Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

SIMULATION OPTIMIZATION WITH MULTIPLE ATTEMPTS

Jingjun Men
Zhihao Liu
Songhao Wang

SUSTech Business School
Southern University of Science and Technology
1088 Xueyuan Avenue
Shenzhen 518055, CHINA

Haowei Wang

Rice-Rick Digitalization PTE. Ltd.
51 Goldhill Plaza
Singapore 308900, SINGAPORE

ABSTRACT

Simulation optimization is a widely utilized approach that allows decision-makers to test various decision variable settings in simulators before implementing a final recommended action on the real systems. In some real-world scenarios, the recommended action can be executed multiple times and the performance is evaluated as the best one among these multiple attempts. In this paper, we introduce such simulation optimization problems with multiple attempts and provide insights into the problem through comparison to risk-averse decision-making problems. We propose a surrogate-assisted algorithm based on the Gaussian process model and the upper confidence bound criterion for efficiently solving such problems. We demonstrate the efficiency and effectiveness of the proposed approach with several numerical examples.

1 INTRODUCTION

Computer simulation is a powerful tool for the analysis of complex stochastic systems and simulation-based optimization provides an effective decision-making approach for these systems. A stochastic simulation optimization problem can be formulated as follows:

$$\max_{x \in D} \mathbb{E}_{\eta}[Y(x, \eta(x))], \tag{1}$$

where x denotes the decision variable within the design space D, $Y(\cdot)$ is the stochastic simulation output and $\eta(\cdot)$ represents the randomness in Y. In problem (1), the noise is heteroscedastic, i.e., the noise in the stochastic output can be different at different design points.

Simulation optimization is often used when the decision makers can try several different settings of decision variables in simulators and then deploy a final recommended action on the real systems. In this work, we consider a scenario where the action on the real system is conducted multiple times when deployed. The performance of this action is evaluated as the best one among these multiple attempts. Taking the best from multiple attempts is quite often in daily life, such as when students can take the same exam multiple times and their final grade is based on the highest score achieved, or when athletes are given multiple attempts in events like the high jump and long jump and their final achievement is determined by their best performance. In the design of biological experiments, each setting is often experimented several times for observations (Hurlbert 1984; Quinn and Keough 2002). If we use simulation optimization to select the design/control parameter for a bomber aircraft to destroy assets of the enemy, typically an aircraft will take multiple bombs, and a hit is scored if any bomb hits the target. In the aforementioned examples, the decision is evaluated by the best performance among multiple attempts. In this work, we consider the simulation optimization problem with multiple attempts and we assume that the number of attempts, m, is

979-8-3503-6966-3/23/$31.00 ©2023 IEEE

predefined. In this setting, the optimization problem can be formulated as:

$$\max_{x\in D}\mathbb{E}_{\eta}[\max\{Y(x,\eta_1(x)),Y(x,\eta_2(x)),\ldots,Y(x,\eta_m(x))\}], \tag{2}$$

where $Y(x,\eta_1(x)),\ldots,Y(x,\eta_m(x))$ are independent and identically distributed (i.i.d.) simulation outputs under variable x. This measure represents the expectation of the maximum of m i.i.d outputs with the same design x. To the best of our knowledge, problem (2) has not been studied in depth in simulation optimization.

If we define $\Psi(x)=(\eta_1(x),\ldots,\eta_m(x))$, which is the multivariate random term consisting of the m i.i.d. $\eta(x)$, and $H(x,\Psi(x))=\max\{Y(x,\eta_1(x)),Y(x,\eta_2(x)),\ldots,Y(x,\eta_m(x))\}$, problem (2) can be reformulated as:

$$\max_{x\in D}\mathbb{E}_{\Psi(x)}[H(x,\Psi(x))] \tag{3}$$

Here, $H(x)$ is a heteroscedastic stochastic response, and problem (2) is then reduced to a general stochastic optimization problem similar to (1). Therefore, we can adopt approaches for stochastic simulation optimization to solve (2). However, optimizing (2) directly will ignore the structural information of $H(x)$ by treating it as a pure black box. The function $H(\cdot)$, with the structural information, essentially is a grey-box function (Astudillo and Frazier 2021). In this work, we will propose a more efficient approach to solve (2). We also notice that problem (2) is related to the risk-averse decision-making. The decision makers take the recommended actions multiple times to reduce the risk that one attempt might miss the target, such as the case in the bomber aircraft example. However, through our analysis, we find that in problem (2), the decision makers are in favor of actions with large uncertainty, which is quite different from that in a traditional problem which optimizes a risk measure, although the decision makers in the two kinds of problems are all risk-averse. A detailed discussion is presented in Section 3.

