developing variants of the expected improvement criterion originally proposed for Bayesian optimization (BO) to select the next input point for addressing LSE problems [6, 31, 32]. Other studies propose sequential sampling criteria that greedily maximize the reduction in, for example, the stepwise uncertainty [5], the maximum predictive variance, the largest classification ambiguity [19], and the truncated predictive variance [9]. Additionally, some studies propose criteria that myopically maximize the expected classification improvement [45]. Recently, there has been a growing trend in research aimed at enhancing the robustness and applicability of LSE approaches by incorporating various features. For example, some studies have started to consider the impact of input uncertainty on LSE, as seen in Chevalier et al. and Inatsu et al. [11, 20]. Additionally, there has been research focusing on the batching selection of input points, as explored in Lyu et al. and Lyu and Ludkovski [26, 27]. However, the work mentioned above assumes that the simulation outputs are noise-free or subject to homoscedastic noise (i.e., the noise variance is identical over the input space). *To the best of our knowledge, there has been limited research on developing metamodel-based approaches to address LSE problems under heteroscedasticity, which is particularly relevant in the stochastic simulation setting.*

The existing methods reviewed above typically perform LSE relative to a given threshold in two ways: using either point estimates or confidence bounds of the mean system performance obtained at individual alternatives/input points. A majority of existing metamodel-based methods use classical pointwise confidence bounds to address LSE at individual input points [45], but some work also adopts uniform bounds [9, 19]. Decisions made using uniform bounds tend to be more robust because these bounds are designed to contain the true system performance at all input points simultaneously and throughout the sampling process with a prescribed high probability guarantee.

1.2 | Our contributions

In this work, we consider a popular heteroscedastic GPR metamodeling approach, stochastic kriging (SK, [3]), which can accurately approximate the mean function implied by a stochastic simulation model. We develop

sequential SK-based procedures for LSE that rely on a suitable uniform bound for the mean function constructed based on SK. This uniform bound can provide a high-probability confidence bound of the mean function value at any input point across the input space and throughout the sequential sampling process. This is the first attempt to use a uniform bound based on SK to tackle LSE problems arising in the stochastic simulation setting. Specifically, we propose two SK-based sequential procedures for LSE that rely on this uniform bound: predictive variance reduction (PVR) and expected classification improvement (ECI), respectively inspired by Bogunovic et al. [9] and Zanette et al. [45]. While Bogunovic et al. adopt some uniform bounds to tackle LSE and BO, they do not specifically consider heteroscedasticity in the theoretical development and the numerical implementation. On the other hand, Zanette et al. do not adopt uniform bounds for LSE, nor do they account for the impact of heteroscedasticity.

The main contributions of this work are summarized as follows. First, relying on the rigorously established uniform bound based on SK, we prove that PVR and ECI possess desirable theoretical performance guarantees for addressing LSE problems. In particular, we provide a high-probability lower bound for the number of iterations PVR takes to achieve a prescribed LSE accuracy level. Moreover, we show that the maximum LSE error achieved by ECI is bounded by a given tolerance parameter with a prescribed high probability. Second, we derive closed-form expressions for their respective sequential sampling criteria to seek the next design point for performing simulation runs, allowing computationally efficient one-iteration look-ahead updates. The criteria for PVR and ECI differentiate their form between sampling from an existing design point and a new one, inherently balancing between adding more replications at an existing design point and exploring a new design point to optimize performance greedily. Additionally, we provide insights into the connection between PVR and ECI's sequential sampling criteria, facilitating a deeper understanding. Lastly, we propose incorporating a budget allocation feature with PVR and ECI, enhancing computational efficiency and potentially increasing robustness to the impacts of heteroscedasticity. Numerical evaluations focus on comparing PVR and ECI, along with their respective generalizations, with state-of-the-art LSE methods, demonstrating their superiority in addressing LSE problems when given a fixed simulation budget.

The remainder of the paper is organized as follows. Section 2.1 reviews stochastic kriging and a uniform bound for the mean function constructed based on SK. Section 3 details the two proposed LSE methods and their generalizations. Section 4 provides numerical evaluations of the proposed methods against state-of-the-art benchmarking approaches. Section 5 concludes the work.

2 | BACKGROUND REVIEW

This section first provides an overview of heteroscedastic simulation metamodeling, with a focus on SK. Then, we briefly review a uniform bound established based on SK, which serves as the foundation for the proposed LSE methods presented in the following section.

2.1 | Review of stochastic kriging

Heteroscedastic simulation metamodeling focuses on using the simulation outputs obtained at a set of design points to approximate the mean response surface implied by a stochastic simulation model, where the simulation output variance varies across the input space. Several methods have gained considerable attention, which include, but are not limited to, the Markov chain Monte Carlo-based fully Bayesian approach [18], the maximum a posteriori-based GP [21], SK [3], practical heteroscedastic GP modeling [7], and more recently, the variational inference-based heteroscedastic GP modeling [39]. Among them, SK stands out by striking a good balance between computational efficiency and statistical accuracy. We will consider SK as the metamodeling tool to tackle LSE in this work.

Before delving into a brief review of SK, we first introduce the notation system to facilitate the exposition. Consider a sequential sampling process for running the simulation model of interest to construct an SK model. Denote the input space as $\mathcal{X} \subset \mathfrak{R}^d$. On each iteration (say, the tth one for $t \geq 1$), a design point is selected from $\mathcal{X}$ to perform simulation runs. Since some design points may be chosen more than once as the iteration proceeds, we denote the number of distinct design points selected up to the tth iteration as $k(t)$. Let $n_{t,i}$ denote the total number of simulation replications allocated to the tth design point up to the tth iteration, and let $\mathcal{D}_t := \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{k(t)}\}$ denote the design-point set formed by iteration t.

Suppose that the simulation output obtained at design point $\mathbf{x}_i \in \mathcal{X}$ on the jth simulation replication can be described by the following model:

$$y_{t,j}(\mathbf{x}_i) = f_0(\mathbf{x}_i) + \epsilon_j(\mathbf{x}_i), \quad j = 1, 2, \ldots, n_{t,i}, \tag{1}$$

where $f_0(\cdot)$ denotes the true unknown mean response function that we intend to estimate and $\epsilon_j(\mathbf{x}_i)$ denotes the simulation noise or sampling error in the output. Assume that the simulation noise terms incurred on different

replications at $\mathbf{x}_i$, $\varepsilon_1(\mathbf{x}_i), \varepsilon_2(\mathbf{x}_i), \ldots$ are independent and identically distributed (i.i.d.) random variables with zero mean and variance $\mathrm{V}(\mathbf{x}_i)$, for $i = 1, 2, \ldots, k(t)$, and that the simulation noise variance function $\mathrm{V}(\cdot)$ satisfies $\sup_{\mathbf{x}\in\mathcal{X}} \mathrm{V}(\mathbf{x}) < \infty$.

Given the simulation outputs generated up to the tth iteration, we can obtain the average simulation output at $\mathbf{x}_i$ based on (1) as

$$\bar{y}_t(\mathbf{x}_i) = \frac{1}{n_{t,i}} \sum_{j=1}^{n_{t,i}} y_{t,j}(\mathbf{x}_i) = f_0(\mathbf{x}_i) + \bar{\varepsilon}(\mathbf{x}_i),$$

where $\bar{\varepsilon}(\mathbf{x}_i) = n_{t,i}^{-1} \sum_{j=1}^{n_{t,i}} \varepsilon_j(\mathbf{x}_i)$ denotes the average simulation noise incurred at design point $\mathbf{x}_i$, for $i = 1, 2, \ldots, k(t)$. Denote the $k(t) \times 1$ vector of average outputs as $\bar{\mathbf{y}}_t := \left(\bar{y}_t(\mathbf{x}_1), \bar{y}_t(\mathbf{x}_2), \ldots, \bar{y}_t\left(\mathbf{x}_{k(t)}\right)\right)^\tau$ and the $k(t) \times 1$ vector of the average simulation noise terms as $\varepsilon_t := \left(\bar{\varepsilon}(\mathbf{x}_1), \bar{\varepsilon}(\mathbf{x}_2), \ldots, \bar{\varepsilon}\left(\mathbf{x}_{k(t)}\right)\right)^\tau$.

Parallel with the treatment in the standard GP modeling literature [34, 42], SK assumes that the underlying mean response function $f_0(\cdot)$ is a sample of a zero-mean Gaussian process, denoted by $f_0(\cdot) \sim \mathrm{GP}\left(0, \tau^2 K(\cdot\cdot)\right)$, where $\tau^2 > 0$ denotes the process variance and $K(\cdot, \cdot)$ is the kernel function. Specifically, the covariance between the values of f_0 at any two input points $\mathbf{x}', \mathbf{x}'' \in \mathcal{X}$ can be modeled as

$$\mathrm{Cov}\left(f_0(\mathbf{x}'), f_0(\mathbf{x}'')\right) = \tau^2 K\left(\mathbf{x}', \mathbf{x}''\right).$$

Commonly used kernel functions include the squared exponential kernel or Gaussian kernel, the Matérn kernel, etc. [46]. On the tth iteration, SK adopts the following predictive mean as the point estimator of $f_0(\mathbf{x})$ at any given $\mathbf{x} \in \mathcal{X}$:

$$\mu_t(\mathbf{x}) = K(\mathbf{x}, \mathbf{X}_t)\left(K(\mathbf{X}_t, \mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\right)^{-1}\bar{\mathbf{y}}_t, \qquad (2)$$

with the corresponding predictive variance given by

$$\sigma_t^2(\mathbf{x}) = \tau^2\Big(K(\mathbf{x}, \mathbf{x}) - K(\mathbf{x}, \mathbf{X}_t)\left(K(\mathbf{X}_t\mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\right)^{-1} \times K(\mathbf{x}, \mathbf{X}_t)^\tau\Big), \qquad (3)$$

where $\mathbf{X}_t := \left(\mathbf{x}_1^\tau, \mathbf{x}_2^\tau, \ldots, \mathbf{x}_{k(t)}^\tau\right)^\tau$ denotes the $k(t) \times d$ design matrix consisting of the $k(t)$ design points accumulated in $\mathcal{D}_t$ up to iteration t. The $1 \times k(t)$ vector $K(\mathbf{x}, \mathbf{X}_t) := \left(K(\mathbf{x}\mathbf{x}_1), K(\mathbf{x}\mathbf{x}_2), \ldots, K\left(\mathbf{x}\mathbf{x}_{k(t)}\right)\right)$ contains the correlations between $\mathbf{x}$ and the $k(t)$ design points, and $K(\mathbf{X}_t, \mathbf{X}_t)$ denotes the $k(t) \times k(t)$ matrix of correlations across the $k(t)$ design points whose (i, j)th entry is given by $K\left(\mathbf{x}_i, \mathbf{x}_j\right)$, $i, j = 1, 2, \ldots, k(t)$. The $k(t) \times k(t)$ matrix $\Sigma_{\epsilon,k(t)}$ represents the variance–covariance matrix of the average simulation noise vector $\bar{\varepsilon}_t$. Since the use of common random numbers (CRN) does not necessarily help improve the predictive performance of SK [10], we assume that CRN is not applied in the simulation experiments in this work. Hence, $\Sigma_{\epsilon,k(t)}$ reduces to a diagonal matrix, that is, $\Sigma_{\epsilon,k(t)} = \mathrm{diag}\left(\mathrm{V}(\mathbf{x}_1)/n_{t,1}, \mathrm{V}(\mathbf{x}_2)/n_{t,2}, \ldots, \mathrm{V}\left(\mathbf{x}_{k(t)}\right)/n_{t,k(t)}\right)$.

Sequential design strategies for applying SK metamodeling have been extensively studied in the stochastic simulation setting, primarily focusing on prediction [1] or optimization tasks [30]. However, there has been limited research on developing efficient SK-based methods specifically tailored for addressing LSE problems.

2.2 | A uniform bound for the mean function based on stochastic kriging

There exist several methods for constructing a confidence bound for the unknown mean response function, including the classical pointwise confidence interval [34], the simultaneous confidence region relying on bootstrapping or the Bonferroni [24] and Šidák corrections [13], and the uniform confidence bounds derived either using the frequentist kernel methods [23] or from the Bayesian GP modeling perspective [43]. We adopt the uniform bound for heteroscedastic metamodeling approaches (including SK) proposed by Kirschner and Krause [23] which holds true with a prescribed high probability across the input space $\mathcal{X}$ and through all iterations $t \geq 1$. We formally state the uniform bound and the underlying assumption stipulated next.

> **Assumption 1.** The simulation noise terms $\varepsilon_1(\mathbf{x}), \varepsilon_2(\mathbf{x}), \ldots$ incurred at $\mathbf{x} \in \mathcal{X}$ are sub-Gaussian, namely, $\mathbb{E}\left[\exp\left(\varepsilon_j(\mathbf{x})\right)\right] \leq \exp\left(\lambda^2 \mathrm{V}(\mathbf{x})/2\right)$, for all $\lambda \in \mathfrak{R}$.

We note that Assumption 1 is less restrictive than modeling the simulation errors as Gaussian random variables, a commonly adopted assumption in the GP modeling literature [42]. The class of sub-Gaussian random variables include Gaussian random variables, any bounded random variables, any random variables with strongly log-concave density, etc.

> **Proposition 1.** *(Lemma 7 in [23]) Let $\delta \in (0, 1)$. Suppose the mean response function $f_0 \in \mathcal{H}_K$, the reproducing kernel Hilbert space (RKHS) corresponding to kernel K, and Assumption 1 is fulfilled. Then, the following uniform error bound holds true with probability at least $1 - \delta$ for all $k(t) \geq 1$ and any $\mathbf{x} \in \mathcal{X}$:*

$$|\mu_t(\mathbf{x}) - f_0(\mathbf{x})| \leq \underbrace{\left(\sqrt{\log\left(|\mathbf{I}_{k(t)} + \tau^2\Sigma_{\varepsilon,k(t)}^{-1}K(\mathbf{X}_t\mathbf{X}_t)|\right) - 2\log\delta} + \tau^{-1} \parallel f_0 \parallel_K\right)}_{:=\beta_{k(t)}}\sigma_t(\mathbf{x}). \tag{4}$$

where $\mathbf{I}_{k(t)}$ denotes the $k(t)\times k(t)$ identity matrix, $|\mathbf{A}|$ denotes the determinant of matrix $\mathbf{A}$, and $\parallel f_0 \parallel_K$ is the RKHS norm.

The uniform error bound in (4) is a direct extension of Theorem 3.11 in [2]. Its connections to the Bayesian GP setting is discussed in Kirschner and Krause [23]. For completeness, we provide the proof of Proposition 1 in Appendix A.

A uniform bound for the mean response function $f_0(\cdot)$ directly follows from Proposition 1. That is, it holds with probability at least $1-\delta$ for all $k(t) \geq 1$ and any $\mathbf{x} \in \mathcal{X}$ that

$$\begin{aligned} l_t(\mathbf{x}) &:= \mu_t(\mathbf{x}) - \beta_{k(t)}\sigma_t(\mathbf{x}) \leq f_0(\mathbf{x}) \leq u_t(\mathbf{x}) \\ &:= \mu_t(\mathbf{x}) + \beta_{k(t)}\sigma_t(\mathbf{x}). \end{aligned} \tag{5}$$

The uniform bound $\left[l_t(\mathbf{x}), u_t(\mathbf{x})\right]$ covers the unknown mean function value $f_0(\mathbf{x})$ at any $\mathbf{x} \in \mathcal{X}$ on any iteration t with a prescribed high probability at least $1-\delta$. Therefore, any LSE decision made at any input point based on this uniform bound throughout the sequential sampling process is valid with a prescribed high-probability guarantee, and tractable analyses of the corresponding metamodel-based LSE method's performance follow conveniently. We will exploit this uniform bound in devising the proposed procedures to be detailed in the next section.

3 | SEQUENTIAL METAMODEL-BASED LEVEL-SET ESTIMATION UNDER HETEROSCEDASTICITY

This section presents two sequential metamodel-based LSE methods suitable for the stochastic simulation setting. Following the problem setup given in Subsection 3.1, Subsections 3.2 and 3.3 respectively detail the proposed methods, predictive variance reduction and expected classification improvement, and reveal the connections between them. Subsection 3.4 introduces a budget allocation feature to be incorporated with PVR and ECI, aiming to potentially mitigate the impact of heteroscedasticity.

3.1 | Problem setup

We consider the LSE problem of identifying those input points in a prediction set $\mathcal{P} \subset \mathfrak{R}^d$ whose corresponding mean values implied by stochastic simulations are smaller (or greater) than a known threshold [19]. That is, given a threshold $h \in \mathcal{T}$, LSE seeks to determine the true super-level set $H := \{\mathbf{x} \in \mathcal{P} | f_0(\mathbf{x}) > h\}$ and the true sub-level set $L := \{\mathbf{x} \in \mathcal{P} | f_0(\mathbf{x}) < h\}$, where $f_0(\cdot)$ denotes the underlying true mean response function and $\mathcal{T} \subset \mathfrak{R}$ represents the range of $f_0(\cdot)$.

We begin by stating some definitions necessary for our development and the basic setup for explaining a generic sequential LSE method.

Definition 1. Given a threshold $h \in \mathcal{T} \subset \mathfrak{R}$, the estimated super-level set H_t, the estimated sub-level set L_t, and the uncertain set M_t obtained up to the tth iteration ($t \geq 1$) are respectively defined as

$$H_t = \{\mathbf{x} \in M_{t-1} | l_t(\mathbf{x}) > h - \epsilon_0\} \cup H_{t-1}, \tag{6}$$

$$L_t = \{\mathbf{x} \in M_{t-1} | u_t(\mathbf{x}) < h + \epsilon_0\} \cup L_{t-1}, \tag{7}$$

$$M_t = \{\mathbf{x} \in M_{t-1} | l_t(\mathbf{x}) \leq h - \epsilon_0, u_t(\mathbf{x}) \geq h + \epsilon_0\}, \tag{8}$$

where recall that $u_t(\mathbf{x})$ and $l_t(\mathbf{x})$ are the upper and lower limits of the uniform bound for $f_0(\mathbf{x})$ defined in (5), and $\epsilon_0 \geq 0$ denotes a prescribed LSE error tolerance parameter. Define $H_0 = L_0 := \emptyset$, an empty set, and $M_0 := \mathcal{P} \subset \mathfrak{R}^d$, with $\mathcal{P}$ denoting the set of input points for level-set estimation.

Definition 2. On the tth iteration, the estimated level-set triplet (M_t, H_t, L_t) is considered ϵ-accurate if $H_t \subseteq H$, $L_t \subseteq L$, and for $\forall \mathbf{x} \in M_t$, $|f_0(\mathbf{x}) - h| < \epsilon/2$, for a prescribed accuracy level $\epsilon > 0$.

