Proceedings of the 2025 Winter Simulation Conference
E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.

NESTED SUPERLEVEL SET ESTIMATION FOR SIMULATION OPTIMIZATION UNDER PARAMETER UNCERTAINTY

Dongjoon Lee[1], and Eunhye Song[1]

[1]School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA

ABSTRACT

This paper addresses the challenge of reliably selecting high-performing solutions in simulation optimization when model parameters are uncertain. We infer the set of solutions whose probabilities of performing better than a user-defined threshold are above a confidence level given the uncertainty about the parameters. We show that this problem can be formulated as a nested superlevel set estimation problem and propose a sequential sampling framework that models the simulation output mean as a Gaussian process (GP) defined on the solution and parameter spaces. Based on the GP model, we introduce a set estimator and an acquisition function that evaluates the expected number of solutions whose set classifications change should each solution-parameter pair be sampled. We also provide approximation schemes to make the acquisition function computation more efficient. Based on these, we propose a sequential sampling algorithm that effectively reduces the set estimation error and empirically demonstrate its performance.

1 INTRODUCTION

In this paper, we study the problem of inferring the performances of the candidate solutions of a simulation optimization problem, where all solutions are evaluated with the same simulation model that requires a parameter vector whose exact value is uncertain. For instance, the vector may parameterize the input distributions of the simulation model or the internal functions of the simulator. In both cases, the parameters determine the distributional properties of the simulation outputs. Such parametric uncertainty poses a challenge in decision making when the simulator is adopted for optimization as the optimal solution's identity may depend on the parameter setting. We approach this problem from an inference perspective by providing *a set of solutions whose probabilities of performing better than a user-defined threshold are above a confidence level* given the uncertainty about the parameter. In particular, we focus on the case where the feasible solution set is a finite subset of the Euclidean space.

A set inference problem has long been studied in the simulation optimization literature. Dating back to Gupta (1965), subset selection procedures that return a set of solutions guaranteed to contain the best solution with a prespecified probability have been proposed. Multiple comparisons with the best (Hsu 1981) is closely related to subset selection where the goal is to construct a simultaneous confidence intervals (CIs) for the differences between the means of each solution and the best of the rest. In addition to the CIs, the procedure also returns a set of solutions that contains the best with the same coverage guarantee as the CIs. More recently, Eckman et al. (2022) propose a framework that screens out the solutions implausible to be optimal from the finite feasible set exploiting the objective function's functional properties (e.g., convexity, Lipschitz continuity, etc.) with a desired level of statistical confidence.

Extending the classical methods, recent studies reflect the uncertainty in estimated input models from finite data, i.e., input uncertainty, in subset selection. Corlu and Biller (2013) adopt a Bayesian framework that models the effect of the input model's parameter uncertainty on the simulation output. Song and Nelson (2019) extend the classical multiple comparisons with the best framework to incorporate input uncertainty and provide the asymptotic consistency guarantee. Wu et al. (2024) develop sequential elimination procedures

under input uncertainty that adaptively incorporate simulation output and incoming input data to refine the set of plausible solutions, while maintaining statistical validity through adjusted confidence bounds.

Another problem with a close connection to our research is estimating a superlevel set of a black-box function, where the goal is to find a preimage of an output of the function no less than a threshold. When the function is expensive to evaluate or its domain has infinite elements, estimating the superlevel set becomes challenging as one may not be able to evaluate all points in the domain. This challenge has led to the development of active learning strategies that sequentially fit a model to approximate the function, estimate the set, and select the most informative points to reduce the error in the estimated set.

Bryan et al. (2005) propose the *straddle* heuristic, which selects the points for evaluation that lie near the boundary of the superlevel set, where the function value is most uncertain, and thereby focusing the search on areas with the greatest potential for information gain. Gotovos (2013) models the function as a realization of a Gaussian Process (GP) and applies GP-derived confidence bounds to guide the sampling process. Focusing on improving the robustness of the superlevel set estimator, Zanette et al. (2018) propose an acquisition function designed to maximize the expected size of the superlevel set in the next iteration. Iwazaki et al. (2020) tackle the problem when there is control uncertainty such that the output cannot be observed at the exact intended input value. Their goal is to infer a set of control values exceeding a threshold with a pre-specified probability.

In this work, we also adopt a GP model to infer the simulation output mean function at any parameter-solution pair. Unlike most of the reviewed work, the set we estimate has a nested superlevel set structure. For each solution, estimating the probability of exceeding the threshold is equivalent to estimating a superlevel set of parameters that returns the function value—simulation output performance measure—no less than the threshold and computing the probability of the set. Then, the set is formed by classifying the solutions whose estimated probabilities are above a probability threshold, which is indeed a superlevel set in the solution space. We introduce a set estimator based on confidence bounds on each solution's exceedance probability computed from the GP model. Then, we propose an acquisition function that effectively reduces the estimation error of the superlevel set by selecting a solution-parameter pair expected to cause the largest classification changes of the solutions should it be sampled.

Among the reviewed work, the problem in Iwazaki et al. (2020) also has a nested structure as it requires estimating an exceedance probability before forming a superlevel set. However, their sampling decision is still made on the control space only, whereas ours must be made for both solution and parameter spaces.

The rest of the paper is organized as follows. Section 2 mathematically defines our problem. Section 3 introduces the GP model we adopt and the set estimator constructed from it. In Section 4, we discuss our acquisition function and its computation. Section 5 presents our sequential sampling algorithm to estimate the superlevel set followed by its empirical demonstrations in Section 6.

2 PROBLEM DEFINITION

Consider a simulation optimization problem,

$$x^c = \arg\max_{x \in \mathscr{X}} \mathrm{E}[Y(x;\theta^c)], \tag{1}$$

where $\mathscr{X}$ is a finite feasible solution space in $\mathbb{R}^n$, θ^c is the true parameter vector, and $Y(x;\theta^c)$ is the simulation output at Solution $x \in \mathscr{X}$. The goal of (1) is to find optimal solution x^c that maximizes the expected performance of a simulated system.

In many practical scenarios, θ^c is unknown. In some cases, data is available to estimate θ^c by exploiting a parametric model, but in others, the decision maker may have to model its uncertainty without data. Even in the former case, the uncertainty about θ^c is not fully resolved since there is estimation error caused by the finiteness of the data. In this work, we assume that we are given distribution $f(\theta)$ defined on its support $\Theta = \{\theta \in \mathbb{R}^d : f(\theta) > 0\}$, which models the uncertainty about θ^c. For instance, if θ is a parameter vector of the input distribution of the simulator whose uncertainty is modeled with a Bayesian prior before collecting any data, then $f(\theta)$ may represent the posterior distribution of θ given the data.

Several papers study how to reflect the uncertainty modeled by $f(\theta)$ in Problem (1) by modifying the problem formulation as reviewed in Section 1. In this work, we approach Problem (1) from the perspective of statistical inference. To facilitate the discussion, let us define the simulation output of Solution x run with parameter $\theta \in \Theta$ by $Y(x;\theta) = \mathrm{E}[Y(x;\theta)|\theta] + \varepsilon(x;\theta)$, where $\varepsilon(x;\theta)$ is the simulation error with mean 0 and variance $0 < v(x,\theta) < \infty$ conditional on θ. We refer to $\mathrm{E}[Y(x;\theta)|\theta]$ as the *conditional response* at x given θ, which maps (x,θ) to $\mathbb{R}$. Our objective is to assess the risk caused by unknown θ^c by classifying the feasible solutions in $\mathscr{X}$ according to the probability that their conditional responses exceed a user-chosen threshold, δ. For each $x \in \mathscr{X}$, let us define the exceedance probability,

$$p_\delta(x) := \Pr\{\mathrm{E}[Y(x;\theta)|\theta] > \delta\}, \tag{2}$$

where the probability is taken with respect to $f(\theta)$. Namely, $p_\delta(x)$ is the probability that the conditional response at x exceeds the threshold, δ, under all possible realizations of θ prescribed by $f(\theta)$. Our goal is to identify the set of solutions given the user-specified probability threshold, $0 < \alpha < 1$:

$$S_\alpha(\delta) := \{x \in X | p_\delta(x) > \alpha\}. \tag{3}$$

In words, $S_\alpha(\delta)$ is the set of solutions whose response exceeds δ with a significant probability ($> \alpha$) measured with $f(\theta)$.