Compared with traditional stochastic simulation problem (1), problem (2) is more challenging as the expectation of the maximum over multiple attempts is difficult to estimate, which will take more simulation runs compared with estimating the expectation of a single simulation output. To tackle this challenge, we develop a surrogate-based simulation optimization approach. A surrogate model summarizes the information from the existing simulation results through a statistical model. It can be used to guide the future optimization process more efficiently. In this work, we adopt the Gaussian process (GP) model as the surrogate, which provides a built-in estimate of the uncertainty in the surrogate model. The GP model has been successfully used in the well-recognized black-box optimization procedure, Bayesian optimization (BO), together with several optimal points searching criteria. Popular searching criteria, also termed acquisition functions, include expected improvement (EI) (Jones et al. 1998), upper confidence bound (UCB) (Srinivas et al. 2012) and knowledge gradient (KG) (Frazier et al. 2008). BO has seen extensive applications in the area of operations research, simulation optimization, and computer science (see Shahriari et al. (2015) and Frazier (2018) for reviews). To apply BO in stochastic simulation optimizations, a generalized GP model for heteroscedastic noise cases, the stochastic kriging (SK) model, is proposed (Ankenman et al. 2010; Yin et al. 2011). In this work, we utilize the SK model with UCB acquisition function in our algorithm.

The remainder of this paper is organized as follows: Section 2 briefly introduces the necessary background on GP Model and GP-UCB algorithm. Section 3 reformulates the simulation optimization problem with multiple attempts. Section 4 presents the UCB-based algorithm for solving such problems. Section 5 details two numerical experiments. Finally, Section 6 concludes the paper and discusses future research directions.

2 A REVIEW OF STOCHASTIC GP MODEL AND GP-UCB ALGORITHM

In this section, we first provide a brief review of GP model in Section 2.1, which is used as the surrogate model in our proposed algorithm. In Section 2.2, we review the GP-UCB algorithm. The traditional GP-UCB algorithm is primarily designed to tackle online problems, thereby guaranteeing a sub-linear cumulative regret rather than directly addressing simple regret. In this paper, we utilize the UCB principle to solve the simulation optimization problems with multiple attempts.

2.1 GP Model Basics in the Stochastic Simulation

The response of the stochastic simulation can be represented by the following form in the standard stochastic GP model:

$$Y(x,\eta(x)) = F(x) + \xi(x) = M(x)^T\beta + Z(x) + \xi(x), \tag{4}$$

where $F(x) = \mathbb{E}_\eta[Y(x,\eta)]$ represents the randomness-free response which can be further decomposed into the mean function, $M(x)$, and a sample from a second-order stationary GP, $Z(x)$. $M(x)$ is a vector of known functions representing our belief of the mean performance of $F(x)$. A GP is a collection of dependent random variables, every finite subset of which is multivariate Gaussian distributed in an overall consistent way (Williams and Rasmussen 2006). A GP distribution $Z(\cdot) \sim \mathscr{GP}(\mu^Z(\cdot), k^Z(\cdot,\cdot))$ is characterized by its mean function $\mu^Z(x) = \mathbb{E}[Z(x)]$ and covariance function $k^Z(x,x') = \text{cov}(Z(x), Z(x'))$. A popular choice of covariance function is $k^Z(x,x') = \exp\left(-\left(2l^2\right)^{-1}\|x-x'\|^2\right)$ with l as the length-scale parameter. Without loss of generality, we assume little prior knowledge about $M(x)$ and choose a zero mean of GP with $M(x) = 0$, which is a common practice in GP-based optimization problems. In (4), $\xi(x)$ is the randomness brought by η. In the heteroscedastic noise case, it is assumed to be Gaussian distributed with mean zero and variance $\tau^2(x)$.