Before delving into the two proposed LSE methods to be detailed in the next two subsections, we point out that there is a common level set updating step underlying them. On each iteration t, the estimated level-set triplet (M_t, H_t, L_t) is updated according to (6), (7), and (8) using the uniform bound $\left[l_t(\mathbf{x}), u_t(\mathbf{x})\right]$ obtained on the tth iteration for all $\mathbf{x} \in M_{t-1}$. That is, we are only interested in classifying the input points in M_{t-1} on iteration t, since the input points already classified into either H_{t-1} or L_{t-1} will remain there till the end of the implementation. Therefore, the estimated sub- and super-level sets L_t and H_t are non-decreasing and the uncertain set M_t is non-increasing in size. The major difference between the two proposed LSE approaches, PVR and ECI, lies in their sequential selection criteria for choosing the next design point to run the simulation model. Subsections 3.2 and 3.3 respectively provide details about PVR and ECI.

3.2 | Predictive variance reduction

This subsection elaborates on the PVR approach for LSE. We first describe the essential steps of PVR and provide closed-form expressions to facilitate its efficient implementation. Then, we prove that PVR can achieve the ϵ-accuracy guarantee with a prescribed high probability under some technical conditions.

The core of PVR lies in its sequential selection criterion, which chooses the next design point to achieve the maximum reduction in the predictive uncertainty at all input points yet to be classified. The details of PVR are summarized in Algorithm 1. Specifically, PVR proceeds in epochs. Each epoch (say, the ith one) comprises a number of iterations which is unknown up front. On each iteration (say, the tth one) within the ith epoch, PVR selects the design point $\mathbf{x}_t \in \mathcal{X}$ that maximizes the standardized reduction in the sum of truncated predictive variances multiplied by $\beta_{(i)}^2$ over the uncertain set (with ties broken arbitrarily):

$$\mathbf{x}_t = \arg\max_{\mathbf{x}^+ \in \mathcal{X}} \left(\sum_{\mathbf{x} \in M_{t-1}} \max\left\{ \beta_{(i)}^2 \sigma_{t-1}^2(\mathbf{x}), \eta_{(i)}^2 \right\} - \sum_{\mathbf{x} \in M_{t-1}} \max\left\{ \beta_{(i)}^2 \sigma_{t-1|\mathbf{x}^+}^2(\mathbf{x}), \eta_{(i)}^2 \right\} \right) / c(\mathbf{x}^+), \quad (9)$$

where $\beta_{(i)}$ is the uniform bound coefficient fixed for the ith epoch, $\sigma_{t-1|\mathbf{x}^+}^2(\mathbf{x})$ is the predictive variance upon obtaining simulation outputs generated up to iteration $t-1$ and given that $\mathbf{x}^+$ were selected on the tth iteration, $\eta_{(i)}$ is the truncation parameter for the predictive variance, and $c(\mathbf{x}^+)$ is the cost to run the simulation model at $\mathbf{x}^+$. The simulation runs are conducted at $\mathbf{x}_t$, which is chosen according to (9) with n_0 replications allocated there (Step 5), and the outputs are used to update the metamodel, the predictive mean and variance (Step 6), and the estimated level-set triplet (Step 7) based on the updated uniform bound. The ith epoch continues until the maximum half-width of the uniform bound at all points in the uncertain set diminishes to a given threshold $(1+\bar{\delta})\eta_{(i)}$ (Step 8), where $\bar{\delta} > 0$ is a tolerance parameter for the truncation target. Then, PVR enters the $(i+1)$th epoch (Step 9) and the threshold further decreases with $\eta_{(i)}$ shrinking by a factor of r with $r \in (0,1)$ (Step 10). We set the parameter values of $\bar{\delta}, r, \eta_{(i)}$, etc., following those adopted in [9].

Successfully obtaining $\mathbf{x}_t$ according to (9) relies on the ability to compute $\sigma_{t-1|\mathbf{x}^+}^2(\mathbf{x})$ for every candidate design point $\mathbf{x}^+$ efficiently. Because the closed-form expression of $\Delta_{t-1|\mathbf{x}^+}(\mathbf{x}) := \sigma_{t-1}^2(\mathbf{x}) - \sigma_{t-1|\mathbf{x}^+}^2(\mathbf{x})$ can be derived, the predictive variance conditional on $\mathbf{x}^+$ being the next design point, $\sigma_{t-1|\mathbf{x}^+}^2(\mathbf{x})$, can be conveniently obtained without actually performing simulation runs at each candidate design point $\mathbf{x}^+$. In particular, there are two cases to consider when deriving the expression for $\Delta_{t-1|\mathbf{x}^+}(\mathbf{x})$: the given candidate point $\mathbf{x}^+$ is either a new design point ($\mathbf{x}^+ \notin \mathcal{D}_{t-1}$) or an existing one ($\mathbf{x}^+ \in \mathcal{D}_{t-1}$). Proposition 2 summarizes the details whose proof is provided in Appendix B.

Algorithm 1. The predictive variance reduction approach for sequential level-set estimation

Input: Input space $\mathcal{X}$, the tolerance parameter for the truncation target $\bar{\delta} > 0$, the confidence-bound related parameters $r \in (0,1)$, $\{\beta_{(i)}\}_{i \geq 1}$, $\eta_{(1)} > 0$, the set of input points to be classified $\mathcal{P}$, the design related parameters K_0, n_0, and the threshold h.

1: Generate an initial design-point set $\mathcal{D}_0$ which consists of K_0 design points from $\mathcal{X}$ and set $\epsilon_0 = 0$.
2: Perform n_0 simulation replications at each design point in $\mathcal{D}_0$ and obtain the initial SK metamodel.
3: Set the epoch index $i = 1$, and initialize the uncertain set $M_0 = \mathcal{P}$ and the estimated level sets $H_0 = L_0 = \emptyset$.
4: **for** $t = 1, 2, \ldots$ **do.**
5: Select $\mathbf{x}_t$ according to (9), perform n_0 replications at $\mathbf{x}_t$, obtain the simulation output vector $\mathbf{y}(\mathbf{x}_t) = (y_1(\mathbf{x}_t), y_2(\mathbf{x}_t), \ldots, y_{n_0}(\mathbf{x}_t))$, and update the design-point set $\mathcal{D}_t = \mathcal{D}_{t-1} \cup \{\mathbf{x}_t\}$.
6: Update $\mu_t(\cdot)$ and $\sigma_t^2(\cdot)$ according to (2) and (3).
7: Update the estimated level-set triplet (H_t, L_t, M_t) according to (6), (7), and (8).
8: **while** $\max_{\mathbf{x} \in M_t} \beta_{(i)} \sigma_t(\mathbf{x}) \leq (1+\bar{\delta})\eta_{(i)}$ **do.**
9: $i \leftarrow i+1$;
10: $\eta_{(i)} \leftarrow r\eta_{(i-1)}$.
11: Update $\beta_{(i)}$ according to (10).
12: **end while**
13: **end for**

Proposition 2. *Denote the difference between the predictive variance at any input point $\mathbf{x} \in \mathcal{X}$ obtained on the $(t-1)$th iteration and that would be obtained on the tth iteration if $\mathbf{x}^+$ were selected to perform simulation runs as $\Delta_{t-1|\mathbf{x}^+}(\mathbf{x}) := \sigma_{t-1}^2(\mathbf{x}) - \sigma_{t-1|\mathbf{x}^+}^2(\mathbf{x})$. Then, if $\mathbf{x}^+ \notin \mathcal{D}_{t-1}$, $\Delta_{t-1|\mathbf{x}^+}(\mathbf{x})$ can be expressed as*

$$\Delta_{t-1|\mathbf{x}^+}(\mathbf{x}) = \frac{\mathrm{Cov}_{t-1}^2(\mathbf{x}, \mathbf{x}^+)}{\sigma_{t-1}^2(\mathbf{x}^+) + \frac{\mathrm{V}(\mathbf{x}^+)}{n_{t,\mathbf{x}^+}}},$$

where $\mathrm{Cov}_{t-1}(\mathbf{x}, \mathbf{x}^+) = \tau^2\big(K(\mathbf{x}, \mathbf{x}^+) - K(\mathbf{x}^+, \mathbf{X}_{t-1})\big(K(\mathbf{X}_{t-1}\mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\big)^{-1} K(\mathbf{x}^+, \mathbf{X}_{t-1})^\tau\big)$, *$n_{t,\mathbf{x}^+}$ denotes the number of replications to be allocated to $\mathbf{x}^+$, and $\sigma_{t-1}^2(\mathbf{x}^+)$ can be computed following (3) which equals* $\mathrm{Cov}_{t-1}(\mathbf{x}^+, \mathbf{x}^+)$.

If $\mathbf{x}^+ \in \mathcal{D}_{t-1}$, *without loss of generality, assume that it is the ith existing design point,* $\mathbf{x}_i$. *Then, for any* $\mathbf{x} \in \mathcal{X}$, $\Delta_{t-1|\mathbf{x}^+}(\mathbf{x}) = \Delta_{t-1|\mathbf{x}_i}(\mathbf{x})$ *and it takes the following form:*

$$\Delta_{t-1|\mathbf{x}_i}(\mathbf{x}) = \frac{\left(\left[K(\mathbf{x}, \mathbf{X}_{t-1})\left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)^{-1}\right]_{(i)}\right)^2}{\left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)^{-1}_{i,i} - \left(\tau^{-2}\left(\frac{\mathrm{V}(\mathbf{x}_i)}{n_{t-1,i}} - \frac{\mathrm{V}(\mathbf{x}_i)}{n_{t,i}}\right)\right)},$$

where $[\mathbf{b}]_{(i)}$ *represents the ith entry in vector* $\mathbf{b}$, $(\mathbf{A})_{ii}$ *denotes the ith diagonal entry of matrix* $\mathbf{A}$, *and* $n_{t,i}$ *denotes the number of replications allocated to the ith design point by the end of iteration t.*

We note that, despite that the next design point $\mathbf{x}_t$ is selected in Step 5 of Algorithm 1 while assuming n_0 simulation replications are to be allocated to each candidate point $\mathbf{x}^+$, one can adopt other choices of number of replications in Step 5 and leverage the general result given in Proposition 2 to facilitate efficient computations.

We next show that PVR can achieve the prescribed ϵ-accuracy guarantee for LSE under some conditions. We first state Lemma 1, which upper bounds $\beta_{k(t)}$, the high-probability uniform bound coefficient on iteration t as a function of the number of design points, $k(t)$. Such an upper bound helps define $\beta_{(i)}$ used in (9). The detailed proof for Lemma 1 is provided in Appendix C.1.

Lemma 1. *The coefficient of the uniform bound in* (4) *can be upper bounded as follows*:

$$\beta_{k(t)} \le \sqrt{k(t)\log\left(1 + \frac{\tau^2}{\mathrm{V}_{\min}(t)}\right) - 2\log\delta} + \tau^{-1}\|f\|_K, \tag{10}$$

where recall that $k(t)$ *denotes the number of design points up to iteration* t *and* $\mathrm{V}_{\min}(t) := \min_{i=1,2,\ldots,k(t)} \mathrm{V}(\mathbf{x}_i)/n_{t,i}$.

Lemma 1 is used in developing a form for $\beta_{(i)}$ to adopt in (9) for implementation. The coefficient of the uniform bound fixed for the ith epoch, $\beta_{(i)}$, should dominate $\beta_{k(t)}$ across all iterations within the ith epoch. Theorem 1 provides a specific form of $\beta_{(i)}$ based on Lemma 1. Before delving into Theorem 1, we first state Definition 3 and Assumption 2 which are used in developing Theorem 1.

Definition 3. A function F is submodular if and only if for all $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{X}$ and $\mathbf{x}' \in \mathcal{X}\backslash\mathcal{B}$, it holds that

$$F(\mathcal{A} \cup \{\mathbf{x}'\}) - F(\mathcal{A}) \ge F(\mathcal{B} \cup \{\mathbf{x}'\}) - F(\mathcal{B}).$$

Assumption 2. The kernel K satisfies that the predictive variance reduction function, $\psi_{t,\mathbf{x}}(S) = \sigma^2_t(\mathbf{x}) - \sigma^2_{t|S}(\mathbf{x})$, is submodular [25], for any iteration t, any design points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{k(t)}$, and any prediction point $\mathbf{x}$. Here, $S \subset \mathcal{X}$ is a set of input points to be added to the design-point set, and $\sigma^2_{t|S}(\mathbf{x})$ is the predictive variance upon performing simulation runs up to iteration t and at those input points in set S.

Assumption 2 has been shown to hold for the predictive variance of a GP with commonly used kernels K, such as squared exponential and Matérn kernels [12, 36]. The submodularity of the predictive variance essentially means that it satisfies the diminishing returns property: adding new design points reduces the predictive variance at any prediction point more if the existing design-point set is smaller.

Theorem 1. *Fix any* $\epsilon > 0$ *and* $\delta \in (0,1)$. *Suppose that Assumption 2 is fulfilled and that there exist sequences* $\{C_{(i)}\}$ *and* $\{\beta_{(i)}\}$ *such that*

$$C_{(i)} \ge C^*\left(\frac{\eta_{(i)}}{\beta_{(i)}}, \overline{M}_{(i-1)}\right)\log\left(\frac{|\overline{M}_{(i-1)}|\beta^2_{(i)}}{\bar{\delta}^2\eta^2_{(i)}}\right) + c_{\max}, \tag{11}$$

where

$$C^*(\xi, \mathcal{M}) := \min_{S\subset\mathcal{X}}\left\{c(S) : \max_{\mathbf{x}\in\mathcal{M}} \sigma_{0|S}(\mathbf{x}) \le \xi\right\} \tag{12}$$

is the minimum cost to achieve a predictive standard deviation of at most ξ within set $\mathcal{M}$, $|\mathcal{A}|$ denotes the cardinality of set $\mathcal{A}$, $c(S) = \sum_{j=1}^{|S|} c(\mathbf{x}'_j)$ denotes the total sampling cost for performing simulation runs at the input points in $S := \{\mathbf{x}'_1, \mathbf{x}'_2, \ldots, \mathbf{x}'_{|S|}\}$, $\overline{M}_{(i)} := \left\{\mathbf{x} \in \mathcal{X} \middle| |f(\mathbf{x}) - h| \le 2(1+\bar{\delta})\eta_{(i)}\right\}$, $c_{\max} := \max_{\mathbf{x}\in\mathcal{X}} c(\mathbf{x})$, and the sequence $\{\beta_{(i)}\}$ satisfies

$$\beta_{(i)} \ge \sqrt{\frac{\sum_{i'\le i} C_{(i')}}{c_{\min}}\log\left(1 + \frac{\tau^2}{\mathrm{V}_{\min}(t)}\right) - 2\log\delta} + \tau^{-1}\|f\|_K, \tag{13}$$

with $c_{\min} := \min_{\mathbf{x}\in\mathcal{X}} c(\mathbf{x})$. If PVR runs with such a choice of $\{\beta_{(i)}\}$, it achieves the ϵ-accuracy guarantee with probability at least $1-\delta$ given that the cumulative cost reaches $C_\epsilon = \sum_{i:4(1+\bar{\delta})\eta_{(i-1)}>\epsilon} C_{(i)}$.

Some remarks follow from Theorem 1 immediately. To achieve the ϵ-accuracy guarantee with a lower cumulative

cost C_ϵ, one can reduce the costs $C_{(i)}$, for $i = 1, 2, \ldots$, which depend on $C^*(\xi, \mathcal{M})$ and parameters such as $\{\beta_{(i)}\}$, $\{\eta_{(i)}\}$, and $\bar{\delta}$. According to the definition of $C^*(\xi, \mathcal{M})$ in (12), we see that two factors can potentially reduce its magnitude: (1) lower sampling costs at individual input points and hence a lower value of $c(S)$, and (2) lower noise variances at individual input points and hence a smaller set S (i.e., smaller $|S|$) to meet the target ξ in (12).

Next we consider an important special case of unit cost where $c(\mathbf{x}) = 1$ for all $\mathbf{x} \in \mathcal{X}$ which holds true in a wide range of stochastic simulation models. In this case, the total cumulative cost becomes proportional to T, *the total number of iterations consumed by PVR*. It follows from (11) and (13) that

$$T_{(i)} \geq T^*\left(\frac{\eta_{(i)}}{\beta_{(i)}}, \overline{M}_{(i-1)}\right) \log \frac{|M_{(i-1)}|\beta_{(i)}^2}{\bar{\delta}^2 \eta_{(i)}^2} + 1,$$

$$\beta_{(i)} \geq \left(\sum_{i' \leq i} T_{(i')} \cdot \log\left(1 + \frac{\tau^2}{\mathrm{V}_{\min}(t)}\right)\right)^{\frac{1}{2}} + \tau^{-1}\|f\|_K,$$

where $T_{(i)}$ denotes the index of the last iteration in the ith epoch for $i \geq 1$, and $T^*(\xi, \mathcal{M})$ represents the minimum number of iterations to achieve a predictive standard deviation of at most ξ within set $\mathcal{M}$. Moreover, to achieve the ϵ-accuracy guarantee, the total number of iterations T required by PVR follows as

$$T = \sum_{i:\, 4(1+\bar{\delta})\eta_{(i-1)} > \epsilon} T_{(i)}.$$

The following result presents an interpretable lower bound on the total number of iterations to achieve the ϵ-accuracy guarantee. It reveals the dependence on the minimum noise variance level and helps understand the corresponding impact of the noise in the simulation outputs. The proof of Corollary 3.2 is similar to that of Corollary D.1 in Appendix D of [9]. For the sake of brevity, we omit the details here.

Corollary 1. *Fix any* $\epsilon > 0$ *and* $\delta \in (0, 1)$. *Define* $\beta_T := \sqrt{T \log\left(1 + \frac{\tau^2}{\mathrm{V}_{\min}(T)}\right)} + \tau^{-1}\|f\|_K$, *and set* $\eta_{(1)} = 1$ *and* $r = 1/2$. *There exist choices of* $\beta_{(i)}$ (*not depending on* T) *such that PVR achieves the* ϵ*-accuracy guarantee with probability at least* $1 - \delta$ *if the total number of iterations satisfies*

$$T \geq \left(C_1 \gamma_T \beta_T^2 \frac{96(1+\bar{\delta})^2}{\epsilon^2} + 2\left\lceil \log_2 \frac{8(1+\bar{\delta})}{\epsilon} \right\rceil\right) \times \log\left(\frac{16(1+\bar{\delta})^2 |\mathcal{P}| \beta_T^2}{\bar{\delta}^2 \epsilon^2}\right),$$

where $C_1 = \left(\log\left(1 + \tau^2/\mathrm{V}_{\min}(T)\right)\right)^{-1}$, $\lceil a \rceil$ gives the least integer greater than or equal to $a \in \Re$, and $\gamma_T = \max_{D_T \subseteq \mathcal{X}} \frac{1}{2} \log\det\left(\mathbf{I}_{k(T)} + \tau^2 \Sigma_{\epsilon,k(T)}^{-1} K(\mathbf{X}_T, \mathbf{X}_T)\right)$. That is, $T \geq \Omega^*\left(C_1 \gamma_T \beta_T^2 \epsilon^{-2} + 1\right)$.