Although our discussion above focuses on the objective function of (1), the condition, $p_\delta(x) > \alpha$, adopted to define $S_\alpha(\delta)$ can be interpreted as a chance constraint on the solutions' acceptable mean performances ($> \delta$) in the presence of uncertainty about θ^c. With this interpretation, $S_\alpha(\delta)$ represents a set of solutions that satisfy the chance constraint.

The definition in (3) can be further modified to make other useful inferences for (1). Suppose the decision maker selects $\hat{x}$ as a plausible solution to implement in the system. This may be a solution returned by a simulation optimization algorithm designed to solve a modified version of (1) incorporating $f(\theta)$. For instance, Kim et al. (2025) propose to approximate x^c with the solution that has the largest probability of being optimal with respect to $f(\theta)$. Before adopting $\hat{x}$, the decision maker may want to assess the risk of implementing $\hat{x}$ caused by uncertain θ^c. If we replace $Y(x;\theta)$ in (2) with $Y(x;\theta) - Y(\hat{x};\theta)$ and let $\delta > 0$, then $p_\delta(x)$ can be interpreted as the probability that x performs better than $\hat{x}$ by more than δ. Consequently, the resulting $S_\alpha(\delta)$ contains the solutions whose probability of performing more than δ better than $\hat{x}$ is at least α. If δ is chosen as the smallest difference in performance that the decision maker cares to differentiate, then $S_\alpha(\delta)$ is the set of solutions that are practically better than $\hat{x}$ with significant probability α (Song 2021). We leave the opportunities to explore these and more variances of (2) and (3) to later work, and focus on estimating (3) in this paper.

3 GAUSSIAN PROCESS MODEL AND NESTED LEVEL SET ESTIMATOR

In general, a superlevel set of function $g : \mathscr{D} \to \mathbb{R}$ refers to a subset in domain $\mathscr{D}$ that produces an output exceeding a threshold. Observe that $S_\alpha(\delta)$ has a nested superlevel set structure. At the outer-level, it is defined as a superlevel set of $x \in \mathscr{X}$ with respect to $p_\delta(x)$ exceeding the probability threshold, α. At each solution level, estimating $p_\delta(x)$ involves identifying a superlevel set of θ that satisfies $\{\mathrm{E}[Y(x;\theta)|\theta] > \delta\}$ and calculating the probability of the set with respect to $f(\theta)$. Thus, characterizing $S_\alpha(\delta)$ can be viewed as a nested superlevel set estimation problem.

As reviewed in Section 1, there are several papers focusing on the superlevel set estimation problem in statistics and machine learning. However, extending it to the nested setting is not straightforward as both p_δ and $\mathrm{E}[(x;\theta)|\theta]$ need to be estimated from the simulation outputs. Moreover, the estimation error in the former propagates to the latter. To estimate $S_\alpha(\delta)$ efficiently, it is important to adopt an estimator of $\mathrm{E}[(x;\theta)|\theta]$ for each x that allows us to quantify its estimation error as well as how it propagates to the estimation error of p_α.

To this end, in Section 3.1, we introduce a Gaussian process (GP) model that takes (x,θ) as input and models $\mathrm{E}[Y(x;\theta)|\theta]$ as an output. Section 3.2 introduces the estimator of $S_\alpha(\delta)$ based on the GP model.

3.1 Gaussian Process Model

Suppose that we regard $\mathrm{E}[Y(x;\theta)|\theta]$ as a realization of a stochastic process that maps $\mathscr{X}\times\Theta$ to $\mathbb{R}$. Specifically, we assume that $\{\mathrm{E}[Y(x;\theta)|\theta]\}_{x\in\mathscr{X},\theta\in\Theta}$ is sampled from a prior GP, $GP(\mu_0(x,\theta),k_0(x,\theta;x',\theta'))$, where μ_0 is the mean function and k_0 is the covariance kernel $k_0:(\mathscr{X}\times\Theta)\times(\mathscr{X}\times\Theta)\to\mathbb{R}$. To distinguish the true conditional response surface from its GP model, we introduce notation $\eta_t(x,\theta)$, which is the GP estimator for $\mathrm{E}[Y(x;\theta)|\theta]$ after observing the simulation outputs at solution-parameter pairs until the tth iteration. Thus, the prior GP can be written as

$$\eta_0(x,\theta)\sim GP(\mu_0(x,\theta),k_0(x,\theta;x',\theta')) \quad \text{for} \quad x,x'\in\mathscr{X}. \tag{4}$$

The mean and covariance functions, μ_0 and k_0, can be parameterized by hyperparamters. We adopt $\mu_0(x,\theta)=\beta_0\in\mathbb{R}$ as a constant prior mean. For the covariance function, we impose $k_0(x,\theta;x',\theta')=\tau^2\gamma_{\mathscr{X}}(x,x')\gamma_\Theta(\theta,\theta')$, where $\tau^2\in\mathbb{R}$ is the marginal variance of the prior GP, and $\gamma_{\mathscr{X}}$ and γ_Θ are correlation kernels defined on $\mathscr{X}\times\mathscr{X}$ and $\Theta\times\Theta$, respectively. We assume that both $\gamma_{\mathscr{X}}$ and γ_Θ are positive definite kernels, which makes k_0 a positive definite kernel. The parameters of the prior GP can be estimated through maximum likelihood estimation (MLE) after sampling n_0 initial solution-parameter pairs $(x_1,\theta_1),(x_2,\theta_2),\ldots,(x_{n_0},\theta_{n_0})$ and simulating r replications at each pair (x_i,θ_i). We define the average of the simulation output at (x_i,θ_i), $\bar{Y}_i:=\sum_{j=1}^r Y_j(x_i,\theta_i)/r$ where $Y_j(x_i,\theta_i)$ is the jth simulation output at (x_i,θ_i). Then, the GP prior is updated to the GP posterior conditional on $\mathscr{F}_0=\{(x_1,\theta_1,\bar{Y}_1),(x_2,\theta_2,\bar{Y}_2),\ldots,(x_{n_0},\theta_{n_0},\bar{Y}_{n_0})\}$.

We define the simulation history at the tth iteration, $\mathscr{F}_t:=\mathscr{F}_{t-1}\cup\{(x_1,\theta_1,\bar{Y}_1),(x_2,\theta_2,\bar{Y}_2),\cdots,(x_{n_t},\theta_{n_t},\bar{Y}_{n_t})\}$, where n_t is the number of solution-parameter pairs simulated at the tth iteration. In this paper, we assume only one solution-parameter pair is sampled at each iteration, i.e., $n_t=1$, for $t=1,2,\cdots$. This simplifies $\mathscr{F}_t=\{(x_1,\theta_1,\bar{Y}_1),(x_2,\theta_2,\bar{Y}_2),\cdots,(x_{n_0+t},\theta_{n_0+t},\bar{Y}_{n_0+t})\}$.