The major advantage of working with GPs is the existing analytical formula for the mean and covariance function of the posterior distribution. Denote x_i, $i = 1,\ldots s_t$ as the selected design variables up to iteration t, with $n_i(t)$ number of observations. Denote $\bar{\mathbf{y}}_t = \left(\bar{y}_t(x_1),\ldots,\bar{y}_t(x_{s_t})\right)^\top$ as the observations vector at the design points where $\bar{y}_t(x_i) = \frac{1}{n_i(t)}\sum_{j=1}^n y(x_i,\eta_j(x_i))$. Then, the posterior distribution of Z is again a GP with mean function $\mu_t(\cdot)$ and covariance function $k_t(\cdot,\cdot)$:

$$\begin{aligned} \mu_t^Z(x) &= \mathbf{k}_t^Z(x)^\top \left(\mathbf{K}_t^Z + \text{diag}\left\{\tau^2(x_1)/n_1(t),\ldots,\tau^2(x_{s_t})/n_{s_t}(t)\right\}\right)^{-1}\bar{\mathbf{y}}_t \\ k_t^Z(x,x') &= k^Z(x,x') - \mathbf{k}_t^Z(x)^\top \left(\mathbf{K}_t^Z + \text{diag}\left\{\tau^2(x_1)/n_1(t),\ldots,\tau^2(x_{s_t})/n_{s_t}(t)\right\}\right)^{-1}\mathbf{k}_t(x'), \\ (\sigma_t^Z(x))^2 &= k_t^Z(x,x) \end{aligned} \tag{5}$$

where $\mathbf{k}_t^Z(x) = \left[k^Z(x_1,x),\ldots,k^Z(x_{s_t},x)\right]^\top$, $\mathbf{K}_t^Z$ is the positive definite kernel matrix $\left[k(x_i,x_j)\right]_{i,j=1,\ldots,t}$.

In practical applications, the model inputs consist of the design set $D = \{x_1,\ldots,x_{s_t}\}$, the observation vector $\bar{\mathbf{y}}_t$ and the noise variance matrix $\text{diag}\left\{\tau^2(x_1),\ldots,\tau^2(x_t)\right\}$. Typically, $\tau^2(x_i)$ is estimated using the sample variance from the simulation outputs $y(x_i,\eta_1(x_i)),\ldots,y(x_i,\eta_{n_i(t)}(x_i))$. The hyperparameter l can then be obtained by maximizing the likelihood of $\bar{\mathbf{y}}_t$.

2.2 GP-UCB Algorithm

The Upper-Confidence-Bound (UCB) algorithm is commonly used in reinforcement learning to tackle the exploration-exploitation dilemma. Srinivas et al. (2012) introduced the UCB algorithm to GP-based optimization problems with their proposed GP-UCB algorithm. It efficiently solves the problem of sequentially optimizing an unknown reward function $Z: D \to \mathbb{R}$, where $Z(\cdot)$ is a sample from some GP, denoted as $\mathscr{GP}(0, k^Z(\cdot,\cdot))$ or lies in some reproducing kernel Hilbert space (Wahba 1990). At each iteration $t = 1,\ldots,T$, the decision maker selects a point $x_t \in D$ and receives a realized reward disturbed by a Gaussian white noise with known variance: $y_t = Z(x) + \varepsilon_t$, where $\varepsilon_t \sim N(0,\tau^2)$. The goal is to perform as well as possible, effectively achieving $x^* = \arg\max_{x\in D} Z(x)$. In this context, the natural performance metric is the cumulative regret, which measures the loss in reward due to not knowing x^* beforehand:

$$R_T = \sum_{t=1}^T r_t = \sum_{t=1}^T \left(Z(x^*) - Z(x_t)\right), \tag{6}$$

where r_t is the instantaneous regret incurred at iteration t.

Since the exact value of $Z(x_t)$ is not observed, the decision maker faces an exploration-exploitation trade-off. They can choose the point with the maximum estimated $Z(x)$ based on all available information up to iteration t (exploitation), or select a point with large uncertainty in less explored regions for better solutions than the current best (exploration). The UCB-based algorithms follow the principle of optimism in the face of uncertainty, assuming that the decision x_t will perform as well as possible within a reasonable range when there is uncertainty about it. Suppose at iteration t, a confidence bound of $Z(x)$ is obtained for any $x \in D$, denoted as $[\text{LCB}_t(x_t), \text{UCB}_t(x)]$. The GP-UCB algorithm suggests selecting $x_t = \arg\max_{x\in D} \text{UCB}_t(x)$.

If these confidence bounds hold jointly across all $x \in D$ and all $t = 1, \dots, T$ with probability no smaller than $1-\delta$ for some $\delta \in (0,1)$, i.e.,

$$P\{Z(x) \in [\text{LCB}_t^Z(x), \text{UCB}_t^Z(x)], \forall x \in D \text{ and } \forall t = 1, \dots, T\} \leq 1-\delta. \tag{7}$$

Then with the same probability confidence level, we can bound the cumulative regret by the summation of the length of the confidence bounds:

$$R_T = \sum_{t=1}^{T} (Z(x^*) - Z(x_t)) \leq \sum_{t=1}^{T} \left(\text{UCB}_t^Z(x^*) - Z(x_t)\right) \leq \sum_{t=1}^{T} \left(\text{UCB}_t^Z(x_t) - \text{LCB}_t^Z(x_t)\right), \tag{8}$$

where the second inequality follows from the UCB principle.