Despite fixing $\eta_{(1)} = 1$ and $r > 1/2$ in Corollary 1 for ease of analysis, we note that similar results can be obtained for other choices of $\eta_{(1)} > 0$ and $r \in (0, 1)$. Some insights into the impact of the simulation noise on T follow immediately. As $\mathrm{V}_{\min}(T) \to \infty$ (i.e., the minimum noise variance is high and hence the noise level is high across the input space), the lower bound on T has noise dependence $\mathcal{O}^*(\mathrm{V}_{\min}(T))$ since $\log\left(1 + \alpha^{-1}\right) = \mathcal{O}\left(\alpha^{-1}\right)$ as $\alpha \to \infty$. On the other hand, as $\mathrm{V}_{\min}(T) \to 0$ (i.e., the minimum noise variance is low), the impact of the simulation noise on the lower bound on T becomes negligible.

3.3 | Expected classification improvement

This subsection details the ECI approach for LSE. We first explain the key steps of ECI and provide closed-form expressions that facilitate its efficient implementation. Furthermore, we show that the maximum LSE error achieved by ECI is bounded by the given LSE error tolerance parameter ϵ_0 with a prescribed high probability. Last but not least, we close this section by unfolding a connection between PVR and ECI.

ECI intends to greedily minimize the number of input points that remain unclassified on each iteration. The detailed steps of ECI are summarized in Algorithm 2. On the tth iteration, ECI chooses the design point to perform simulation runs according to the following criterion:

$$\mathbf{x}_t := \arg\max_{\mathbf{x}^+ \in \mathcal{X}} \mathbb{E}\left[\left|HL_{t-1}\left(\mathbf{x}^+, \mathbf{y}(\mathbf{x}^+)\right)\right|\right] - |HL_{t-1}|, \quad (14)$$

where $HL_{t-1} := H_{t-1} \cup L_{t-1}$, $|HL_{t-1}|$ denotes the number of input points classified up to the $(t-1)$th iteration, and $\mathbb{E}\left[\left|HL_{t-1}\left(\mathbf{x}^+\mathbf{y}(\mathbf{x}^+)\right)\right|\right]$ denotes the expected number of input points that would be classified by the end of the tth iteration if $\mathbf{x}^+$ were chosen as the design point to perform simulation runs on iteration t. We note that $|HL_{t-1}|$ is fixed conditional on the sample path leading up to the tth iteration. The expectation in (14) is taken with respect to the distribution of $\mathbf{y}(\mathbf{x}^+)$ at the candidate design point $\mathbf{x}^+$. Once the design point $\mathbf{x}_t$ is chosen according to (14), n_0 replications are allocated to run the simulation model at $\mathbf{x}_t$.

Algorithm 2. The expected classification improvement approach for sequential level-set estimation

Input: Input space $\mathcal{X}$, the set of input points to be classified $\mathcal{P}$, the design related parameters K_0, n_0, the threshold h, and the LSE error tolerance parameter ϵ_0.

1: Generate an initial design point set $\mathcal{D}_0$ consisting of K_0 design points.
2: Perform n_0 simulation replications at each design point in $\mathcal{D}_0$ and obtain the initial SK metamodel.
3: Initialize the uncertain set $M_0 = \mathcal{P}$ and the estimated level sets $H_0 = L_0 = \emptyset$.
4: **for** $t = 1, 2, \ldots$ **do.**
5: Select $\mathbf{x}_t$ according to (14), perform n_0 replications at $\mathbf{x}_t$, obtain $\mathbf{y}(\mathbf{x}_t) := \left(y_1(\mathbf{x}_t), y_2(\mathbf{x}_t), \ldots, y_{n_0}(\mathbf{x}_t)\right)$, and update $\mathcal{D}_t = \mathcal{D}_{t-1} \cup \{\mathbf{x}_t\}$.
6: Update $\mu_t(\cdot)$ and $\sigma_t^2(\cdot)$ according to (2) and (3).
7: Update the estimated level-set triplet (H_t, L_t, M_t) according to (6), (7), and (8).
8: **end for**

To facilitate efficient implementation of ECI, we can obtain closed-form expressions for $\mathbb{E}|HL_{t-1}\left(\mathbf{x}^+, \mathbf{y}\left(\mathbf{x}^+\right)\right)|$ in (14) under Assumption 3 on normality of the simulation noise terms. As in Section 3.2, we consider two cases: the given candidate point $\mathbf{x}^+$ is either a new design point $\left(\mathbf{x}^+ \notin \mathcal{D}_{t-1}\right)$ or an existing design point $\left(\mathbf{x}^+ \in \mathcal{D}_{t-1}\right)$. Proposition 3 provides details, whose proof is given in Appendix D.

Assumption 3. The simulation noise terms $\varepsilon_1(\mathbf{x}), \varepsilon_2(\mathbf{x}), \ldots$ incurred at any $\mathbf{x} \in \mathcal{X}$ follow a normal distribution $\mathcal{N}(0, \mathrm{V}(\mathbf{x}))$.

Proposition 3. *Suppose that Assumption 3 is fulfilled. Denote the candidate design point on iteration t as $\mathbf{x}^+$. If $\mathbf{x}^+ \notin \mathcal{D}_{t-1}$, $\mathbb{E}\left[|HL_{t-1}(\mathbf{x}^+, \mathbf{y}\left(\mathbf{x}^+\right))|\right]$ can be written as*

$$\mathbb{E}\left[|HL_{t-1}\left(\mathbf{x}^+, \mathbf{y}\left(\mathbf{x}^+\right)\right)|\right] = \sum_{\mathbf{x}\in\mathcal{P}} \Phi\left(\frac{\sqrt{\sigma_{t-1}^2(\mathbf{x}^+) + \left(\mathrm{V}(\mathbf{x}^+)/n_{t,\mathbf{x}^+}\right)}}{|\mathrm{Cov}_{t-1}(\mathbf{x}, \mathbf{x}^+)|} \times c_{t-1}^+\left(\mathbf{x}|\mathbf{x}^+\right)\right) + \sum_{\mathbf{x}\in\mathcal{P}} \Phi\left(\frac{\sqrt{\sigma_{t-1}^2(\mathbf{x}^+) + \left(\mathrm{V}(\mathbf{x}^+)/n_{t,\mathbf{x}^+}\right)}}{|\mathrm{Cov}_{t-1}(\mathbf{x}, \mathbf{x}^+)|} \times c_{t-1}^-\left(\mathbf{x}|\mathbf{x}^+\right)\right),$$

where $c_{t-1}^+\left(\mathbf{x}\,|\,\mathbf{x}^+\right) := \mu_{t-1}(\mathbf{x}) - \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}^+}(\mathbf{x}) - h + \epsilon_0$, $c_{t-1}^-\left(\mathbf{x}|\mathbf{x}^+\right) := -\mu_{t-1}(\mathbf{x}) - \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}^+}(\mathbf{x}) + h + \epsilon_0$, *and* $\Phi(\cdot)$ *is the cumulative distribution function of the standard normal distribution. Recall that $n_{t,\mathbf{x}^+}$ denotes the number of replications to be allocated to $\mathbf{x}^+$.*

If $\mathbf{x}^+ \in \mathcal{D}_{t-1}$, without loss of generality, assume that $\mathbf{x}^+$ is the ith existing design point, $\mathbf{x}_i$. In this case, $\mathbf{X}_t = \mathbf{X}_{t-1}$. Then $\mathbb{E}\left[|HL_{t-1}\left(\mathbf{x}^+, \mathbf{y}\left(\mathbf{x}^+\right)\right)|\right]$ can be written as

$$\mathbb{E}\left[|HL_{t-1}(\mathbf{x}_i\mathbf{y}(\mathbf{x}_i))|\right] = \sum_{\mathbf{x}\in\mathcal{P}} \Phi\left(-\frac{R_i^+(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma_{t-1}^2(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right) + \sum_{\mathbf{x}\in\mathcal{P}} \Phi\left(\frac{R_i^-(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma_{t-1}^2(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right),$$

where

$$R_i^+(\mathbf{x}) = \frac{h - \epsilon_0 + \beta_{k(t-1)}\sigma_t(\mathbf{x}) - \mu_{t-1}(\mathbf{x}) - K(\mathbf{x}, \mathbf{X}_{t-1})D_i\bar{\mathbf{y}}_{t-1}}{|\frac{\Delta n_{t,i}C_i}{n_{t,i}}|} + \bar{y}_{t-1}(\mathbf{x}_i),$$

$$R_i^-(\mathbf{x}) = \frac{h + \epsilon_0 - \beta_{k(t-1)}\sigma_t(\mathbf{x}) - \mu_{t-1}(\mathbf{x}) - K(\mathbf{x}, \mathbf{X}_{t-1})D_i\bar{\mathbf{y}}_{t-1}}{|\frac{\Delta n_{t,i}C_i}{n_{t,i}}|} + \bar{y}_{t-1}(\mathbf{x}_i),$$

$$C_i = \left[K(\mathbf{x}, \mathbf{X}_{t-1})\left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\varepsilon,k(t-1)}\right)^{-1}\right]_{(i)},$$

$$D_i = \left(K(\mathbf{X}_t, \mathbf{X}_t) + \tau^{-2}\Sigma_{\varepsilon,k(t)}\right)^{-1} - \left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\varepsilon,k(t-1)}\right)^{-1},$$

$\bar{\mathbf{y}}_{t-1}$ *is a $k(t-1)\times 1$ vector of average outputs, $\Delta n_{t,i} := n_{t,i} - n_{t-1,i}$ denotes the number of replications allocated to $\mathbf{x}_i$ on the tth iteration, and recall that $n_{t,i}$ denotes the number of replications allocated to the ith design point by the end of iteration t.*

It is worthwhile noting that although the next design point $\mathbf{x}_t$ is selected in Step 5 of Algorithm 2 while assuming n_0 simulation replications are to be allocated to each candidate point $\mathbf{x}^+$, the number of replications can be adjusted in each iteration, and one can leverage the general result given in Proposition 3 to facilitate efficient computations.

Define the LSE error incurred at each $\mathbf{x} \in \mathcal{P}$ as $e_h(\mathbf{x}) = \max\{0, f_0(\mathbf{x}) - h\}$ if $\mathbf{x}$ is classfied into the estimated sub-level set and $e_h(\mathbf{x}) = \max\{0, h - f_0(\mathbf{x})\}$ if $\mathbf{x}$ is classified into the estimated super-level set. The next result reveals that the maximum LSE error achieved by ECI is bounded by ϵ_0 with a presecribed high probability.

Theorem 2. *Given $h \in \mathcal{T}$, fix any $\delta \in (0,1)$ and $\epsilon_0 > 0$. If the uniform bound in (5) is adopted for implementing ECI, $\mathrm{P}(\max_{\mathbf{x}\in\mathcal{P}} e_h(\mathbf{x}) \le \epsilon_0) \ge 1-\delta$ when all input points in $\mathcal{P}$ are classified by Algorithm 2.*

The proof of Theorem 2 follows from Definition 1 and that the uniform bound in (5) covers $f_0(\mathbf{x})$ at all $\mathbf{x} \in \mathcal{P}$ on iteration t for all $t \ge 1$.

We close this section by revealing the connection between the two proposed methods, PVR and ECI. The specific insight is articulated in Proposition 4, whose proof is provided in Appendix E.

Proposition 4. *Suppose that the sequential selection criterion of ECI is modified to the following one:*

$$\mathbf{x}_t = \arg\max_{\mathbf{x}^+\in\mathcal{X}} \mathbb{E}\int_{\mathcal{T}} \left(\left|HL_{t-1}^h\left(\mathbf{x}^+, \mathbf{y}(\mathbf{x}^+)\right)\right| - \left|HL_{t-1}^h\right|\right) dh \tag{15}$$

where h denotes the given LSE threshold, $|HL_{t-1}^h|$ denotes the number of input points classified up to the $(t-1)$th iteration for the given threshold h, and $|HL_{t-1}^h(\mathbf{x}^+, \mathbf{y}(\mathbf{x}^+))|$ is the number of input points classified up to the tth iteration if $\mathbf{x}^+$ were selected on the tth iteration. The modified ECI criterion in (15) is equivalent to

$$\mathbf{x}_t = \arg\min_{\mathbf{x}^+\in\mathcal{X}} \sum_{\mathbf{x}\in\mathcal{P}} \sigma_{t-1|\mathbf{x}^+}(\mathbf{x}). \tag{16}$$

Proposition 4 reveals that if we modify the selection criterion of ECI from (14) to the expectation of all possible threshold values as given in (15), then the modified ECI criterion seeks the next design point that minimizes the 1-norm of the predictive standard deviation. Recall that for PVR, if we deal with the unit cost case (i.e., $c(\mathbf{x}) = 1$) and ignore the truncated variance target $\eta_{(i)}$, the sequential selection criterion of PVR in (9) then reduces to $\arg\min_{\mathbf{x}^+\in\mathcal{X}} \sum_{\mathbf{x}\in M_{t-1}} \sigma^2_{t-1|\mathbf{x}^+}(\mathbf{x})$, which is in a similar spirit as (16), namely, focusing on predictive uncertainty reduction. The PVR selection criterion in (9) focuses on reducing the predictive uncertainty at the input points in the uncertain set, while the modified ECI criterion in (16) examines the entire prediction set $\mathcal{P}$. It is worth noting that ECI with the original selection criterion as given in (14) only considers one particular threshold h specified. Hence, we conjecture that, while tackling the LSE problems, PVR emphasizes more on the mean response function estimation accuracy achieved within the uncertain set while ECI focuses more on the classification accuracy at the input points regarding the given threshold h.

3.4 | Generalization of PVR and ECI

To better mitigate the impact of heteroscedasticity inherent in stochastic simulation outputs and further enhance the computational efficiency, we propose to incorporate a budget allocation scheme with PVR and ECI. On each iteration, upon selecting the next design point following either the PVR or ECI selection criterion and performing simulation runs at the selected design point, this scheme allocates additional Δn replications to each design point in set $\mathcal{S}$, the definition of which will be specified later in this subsection. Algorithm 3 details the major steps of the generalized LSE approaches which incorporate the budget allocation scheme.

Algorithm 3. A generalized LSE method with a budget allocation feature

Input: Input space $\mathcal{X}$, the design related parameters K_0, n_0, the additional number of replications Δn, and the threshold h.

1: Generate an initial design set $\mathcal{D}_0$ consisting of K_0 design points, perform n_0 simulation replications at each design point in $\mathcal{D}_0$, and obtain the initial SK metamodel.
2: Initialize the uncertain set $M_0 = \mathcal{P}$ and the estimated level sets $H_0 = L_0 = \emptyset$.
3: **for** $t = 1, 2, \ldots$ **do**.
4: Choose $\mathbf{x}_t$ according to either (9) for PVR or (14) for ECI, obtain the simulation outputs $\mathbf{y}(\mathbf{x}_t) := \left(y_1(\mathbf{x}_t), y_2(\mathbf{x}_t), \ldots, y_{n_0}(\mathbf{x}_t)\right)$, and update the design-point set $\mathcal{D}_t = \mathcal{D}_{t-1} \cup \{\mathbf{x}_t\}$.
5: Allocate $\Delta n_{t,i}$ replications according to (18) to each design point in set $\mathcal{S}$, obtain additional simulation outputs and update the sample mean and sample variances at each design point.
6: Update $\mu_t(\cdot)$ and $\sigma_t^2(\cdot)$ according to (2) and (3).
7: Update the estimated level-set triplet (H_t, L_t, M_t) according to (6), (7), and (8).
8: **end for**

The budget allocation scheme is incorporated in Step 5 of Algorithm 3, where the additional number of simulation replications to be allocated to each existing design point can be computed as follows. Inspired by the allocation rule adopted by Dieker and Kim [14], Frazier, and Xie and Chen [16, 43], we first calculate the number of replications that should be allocated to the ith design point in set $\mathcal{S}$ by the end of iteration t as follows:

$$n_{t,i}^* = \frac{\mathrm{V}(\mathbf{x}_i)}{\sum_{j\in\mathcal{S}} \mathrm{V}(\mathbf{x}_j)}(B_{\mathcal{S}} + \Delta n), \tag{17}$$

where $\mathcal{S}$ is the set of the existing yet unclassified design points, $B_{\mathcal{S}}$ is the total number of replications that have already been allocated to all points in $\mathcal{S}$, and Δn denotes the additional budget to be expended on each iteration. The allocation in (17) aims at making the variance of the sample means at all design points identical [14, 16] and hence potentially better mitigates the impact of heteroscedasticity. Thus, the number of additional replications to be allocated to each design point in set $\mathcal{S}$ in Step 5 follows as

$$\Delta n_{t,i} = \max\left\{0, n^*_{t,i} - n_{t,i}\right\}, \tag{18}$$

where $n_{t,i}$ denotes the number of replications that have actually been allocated to the ith design point in set $\mathcal{S}$. In implementation, we replace the unknown true noise variance $\mathrm{V}(\mathbf{x}_i)$ in (17) by the sample variance $\widehat{\mathrm{V}}(\mathbf{x}_i)$ at each design point $\mathbf{x}_i$.

Upon incorporating the budget allocation scheme, each iteration of the proposed LSE methods involves two steps: seeking the next design point and adjusting the budget allocated thus far to mitigate the impact of heteroscedasticity. We denote PVR and ECI with the budget allocation scheme incorporated as PVR-ts and ECI-ts, in contrast to their original versions introduced in Subsections 3.2 and 3.3. It is worth noting that the generalizations of PVR and ECI, PVR-ts and ECI-ts, possess the same theoretical performance guarantees as PVR and ECI do. This is because incorporating a budget allocation scheme at the design points does not impact the respective assumptions stipulated for PVR and ECI.

In practice, incorporating this additional budget allocation step does impact the performance of PVR and ECI given a fixed simulation budget. Specifically, this additional budget allocation step expedites the budget consumption by exploiting more at existing design points to tackle the impact of heteroscedasticity, leading to PVR-ts and ECI-ts terminating in fewer iterations and with a smaller design-point set. Denote T as the stopping time of a given method (i.e., the index of the iteration on which the simulation budget is exhausted). On the one hand, a smaller number of distinct design points $k(T)$ leads to less exploration over the input space, hence a higher predictive standard deviation $\sigma_T(\mathbf{x})$ at any $\mathbf{x} \in \mathcal{P}$, but also a lower bound coefficient $\beta_{k(T)}$. The net effect on the resulting LSE performance, relying on the uniform bound given in (5), is hard to determine. On the other hand, the additional budget allocation step can mitigate the impact of heteroscedasticity and potentially reduce the uniform bound width at input points in the neighborhood of the existing design points in the uncertain set. This could result in more input points being classified correctly, thereby enhancing the LSE performance relying on the uniform bound given in (5), especially when the impact of heteroscedasticity is severe.

The overall impacts on the performance of PVR and ECI are intractable to assess analytically but can be evaluated empirically. In the next section, we will focus on investigating the performance of the PVR-type and ECI-type methods in detail, comparing them with state-of-the-art LSE methods.

4 | NUMERICAL EXPERIMENTS

This section provides three numerical examples to evaluate the performance of proposed PVR-type and ECI-type methods in comparison with state-of-the-art benchmarking approaches. We start by giving a description of the general experimental setup, the benchmarking methods, the implementation details, and the performance metrics adopted. The detailed experimental settings and results for specific examples are presented in Sections 4.5 to 4.9.