There are two advantages of adopting the GP model: 1) the posterior model after observing $\mathscr{F}_t$ is also a GP, and 2) the posterior GP is completely specified by the mean and covariance functions. Given stochastic sample $Y_t:=(\bar{Y}_1,\ldots,\bar{Y}_{n_0+t})^\top$ observed at $X_t:=\{(x_1,\theta_1),(x_2,\theta_2),\ldots,(x_{n_0+t},\theta_{n_0+t})\}$, the posterior GP's mean $\mu_t(x,\theta)$ and covariance $k_t(x,\theta;x',\theta')$ can be computed as

$$\begin{aligned}\mu_t(x,\theta)&=\beta_0+\Sigma_t(x,\theta)^\top(\Sigma_t+\Sigma_t^\varepsilon)^{-1}(Y_t-\beta_0 1_{n_0+t}),\\ k_t(x,\theta;x',\theta')&=k_0(x,\theta;x',\theta')-\Sigma_t(x,\theta)^\top(\Sigma_t+\Sigma_t^\varepsilon)^{-1}\Sigma_t(x',\theta'),\end{aligned} \tag{5}$$

where $\Sigma_t(x,\theta)$ is a (n_0+t)-dimensional vector of covariances between (x,θ) and X_t stipulated by kernel k_0, $\Sigma_t:=[k_0(x,\theta;x',\theta')]_{(x,\theta),(x',\theta')\in X_t}$, Σ_t^ε is the variance-covariance matrix of the simulation errors of Y_t, and 1_n is an n-dimensional vector of ones. From (5), the posterior variance at (x,θ) is computed as $\sigma_t^2(x,\theta)=k_t(x,\theta;x,\theta)$. The posterior GP given $\mathscr{F}_t$ can be written as $\eta_t(x,\theta)\sim GP(\mu_t(x,\theta),k_t(x,\theta;x',\theta'))$ for $x,x'\in\mathscr{X}$ and $\theta,\theta'\in\Theta$. Moreover, η_t, evaluated at the finite set of t distinct (x,θ) pairs, $(x_1,\theta_1),(x_2,\theta_2),\ldots,(x_t,\theta_t)$, follows a multivariate normal distribution: $(\eta_t(x_1,\theta_1),\ldots,\eta_t(x_t,\theta_t))^\top\sim\mathscr{N}(\mu_t,K_t)$, where $\mu_t=(\mu_t(x_1,\theta_1),\ldots,\mu_t(x_t,\theta_t))^\top$ and K_t is a $t\times t$ covariance matrix whose (i,j)th element is $k_t(x_i,\theta_i;x_j,\theta_j)$.

3.2 Nested Level Set Estimator

To infer $S_\alpha(\delta)$, we first estimate $p_\delta(x)$ in (2) for each $x\in\mathscr{X}$. We define the estimator of $p_\delta(x)$ from the posterior GP model, $\eta_t(x,\theta)$, as

$$p_{t,\delta}(x):=\Pr_{\theta\sim f(\theta)}\{\eta_t(x,\theta)>\delta\}=\int_\Theta \mathbb{1}\{\eta_t(x,\theta)>\delta|\theta\}f(\theta)d\theta. \tag{6}$$

Note that $p_{t,\delta}(x)$ is a random variable that depends on the sample path of η_t. We denote the mean and variance of (6) by $\mu_t^{(p)}(x):=\mathrm{E}_{GP}[p_{t,\delta}(x)]$ and $\gamma_t^2(x):=V_{GP}[p_{t,\delta}(x)]$, respectively, where E_{GP} and V_{GP} are

taken with respect to the posterior GP. We further derive

$$\mu_t^{(p)}(x) = \mathrm{E}_{GP}\left[\int_\Theta \mathbb{1}\{\eta_t(x,\theta) > \delta|\theta\} f(\theta) d\theta\right] = \int_\Theta \Pr_{GP}\{\eta_t(x,\theta) > \delta|\theta\} f(\theta) d\theta$$
$$= \int_\Theta \left\{1 - \Phi\left(\tfrac{\delta - \mu_t(x,\theta)}{\sigma_t(x,\theta)}\right)\right\} f(\theta) d\theta = \int_\Theta (1 - \Phi_t(x,\theta)) f(\theta) d\theta, \tag{7}$$

where the second equality follows from interchanging the two expectations via Fubini's theorem, and $\Phi(\cdot)$ is the cumulative distribution function (cdf) of the standard normal distribution. The last equality follows from defining $\Phi_t(x,\theta) := \Phi\left(\tfrac{\delta - \mu_t(x,\theta)}{\sigma_t(x,\theta)}\right)$. For the variance of $p_{t,\delta}(x)$, we have

$$\begin{aligned}
\gamma_t^2(x) &= V_{GP}[p_{t,\delta}(x)] = \mathrm{E}_{GP}[p_{t,\delta}(x)^2] - \{\mathrm{E}_{GP}[p_{t,\delta}(x)]\}^2 \\
&= \mathrm{E}_{GP}\left[\int_\Theta \int_\Theta \mathbb{1}\{\eta_t(x,\theta) > \delta|\theta\} \mathbb{1}\{\eta_t(x,\theta') > \delta|\theta'\} f(\theta) f(\theta') d\theta d\theta'\right] - \left\{\int_\Theta (1 - \Phi_t(x,\theta)) f(\theta) d\theta\right\}^2 \\
&= \int_\Theta \int_\Theta \left\{\Pr_{GP}\left\{\eta_t(x,\theta) > \delta, \eta_t(x,\theta') > \delta|\theta,\theta'\right\} - (1 - \Phi_t(x,\theta))(1 - \Phi_t(x,\theta'))\right\} f(\theta) f(\theta') d\theta d\theta' \\
&= \int_\Theta \int_\Theta \left\{\Phi_t(x,\theta;x,\theta') - \Phi_t(x,\theta)\Phi_t(x,\theta')\right\} f(\theta) f(\theta') d\theta d\theta',
\end{aligned} \tag{8}$$

where the third equality follows by rewriting the second moment as the double integral. The two random variables, Z and Z', are bivariate standard normal random variables with correlation $\frac{k_t(x,\theta;x,\theta')}{\sigma_t(x,\theta)\cdot\sigma_t(x,\theta')}$. The last equality follows from the inclusion–exclusion principle, i.e., $\Pr(A \cap B) = 1 - \Pr(A^c) - \Pr(B^c) + \Pr(A^c \cap B^c)$, and by defining $\Phi_t(x,\theta;x,\theta') := \Pr\left\{Z \le \tfrac{\delta - \mu_t(x,\theta)}{\sigma_t(x,\theta)}, Z' \le \tfrac{\delta - \mu_t(x,\theta')}{\sigma_t(x,\theta')} \middle| \theta,\theta'\right\}$.

Computing the integrals in (7) and (8) exactly is difficult in general. Instead, we approximate them with their Monte Carlo estimates by drawing a finite random sample, $\theta_1, \theta_2, \ldots, \theta_B \overset{\text{i.i.d.}}{\sim} f(\theta)$. Essentially, we replace Θ with the size-B sample and regard each θ_b equally likely. This also benefits the acquisition function optimization in the sequential sampling algorithm discussed in Section 4 as the candidate θs can be restricted to a finite set. Replacing Θ with the size-B sample, $p_\delta(x)$ for each x can be approximated by

$$\tilde{p}_\delta(x) := \frac{1}{B}\sum_{b=1}^B \mathbb{1}\{\mathrm{E}[Y(x;\theta_b)|\theta_b] > \delta\}. \tag{9}$$

How closely $\tilde{p}_\delta(x)$ approximates $p_\delta(x)$ depends on the sample size, B. It is easy to see that $\tilde{p}_\delta(x)$ is an unbiased estimator of $p_\delta(x)$. Lemma 1 ensures that the estimation error of $\tilde{p}_\delta(x)$ diminishes exponentially fast in B.

Lemma 1 For all $\tau > 0$, $\Pr\{|\tilde{p}_\delta(x) - \mathrm{E}[\tilde{p}_\delta(x)]| \ge \tau\} \le 2\exp\left(-2B\tau^2\right)$.

Proof. Let $X_b = \mathbb{1}\{\mathrm{E}[Y(x;\theta_b)|\theta_b] > \delta\}$. Then, $\frac{X_1}{B}, \ldots, \frac{X_B}{B}$ are i.i.d. random variables such that $0 \le \frac{X_b}{B} \le \frac{1}{B}$ almost surely. Let $S_B = \sum_{b=1}^B \frac{X_b}{B}$. From Hoeffding's inequality,

$$\Pr\{|S_B - \mathrm{E}[S_B]| \ge \tau\} = \Pr\{|\tilde{p}_\delta(x) - \mathrm{E}[\tilde{p}_\delta(x)]| \ge \tau\} \le 2\exp\left(-\frac{2\tau^2}{\sum_{i=1}^B (\frac{1}{B} - 0)^2}\right) = 2\exp\left(-2B\tau^2\right). \qquad \square$$

In the remainder of the paper, we assume that $\{\theta_1, \theta_2, \ldots, \theta_B\}$ is fixed and replaces Θ. Any statistical statement is conditional on the sample unless otherwise mentioned.

The expressions in (7) and (8) can be rewritten respectively as:

$$\mu_t^{(p)}(x) = \frac{1}{B}\sum_{b=1}^B (1 - \Phi_t(x,\theta_b)), \quad \gamma_t^2(x) = \frac{1}{B^2}\sum_{i=1}^B \sum_{j=1}^B \{\Phi_t(x,\theta_i;x,\theta_j) - \Phi_t(x,\theta_i)\cdot\Phi_t(x,\theta_j)\}, \tag{10}$$

which can be computed easily. The following lemma states the strong consistency of $\mu_t^{(p)}(x)$.

Lemma 2 As $t \to \infty$, if x is simulated infinitely often at all $\theta_b, 1 \le b \le B$, then $\mu_t^{(p)}(x) \xrightarrow{a.s.} \tilde{p}_\delta(x)$ and $\gamma_t^2(x) \xrightarrow{a.s.} 0$.

Although we omit its proof due to the space limit here, Lemma 2 follows straightforwardly from the consistency of the posterior mean of the GP.

From (10), we construct the following interval estimate for $\tilde{p}_\delta(x)$ for each $x \in \mathscr{X}$:

$$Q_t(x) = \left[\mu_t^{(p)}(x) \pm \beta_t^{1/2}\gamma_t(x)\right], \tag{11}$$

where $\beta_t > 0$ plays a similar role as the critical value of a CI that determines its coverage. The larger β_t is, the more weight is assigned to the uncertainty of $p_{t,\delta}(x)$ due to its prediction error characterized by $\gamma_t^2(x)$. By comparing each solution x's lower bound given by $Q_t(x)$ with the threshold probability, α, we define the following estimator of $S_\alpha(\delta)$ in (3):

$$\tilde{S}^t := \{x \in \mathscr{X} | \mu_t^{(p)}(x) - \beta_t^{1/2}\gamma_t(x) > \alpha - \varepsilon\}, \tag{12}$$

where $\varepsilon > 0$ the error tolerance in classifying the solutions based on the estimated $\tilde{p}_\delta(x)$. We suppress the dependence of $\tilde{S}^t$ on α and δ for notational convenience. Similarly, we define the estimator of $S_\alpha^c(\delta)$ by comparing the upper bound of $Q_t(x)$ with α:

$$\tilde{C}^t := \{x \in \mathscr{X} | \mu_t^{(p)}(x) + \beta_t^{1/2}\gamma_t(x) \le \alpha + \varepsilon\}.$$

Again, α is relaxed to $\alpha + \varepsilon$. In both $\tilde{S}^t$ and $\tilde{C}^t$, ε makes the classifications of solutions more conservative. To understand why such ε is needed, suppose that for some $x \in \mathscr{X}$, $\tilde{p}_\delta(X) = \alpha$. Then, for any finite t, x is incorrectly classified to $\tilde{S}_\alpha^t(\delta)$ with probability 0.5 even if $\mu_t^{(p)}(x) = \tilde{p}_\delta(x)$. By introducing ε, we allow the solutions whose $\tilde{p}_\delta(x)$ fall within $[\alpha - \varepsilon, \alpha + \varepsilon]$ to be classified in both sets as $t \to \infty$ (cf. Lemma 2).

For any finite t, $\tilde{S}_\alpha^t(\delta)$ and $\tilde{C}_\alpha^t(\delta)$ do not necessarily span $\mathscr{X}$, i.e., some solutions may remain unclassified. Indeed, when $\gamma_t(x)$ is large enough so that $Q(x)$ covers the interval, $[\alpha - \varepsilon, \alpha + \varepsilon]$, then x remains unclassified at t. As $t \to \infty$, all solutions are classified according to Lemma 2. Therefore, we adopt the stopping criterion that the unclassified set is empty for our sequential sampling algorithm.

The choice of ε should depend on the user's error tolerance level. Nevertheless, we provide general guidelines here: 1) choosing larger ε makes the algorithm terminate earlier while tolerating a maximum classification error of ε and 2) smaller ε requires more sampling to tighten the estimation error bounds around α, resulting in higher accuracy at a higher sampling cost.

4 ACQUISITION FUNCTIONS

An efficient sequential sampling algorithm would make sampling decisions to quickly classify the unclassified solutions in subsequent iterations. Unlike a typical level set estimation problem, our problem has a nested structure where we need to not only choose which solution x to sample, but also decide which θ to sample with x. In this section, we introduce an acquisition function that guides our algorithm to select solution-parameter pairs to sample.

In their superlevel set estimation problem, Zanette et al. (2018) adopt the acquisition function that maximizes the expected size of the estimated superlevel set in the next iteration. Similar approaches are taken by Iwazaki et al. (2020). However, doing so penalizes the false negatives in identifying the superlevel set, not the false positives. To see this, suppose solution x's function value is overestimated, and thus x is incorrectly classified in the superlevel set. Then, even if $\gamma_t^2(x)$ is large, this acquisition function may assign a small value to x.

To reduce both false positives and negatives, our acquisition function A_t evaluates the expected number of solutions whose classifications change from being included in $\tilde{S}^t_\delta$ to be excluded from it (and vice versa) after sampling (x', θ') at the tth iteration for each pair of $x' \in \mathscr{X}$ and $\theta' \in \Theta$. To mathematically define A_t, let $\triangle$ denote the set difference operator such that $E \triangle F := (E \setminus F) \cup (F \setminus E)$ for two sets E and F. Namely, $E \triangle F$ is the set of all elements included or excluded by only one of E and F. After sampling some (x', θ') at the tth iteration, suppose we update $\tilde{S}^t$ to $\tilde{S}^{t+1}$, then the number of solutions whose classifications change to be in/out of the superlevel set is equal to $|\tilde{S}^{t+1} \triangle \tilde{S}^t| = |\tilde{S}^{t+1} \setminus \tilde{S}^t| + |\tilde{S}^t \setminus \tilde{S}^{t+1}|$. Therefore, the acquisition function, A_t, can be written as

$$A_t(x', \theta') := \mathrm{E}\left[|\tilde{S}^{t+1} \triangle \tilde{S}^t| \,\middle|\, \mathscr{F}_t, a_t = (x', \theta') \right]. \tag{13}$$

We choose the next sampling pair that maximizes A_t: $(x^*, \theta^*) = \arg\max_{(x', \theta') \in \mathscr{X} \times \Theta} A_t(x', \theta')$. In words, sampling (x', θ') is expected to make the largest classification change in $\tilde{S}^{t+1}$ from $\tilde{S}^t$ given $\mathscr{F}_t$.