4.1 | General experimental setup

In each example, we consider the LSE task of classifying the input points in a prediction set $\mathcal{P} \subset \mathcal{X}$ relative to a given threshold h with a fixed simulation budget. The values of h are specified corresponding to small, medium, and large mean response values in each example. The prediction set $\mathcal{P}$ consists of N input points, and the fixed simulation budget is specified as a total of B simulation replications.

4.2 | Methods in comparison

We consider comparing the performance of the proposed LSE methods with three state-of-the-art benchmarking methods, among which one is a model-free sampling method and the other two are metamodel-based approaches. Specifically, the model-free benchmarking method is the finite-budget allocation rule proposed in Shi et al. [35]. Given a fixed budget, FAR sequentially allocates simulation replications and has shown to outperform competing model-free sampling methods. FAR possesses both desirable finite-budget properties and asymptotic optimality for maximizing the posterior performance. Upon termination, FAR classifies the input points in $\mathcal{P}$ based on the sample means obtained at individual input points. The two metamodel-based benchmarking methods are respectively VAR and GCHK. VAR [19] sequentially selects the design point with the maximum predictive variance and hence is referred to as VAR:

$\mathbf{x}_t = \arg\ \max_{\mathbf{x}\in\mathcal{X}} \sigma_{t-1}(\mathbf{x})$. GCHK [19] is named after the authors' initials, which sequentially selects the next design point according to $\mathbf{x}_t = \arg\ \max_{\mathbf{x}\in M_t} a_t(\mathbf{x})$, where $a_t(\mathbf{x}) \coloneqq \min\{\max(C_t(\mathbf{x})) - h, h - \min(C_t(\mathbf{x}))\}$, $C_t(\mathbf{x}) = \cap_{i=1}^{t} Q_i(\mathbf{x})$, and $Q_i(\mathbf{x}) = \left[l_i(\mathbf{x}), u_i(\mathbf{x})\right]$ denotes the confidence bound obtained for the response of interest at $\mathbf{x}$ on the ith iteration. We note that VAR and GCHK classify the input points in $\mathcal{P}$ using confidence bounds. Regarding the proposed LSE methods in this work, we consider the PVR-type and ECI-type methods which include PVR, ECI, and their respective variants with the budget allocation scheme incorporated, i.e., PVR-ts and ECI-ts; see Section 3.4 for more details.

4.3 | Implementation details

The implementation of all methods starts with an initial stage followed by a main stage which comprises sequential iterations until the fixed budget is exhausted. For FAR, the initial stage estimates the mean response at each input point in $\mathcal{P}$ with two replications. On each iteration within the main stage, FAR allocates one simulation replication to the chosen input point from $\mathcal{P}$, which is selected by optimizing the posterior expectation of the classification accuracy [35]. Regarding VAR and GCHK, the initial stage estimates the metamodel hyper-parameters based on simulation output data obtained at K_0 initial design points, which are a Latin hypercube sample from the input space $\mathcal{X}$, with n_0 simulation replications allocated to each design point. On each iteration within the main stage of VAR and GCHK, one input point is chosen from $\mathcal{X}$ according to their respective point selection criteria and n_0 simulation replications are allocated there. The proposed PVR-type and ECI-type methods are implemented according to Algorithms 1 to 3. For fair comparisons, we employ the SK metamodel with a squared exponential kernel and the uniform bound given in (5) for all metamodel-based LSE methods (i.e., VAR, GCHK, and the proposed PVR-type and ECI-type methods) to classify the input points on each iteration, with the nominal confidence level set to $1 - \delta = 0.95$. For the PVR-type methods, we adopt $\overline{\delta} = 0$, $r = 0.5$, $\eta_{(0)} = 1$, and $\epsilon_0 = 0$ following [9]. For the ECI-type methods, the tolerance parameter is set to $\epsilon_0 = 0.05$.

4.4 | Evaluation metrics

A total of $M = 50$ independent macro-replications are conducted for comparing performance of all methods under consideration. We adopt the F1 score which is a widely used performance metric for evaluating LSE performance in the literature [9, 19, 45]. The F1 score balances precision and recall, giving equal consideration to false positives and false negatives. This feature enables the F1 score to more accurately and reliably capture a candidate method's performance, especially when dealing with imbalanced super-level and sub-level sets. On each macro-replication, we record the evolving F1 score of each metamodel-based method throughout iterations, with the F1 score on the tth iteration defined as

$$F1_t = \frac{|H_t \cap H|}{|H_t \cap H| + (|H_t \cap L| + |L_t \cap H| + |M_t|)/2}, \quad t \geq 1. \tag{19}$$

We refer to $F1_t$ in (19) as the *conservative F1 score* since all input points in the uncertain set M_t obtained by a given metamodel-based LSE method are counted as wrongly classified if $M_t \neq \emptyset$. Notice that FAR uses the point estimates obtained at the input points for LSE and its corresponding $M_t \equiv \emptyset$ for $t \geq 1$. Hence, we do not evaluate FAR using the its use within the metamodel-based methods to avoid an unfair comparison of FAR and the metamodel-based methods.

For a fair comparison of all methods under consideration, inspired by the metric adopted by [9], we use the *final F1 score* achieved when the budget is exhausted on each macro-replication, which is defined as

$$\tilde{F1} = \frac{|\tilde{H}_T \cap H|}{|\tilde{H}_T \cap H| + \left(|\tilde{H}_T \cap L| + |\tilde{L}_T \cap H|\right)/2}, \tag{20}$$

where recall that T denotes the stopping time of a given method (i.e., the index of the last iteration upon termination), and $\tilde{H}_T \coloneqq H_T \cup MH_T$ with MH_T denoting the set of input points in M_T that are classified to the super-level set by the point estimate given in (2) for the metamodel-based methods upon termination, and $\tilde{L}_T \coloneqq L_T \cup ML_T$ with ML_T denoting the set of input points in M_T that are classified to the sub-level set using the point estimate given in (2) upon termination. For FAR, since all points are classified on each iteration, we have $\tilde{H}_T = H_T$ and $\tilde{L}_T = L_T$. Lastly, to evaluate the point estimation accuracy upon termination of each given method, we adopt the root mean squared error (RMSE), which is defined as

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(f_0(\mathbf{x}_i) - \hat{f}_0(\mathbf{x}_i)\right)^2},$$

where $\hat{f}_0(\mathbf{x}_i)$ denotes the point estimate obtained at input point $\mathbf{x}_i$, and recall $N = |\mathcal{P}|$ denotes the number of input points in the prediction set $\mathcal{P}$.

4.5 | Numerical Examples

4.5.1 | A one-dimensional trigonometric function example

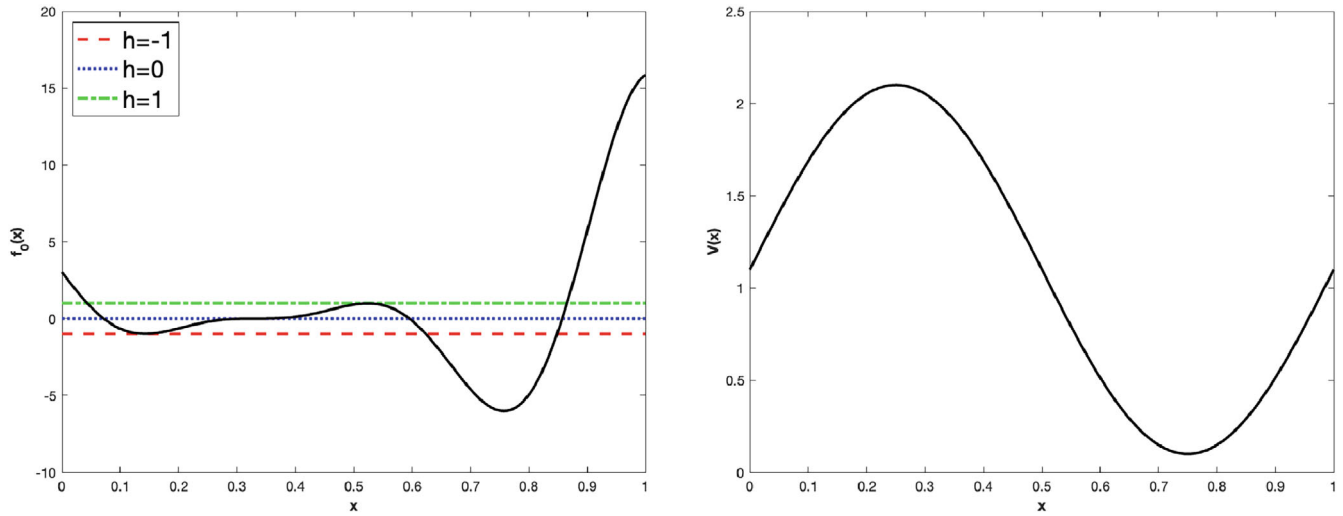
Consider the following 1-D trigonometric function example, where the mean response surface of interest is $f_0(x) = (6x-2)^2 \sin(12x-4)$ and the noise variance function is $\mathrm{V}(x) = 1.1 + \sin(2\pi x)$, for $x \in \mathcal{X} = [0,1]$. The simulation outputs at x are generated according to (1) where the simulation noise is normally distributed with mean zero and variance $\mathrm{V}(x)$. We seek to perform LSE regarding a given threshold h for the input points in set $\mathcal{P}$ which consists of a grid of $N = 1000$ equispaced points in $\mathcal{X}$. Three threshold values are considered, respectively, $h = -1$, 0, and 1. The total budget is $B = 5100$ simulation replications, and the parameters for metamodel-based methods are set to $K_0 = 10$, $n_0 = 10$, and $\Delta n = 100$.

The mean and noise variance functions are respectively illustrated in Figure 1A,B. We see from Figure 1 that identifying level sets is particularly challenging for the first half of the input space, [0,0.5], due to high variances and low absolute mean values. Additionally, among the three thresholds considered, $h = 0$ is the most challenging case because a greater proportion of the input points in $\mathcal{P}$ have their true mean values around $h = 0$; see Figure 1A for details.

Summary of results

Figure 2 shows the evolving performance of all metamodel-based methods (i.e., VAR, GCHK, PVR-type, and ECI-type methods) in terms of the conservative F1 scores obtained on an arbitrarily chosen macro-replication, which is representative of the 50 independent macro-replications. The following observations can be made. First, the ECI-type methods deliver the best performance, followed by VAR and the PVR-type methods, and GCHK ranks last. Second, compared to PVR (respectively, ECI), PVR-ts (resp., ECI-ts) improves the conservative F1 scores by incorporating the budget allocation step. Lastly, GCHK's performance stays at a non-competitive level with little changes despite the increase in the allocated budget. A similar pattern is observed for PVR. However, thanks to the budget allocation step, PVR-ts improves the conservative F1 scores by a substantial margin.

Table 1 summarizes the final F1 scores achieved by all methods upon termination in 50 independent macro-replications. Recall the definition of the final F1 score in (20); it reflects the point estimation accuracy achieved by FAR and by the metamodel-based methods (to some extent) upon termination, because the latter adopts the point estimates in (2) to classify those input points remaining in the uncertain set upon termination. We have the following observations from Table 1. First, compared to the metamodel-based LSE methods, FAR yields the lowest final F1 scores, together with a small standard error, indicating that FAR's performance has slight variation and our conclusions regarding its performance relative to the metamodel-based methods are robust. Second, among the metamodel-based LSE methods, the PVR-type methods and GCHK perform best, immediately followed by the ECI-type methods and VAR. Third, incorporating the budget allocation step helps ECI-ts achieve higher final F1



(A) The mean function f_0 (B) The noise variance function V

FIGURE 1 The true mean and noise variance functions in the 1-D trigonometric function example. In (A), the solid line represents the mean function; and the dashed, dotted, and dashed-dotted lines respectively correspond to the three threshold values, $h = -1$, 0, and 1.

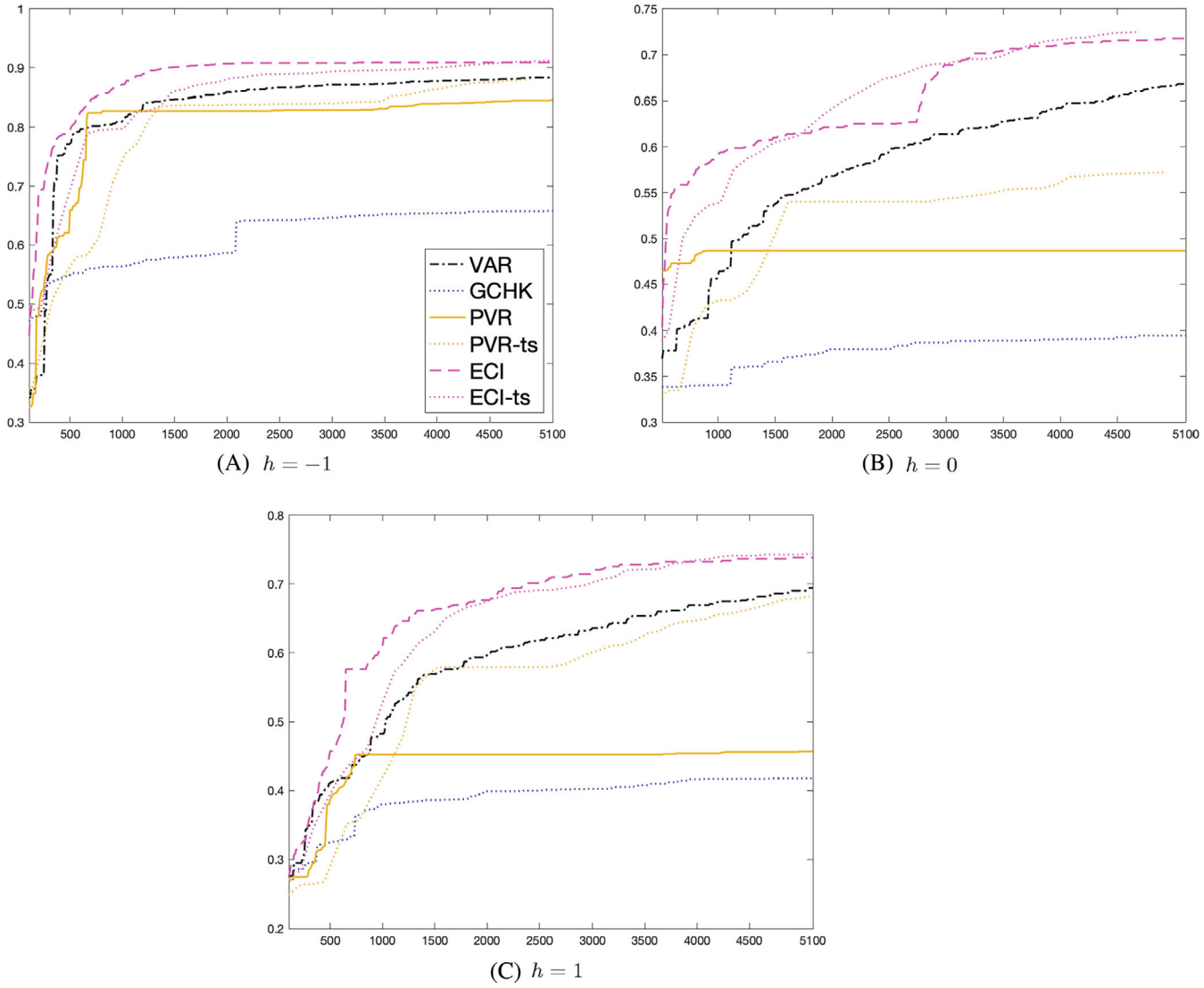


FIGURE 2 The 1-D trigonometric function example: the conservative F1 scores obtained by the metamodel-based methods against the consumed simulation budget on an arbitrarily chosen macro-replication.

TABLE 1 Summary of the average and the standard error (in parentheses) of the final F1 scores achieved by different LSE methods across 50 macro-replications in the 1-D trigonometric function example.

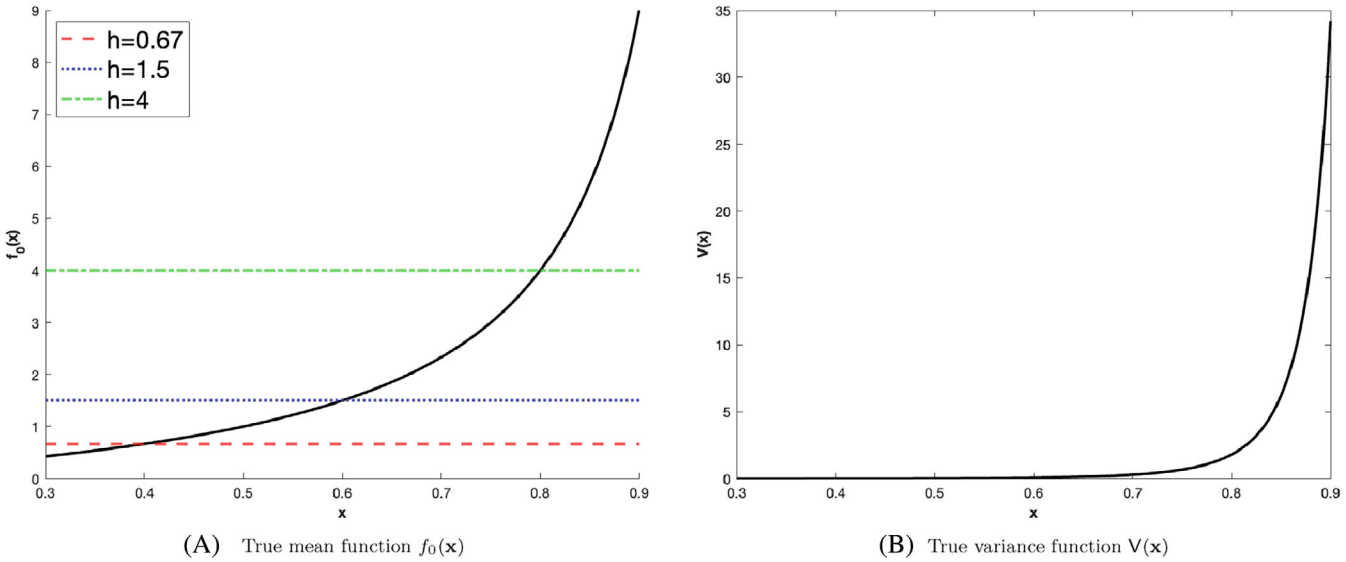
h	FAR	VAR	GCHK	PVR	PVR-ts	ECI	ECI-ts
−1	0.953	0.990	0.999	0.995	0.976	0.990	0.993
	(0.001)	(0.002)	(0.001)	(0.001)	(0.002)	(0.002)	(0.001)
0	0.855	0.947	0.961	0.949	0.964	0.927	0.949
	(0.002)	(0.003)	(0.003)	(0.002)	(0.002)	(0.004)	(0.004)
1	0.835	0.967	0.993	0.980	0.949	0.982	0.986
	(0.002)	(0.006)	(0.002)	(0.004)	(0.007)	(0.004)	(0.003)

scores than ECI. On the other hand, the budget allocation step does not seem to bring as much benefit to PVR in terms of the final F1 scores, except in the case of $h = 0$, which is also the most challenging case to tackle.