To discuss the computation of (13), we first rewrite it as

$$\begin{aligned}
(13) &= \mathrm{E}\left[\textstyle\sum_{x \notin \tilde{S}^t} \mathbb{1}[x \in \tilde{S}^{t+1}] \,\middle|\, \mathscr{F}_t, a_t = (x', \theta') \right] + \mathrm{E}\left[\textstyle\sum_{x \in \tilde{S}^t} \mathbb{1}[x \notin \tilde{S}^{t+1}] \,\middle|\, \mathscr{F}_t, a_t = (x', \theta') \right] \\
&= \textstyle\sum_{x \notin \tilde{S}^t} \Pr\{ \mu^{(p)}_{t+1}(x) - \beta^{1/2}_{t+1} \gamma_{t+1}(x) > \alpha - \varepsilon | \mathscr{F}_t, a_t = (x', \theta') \} \\
&\quad + \textstyle\sum_{x \in \tilde{S}^t} \Pr\{ \mu^{(p)}_{t+1}(x) - \beta^{1/2}_{t+1} \gamma_{t+1}(x) \leq \alpha - \varepsilon | \mathscr{F}_t, a_t = (x', \theta') \}
\end{aligned} \tag{14}$$

where the second equality follows from the definition in (12). Computing the probabilities in (14) requires the predictive distribution of $\mu^{(p)}_{t+1}(x) - \beta^{1/2}_{t+1} \gamma_{t+1}(x)$ conditional on sampling $a_t = (x', \theta')$. Let $\mu_t \in \mathbb{R}^{|\mathscr{X}|B}$ and $V_t \in \mathbb{R}^{|\mathscr{X}|B \times |\mathscr{X}|B}$ denote the mean vector and the variance-covariance matrix of the posterior GP at all $\mathscr{X} \times \{\theta_1, \theta_2, \ldots, \theta_B\}$ at the tth iteration. From their respective expressions in (10), observe that $\mu^{(p)}_{t+1}(x)$ and $\gamma_{t+1}(x)$ are completely specified by μ_{t+1} and V_{t+1}. Given $a_t = (x', \theta')$, it can be shown that μ_{t+1} is a multi-variate normal vector, while V_{t+1} is derived deterministically regardless of the simulation result at a_t (Xie et al. 2016):

$$\mu_{t+1} \sim \mathscr{N}(\mu_t, V_t - V_{t+1}), \quad V_{t+1} = V_t - \frac{V_t e_{(x', \theta')} e^\top_{(x', \theta')} V_t^\top}{v(x', \theta')/r + e^\top_{(x', \theta')} V_t^\top e_{(x', \theta')}}, \tag{15}$$

where $e_{(x,\theta)} \in \mathbb{R}^{|\mathscr{X}|B}$ is a standard basis vector that has one corresponding to (x, θ) and zero elsewhere. The marginal predictive variance at (x, θ) can be computed from V_{t+1}: $\sigma^2_{t+1}(x, \theta) = e^\top_{(x,\theta)} V_{t+1} e_{(x,\theta)}$.

Because $\mu^{(p)}_{t+1}(x) - \beta^{1/2}_{t+1} \gamma_{t+1}(x)$ is a nonlinear function of μ_{t+1}, deriving its exact distribution is challenging. Instead, we approximate $\mu^{(p)}_{t+1}(x) - \beta^{1/2}_{t+1} \gamma_{t+1}(x)$ by a linear function in μ_{t+1} around $\mu_{t+1} = \mu_t$ and derive its distribution. First, we have the following approximation for $\mu^{(p)}_{t+1}(x)$

$$\mu^{(p)}_{t+1}(x) \approx \frac{1}{B} \sum_{b=1}^{B} \Phi\left(\frac{\mu_t(x, \theta_b) - \delta}{\sigma_{t+1}(x, \theta_b)} \right) + \frac{1}{B} \sum_{b=1}^{B} \frac{1}{\sigma_{t+1}(x, \theta_b)} \phi\left(\frac{\mu_t(x, \theta_b) - \delta}{\sigma_{t+1}(x, \theta_b)} \right) \{ \mu_{t+1}(x, \theta_b) - \mu_t(x, \theta_b) \}, \tag{16}$$

where $\phi(\cdot)$ is the standard normal probability density function. Similarly, taking the linear approximation of $\gamma_{t+1}(x)$ in μ_{t+1} at $\mu_{t+1} = \mu_t$ gives

$$\begin{aligned}
\gamma_{t+1}(x) \approx &\left\{ \frac{1}{B^2} \sum_{i=1}^{B} \sum_{j=1}^{B} \left[\Pr\left\{ Z_i \leq \frac{\delta - \mu_t(x, \theta_i)}{\sigma_{t+1}(x, \theta_i)}, Z_j \leq \frac{\delta - \mu_t(x, \theta_j)}{\sigma_{t+1}(x, \theta_j)} \right\} - \Phi\left(\frac{\delta - \mu_t(x, \theta_i)}{\sigma_{t+1}(x, \theta_i)} \right) \Phi\left(\frac{\delta - \mu_t(x, \theta_j)}{\sigma_{t+1}(x, \theta_j)} \right) \right] \right\}^{1/2} \\
&+ \frac{1}{2} \frac{(\nabla_{\mu_{t+1}(x)} \gamma^2_{t+1}(x)|_{\mu_{t+1}(x) = \mu_t(x)})^\top (\mu_{t+1}(x) - \mu_t(x))}{\left\{ \frac{1}{B^2} \sum_{i=1}^{B} \sum_{j=1}^{B} \left[\Pr\left\{ Z_i \leq \frac{\delta - \mu_t(x, \theta_i)}{\sigma_{t+1}(x, \theta_i)}, Z_j \leq \frac{\delta - \mu_t(x, \theta_j)}{\sigma_{t+1}(x, \theta_j)} \right\} - \Phi\left(\frac{\delta - \mu_t(x, \theta_i)}{\sigma_{t+1}(x, \theta_i)} \right) \Phi\left(\frac{\delta - \mu_t(x, \theta_j)}{\sigma_{t+1}(x, \theta_j)} \right) \right] \right\}^{1/2}},
\end{aligned} \tag{17}$$

where $\mu_t(x) \in \mathbb{R}^B$ is a subvector of μ_t corresponding to x, and Z_i and Z_j are bivariate standard normal variables with correlation $\frac{k_{t+1}(x,\theta_i;x,\theta_j)}{\sigma_{t+1}(x,\theta_i)\sigma_{t+1}(x,\theta_j)}$. From (16) and (17), it follows that $\mu_{t+1}^{(p)}(x) - \beta_{t+1}^{1/2}\gamma_{t+1}(x)$ is approximately normally distributed as

$$\mu_{t+1}^{(p)}(x) - \beta_{t+1}^{1/2}\gamma_{t+1}(x) \approx h_x + g_x^\top(\mu_{t+1}(x) - \mu_t(x)) \sim \mathcal{N}(h_x, (g_x^\top w_x)^2), \tag{18}$$

where the expressions for $h_x \in \mathbb{R}$ and $g_x, w_x \in \mathbb{R}^B$ are given in Appendix A.

From (18), we can approximate $A_t(x', \theta')$ in (14) by

$$\tilde{A}_t(x', \theta') := \sum_{x \notin \tilde{S}^t} \Phi\left(-\frac{\alpha - \varepsilon - h_x}{|g_x^\top w_x|}\right) + \sum_{x \in \tilde{S}^t} \Phi\left(\frac{\alpha - \varepsilon - h_x}{|g_x^\top w_x|}\right). \tag{19}$$

Therefore, the next sampling pair can be determined by solving $(x^*, \theta^*) = \arg\max_{(x',\theta') \in \mathscr{X} \times \Theta} \tilde{A}_t(x', \theta')$.

Computing $\tilde{A}_t$ for all $B|\mathscr{X}|$ pairs at each t is expensive. To reduce the cost, we first select a 'good' parameter for each solution x to sample and then evaluate $\tilde{A}_t$ for the down-selected solution-parameter pairs. Let $\theta_{t,x}$ represent the parameter selected for x at the tth iteration according to some criterion. Then, our sampling decision with the reduced action space becomes $(x^*, \theta^*) = \arg\max_{x \in \mathscr{X}} \tilde{A}_t(x, \theta_{t,x})$. While other criteria are possible, in our empirical studies in Section 6, we select $\theta_{t,x}$ to be the posterior variance-maximizing parameter of the GP model at x, i.e., $\theta_{t,x} = \arg\max_{\theta_b \in \{\theta_1, \theta_2, \ldots, \theta_B\}} \sigma_t(x, \theta_b)$.

5 SEQUENTIAL SAMPLING ALGORITHM

Algorithm 1 presents our procedure that fits and updates the GP model in Section 3.1, estimates $S_\alpha(\delta)$ from (12), and selects the next sampling solution-parameter pair by applying the acquisition function introduced in Section 4.

At the beginning of the algorithm and every p iterations thereafter, it refits the GP hyperparameters via maximum likelihood estimation. We compute $\mu_t^{(p)}(x)$ and $\gamma_t(x)$ for each $x \in \mathscr{X}$ and construct the CI, $Q_t(x)$, in Lines 9-11. From these CIs, we update $\tilde{S}^t$ and $\tilde{C}^t$ (Line 12), and find $\theta_{t,x}$ for each $x \in \mathscr{X}$. We then evaluate the approximate acquisition function, $\tilde{A}_t(x, \theta_{t,x})$, for each $x \in \mathscr{X}$ and select the maximizer (x^*, θ^*) for sampling (Lines 13-15). The algorithm terminates when the unclassified set is empty or the simulation budget is exhausted (Line 19).

Recall that we introduce ε and relax the set classification criteria in Section 4. The following theorem states the statistical guarantee that Algorithm 1 provides.

Theorem 1 Suppose that $\{\mathrm{E}[Y(x,\theta)|\theta]\}_{x \in \mathscr{X}, \theta \in \Theta}$ is a realization of the prior GP η_0 in (4). For any $\alpha \in (0,1)$, $\xi \in (0,1)$, $\varepsilon > 0$ and $\Theta = \{\theta_1, \theta_2, \ldots, \theta_B\}$, if $\beta_t = |\mathscr{X}|/\xi$, then Algorithm 1 terminates after a finite number of iterations, T. Moreover, with probability no less than $1 - \xi$, the following holds simultaneously for every $x \in \mathscr{X}$: if $\tilde{p}_\delta(x) > \alpha + \varepsilon$ then $x \in \tilde{S}^T \setminus \tilde{C}^T$, and if $\tilde{p}_\delta(x) < \alpha - \varepsilon$ then $x \in \tilde{C}^T \setminus \tilde{S}^T$.

The proof follows by extending the results in Iwazaki et al. (2020), which we omit here due to the page limit. Theorem 1 guarantees that Algorithm 1 terminates and classifies all solutions in finite time. Moreover, Theorem 1 ensures that we can identify all solutions whose exceedance probabilities are greater than $\alpha + \varepsilon$ in finite time with probability at least $1 - \xi$.

6 EMPIRICAL STUDY

In this section, we illustrate the empirical performance of our algorithm using an M/M/1/k queueing system. The decision variable $x = k$ represents the system capacity, and the goal is to minimize the expected net cost per customer, which can be written as: $\mathrm{E}[\text{net cost per customer}] = c\mathrm{E}[\text{waiting time}] - rev(1 - \Pr\{\text{balking}\})$, where $c = 1$ is the cost per unit waiting time per customer, and $rev = 1$ is the revenue per each served

Algorithm 1 Sequential nested superlevel set estimation procedure

Input: $f(\theta)$, number of sampled parameters B, user-specified thresholds δ and α, number of replications r, initial sample size n_0, period for GP parameter update p, simulation budget T.
Sample $\theta_1, \theta_2, \ldots, \theta_B$ from $f(\theta)$.
Select n_0 initial (x,θ) pairs from $\mathscr{X} \times \{\theta_1, \theta_2, \ldots, \theta_B\}$, simulate r replications at each pair to obtain $Y_0 = \{\bar{Y}_1, \ldots, \bar{Y}_{n_0}\}$, and update $\mathscr{F}_0 \leftarrow \{(x_1, \theta_1, \bar{Y}_1), \ldots, (x_{n_0}, \theta_{n_0}, \bar{Y}_{n_0})\}$.
$t \leftarrow 0$.
do
  If $t \bmod p = 0$ **Then**
    Estimate the hyperparameters of the GP via MLE given $\mathscr{F}_t$
  Compute the mean function μ_t and covariance function V_t of GP conditional on $\mathscr{F}_t$ in (5).
  For all $x \in \mathscr{X}$ **do**
    Compute $\mu_t^{(p)}(x)$ and $\gamma_t(x)$ in (10), and construct $Q_t(x)$ in (11).
  end for
  Update $\tilde{S}^t$ and $\tilde{C}^t$.
  For each x, find $\theta_{t,x} = \arg\max_{\theta_b \in \{\theta_1, \theta_2, \ldots, \theta_B\}} \sigma_t(x, \theta_b)$ to construct reduced action space
  For each $x \in \mathscr{X}$, compute $\tilde{A}_t(x, \theta_{t,x})$ in (19).
  Select $(x^*, \theta^*) = \arg\max_{x \in \mathscr{X}} \tilde{A}_t(x, \theta_{t,x})$.
  Run r replications at (x^*, θ^*) and update $\mathscr{F}_{t+1} \leftarrow \mathscr{F}_t \cup \{(x^*, \theta^*, \bar{Y}^*)\}$.
  Update μ_{t+1} and V_{t+1} conditional on $\mathscr{F}_{t+1}$.
  $t \leftarrow t+1$.
While $\tilde{S}^t \cup \tilde{C}^t \neq \mathscr{X}$ and $t < T$
Return $\tilde{S}^{t-1}$

customer. Since our algorithm is designed to maximize the objective, we multiply the expected net cost per customer by -1. We consider the candidate of system capacity $k = 1, \ldots, 50$, so $\mathscr{X} = \{1 \le x \le 50\}$. In this problem, Θ is the parameter set of the distribution of interarrival and service times. Suppose that the real-world interarrival and service times are exponentially distributed with their means denoted by θ_1 and θ_2, respectively. We model our uncertainty by assuming $\theta_1 \sim U(0.9, 1.1)$ and $\theta_2 \sim U(0.9, 1.8)$. To discretize Θ, we draw 11 random samples from each uniform distribution, 11 for θ_1 and 11 for θ_2, and form the Cartesian product. So, we replace Θ with the size-$B = 121$ samples.

To calculate $\tilde{p}_\delta(x)$, we compute the negative expected cost for each of the B sampled parameter vectors and for each capacity $x \in \mathscr{X}$, and compare it against the threshold δ. From (9), we obtain the discrete approximation of the exceedance probability, $\Pr\{\mathrm{E}[\text{net cost}|\theta_1, \theta_2] > \delta\}$, for each x. We assume that the user chose $\delta = 110$ and $\alpha = 0.8$, while $\varepsilon = 0.05$. We define $\tilde{S}_\varepsilon := \{x \in \mathscr{X} | \alpha - \varepsilon \le \tilde{p}_\delta(x) \le \alpha + \varepsilon\}$ whose elements are allowed to be misclassified and two sets, $R := \tilde{S}_\alpha(\delta) \setminus \tilde{S}_\varepsilon$ and $F := \tilde{S}^c_\alpha(\delta) \setminus \tilde{S}_\varepsilon$, where $\tilde{S}_\alpha(\delta) := \{x \in X | \tilde{p}_\delta(x) > \alpha\}$; R consists of the elements that must be included in $\tilde{S}^t$ while Fcontains the elements that must be excluded from $\tilde{S}^t$. These two sets are used to evaluate the classification accuracy. For our problem, $\tilde{S}_\alpha(\delta) = \{x \in \mathscr{X} | \tilde{p}_\delta(x) > 0.8\} = \{4,5,6,7,8,9,10,11,12\}$ and $\tilde{S}_\varepsilon = \{3,9,10,11,12,13,14\}$.