Lastly, Table 2 summarizes the point estimation accuracy in terms of RMSE for all methods, which helps explain observations made earlier. We see that FAR yields the largest RMSEs hence the worst point estimation

TABLE 2 Summary of the average and the standard error (in parentheses) of the root mean squared errors achieved by different level-set estimation methods across 50 macro-replications in the 1-D trigonometric function example.

h	FAR	VAR	GCHK	PVR	PVR-ts	ECI	ECI-ts
−1	0.705	0.108	0.300	0.216	0.264	0.096	0.142
	(0.003)	(0.003)	(0.010)	(0.008)	(0.010)	(0.007)	(0.007)
0	0.727	0.106	0.286	0.204	0.232	0.113	0.139
	(0.002)	(0.003)	(0.001)	(0.008)	(0.009)	(0.005)	(0.008)
1	0.683	0.103	0.281	0.178	0.219	0.087	0.132
	(0.003)	(0.003)	(0.008)	(0.007)	(0.008)	(0.005)	(0.007)



(A) True mean function $f_0(\mathbf{x})$ (B) True variance function $\mathrm{V}(\mathbf{x})$

FIGURE 3 The true mean and noise variance functions for the M/M/1 queueing example. In (A), the solid line represents the mean response surface and the dashed, dotted, and dashed-dotted lines represent the three threshold values, $h = 0.6$, 1.5, and 4, respectively.

accuracy and consequently the lowest final F1 scores as shown in Table 1. Furthermore, we find that the RMSEs of the PVR-type methods and GCHK dominate those of the ECI-type methods and VAR; however, the former's final F1 scores are comparable to, if not much better than, the latter's. This is because the PVR-type methods and GCHK emphasize reducing the uncertainty in estimating the mean function values at points in the uncertain set, which gives them an advantage in terms of final F1 scores. However, they sacrifice estimation accuracy at points already classified, where the RMSEs dominate in scale in this example.

4.5.2 | An M/M/1 queueing example

The M/M/1 queue is a classical one-dimensional example in stochastic simulation, which simulates a single-server queueing system. The customers arrive to the system according to a Poisson process with arrival rate x customers/time and are served according to a first-come, first-served discipline. A single server serves customers one at a time and the service times of the customers are i.i.d. exponentially distributed with rate fixed at one customer/-time. In this example, the input variable is the arrival rate x with the input space $\mathcal{X} = [0.3,0.9]$. The simulation output recorded on each replication at a given design point is the average number of customers in the system from time 0 to T. The mean response surface of interest is the steady-state mean number of customers in the queue $f_0(x) = x/(1-x)$. The noise variance function is $\mathrm{V}(x) \approx 2x(1+x)/\left(T(1-x)^4\right)$ for T large [41]. When simulating the M/M/1 queue at each design point, we initialize each replication in steady state and set the run length of each replication T to 1000 time units.

In this experiment, the total budget is $B = 5100$ simulation replications, and the parameters for metamodel-based methods are set to $K_0 = 10$, $n_0 = 10$, and $\Delta n = 100$. The set of input points for LSE, $\mathcal{P}$, comprises a grid of $N = 1000$ equispaced points in $\mathcal{X}$. Three threshold values are considered in this example, $h = 0.67$, 1.5, and 4, respectively

corresponding to the input values $x = 0.4$, 0.6, and 0.8. The mean and noise variance functions are illustrated in Figure 3A,B. We see that both the mean and the noise variance functions hike with the input x, with the latter's increasing trend being more dramatic; this indicates that identifying level sets becomes progressively challenging as the threshold value h grows.

Summary of results

Figure 4 shows the evolving conservative F1 scores of all metamodel-based methods against the consumed simulation budget on an arbitrarily chosen macro-replication, which is representative of the 50 independent macro-replications. We have the following observations. First, the ECI-type methods outperform the other methods with a noticeable gap. Moreover, the higher conservative F1 scores of ECI-ts (resp., PVR-ts) show that incorporating the budget allocation step brings an improvement to ECI (resp., PVR). Second, regarding the two benchmarking metamodel-based methods, VAR's performance is noncompetitive in all three cases compared to the other metamodel-based methods despite the increase in the allocated budget. On the other hand, GCHK demonstrates satisfactory performance, especially as the allocated budget increases.

Table 3 summarizes the final F1 scores obtained by all methods upon termination in 50 macro-replications. Several observations can be made. First, FAR's performance degrades significantly as the LSE difficulty rises, with its worst performance observed in the case of $h = 4$. Second, among the metamodel-based approaches, the PVR-type methods' performance is comparable to that of the ECI-type methods. Both types of methods outperform GCHK and VAR. Moreover, as h increases, the

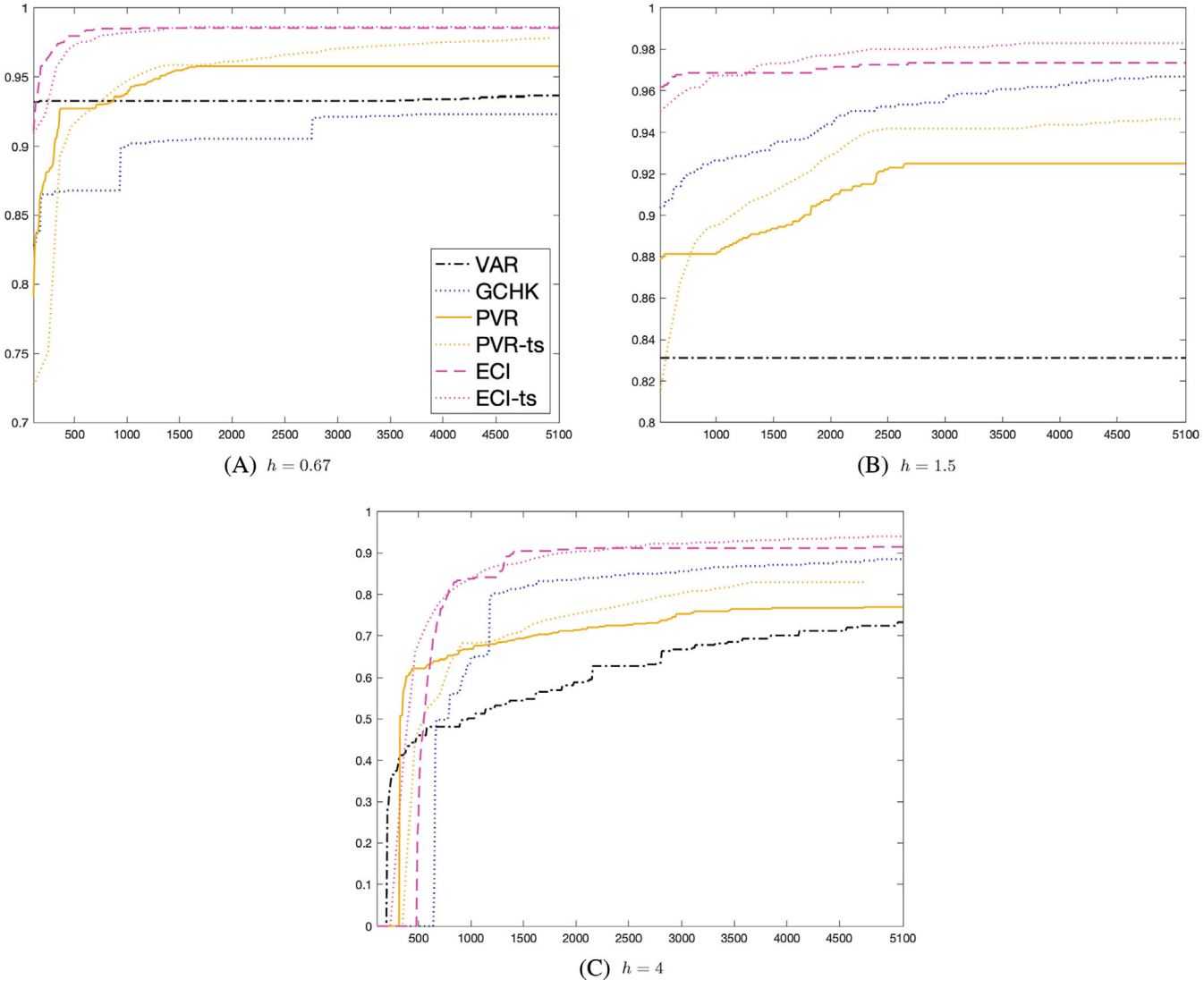


FIGURE 4 The M/M/1 queueing example: the conservative F1 scores obtained by the metamodel-based methods against the consumed simulation budget on an arbitrarily chosen macro-replication.

TABLE 3 Summary of the average and the standard error (in parentheses) of the final F1 scores achieved by all methods across 50 macro-replications in the M/M/1 example.

h	FAR	VAR	GCHK	PVR	PVR-ts	ECI	ECI-ts
0.67	0.990	0.992	0.999	0.999	0.986	0.999	0.999
	(0.001)	(0.001)	(0.001)	(0.001)	(0.012)	(0.001)	(0.001)
1.5	0.978	0.983	0.997	0.998	0.991	0.997	0.999
	(0.001)	(0.001)	(0.001)	(0.001)	(0.004)	(0.001)	(0.001)
4	0.899	0.938	0.963	0.986	0.978	0.988	0.994
	(0.003)	(0.006)	(0.002)	(0.001)	(0.002)	(0.001)	(0.003)

TABLE 4 Summary of the average and the standard error (in parentheses) of the root mean squared errors achieved by all methods across 50 macro-replications in the M/M/1 queueing example.

h	FAR	VAR	GCHK	PVR	PVR-ts	ECI	ECI-ts
0.67	0.865	0.821	0.974	0.579	0.682	0.339	0.326
	(0.009)	(0.034)	(0.030)	(0.027)	(0.065)	(0.022)	(0.021)
1.5	0.852	0.847	0.965	0.436	0.651	0.461	0.415
	(0.009)	(0.033)	(0.035)	(0.024)	(0.054)	(0.028)	(0.024)
4	0.850	0.835	0.693	0.220	0.405	0.238	0.102
	(0.010)	(0.033)	(0.018)	(0.012)	(0.013)	(0.016)	(0.006)

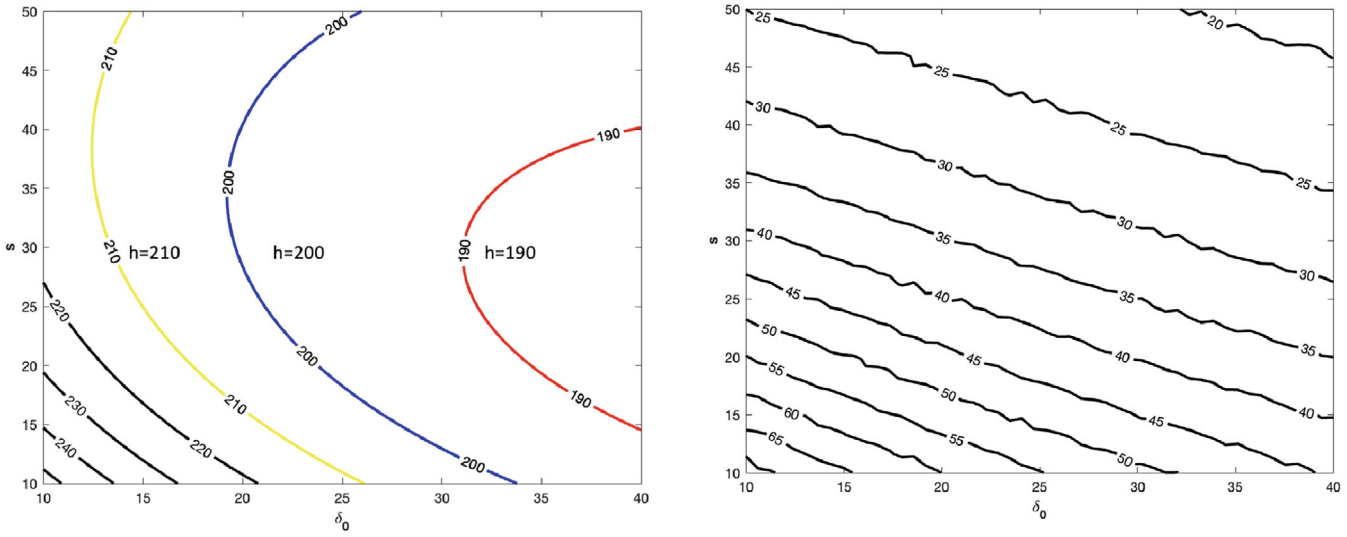
performance of GCHK and VAR deteriorates more rapidly than that of the PVR-type and ECI-type methods. Third, the budget allocation scheme benefits ECI in achieving higher final F1 scores across all three threshold values considered, while this is not observed for PVR.

Table 4 summarizes the point estimation accuracy in terms of RMSE achieved by all methods in 50 macro-replications. First, the RMSEs obtained by FAR are among the worst across all three threshold values, especially in the case of $h = 4$. This helps explain FAR's low final F1 scores as shown in Table 3, as FAR solely uses point estimates for LSE. Second, comparing the RMSEs obtained by the metamodel-based approaches serves as a valuable indicator of their relative performance in terms of the final F1 scores. Specifically, higher RMSEs correspond to lower final F1 scores, especially true in the case of $h = 4$. This relationship is particularly relevant in this M/M/1 queueing example, where the uncertain set includes a considerable number of input points that remain unclassified based on the uniform bound when the simulation budget is exhausted and must be classified by the point estimates. This situation arises due to the high heteroscedasticity present in this example. The advantage of the PVR-type and ECI-type methods is worth noting, as they consistently demonstrate high estimation accuracy. Despite being devised originally for LSE, the PVR-type and ECI-type methods exhibit robust performance in this example, underscoring their versatility and effectiveness in different scenarios.

4.5.3 | A periodic review (s, S) inventory system

In this subsection, we consider a periodic review single-commodity (s, S) inventory system that supplies external demands and receives stock from a production facility. The stock level, s, is at which a new order for the target product should be placed, and S is the maximum stock level that should be maintained. Assume that the system has i.i.d. continuous demands, zero lead times, full backlogging, and linear ordering, holding, and shortage costs, as considered in Fu and Healy and Wang and Chen [17, 40]. The (s, S) inventory system works as follows. Let X_i denote the stock level of the target product. When X_i is below s units, an order of amount $(S - X_i)$ is made which incurs a fixed ordering cost K and a total purchase cost $c_0(S - X_i)$ where c_0 is the unit purchase cost. The unit holding cost is h_0, and the unit shortage cost is p. The cost in period i is the sum of ordering, holding, and backorder costs which is given by

$$J_i = \mathbf{1}\{X_i < s\}(K + c_0(S - X_i)) + h_0 \max\{0, W_i\} + p \max\{0, -W_i\}.$$



(A) True mean function $J(\delta_0, s)$ (B) Estimated noise variance function $\widehat{V}(\delta_0, s)$

FIGURE 5 The contour plots of (A) the mean response function $J(\delta_0, s)$ and (B) the estimated noise variance function $\widehat{V}(\delta_0, s)$ using a large computational budget at each input point in the (s, S) inventory system example.

We are interested in the long-run average cost per period J, which is defined as

$$J = \lim_{n\to\infty} J_n, \quad \text{with } J_n = \frac{1}{n}\sum_{i=1}^{n} J_i.$$

Assume that the demands are i.i.d. exponentially distributed with mean E[D]. Define $\delta_0 = S - s$ and let $\lambda = 1/\mathrm{E}[D]$. We are interested in estimating the unknown mean response function $J(\delta_0, s)$ whose analytical form is given by

$$J(\delta_0, s) = c_0\mathbb{E}(D) + \frac{K + h_0(s - \mathbb{E}(D) + \lambda\delta_0(s + \delta_0/2)) + (h_0 + p)\mathbb{E}(D)\exp(-\lambda s)}{1 + \lambda\delta_0}.$$

In our experiment, we adopt $c_0 = 5$, $\mathbb{E}(D) = 20$, $K = 100$, $h_0 = 1$, and $p = 10$. The input space is $\mathcal{X} := \Omega_{\delta_0} \times \Omega_s = [10, 40] \times [10, 50]$. On each simulation replication at a given design point (δ_0, s), we use the run length of $T = 1000$ periods to estimate $J(\delta_0, s)$. The prediction set $\mathcal{P}$ contains $N = 1000$ input points, comprising 996 Latin-hypercube sampled input points from $\mathcal{X}$ plus the four corner points of $\mathcal{X}$. The three threshold values are $h = 190$, 200, and 210. The total budget is $B = 10500$ simulation replications, and the parameters for metamodel-based methods are set to $K_0 = 25$, $n_0 = 20$, and $\Delta n = 100$. The contour plots for the mean function $J(\delta_0, s)$ and the approximated noise variance function $\widehat{V}(\delta_0, s)$ are shown in Figure 5A,B. Notice that a closed-form expression of $V(\delta_0, s)$ is unavailable and it is estimated by the sample variance $\widehat{V}(\delta_0, s)$ based on 10^5 replications at each input point. We see from Figure 5A,B that as the threshold value h increases, the LSE difficulty rises due to an increasing trend of noise variances.

Summary of results

Figure 6 shows the evolving conservative F1 scores of all metamodel-based methods against the consumed simulation budget on an arbitrarily chosen macro-replication, which is representative of the 50 independent macro-replications. The following observations can be made. First, the ECI-type methods perform best, followed by the PVR-type methods and GCHK; and VAR yields the worst performance. Second, compared to PVR (respectively, ECI), PVR-ts (resp., ECI-ts) improves the conservative F1 scores by incorporating the budget allocation step.

Table 5 summarizes the final F1 scores obtained by all methods upon termination in 50 independent macro-replications. Several observations can be made from the table. First, FAR achieves the lowest final F1 scores due to its low point estimation accuracy. Regarding the metamodel-based approaches, the ECI-type methods perform the best, followed by the PVR-type, GCHK, and VAR. However, the difference in performance between ECI and PVR is small. Third, incorporating the budget allocation step helps ECI-ts achieve higher final F1 scores than ECI. On the other hand, the budget allocation scheme does not seem to bring as much benefit to PVR in terms of the final F1 scores.