We evaluate each capacity x under uncertainty in the mean interarrival and service times by estimating its expected cost via simulations. Each time we simulate some (x,θ), we run $r = 30$ independent replications. In each replication, we simulate 100 customer arrivals: first drawing the system's initial occupancy from its steady-state distribution, then, computing each customer's cost using Lindley's equation. We average the 100 costs to produce a simulation output for one replication. Once (x,θ) is sampled, $v(x,\theta)$ is estimated by the sample variance of the outputs. Our initial design consists of $n_0 = 50$ pairs chosen by Latin hypercube sampling. At each iteration of the algorithm, we fit a GP surrogate to the observed pairs $\{(x,\theta), \bar{Y}, v(x,\theta)\}$

Table 1: Summary statistics averaged over 50 macro runs of Algorithm 1 for $\beta_t = 4, 9$, and 16. The standard errors are presented in parentheses. The last column contains the number of macro runs that terminated after satisfying the stopping criterion before $t = 150$.

β	Estimated set size	FN	FP	# Unclassified sol.	# macros terminated in time
4	11.74 (0.42)	1.00 (0.19)	2.44 (0.33)	1.32 (0.43)	35
9	9.04 (0.34)	0.06 (0.04)	0.42 (0.16)	9.32 (0.96)	4
16	7.64 (0.22)	0.0 (0.0)	0.04 (0.03)	19.36 (1.03)	0

using a Matérn covariance kernel with smoothness parameter $\nu = 2.5$. We test $\beta_t^{1/2} = 2, 3$ and 4, and the simulation budget $T = 150$ for numerical experiments.

We define two penalty functions to assess error in our set estimation problem. Suppose $\hat{T}$ be some estimator. We define false negatives (FN) $:= |R \setminus \hat{T}|$ as the number of required elements that $\hat{T}$ fails to include, and false positives (FP) $:= |F \cap \hat{T}|$ as the number of forbidden elements that $\hat{T}$ mistakenly includes. These two penalties measure omission and commission errors, respectively.

The experiment results are based on 50 macro-runs. The summary results of Algorithm 1 are shown in Table 1. Table 1 summarizes the performance of our sequential sampling algorithm over 50 macro runs for $\beta_t^{1/2} = 2, 3$ and 4. For each β, we report the average estimated set size, the mean number of false negatives (FN) and false positives (FP). If the macro run stops at $T = 150$, then we present the average number of solutions left unclassified at termination $T = 150$. The last column shows the number of macro runs completed within the simulation budget. The standard errors are presented in parentheses.

A larger choice of β makes all confidence bounds $Q_t(x)$ wider, which slows down the rate at which any solution is classified. So, the algorithm must sample more to achieve the same level of confidence in classifying each solution. In other words, larger β makes the algorithm more conservative, reducing the risk of including forbidden elements but requiring more iterations to terminate. In contrast, smaller β results in narrower confidence bounds $Q_t(x)$, quicker classifications, and fewer iterations until stopping, at the cost of potentially higher misclassification error.

This behavior can be observed in Table 1. For $\beta = 4$, the average estimated set size is the largest (11.74), with corresponding higher false negatives (0.84) and false positives (2.44), but the algorithm terminates within 150 iterations in 35 of 50 runs. Increasing $\beta = 9$ shrinks the estimated set size (9.04), reduces false negatives almost to zero (0.02), and reduces false positives (0.42), but only 4 runs meet the stopping criterion, and on average 9.32 solutions remain unclassified. For $\beta = 16$, the algorithm is the most conservative: the set size falls to 7.64, both the average FN and FP are 0.04, but no runs terminate early and nearly 20 solutions on average remain unclassified. Table 1 illustrates the trade-off between classification accuracy and termination time (or convergence speed) as β varies.

To further demonstrate the performance of Algorithm 1, we present the sampling frequencies of all solutions within a single macro run in Figure 1. The left panel shows the sampling frequencies with $\beta = 4$ (terminated after 27 iterations) and the right panel shows the sampling frequencies with $\beta = 16$ (terminated after 150 iterations). In both cases, $k = 1$ is sampled most frequently since the GP has large prediction errors at the lower k of the solution space from the property of M/M/1/k queueing system. Other than $k = 1$, the sampling decision concentrates on the k values near the true superlevel set since these points have stronger influence on whether a solution's exceedance probability crosses the user-specified probability threshold α. For $\beta = 4$, the earlier termination time leads to a wider spread of samples, whereas for $\beta = 16$ the longer termination time allows more focused exploration around the boundary region.

Furthermore, we present the fitted GP within a single run of the algorithm in Figure 2, which displays the posterior GP's mean and its confidence bounds at iteration 1 (blue solid line/shaded region) and at termination time (orange solid line/shaded region). The red line and the purple dotted line represent the

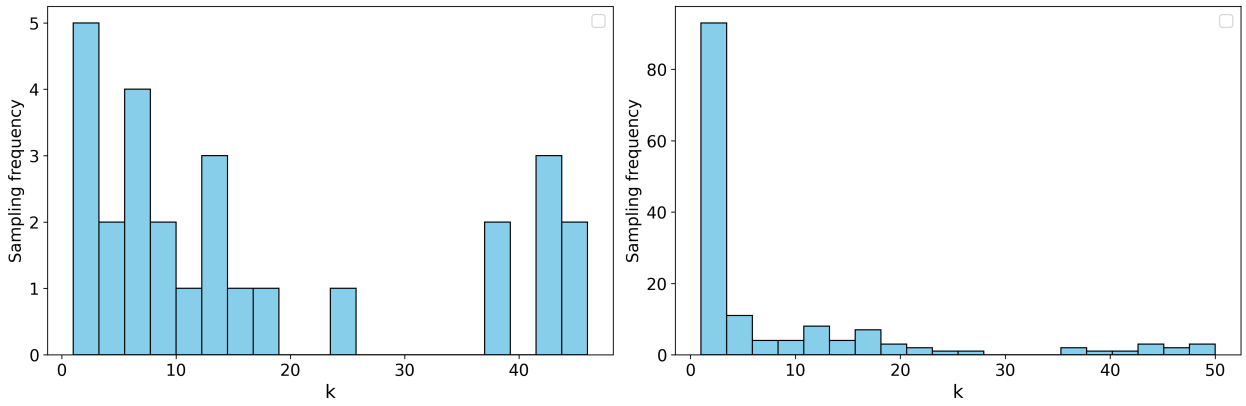

Figure 1: Sampling frequencies of solutions until stopping for $\beta = 4$ (left) and 16 (right).

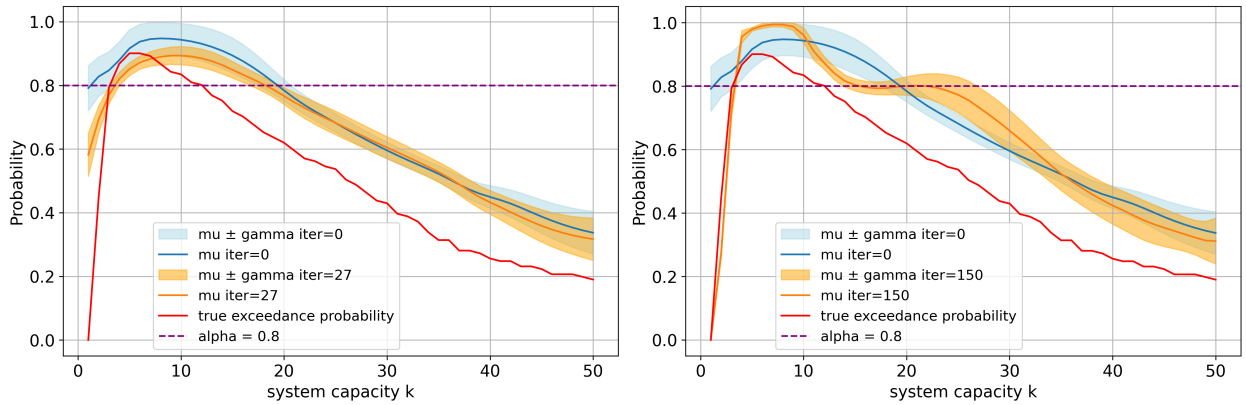

Figure 2: Fitted GP plot for $\beta = 4$ (left) and 16 (right).

true exceedance probability and the user-specified probability threshold $\alpha = 0.8$, respectively. Initially, the GP mean lies above the threshold with large uncertainty. At termination, the GP mean has been fitted downward where the true exceedance probability is below the threshold, and its confidence bound has narrowed near the superlevel set boundary, allowing the algorithm to reliably include or exclude solutions. In the right panel, the closer alignment of the GP mean to the true exceedance probability near the boundary shows how sequential sampling improves estimation accuracy and reduces uncertainty.

7 CONCLUSION

In this paper, we study the problem of identifying high-performing solutions when model parameters are uncertain. We formulate the problem as a nested superlevel set estimation problem and propose a set estimator based on the GP model fit to the simulation output mean as a function of the solution and parameter. The set estimator classifies each solution according to whether its lower confidence bound exceeds the target threshold with error tolerance. We also propose an acquisition function that aims to reduce both false positive and negative classifications and its efficient approximation scheme that keeps the computational cost manageable for a large solution/parameter space. In the extended version of this conference paper, we will provide a more thorough empirical study including benchmarking against the state-of-the-art procedures. Moreover, we plan to investigate the problem of classifying the solutions based on the threshold relative to the best-performing solution's performance measure in future studies.

REFERENCES

Bryan, B., R. C. Nichol, C. R. Genovese, J. Schneider, C. J. Miller, and L. Wasserman. 2005. "Active Learning for Identifying Function Threshold Boundaries". *Advances in neural information processing systems* 18.

Corlu, C. G., and B. Biller. 2013. "A Subset Selection Procedure under Input Parameter Uncertainty". In *2013 Winter Simulations Conference (WSC)*, 463–473 https://doi.org/10.1109/WSC.2013.6721442.

Eckman, D. J., M. Plumlee, and B. L. Nelson. 2022. "Plausible Screening Using Functional Properties for Simulations with Large Solution Spaces". *Operations Research* 70(6):3473–3489.

Gotovos, A. 2013. "Active Learning for Level Set Estimation". Master's thesis, Eidgenössische Technische Hochschule Zürich, Department of Computer Science.

Gupta, S. S. 1965. "On Some Multiple Decision (Selection and Ranking) Rules". *Technometrics* 7(2):225–245.

Hsu, J. C. 1981. "Simultaneous Confidence Intervals for All Distances from the "Best"". *The Annals of Statistics* 9(5):1026–1034.

Iwazaki, S., Y. Inatsu, and I. Takeuchi. 2020. "Bayesian Experimental Design for Finding Reliable Level Set under Input Uncertainty". *IEEE Access* 8:203982–203993 https://doi.org/10.1109/ACCESS.2020.3036863.

Kim, T., K.-K. Kim, and E. Song. 2025. "Selection of the Most Probable Best". *Operations Research* 0(0) https://doi.org/10.1287/opre.2022.0343.

Song, E. 2021. "Sequential Bayesian Risk Set Inference for Robust Discrete Optimization via Simulation". *arXiv preprint arXiv:2101.07466*.

Song, E., and B. L. Nelson. 2019. "Input–output Uncertainty Comparisons for Discrete Optimization via Simulation". *Operations Research* 67(2):562–576.

Wu, D., Y. Wang, and E. Zhou. 2024. "Data–driven Ranking and Selection under Input Uncertainty". *Operations Research* 72(2):781–795.

Xie, J., P. I. Frazier, and S. E. Chick. 2016. "Bayesian Optimization via Simulation with Pairwise Sampling and Correlated Prior Beliefs". *Operations Research* 64(2):542–559.

Zanette, A., J. Zhang, and M. J. Kochenderfer. 2018. "Robust Super-Level Set Estimation Using Gaussian Processes". In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 276–291. Springer.

AUTHOR BIOGRAPHIES

DONGJOON LEE is a first-year Ph.D. student in the H. Milton Stewart School of Industrial and Systems Engineering at Georgia Institute of Technology. His research interests include simulation and Bayesian optimization. His email address is dlee3006@gatech.edu.

EUNHYE SONG is a Coca-Cola Foundation Early Career Professor and Assistant Professor in the H. Milton Stewart School of Industrial and Systems Engineering at Georgia Institute of Technology. Her research interests include simulation model validation/calibration, uncertainty and risk quantification, and simulation optimization. She has several past and current industry collaborations on manufacturing digital twins, energy sustainability, product portfolio optimization and more. Her email address is eunhye.song@isye.gatech.edu. Her website is http://eunhyesong.info.

A APPENDIX A

We provide the expressions for the vectors and scalars that appear in (18). First, the following vectors are all B-dimensional:

$$w_x := \left\{ \frac{e_{(x,\theta_b)}^{\top} V_t e_{(x',\theta')}}{\{v(x',\theta')/r + \sigma_t^2(x',\theta')\}^{1/2}} \right\}_{1\le b\le B}, \quad c_x := \left\{ \frac{1}{B} \frac{1}{\sigma_{t+1}(x,\theta_b)} \phi\left(-\delta_{\theta_b}\right) \right\}_{1\le b\le B},$$

$$d_x := \left\{ \frac{2}{B^2} \frac{1}{\sigma_{t+1}(x,\theta_b)} \phi\left(\delta_{\theta_b}\right) \sum_{i=1}^{B} \left[\Phi\left(\delta_{\theta_i}\right) - \Phi\left(\frac{\delta_{\theta_i} - \rho_{i,b}\delta_{\theta_b}}{(1-\rho_{i,b}^2)^{1/2}} \right) \right] \right\}_{1\le b\le B}, \quad g_x := c_x - \left(\frac{\beta_{t+1}}{4l_0} \right)^{1/2} d_x$$

where $\delta_{\theta_i} := \frac{\delta - \mu_t(x,\theta_i)}{\sigma_{t+1}(x,\theta_i)}$, $\rho_{i,b} := \frac{k_{t+1}(x,\theta_i;x,\theta_b)}{\sigma_{t+1}(x,\theta_i)\sigma_{t+1}(x,\theta_b)}$, and $l_0 := \frac{1}{B^2}\sum_{i=1}^{B}\sum_{j=1}^{B}\left[\Pr\left\{Z_i \le \delta_{\theta_i}, Z_j \le \delta_{\theta_j}\right\} - \Phi\left(\delta_{\theta_i}\right)\cdot\Phi\left(\delta_{\theta_j}\right)\right]$. Lastly, $h_x := \frac{1}{B}\sum_{b=1}^{B}\Phi\left(\frac{\mu_t(x,\theta_b)-\delta}{\sigma_{t+1}(x,\theta_b)}\right) - (\beta_{t+1}l_0)^{1/2}$.