Table 6 summarizes the point estimation accuracy in terms of RMSE achieved by all methods in 50 macro-replications. First, the RMSEs obtained by FAR are among the worst across all three threshold values,

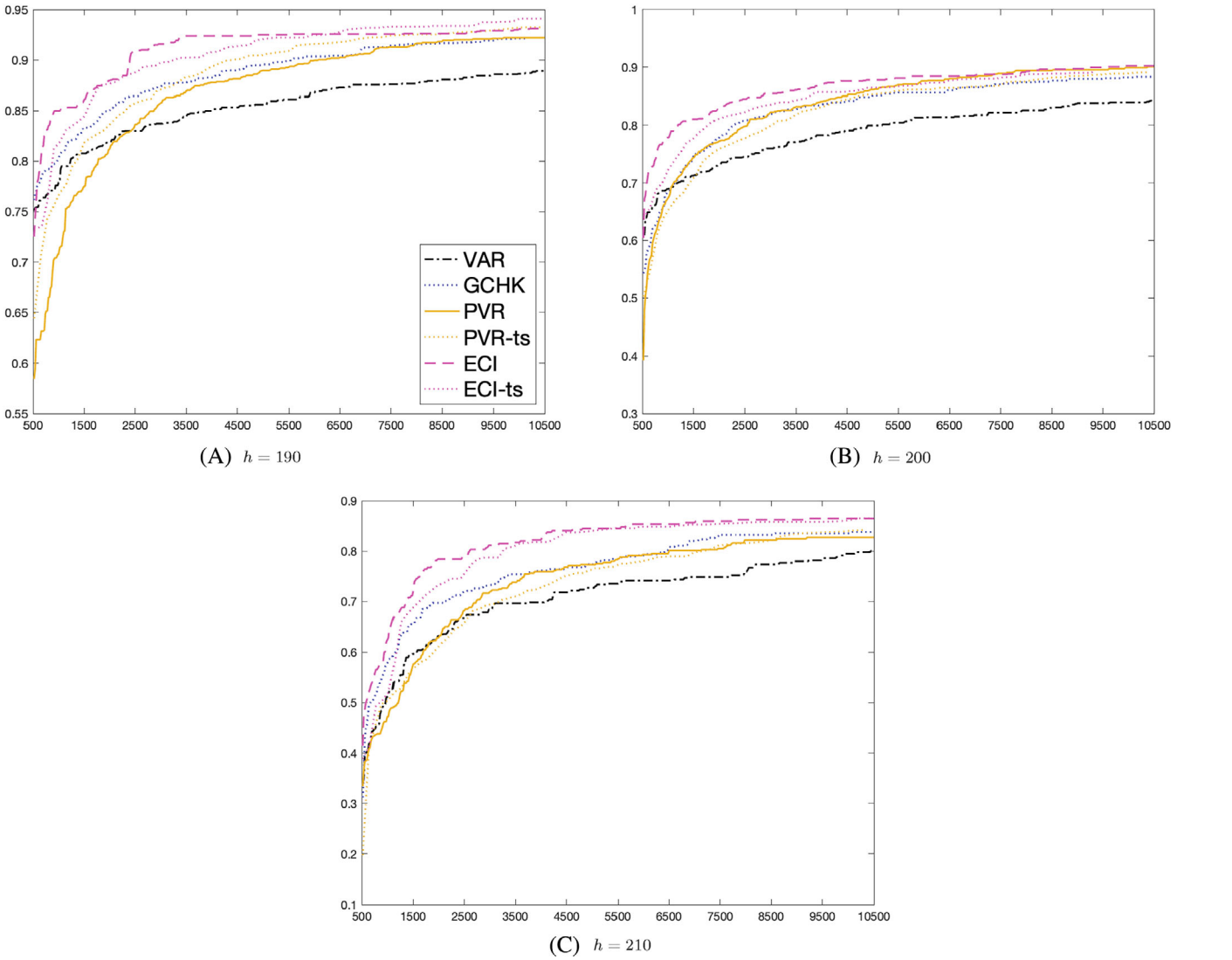


FIGURE 6 The (s, S) inventory example: the conservative F1 scores obtained by the metamodel-based methods against the consumed simulation budget on an arbitrarily chosen macro-replication.

TABLE 5 Summary of the average and the standard error (in parentheses) of the final F1 scores achieved by all methods across 50 macro-replications in the (s, S) inventory system example.

h	FAR	VAR	GCHK	PVR	PVR-ts	ECI	ECI-ts
190	0.963	0.985	0.991	0.992	0.989	0.996	0.997
	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)
200	0.943	0.980	0.984	0.994	0.987	0.994	0.994
	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)
210	0.923	0.958	0.971	0.986	0.970	0.991	0.994
	(0.002)	(0.002)	(0.001)	(0.001)	(0.002)	(0.001)	(0.001)

which helps explain FAR's low final F1 scores as shown in Table 5. Second, the ECI-type and PVR-type methods demonstrate relatively high estimation accuracy. Despite being originally designed for tackling LSE problems, the PVR-type and ECI-type methods exhibit robust performance in this example. Third, in contrast to the M/M/1 queueing example, the relatively large differences in the RMSEs do not get reflected as much in the differences in the final F1 scores. This is because the impact of heteroscedasticity in this example is not as dramatic as in the M/M/1 queueing example, making the LSE task relatively easier to tackle by all methods.

TABLE 6 Summary of the average and the standard error (in parentheses) of the root mean squared errors achieved by all methods across 50 macro-replications in the (s, S) inventory system example.

h	FAR	VAR	GCHK	PVR	PVR-ts	ECI	ECI-ts
190	15.760	0.961	1.366	0.697	0.933	0.424	0.570
	(0.053)	(0.020)	(0.054)	(0.021)	(0.019)	(0.014)	(0.025)
200	15.874	0.973	1.188	0.535	0.924	0.331	0.487
	(0.048)	(0.020)	(0.036)	(0.011)	(0.017)	(0.012)	(0.017)
210	15.449	0.952	1.098	0.560	0.902	0.374	0.522
	(0.237)	(0.017)	(0.032)	(0.019)	(0.026)	(0.015)	(0.014)

We close this section with remarks on the performance of all methods based on the three numerical examples. First, metamodel-based methods yield better performance on final F1 scores than the state-of-the-art model-free sampling method FAR. This is unsurprising since the metamodel-based methods can leverage simulation outputs obtained at all design points for function approximation across the input space and achieve higher estimation accuracy than the sample mean obtained based on outputs at each input point. Second, compared to the metamodel-based benchmarking methods (i.e., VAR and GCHK), the proposed PVR-type and ECI-type methods show more robust performance regarding both the conservative F1 scores and the final F1 scores in different scenarios. Third, incorporating the budget allocation step helps potentially improve PVR's performance regarding the conservative F1 scores and ECI's performance in terms of both the conservative F1 scores and the final F1 scores. Last but not least, our numerical experiments show that the metamodel-based LSE task differs from the metamodel-based prediction task: achieving high point estimation accuracy does not guarantee superior LSE performance, and vice versa.

5 | CONCLUSION

In this work, we proposed two metamodel-based LSE methods suitable for the stochastic simulation setting, PVR and ECI. We provided insights into their respective characteristics and established the connection between these two methods. We also incorporated a budget allocation feature with PVR and ECI to better tackle heteroscedasticity's impact, which is prevalent in stochastic simulation experiments. Numerical examples demonstrated the superior performance of the proposed methods compared to benchmarking approaches. Given a fixed total budget, we found that the metamodel-based LSE methods outperform the state-of-the-art model-free type method, FAR. Moreover, the metamodel-based LSE task differs from the metamodel-based prediction task, which deserves separate, in-depth investigations.

We aim to extend this work in two key directions. The first is to develop theoretically sound and computationally efficient sequential metamodel-based LSE methods that can scale effectively to large heteroscedastic datasets. The second direction is to devise a principled budget allocation scheme that balances exploitation and exploration for metamodel-based LSE, especially when given a fixed simulation budget to expend.

ACKNOWLEDGMENTS

This paper is based upon work supported by the National Science Foundation [IIS-1849300] and NSF CAREER [CMMI-1846663].

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Xi Chen https://orcid.org/0000-0001-7965-9198

REFERENCES

1. X. Chen and Q. Zhou, *Sequential design strategies for mean response surface metamodeling via stochastic kriging with adaptive exploration and exploitation*, Eur. J. Oper. Res. 262 (2017), no. 2, 575–585.
2. Y. Abbasi-Yadkori, *Online learning for linearly parametrized control problems*. PhD thesis, University of Alberta, Edmonton, Alberta, Canada, 2013.
3. B. Ankenman, B. L. Nelson, and J. Staum, *Stochastic kriging for simulation metamodeling*, Oper. Res. 58 (2010), no. 2, 371–382.
4. R. E. Bechhofer and B. W. Turnbull, *Two (k + 1)-decision selection procedures for comparing k normal means with a specified standard*, J. Am. Stat. Assoc. 73 (1978), no. 362, 385–392.
5. J. Bect, D. Ginsbourger, L. Li, V. Picheny, and E. Vazquez, *Sequential design of computer experiments for the estimation of a probability of failure*, Stat. Comput. 22 (2012), no. 3, 773–793.
6. B. Bichon, M. Eldred, L. Swiler, S. Mahadevan, and J. McFarland, *Efficient global reliability analysis for nonlinear implicit performance functions*, AIAA J. 46 (2008), no. 10, 2459–2468.

7. M. Binois, R. B. Gramacy, and M. Ludkovski, *Practical heteroscedastic gaussian process modeling for large simulation experiments*, J. Comput. Graph. Stat. 27 (2018), no. 4, 808–821.
8. M. Binois, J. Huang, R. B. Gramacy, and M. Ludkovski, *Replication or exploration? Sequential design for stochastic simulation experiments*, Technometrics 61 (2019), no. 1, 7–23.
9. I. Bogunovic, J. Scarlett, A. Krause, and V. Cevher, "*Truncated variance reduction: A unified approach to Bayesian optimization and level-set estimation*," *Advances in neural information processing systems*, Vol 29, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), Curran Associates, Inc, Red Hook, NY USA, 2016, pp. 1507–1515.
10. X. Chen, B. E. Ankenman, and B. L. Nelson, *The effects of common random numbers on stochastic kriging metamodels*, ACM Trans. Model. Comput. Simul. 22 (2012), no. 2, 1–20 Article No.:7.
11. C. Chevalier, D. Ginsbourger, J. Bect, and I. Molchanov, "*Estimating and quantifying uncertainties on level sets using the Vorob'ev expectation and deviation with Gaussian process models*," *In mODa 10–advances in model-oriented design and analysis*, Springer, Heidelberg, 2013, pp. 35–43.
12. A. Das and D. Kempe, "*Algorithms for subset selection in linear regression*," *Proceedings of the fortieth annual ACM symposium on theory of computing*, Association for Computing Machinery, New York, NY, USA, 2008, pp. 45–54.
13. K. de Brabanter, J. de Brabanter, J. A. Suykens, and B. de Moor, *Approximate confidence and prediction intervals for least squares support vector regression*, IEEE Trans. Neural Netw. 22 (2010), no. 1, 110–120.
14. A. B. Dieker and S.-H. Kim, *Efficient fully sequential indifference-zone procedures using properties of multidimensional Brownian motion exiting a sphere*, 2021, 1–37. https://arxiv.org/abs/2104.08784.
15. A. Durand, O.-A. Maillard, and J. Pineau, *Streaming kernel regression with provably adaptive mean, variance, and regularization*, J. Mach. Learn. Res. 19 (2018), no. 1, 650–683.
16. P. I. Frazier, *A fully sequential elimination procedure for indifference-zone ranking and selection with tight bounds on probability of correct selection*, Oper. Res. 62 (2014), no. 4, 926–942.
17. M. C. Fu and K. J. Healy, "*Simulation optimization of (s, S) inventory systems*," *The 1992 winter simulation conference*, J. Swain, D. Goldsman, R. Crain, and J. Wilson (eds.), IEEE, Piscataway, New Jersey, 1992, pp. 506–514.
18. P. Goldberg, C. Williams, and C. Bishop, "*Regression with input-dependent noise: a Gcaussian process treatment*," *Advances in neural information processing systems*, Vol 10, M. Jordan, M. Kearns, and S. Solla (eds.), MIT Press, Cambridge, Massachusetts, USA, 1997, pp. 493–499.
19. A. Gotovos, N. Casati, G. Hitz, and A. Krause, "*Active learning for level set estimation*," *The 23rd international joint conference on artificial intelligence*, F. Rossi (ed.), AAAI Press, Menlo Park, California, 2013, pp. 1344–1350.
20. Y. Inatsu, M. Karasuyama, K. Inoue, and I. Takeuchi, *Active learning for level set estimation under input uncertainty and its extensions*, Neural Comput. 32 (2020), no. 12, 2486–2531.
21. K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard, "*Most likely heteroscedastic Gaussian process regression*," *Proceedings of the 24th international conference on machine learning*, Z. Ghahramani (ed.), Association for Computing Machinery, New York, NY, USA, 2007, pp. 393–400.
22. S.-H. Kim, *Comparison with a standard via fully sequential procedures*, ACM Trans. Model. Comput. Simul. 15 (2005), no. 2, 155–174.
23. J. Kirschner and A. Krause, "*Information directed sampling and bandits with heteroscedastic noise*," *Conference on learning theory*, S. Bubeck and P. Rigollet (eds.), Proceedings of Machine Learning Research, USA, 2018, pp. 358–384.
24. J. P. Kleijnen and W. C. van Beers, *Statistical tests for cross-validation of kriging models*, INFORMS J. Comput. 34 (2022), no. 1, 607–621.
25. A. Krause and D. Golovin, "*Submodular function maximization*," *Tractability: Practical approaches to hard problems*, L. Bordeaux, Y. Hamadi, and P. Kohli (eds.), Cambridge University Press, Cambridge, UK, 2014, pp. 71–104.
26. X. Lyu, M. Binois, and M. Ludkovski, *Evaluating Gaussian process metamodels and sequential designs for noisy level set estimation*, Stat. Comput. 31 (2021), no. 4, 1–21 Article No.:43.
27. X. Lyu and M. Ludkovski, *Adaptive batching for Gaussian process surrogates with application in noisy level set estimation*, Stat. Anal. Data Min. ASA Data Sci. J. 15 (2022), no. 2, 225–246.
28. B. L. Nelson and D. Goldsman, *Comparisons with a standard in simulation experiments*, Manag. Sci. 47 (2001), no. 3, 449–463.
29. E. Paulson, *A sequential procedure for comparing several experimental categories with a standard or control*, Ann. Math. Stat. 33 (1962), no. 2, 438–443.
30. V. Picheny, D. Ginsbourger, Y. Richet, and G. Caplin, *Quantile-based optimization of noisy computer experiments with tunable precision*, Technometrics 55 (2013), no. 1, 2–13.
31. V. Picheny, D. Ginsbourger, O. Roustant, R. T. Haftka, and N.-H. Kim, *Adaptive designs of experiments for accurate approximation of a target region*, J. Mech. Des. 132 (2010), no. 7, 071008-1–071008-9.
32. P. Ranjan, D. Bingham, and G. Michailidis, *Sequential experiment design for contour estimation from complex computer codes*, Technometrics 50 (2008), no. 4, 527–541.
33. C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*, MIT Press, Massachusetts, 2006.
34. T. J. Santner, B. J. Williams, and W. I. Notz, *The design and analysis of computer experiments*, 2nd ed., Springer, New York, 2018.
35. Z. Shi, Y. Peng, L. Shi, C.-H. Chen, and M. C. Fu, *Dynamic sampling allocation under finite simulation budget for feasibility determination*, INFORMS J. Comput. 34 (2022), no. 1, 557–568.
36. 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 58 (2012), 3250–3265.
37. R. Szechtman and E. Yucesan, "*A new perspective on feasibility determination*," *Proceedings of the 2008 winter simulation conference*, S. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, and J. W. Fowler (eds.), IEEE, Piscataway, New Jersey, 2008, pp. 273–280.
38. R. Szechtman and E. Yücesan, "*A Bayesian approach to feasibility determination*," *Proceedings of the 2016 winter simulation conference*, T. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. Chick (eds.), IEEE, Piscataway, New Jersey, 2016, pp. 782–790.
39. W. Wang, N. Chen, X. Chen, and L. Yang, *A variational inference-based heteroscedastic Gaussian process approach for*

simulation metamodeling, ACM Trans. Model. Comput. Simul. 29 (2019), no. 1, 1–22 Article No.:6.
40. W. Wang and X. Chen, *An adaptive two-stage dual metamodeling approach for stochastic simulation experiments*, IISE Trans. 50 (2018), no. 9, 820–836.
41. W. Whitt, *Planning queueing simulations*, Manag. Sci. 35 (1989), no. 11, 1341–1366.
42. C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, MIT Press Cambridge, MA, 2006.
43. G. Xie and X. Chen, "*Uniform error bounds for stochastic kriging*," *The 2020 winter simulation conference*, K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing (eds.), IEEE, Piscataway, New Jersey, 2020, pp. 361–372.
44. J. Xie and P. I. Frazier, *Sequential Bayes-optimal policies for multiple comparisons with a known standard*, Oper. Res. 61 (2013), no. 5, 174–1189.
45. A. Zanette, J. Zhang, and M. J. Kochenderfer, "*Robust super-level set estimation using Gaussian processes*," *The joint European conference on machine learning and knowledge discovery in databases*, M. Berlingerio, F. Bonchi, T. Gärtner, N. Hurley, and G. Ifrim (eds.), Springer, Cham, Switzerland, 2018, pp. 276–291.
46. Y. Zhang and X. Chen, "*Information consistency of stochastic kriging and its implications*," *2021 winter simulation conference*, S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo, and M. Loper (eds.), IEEE, Piscataway, New Jersey, 2021, pp. 1–12.

How to cite this article: Y. Zhang and X. Chen, *Sequential metamodel-based approaches to level-set estimation under heteroscedasticity*, Stat. Anal. Data Min.: ASA Data Sci. J. **17** (2024), e11697. https://doi.org/10.1002/sam.11697

APPENDIX A. PROOF OF PROPOSITION 1

The proof for the uniform error bound as stated in Proposition 1 is based on the connection between the least square estimator in reproducing kernel Hilbert space (RKHS) and a GP model. The following outlines the main idea.

Proof. Let $\mathcal{H}$ be the RKHS corresponding to the kernel function $K : \mathfrak{R}^d \times \mathfrak{R}^d \to \mathfrak{R}$ with canonical embeddings $K_{\mathbf{x}} = K(\mathbf{x}, \cdot)$. Let $\mathbf{v} = K_{\mathbf{x}} \in \mathcal{H}$ be the embedding of $\mathbf{x} \in \mathfrak{R}^d$, such that $\langle \mathbf{v}, f_0 \rangle_{\mathcal{H}} = \langle K_{\mathbf{x}}, f_0 \rangle_{\mathcal{H}} = f_0(\mathbf{x})$. Define $(K(\mathbf{X}_t, \mathbf{X}_t))_{i,j} := (\mathcal{M}\mathcal{M}^*)_{i,j} = K(\mathbf{x}_i, \mathbf{x}_j)$ and $(K_{\mathbf{x}})_i := (\mathcal{M}K_{\mathbf{x}})_i = \langle \mathbf{x}_i, \mathbf{x} \rangle = K(\mathbf{x}_i, \mathbf{x})$. Using the reproducing property and the linear operator property $(\mathcal{A}^*\mathcal{A} + \lambda I_{\mathcal{H}})^{-1}\mathcal{A}^* = \mathcal{A}^*(\mathcal{A}\mathcal{A}^* + \lambda \mathbf{I}_{k(t)})^{-1}$ with $\mathcal{A} = \Sigma_{\epsilon,k}^{-1}\mathcal{M}$, we calculate $\mu_t(\mathbf{x})$ for any $\mathbf{x} \in \mathcal{X}$,

$$\begin{aligned}\mu_t(\mathbf{x}) &= \langle \mu_t, K_{\mathbf{x}} \rangle_{\mathcal{H}} \\ &= \left\langle \left(\mathcal{M}^* \Sigma_{\epsilon,k}^{-1} \mathcal{M} + \lambda I_{\mathcal{H}} \right)^{-1} \mathcal{M}^* \Sigma_{\epsilon,k}^{-1} \bar{\mathbf{y}}_t, K_{\mathbf{x}} \right\rangle_{\mathcal{H}} \\ &= \left\langle \mathcal{M}^* \Sigma_{\epsilon,k}^{-1/2} \left(\Sigma_{\epsilon,k}^{-1/2} \mathcal{M}\mathcal{M}^* \Sigma_{\epsilon,k}^{-1/2} + \mathbf{I}_d \right)^{-1} \Sigma_{\epsilon,k}^{-1/2} \bar{\mathbf{y}}_t, K_{\mathbf{x}} \right\rangle_{\mathcal{H}} \\ &= \left\langle \mathcal{M}^* \left(\mathcal{M}\mathcal{M}^* + \lambda \Sigma_{\epsilon,k} \right)^{-1} \bar{\mathbf{y}}_t, K_{\mathbf{x}} \right\rangle_{\mathcal{H}} \\ &= \left\langle \left(\mathcal{M}\mathcal{M}^* + \lambda \Sigma_{\epsilon,k} \right)^{-1} \bar{\mathbf{y}}_t, \mathcal{M}K_{\mathbf{x}} \right\rangle_{\mathfrak{R}^{k(t)}} \\ &= K(\mathbf{x}, \mathbf{X}_t) \left(K(\mathbf{X}_t, \mathbf{X}_t) + \lambda \Sigma_{\epsilon,k} \right)^{-1} \bar{\mathbf{y}}_t. \end{aligned} \tag{A1}$$

Using $\lambda(\mathcal{A}^*\mathcal{A} + \lambda \mathcal{I}_{\mathcal{H}})^{-1} = \mathbf{I}_{k(t)} - \mathcal{A}^*(\mathcal{A}\mathcal{A}^* + \lambda \mathbf{I}_{k(t)})^{-1}\mathcal{A}$ with $\mathcal{A} = \Sigma_{\epsilon,k}^{-1/2}\mathcal{M}$, we have $\lambda \langle K_{\mathbf{x}}, K_{\mathbf{x}'} \rangle_{\mathcal{V}_t^{-1}} = K(\mathbf{x}, \mathbf{x}') - K(\mathbf{x}, \mathbf{X}_t)\left(K(\mathbf{X}_t, \mathbf{X}_t) + \lambda \Sigma_{\epsilon,k} \right)^{-1} K(\mathbf{x}', \mathbf{X}_t)^{\tau}$, where $\mathcal{V}_t = K(\mathbf{X}_t, \mathbf{X}_t) + \lambda \Sigma_{\epsilon,k}$. Hence, by setting $\mathbf{v} = K_{\mathbf{x}} = K_{\mathbf{x}'}$, we can calculate $\|\mathbf{v}\|_{\mathcal{V}_t^{-1}}^2$ as follows:

$$\begin{aligned}\|\mathbf{v}\|_{\mathcal{V}_t^{-1}}^2 = \frac{1}{\lambda} \Big(& K(\mathbf{x}, \mathbf{x}) - K(\mathbf{x}, \mathbf{X}_t) \left(K(\mathbf{X}_t, \mathbf{X}_t) + \lambda \Sigma_{\epsilon,k} \right)^{-1} \\ & \times K(\mathbf{x}, \mathbf{X}_t)^{\tau} \Big) = \sigma_t^2(\mathbf{x}). \end{aligned} \tag{A2}$$

We next draw the connection to the Bayesian setting. Let $f_0 \sim GP(0, \lambda^{-1}K)$ be a sample from a GP, where $\lambda^{-1} = \tau^2 > 0$ is the process variance parameter, and assume that the simulation noise ϵ_j's are independent and following the distribution $\mathcal{N}(0, \mathrm{V}(\mathbf{x}_j))$. Then the posterior distribution of f_0 is also a GP with mean $\mu_t(\mathbf{x})$, and $\sigma_t^2(\mathbf{x})$ is the posterior variance at $\mathbf{x}$.

We have set up the connection between least squares estimation in RKHS under heteroscedasticity and stochastic kriging (SK). Then, the uniform error bound for SK follows from specializing Lemma 6 in [23] to the RKHS setting:

$$\begin{aligned}& |\langle \mathbf{v}, \mu_t \rangle - \langle \mathbf{v}, f_0 \rangle| \\ & \leq \left(\sqrt{2 \log \left(\frac{1}{\delta} \frac{\det\left(\Sigma_{\epsilon,k} + \mathcal{M}\mathcal{V}_0\mathcal{M}^* \right)^{1/2}}{\det\left(\Sigma_{\epsilon,k} \right)^{1/2}} \right)} + \| f_0 \|_{\mathcal{V}_0} \right) \\ & \times \| \mathbf{v} \|_{\mathcal{V}_t^{-1}}, \end{aligned} \tag{A3}$$

where $\mathcal{V}_0 = \lambda \mathbf{I}_{k(t)} = \tau^{-2}\mathbf{I}_{k(t)}$.

APPENDIX B. PROOF OF PROPOSITION 2

To reduce the computational complexity, we need to update the predictive variance by computing the predictive variance difference, instead of directly computing the new predictive variance. In this section, we detail the computation of $\Delta_{t-1|\mathbf{x}^+}(\mathbf{x})$ as given in Proposition 2 for PVR.

Proof. If $\mathbf{x}^+$ has not been visited, then we have

$$\begin{aligned}\Delta_{t-1|\mathbf{x}^+}(\mathbf{x}) &= \sigma_{t-1}^2(\mathbf{x}) - \sigma_{t-1|\mathbf{x}^+}^2(\mathbf{x}) \\ &= \tau^2 \left(K(\mathbf{x}, \mathbf{x}) - K\left(\mathbf{x}, \mathbf{X}_{t-1} \right) \right.\end{aligned}$$

$$
\begin{aligned}
&\times \left(K(\mathbf{X}_{t-1}\mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)^{-1} K\left(\mathbf{x}, \mathbf{X}_{t-1}\right)^{\tau}\Big) \\
&- \tau^2\Big(K(\mathbf{x},\mathbf{x}) - K\left(\mathbf{x}, \mathbf{X}_t\right) \\
&\times \left(K(\mathbf{X}_t\mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\right)^{-1} K\left(\mathbf{x}, \mathbf{X}_t\right)^{\tau}\Big) \\
&= \tau^2\Big(K(\mathbf{x}\mathbf{X}_t)\left(K(\mathbf{X}_t\mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\right)^{-1} K\left(\mathbf{x}, \mathbf{X}_t\right)^{\tau} \\
&- K(\mathbf{x}\mathbf{X}_{t-1})\left(K(\mathbf{X}_{t-1}\mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}{}^{-1}\right) \\
&\times K\left(\mathbf{x}, \mathbf{X}_{t-1}\right)^{\tau}\Big), \qquad \text{(B1)}
\end{aligned}
$$

where $K(\mathbf{x}, \mathbf{X}_t) = \left(K(\mathbf{x}\mathbf{X}_{t-1}), K\left(\mathbf{x}, \mathbf{x}^+\right)\right), K(\mathbf{X}_t, \mathbf{X}_t) = \begin{pmatrix} K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) & K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right)^{\tau} \\ K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right) & K\left(\mathbf{x}^+, \mathbf{x}^+\right) \end{pmatrix}$, $\Sigma_{\epsilon,k(t)} = \begin{pmatrix} \Sigma_{\epsilon,k(t-1)} & \mathbf{0} \\ \mathbf{0} & \mathrm{V}\left(\mathbf{x}^+\right) n_{t,\mathbf{x}^+}^{-1} \end{pmatrix}$. Denote $\mathbf{b} = K(\mathbf{x}, \mathbf{X}_{t-1})$, $d = K\left(\mathbf{x}^+, \mathbf{x}^+\right) + \tau^{-2}\mathrm{V}\left(\mathbf{x}^+\right) n_{t,\mathbf{x}^+}^{-1}$, $\mathbf{e} = \left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)^{-1} \mathbf{b}^{\tau}$, and $g = (d - \mathbf{b}\mathbf{e})^{-1}$. Then,

$$
\begin{aligned}
&\left(K(\mathbf{X}_t, \mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\right)^{-1} \\
&= \begin{pmatrix} K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)} K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right)^{\tau} & \\ & K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right) K\left(\mathbf{x}^+, \mathbf{x}^+\right) + \dfrac{\mathrm{V}\left(\mathbf{x}^+\right)}{n_{t,\mathbf{x}^+}} \end{pmatrix}^{-1} \\
&= \begin{pmatrix} \left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)^{-1} + g\mathbf{e}\mathbf{e}^{\tau} & -g\mathbf{e} \\ -g\mathbf{e}^{\tau} & g \end{pmatrix}. \qquad \text{(B2)}
\end{aligned}
$$

Plugging (B1) into (B2) yields

$$
\begin{aligned}
\Delta_{t-1|\mathbf{x}^+}(\mathbf{x}) &= \tau^2\big(gK\left(\mathbf{x}^+\mathbf{X}_{t-1}\right)\mathbf{e}\mathbf{e}^{\tau} K\left(\mathbf{x}^+\mathbf{X}_{t-1}\right)^{\tau} \\
&\quad + K\left(\mathbf{x}, \mathbf{x}^+\right)(-g\mathbf{e}^{\tau}) K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right)^{\tau} \\
&\quad + K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right)(-g\mathbf{e}) K\left(\mathbf{x}^+, \mathbf{x}^+\right) \\
&\quad + K\left(\mathbf{x}, \mathbf{x}^+\right) g K\left(\mathbf{x}, \mathbf{x}^+\right)\big) \\
&= \tau^2 g\big(K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right)\mathbf{e}\mathbf{e}^{\tau} K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right)^{\tau} \\
&\quad - 2K\left(\mathbf{x}^+, \mathbf{X}_{t-1}\right) e K\left(\mathbf{x}^+, \mathbf{x}^+\right) + K\left(\mathbf{x}, \mathbf{x}^+\right) K\left(\mathbf{x}, \mathbf{x}^+\right)\big) \\
&= \frac{\mathrm{Cov}_{t-1}^2\left(\mathbf{x}, \mathbf{x}^+\right)}{\sigma_{t-1}^2\left(\mathbf{x}^+\right) + \frac{\mathrm{V}\left(\mathbf{x}^+\right)}{n_{t,\mathbf{x}^+}}}.
\end{aligned}
$$

If $\mathbf{x}^+$ has been visited, without loss of generality, assume it to be the ith existing design point, $\mathbf{x}_i$, then $\Delta_{t-1|\mathbf{x}_i}(\mathbf{x})$ can be written as below following the proof in [8],

$$
\Delta_{t-1|\mathbf{x}_i}(\mathbf{x}) = \sigma_{t-1}^2(\mathbf{x}) - \sigma_{t-1|\mathbf{x}_i}^2(\mathbf{x}) = \tau^2\Bigg(K(\mathbf{x}\mathbf{X}_{t-1}) \underbrace{\left(\left(K(\mathbf{X}_t\mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\right)^{-1} - \left(K(\mathbf{X}_{t-1}\mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)^{-1}\right)}_{:=B_k} K\left(\mathbf{x}, \mathbf{X}_{t-1}\right)^{\tau}\Bigg),
$$

where $\mathbf{X}_t = \mathbf{X}_{t-1}$. By the Sherman-Morrison formula, we have

$$
B_k = \frac{\begin{array}{c}\left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)_{\cdot,i}^{-1} \\ \cdot\left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)_{i,\cdot}^{-1}\end{array}}{\begin{array}{c}\left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)_{i,i}^{-1} \\ -\left(\tau^{-2}\left(\frac{\mathrm{V}(\mathbf{x}_i)}{n_{t-1,i}} - \frac{\mathrm{V}(\mathbf{x}_i)}{n_{t,i}}\right)\right)\end{array}},
$$

where $\mathbf{A}_{\cdot,i}$ denotes the ith column of matrix $\mathbf{A}$, and $\mathbf{A}_{i,\cdot}$ denotes the ith row of matrix $\mathbf{A}$. It follows that

$$
\begin{aligned}
&\Delta_{t-1|\mathbf{x}_i}(\mathbf{x}) \\
&= \frac{\tau^2\left(\left[K(\mathbf{x}, \mathbf{X}_{t-1})\left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)^{-1}\right]_{(i)}\right)^2}{\left(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\right)_{i,i}^{-1} - \left(\tau^{-2}\left(\frac{\mathrm{V}(\mathbf{x}_i)}{n_{t-1,i}} - \frac{\mathrm{V}(\mathbf{x}_i)}{n_{t,i}}\right)\right)},
\end{aligned}
$$

where $\mathbf{b}_{(i)}$ denotes the ith entry of vector $\mathbf{b}$.

APPENDIX C. PROOF OF THEOREM 3.2

Lemmas 1 and 2 upper bound $\beta_{k(t)}$ by a function of $k(t)$, the number of design points, which can be easily generalized to a function of t, the number of iterations. Such an upper bound supports the proof of Theorem 1, which starts by transforming the uniform bound given in Proposition 1 to a function of the iteration index t. This helps us find an analytical form of $\beta_{(i)}$. To prove Theorem 1, we first begin with the proof of Lemma 1 using Lemma 2.

C.1. Proof of Lemma 1

The proof for Lemma 1 relies on Lemma 2 which is stated below.

Lemma 2. *In the uniform bound given in (4), $\gamma_{k(t)} := \log\left(|\mathbf{I}_{k(t)} + \tau^2\Sigma_{\epsilon,k(t)}^{-1} K(\mathbf{X}_t\mathbf{X}_t)|\right)$ is equivalent to $\sum_{i=1}^{k} \log\left(1 + \tau^2 n_{t,i}\mathrm{V}(\mathbf{x}_i)^{-1}\sigma_t^2(\mathbf{x}_i)\right)$.*

Proof. The proof follows Appendix A in [15]. We abbreviate $k(t)$, the number of design points on the tth iteration, to k, to ease notation. Define $\mathbf{B}_k = \mathbf{I} + \tau^2\, \boldsymbol{\Phi}_k^{\tau}\Sigma_{\epsilon,k(t)}^{-1}\, \boldsymbol{\Phi}_k$, where $\boldsymbol{\Phi}_k = (\boldsymbol{\Phi}(\mathbf{x}_1), \dots, \boldsymbol{\Phi}(\mathbf{x}_k))^{\tau}$ with $\boldsymbol{\Phi}(\mathbf{x}_i) = (\phi_1(\mathbf{x}_i), \phi_2(\mathbf{x}_i), \dots)$ for $i = 1, 2, \dots, k$ is a $k \times \infty$ matrix. Meanwhile, $K(\mathbf{X}_t, \mathbf{X}_t) = \Phi_k\ \Phi_k^{\tau}$ and $K\left(\mathbf{x}, \mathbf{x}'\right) = \boldsymbol{\Phi}(\mathbf{x})^{\tau}\ \boldsymbol{\Phi}\left(\mathbf{x}'\right)$. Then, we have

$$
\begin{aligned}
\mathbf{B}_k\boldsymbol{\Phi}_k^{\tau} &= \left(\mathbf{I}_k + \tau^2\Sigma_{\epsilon,k(t)}^{-1} K(\mathbf{X}_t, \mathbf{X}_t)\right)\boldsymbol{\Phi}_k^{\tau} \\
&= \boldsymbol{\Phi}_k^{\tau} + \tau^2\boldsymbol{\Phi}_k^{\tau}\Sigma_{\epsilon,k(t)}^{-1}\boldsymbol{\Phi}_k\boldsymbol{\Phi}_k^{\tau} \\
&= \boldsymbol{\Phi}_k^{\tau}\left(\mathbf{I}_k + \tau^2\Sigma_{\epsilon,k(t)}^{-1}\boldsymbol{\Phi}_k\boldsymbol{\Phi}_k^{\tau}\right) \\
&= \boldsymbol{\Phi}_k^{\tau}\left(\mathbf{I}_k + \tau^2\Sigma_{\epsilon,k(t)}^{-1} K(\mathbf{X}_t, \mathbf{X}_t)\right).
\end{aligned}
$$

Since $\det(\mathbf{AC}) = \det(\mathbf{A})\det(\mathbf{C})$, using $\mathbf{A} = \mathbf{B}_{k-1}$ and $\mathbf{C} = 1 + \tau^2 \frac{n_{t,k}}{\mathrm{V}(\mathbf{x}_k)} \|\phi(\mathbf{x}_k)\|_{\mathbf{B}_{k-1}^{-1}}$, we have

$$\begin{aligned}
&\log\left(\det\left(\mathbf{I}_k + \tau^2 \Sigma_{\epsilon,k(t)}^{-1} K(\mathbf{X}_t, \mathbf{X}_t)\right)\right) = \log\left(\det\left(\mathbf{B}_k\right)\right) \\
&= \log\left(\det\left(\mathbf{B}_{k-1} + \tau^2 \frac{n_{t,k}}{\mathrm{V}(\mathbf{x}_k)} \phi(\mathbf{x}_k)\phi(\mathbf{x}_k)^{\mathrm{T}}\right)\right) \\
&= \log\left(\det\left(\mathbf{B}_{k-1}\right)\det\left(1 + \tau^2 \frac{n_{t,k}}{\mathrm{V}(\mathbf{x}_k)} \|\phi(\mathbf{x}_k)\|_{\mathrm{B}_{k-1}^{-1}}\right)\right) \\
&= \log\left(\det\left(\mathbf{B}_0\right)\prod_{i=1}^{k}\left(1 + \tau^2 \frac{n_{t,i}}{\mathrm{V}(\mathbf{x}_i)} \|\phi(x_k)\|_{\mathrm{B}_{i-1}^{-1}}\right)\right) \\
&= \log\left(\prod_{i=1}^{k}\left(1 + \frac{n_{t,i}}{\mathrm{V}(\mathbf{x}_i)}\sigma_t^2(\mathbf{x}_i)\right.\right. \\
&= \sum_{i=1}^{k} \log\left(1 + \frac{n_{t,i}}{\mathrm{V}(\mathbf{x}_i)}\sigma_t^2(\mathbf{x}_i)\right).
\end{aligned}$$

Based on the proof of Lemma 2, Lemma 1 can be proved as follows.

Proof. By Lemma 2, we have

$$\begin{aligned}
\beta_{k(t)} &= \sqrt{\sum_{i=1}^{k(t)} \log\left(1 + \tfrac{n_{t,i}}{\mathrm{V}(\mathbf{x}_i)}\sigma_t^2(\mathbf{x}_i)\right) - 2\log\delta + + \tau^{-1} \| f \|_K} \\
&\le \sqrt{\sum_{i=1}^{k(t)} \log\left(1 + \tfrac{\tau^2}{\mathrm{V}_{\min}(t)}\right) - 2\log\delta + + \tau^{-1} \| f \|_K} \\
&= \sqrt{k(t)\log\left(1 + \tfrac{\tau^2}{\mathrm{V}_{\min}(t)}\right) - 2\log\delta + + \tau^{-1} \| f \|_K},
\end{aligned}$$

where recall that $\mathrm{V}_{\min}(t) \coloneqq \min_{i=1,2,\ldots,k(t)} \mathrm{V}(\mathbf{x}_i)/n_{t,i}$. ∎

C.2. Proof of Theorem 1

Based on Lemmas 1 and 2, we can prove Theorem 1 in the same vein as the proof of Theorem 1 in [9]. The flow of the proof is outlined next, with specific details omitted for the sake of brevity.

The proof consists of three parts, and the extension to the heteroscedastic noise setting concentrates on the second part. The first part is to prove the least cost each epoch requested to achieve $(1 + \bar{\delta})\eta_{(i)}$ confidence for all points in the uncertain set M_t is $c\left(S^*_{t_{(i)}}\right)\log\left(|M_{t_{(i)}-1}|\beta^2_{(i)}\left(\bar{\delta}^2\eta^2_{(i)}\right)^{-1}\right)$, where $S^*_{t_{(i)}}$ is the optimal solution to the problem stated below as given in (C1), and recall that $|M|$ denotes the cardinality of set M, and $t_{(i)}$ denotes the index of the first iteration of the ith epoch. We first define the function $g_t(S)$ as the decrease in the truncated variance across the unclassified points in set M_t after adding points in set S as follows:

$$\begin{aligned}
g_t(S) = &\sum_{\mathbf{x}\in M_{t-1}} \max\left\{\sigma^2_{t-1}(\mathbf{x}), \frac{\eta^2_{(i)}}{\beta^2_{(i)}}\right\} \\
&- \sum_{\mathbf{x}\in M_{t-1}} \max\left\{\sigma^2_{t-1|S}(\mathbf{x}), \frac{\eta^2_{(i)}}{\beta^2_{(i)}}\right\},
\end{aligned}$$

and its maximum $g_{t,\max}$ which can be understood as the excess variance:

$$g_{t,\max} \coloneqq \sum_{\mathbf{x}\in M_{t-1}} \max\left\{0, \sigma^2_{t-1}(\mathbf{x}) - \frac{\eta^2_{(i)}}{\beta^2_{(i)}}\right\}.$$

Thus, in light of Assumption 2, Algorithm 1 can be treated as a greedy rule for solving the following submodular optimization problem:

$$\begin{aligned}
&\min_{S} \; c(S) \\
&\text{subject to} \quad g_t(S) = g_{t,\max}.
\end{aligned} \tag{C1}$$

Then, by Lemma 2 in [25], we have

$$g_{t+1,\max} \le \left(1 - \frac{c(\mathbf{x}_t)}{c\left(S^*_t\right)}\right) g_{t,\max},$$

and hence

$$\frac{g_{t+l,\max}}{g_{t,\max}} \le \exp\left(-\frac{\sum_{t'=t+1}^{t+l} c(\mathbf{x}_{t'})}{c\left(S^*_t\right)}\right) \quad \text{for} \quad l \ge 1,$$

if the tth and the $(t+l)$th iterations are in the same epoch. Thus, if we want to reduce all but a proportion γ of the excessive variance, it takes cost at least $c\left(S^*_{t_{(i)}}\right)\log\gamma^{-1}$. According to Algorithm 1, it requires γ to be $\bar{\delta}^2\eta^2_{(i)}\left(|M_{t_{(i)}-1}|\beta^2_{(i)}\right)^{-1}$.

The second part is to prove the maximum cost $C_{(i)}$ required by each epoch via mathematical induction on the epoch number, which is detailed discussed in [23]. To extend their proof to the heteroscedastic noise case, we require that $\beta_{(i)}$ as given in (10) is an upper bound for all $\beta_{k(t)}$ in the ith epoch. This result can be validated by Lemma 1 and the following two facts: (1) t is always larger than or equal to $k(t)$; and (2) $\sum_{i'\le i} C_{(i')}c^{-1}_{\min}$ is always larger than t.

Finally, by Definition 2 in Section 3, to achieve ϵ-accuracy, we require $2(1+\bar{\delta})\eta_{(i-1)} > \epsilon/2$, hence $4(1+\bar{\delta})\eta_{(i-1)} > \epsilon$. Therefore, if we have $C_\epsilon = \sum_{i:4(1+\bar{\delta})\eta_{(i-1)}>\epsilon} C_{(i)}$, Algorithm 1 can be validated to achieve the ϵ-accuracy guarantee.

APPENDIX D. PROOF OF PROPOSITION 3

Proof. If the selected point on the tth iteration is an existing point, assume it to be the ith point, $\mathbf{x}_i$. Then the predictive mean can be written as:

$$\begin{aligned}\mu_t(\mathbf{x}) &= \mu_{t-1}(\mathbf{x}) + K(\mathbf{x}, \mathbf{X}_{t-1})\cdot \\ &\quad \Big(\big(K(\mathbf{X}_t, \mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\big)^{-1}\big(\overline{\mathbf{y}}_{t-1} + \Delta\mathbf{y}_{t-1}\big) \\ &\quad -\big(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\big)^{-1}\overline{\mathbf{y}}_{t-1}\Big) \\ &= \mu_{t-1}(\mathbf{x}) + \underbrace{K(\mathbf{x}, \mathbf{X}_{t-1})\Big(\big(K(\mathbf{X}_t, \mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\big)^{-1} \\ &\quad -\big(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\big)^{-1}\Big)}_{:=D_i}\overline{\mathbf{y}}_{t-1} \\ &\quad + K(\mathbf{x}, \mathbf{X}_{t-1})\big(K(\mathbf{X}_t, \mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\big)^{-1}\Delta\mathbf{y}_{t-1} \\ &= \mu_{t-1}(\mathbf{x}) + K(\mathbf{x}, \mathbf{X}_{t-1})D_i\overline{\mathbf{y}}_{t-1} + K(\mathbf{x}, \mathbf{X}_{t-1}) \\ &\quad \big(K(\mathbf{X}_t, \mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\big)^{-1}\Delta\mathbf{y}_{t-1},\end{aligned} \tag{D1}$$

where $\Delta\mathbf{y}_{t-1} = \overline{\mathbf{y}}_t - \overline{\mathbf{y}}_{t-1} = \Big(\underbrace{0, \dots, 0}_{i-1}, \frac{n_{t-1,i}\overline{y}_{t-1,i} + \Delta n_{t-1,i}\Delta\overline{y}_{t-1,i}}{n_{t,i}} - \overline{y}_{t,i}, \underbrace{0, \dots, 0}_{k-i}\Big)$, $\Delta\overline{y}_{t-1,i}$ denotes the sample average at the ith design point collected on the tth iteration. By Assumption 3, we have $\Delta\overline{y}_{t-1,i} \sim \mathcal{N}\left(\mu_{t-1}(\mathbf{x}_i), \sigma^2_{t-1}(\mathbf{x}_i) + \mathrm{V}(\mathbf{x}_i)/\Delta n_{t,i}\right)$. Thus, we have

$$\begin{aligned}&K(\mathbf{x}, \mathbf{X}_{t-1})\big(K(\mathbf{X}_t, \mathbf{X}_t) + \tau^{-2}\Sigma_{\epsilon,k(t)}\big)^{-1}\Delta\mathbf{y}_{t-1} \\ &= \underbrace{\Big[K(\mathbf{x}, \mathbf{X}_{t-1})\big(K(\mathbf{X}_{t-1}, \mathbf{X}_{t-1}) + \tau^{-2}\Sigma_{\epsilon,k(t-1)}\big)^{-1}\Big]_{(i)}}_{:=C_i} \\ &\quad \left(\frac{n_{t-1,i}\overline{\mathbf{y}}_{t-1,i} + \Delta n_{t-1,i}\Delta\overline{y}_{t-1,i}}{n_{t,i}} - \overline{y}_{t-1,i}\right) \\ &= \frac{\Delta n_{t-1,i}C_i}{n_{t,i}}\big(\Delta\overline{y}_{t-1,i} - \overline{y}_{t,i}\big),\end{aligned} \tag{D2}$$

where recall that $[\mathbf{b}]_{(i)}$ is the ith element of vector $\mathbf{b}$.

Then, for the super-level set H_t, we have

$$\mu_t(\mathbf{x}) - \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}_i}(\mathbf{x}) > h - \epsilon_0. \tag{D3}$$

Plugging (D2) into (D1) and combining with (D3) yields

$$\begin{aligned}&\mu_{t-1}(\mathbf{x}) + K(\mathbf{x}, \mathbf{X}_{t-1})D_i\overline{\mathbf{y}}_{t-1} + \frac{\Delta n_{t-1,i}C_i}{n_{t,i}}\big(\Delta\overline{y}_{t-1,i} - \overline{\mathbf{y}}_{t-1,i}\big) \\ &- \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}_i}(\mathbf{x}) > h - \epsilon_0\end{aligned} \tag{D4}$$

$$\begin{aligned}\Rightarrow\ &\frac{\Delta n_{t-1,i}C_i}{n_{t,i}}\big(\Delta\overline{y}_{t-1,i} - \overline{y}_{t-1,i}\big) > h - \epsilon_0 \\ &+ \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}_i}(\mathbf{x}) - \mu_{t-1}(\mathbf{x}) - K(\mathbf{x}, \mathbf{X}_{t-1})D_i\overline{\mathbf{y}}_{t-1}.\end{aligned} \tag{D4}$$

If $\Delta n_{t-1,i}C_i\big(n_{t,i}\big)^{-1} > 0$, we can write (D4) as

$$\Delta\overline{y}_{t-1,i} > \frac{h - \epsilon_0 + \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}_i}(\mathbf{x}) - \mu_{t-1}(\mathbf{x}) - K(\mathbf{x}, \mathbf{X}_{t-1})D_i\overline{\mathbf{y}}_{t-1}}{\frac{\Delta n_{t-1,i}C_i}{n_{t,i}}} + \overline{y}_{t-1,i}.$$

Then, the expectation that any point in the prediction set (i.e., $\mathbf{x} \in \mathcal{P}$) is included in the super-level set is given by

$$\begin{aligned}&\int_{-\infty}^{\infty} 1\big\{\mu_t(\mathbf{x}) - \beta_{k(t-1)}\sigma_t(\mathbf{x}) > h - \epsilon_0\big\}p\big(\Delta\overline{y}_{t-1,i}\big)d\Delta\overline{y}_{t-1,i} \\ &= \int_{-\infty}^{\infty} 1\Bigg\{\Delta\overline{y}_{t-1,i} > \underbrace{\frac{h - \epsilon_0 + \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}_i}(\mathbf{x}) - \mu_{t-1}(\mathbf{x}) - K(\mathbf{x}, \mathbf{X}_{t-1})D_i\overline{\mathbf{y}}_{t-1}}{\left|\frac{\Delta n_{t-1,i}C_i}{n_{t,i}}\right|} + \overline{y}_{t-1,i}}_{:=R_i^+(\mathbf{x})}\Bigg\} \\ &\quad \times p\big(\Delta\overline{y}_{t-1,i}\big)d\Delta\overline{y}_{t-1,i} \\ &= \int_{-\infty}^{+\infty} 1\left\{z > \frac{R_i^+(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma^2_{t-1}(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right\}\phi(z)dz \\ &= \int_{-\infty}^{+\infty} 1\left\{-z < -\frac{R_i^+(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma^2_{t-1}(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right\}\phi(z)dz \\ &= \Phi\left(-\frac{R_i^+(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma^2_{t-1}(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right),\end{aligned} \tag{D5}$$

where $p\big(\Delta\overline{y}_{t-1,i}\big)$ is the probability density function of $\Delta\overline{y}_{t-1,i}$. If $\Delta n_{t-1,i}C_i\big(n_{t,i}\big)^{-1} < 0$, the expectation that any point in the prediction set (i.e., $\mathbf{x} \in \mathcal{P}$) is included in the super-level set is given by

$$\begin{aligned}&\int_{-\infty}^{\infty} 1\big\{\mu_t(\mathbf{x}) - \beta_{k(t-1)}\sigma_{t-1|x_i}(\mathbf{x}) > h - \epsilon_0\big\}p\big(\Delta\overline{y}_{t-1,i}\big)d\Delta\overline{y}_{t-1,i} \\ &= \int_{-\infty}^{\infty} 1\Bigg\{\Delta\overline{y}_{t-1,i} < \frac{h - \epsilon_0 + \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}_i}(\mathbf{x}) - \mu_{t-1}(\mathbf{x}) - K(\mathbf{x}, \mathbf{X}_{t-1})D_i\overline{\mathbf{y}}_{t-1}}{\left|\frac{\Delta n_{t-1,i}C_i}{n_{t,i}}\right|} + \overline{y}_{t-1,i}\Bigg\} \\ &\quad \times p\big(\Delta\overline{y}_{t-1,i}\big)d\Delta\overline{y}_{t-1,i}\end{aligned}$$

$$= \int_{-\infty}^{+\infty} 1\left\{z < -\frac{R_i^+(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma_{t-1}^2(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right\}\phi(z)dz$$

$$= \Phi\left(-\frac{R_i^+(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma_{t-1}^2(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right). \tag{D6}$$

Equations (D5) and (D6) are in the same form. Hence, the probability that any point in the prediction set (i.e., $\mathbf{x} \in \mathcal{P}$) is classified to the super-level set is given by

$$\Phi\left(-\frac{R_i^+(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma_{t-1}^2(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right). \tag{D7}$$

Similarly, the probability that any point in the prediction set (i.e., $\mathbf{x} \in \mathcal{P}$) belongs to the sub-level set can be shown to be

$$\Phi\left(\frac{R_i^-(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma_{t-1}^2(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right), \tag{D8}$$

where

$$R_i^-(\mathbf{x}) = \frac{\begin{array}{c} h + \epsilon_0 - \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}_i}(\mathbf{x}) \\ -\mu_{t-1}(\mathbf{x}) - K(\mathbf{x}, \mathbf{X}_{t-1})D_i\bar{\mathbf{y}}_{t-1} \end{array}}{|\frac{\Delta n_{t-1,i}C_i}{n_{t,i}}|} + \bar{y}_{t-1,i}.$$

Finally, the expected number of points in the super-level set and the sub-level set follows as the summation of (D7) and (D8) over the prediction set $\mathcal{P}$:

$$\mathbb{E}\left[|HL_t(\mathbf{x}_i\mathbf{y}(\mathbf{x}_i))|\right] = \sum_{\mathbf{x}\in\mathcal{P}}\Phi\left(-\frac{R_i^+(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma_{t-1}^2(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right) + \sum_{\mathbf{x}\in\mathcal{P}}\Phi\left(\frac{R_i^-(\mathbf{x}) - \mu_{t-1}(\mathbf{x}_i)}{\sqrt{\sigma_{t-1}^2(\mathbf{x}_i) + \frac{\mathrm{V}(\mathbf{x}_i)}{\Delta n_{t,i}}}}\right).$$

APPENDIX E. PROOF OF PROPOSITION 4

Proof. We compute the expected integral on the number of classified points over $h \in \mathcal{T}$ as follows. We separate the calculation into two parts: one for the super-level set and the other for the sub-level set. Following the proof process in Appendix D for each part leads to the final expression.

$$\begin{aligned}
&\mathbb{E}\int_{-\infty}^{\infty}|HL_{t-1}^h(\mathbf{x}^+, \mathbf{y}(\mathbf{x}^+))| - |HL_{t-1}^h|dh \\
&= \mathbb{E}\int_{-\infty}^{\infty}\sum_{\mathbf{x}\in\mathcal{P}} 1\left\{\mu_t(\mathbf{x}) - \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}^+}(\mathbf{x}) > h - \epsilon_0\right\} \\
&\quad - 1\left\{\mu_{t-1}(\mathbf{x}) - \beta_{k(t-1)}\sigma_{t-1}(\mathbf{x}) > h - \epsilon_0\right\}dh \\
&\quad + \mathbb{E}\int_{-\infty}^{\infty}\sum_{\mathbf{x}\in\mathcal{P}} 1\left\{\mu_t(\mathbf{x}) + \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}^+} < h + \epsilon_0\right\} \\
&\quad - 1\left\{\mu_{t-1}(\mathbf{x}) + \beta_{k(t-1)}\sigma_{t-1}(\mathbf{x}) < h + \epsilon_0\right\}dh \\
&= \mathbb{E}\sum_{\mathbf{x}\in\mathcal{P}}\int_{-\infty}^{\infty} 1\left\{h < \mu_t(\mathbf{x}) - \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}^+}(\mathbf{x}) + \epsilon_0\right\} \\
&\quad - 1\left\{h < \mu_{t-1}(\mathbf{x}) - \beta_{k(t-1)}\sigma_{t-1}(\mathbf{x}) + \epsilon_0\right\}dh \\
&\quad + \mathbb{E}\sum_{\mathbf{x}\in\mathcal{P}}\int_{-\infty}^{\infty} 1\left\{h > \mu_t(\mathbf{x}) + \beta_{k(t-1)}\sigma_{t-1|\mathbf{x}^+} - \epsilon_0\right\} \\
&\quad - 1\left\{h > \mu_{t-1}(\mathbf{x}) + \beta_{k(t-1)}\sigma_{t-1}(\mathbf{x}) - \epsilon_0\right\}dh \\
&= \mathbb{E}\sum_{\mathbf{x}\in\mathcal{P}}\int_{\mu_{t-1}(\mathbf{x})-\beta_{k(t-1)}\sigma_{t-1}(\mathbf{x})+\epsilon_0}^{\mu_t(\mathbf{x})-\beta_{k(t-1)}\sigma_{t-1|\mathbf{x}^+}(\mathbf{x})+\epsilon_0} dh \\
&\quad + \mathbb{E}\sum_{\mathbf{x}\in\mathcal{P}}\int_{-(\mu_{t-1}(\mathbf{x})+\beta_{k(t-1)}\sigma_{t-1}(\mathbf{x})-\epsilon_0)}^{-(\mu_t(\mathbf{x})+\beta_{k(t-1)}\sigma_{t-1|\mathbf{x}^+}-\epsilon_0)} dh \\
&= 2\beta_{k(t-1)}\sum_{\mathbf{x}\in\mathcal{P}}\left(\sigma_{t-1}(\mathbf{x}) - \sigma_{t-1|\mathbf{x}^+}(\mathbf{x})\right).
\end{aligned}$$

Then, since $\beta_{k(t-1)}$ and $\sigma_{t-1}(\mathbf{x})$ are fixed on the tth iteration, we have

$$\begin{aligned}
&\arg\max_{\mathbf{x}^+\in\mathcal{X}} 2\beta_{k(t-1)}\sum_{\mathbf{x}\in\mathcal{P}}\left(\sigma_{t-1}(\mathbf{x}) - \sigma_{t-1|\mathbf{x}^+}(\mathbf{x})\right) \\
&\quad \Leftrightarrow \arg\min_{\mathbf{x}^+\in\mathcal{X}}\sum_{\mathbf{x}\in\mathcal{P}}\sigma_{t-1|\mathbf{x}^+}(\mathbf{x}).
\end{aligned}$$

□