In the GP-UCB algorithm, the confidence bound takes the form:

$$\begin{aligned} \text{LCB}_t^Z &= \mu_{t-1}^Z(x) - \beta_t^Z \sigma_{t-1}^Z(x,x) \\ \text{UCB}_t^Z &= \mu_{t-1}^Z(x) + \beta_t^Z \sigma_{t-1}^Z(x,x), \end{aligned} \tag{9}$$

for some parameter $\beta_t > 0$, where $\mu_{t-1}(x)$ and $\sigma_{t-1}^2(x) = k_{t-1}(x,x)$ is the mean and variance of the posterior distribution of Z conditional on $\{y(x_1), \dots, y(x_{t-1})\}$ according to Equation (5). By appropriately choosing β_t, one can bound the cumulative regret using the quantity maximal information gain, which depends on the choice of $k(\cdot,\cdot)$. Notably, it is found that the cumulative regret exhibits sublinear growth for a broad range of kernel functions.

3 PROBLEM REFORMULATION & MODEL INSIGHTS

In this section, we reformulate the problem and provide an in-depth analysis of the problem from the reformulation.

We consider the proposed optimization problem (2) in Section 1 where m is a predetermined number of attempts, and $Y(x,\eta_1(x)), \dots, Y(x,\eta_m(x))$ are independent and identically distributed Gaussian simulation outputs for input $x \in D$. We assume that $D \subset [0,1]^d$ is a compact set for some constant $d \in \mathbb{N}+$. We further define $Z(x) = \mathbb{E}[Y(x,\eta(x))]$ as the expected value of the simulation output and $\xi(x) = Y(x,\eta(x)) - Z(x)$ is a normal noise, which possesses an expected value of zero and variance of $\tau^2(x)$. As mentioned in Section 1, we denote $H(x,\Psi(x)) = \max\{Y(x,\eta_1), Y(x,\eta_2), \dots, Y(x,\eta_m)\}$ and $G(x) = \mathbb{E}_{\Psi(x)}[H(x,\Psi(x))]$. Define $x^* = \max_{x\in D} G(x)$. Thus, (2) is equivalent to $\max_{x\in D} G(x)$.

We introduce a constant

$$E_m = \mathbb{E}[\max\{W_1, W_2, \dots, W_m\}] \tag{10}$$

where $W_1, \dots, W_m$ are independent and identically distributed standard Gaussian random variables. Utilizing this constant, we can express $G(x)$ as follows:

$$G(x) = \mathbb{E}[\max\{Y(x,\eta_1), Y(x,\eta_2), \dots, Y(x,\eta_m)\}] = Z(x) + \tau(x) E_m. \tag{11}$$

Here, $G(x)$ can be partitioned into the expectation of $Y(x)$ and the product of its standard deviation with E_m. Thus, the problem (2) can be reformulated into

$$\max_{x\in D} Z(x) + \tau(x) E_m. \tag{12}$$

Forms analogous to (12) are frequently employed in risk management (Chiu and Choi 2016) when a mean-variance risk measure is applied. As mentioned in Section 1, in risk management problems such as portfolio selection (Simaan 1997) and supply chain inventory (Lau 1980), the risk-averse decision-maker aims to maximize $Z(x) - c \cdot \tau(x)$, where $Z(x)$ represents the expected profit and $\tau(x)$ denotes the standard deviation of response at x. In this context, c is a constant that measures the decision-maker's risk tolerance, with a positive value indicating a risk-averse attitude. Hence, decision-makers prefer decisions with smaller uncertainty.

Considering the proposed multi-attempt problem (12), E_m is a positive constant that is parallel to the risk tolerance parameter c in traditional risk management problems. It seems that (2) is reduced to a risk-seeking problem when searching for the optimal solution. However, the proposed problem has the following two main differences from the risk-seeking problem. First, the decision makers here are essentially risk-averse, so they try multiple times to reduce the risk that one attempt of the decision has inferior performance, as illustrated by the bombing mission example in the introduction where several bombs are dropped simultaneously to decrease the failure rate. Compared with the traditional risk-averse problem, the problem we investigate addresses risk aversion from distinct perspectives. In the conventional risk management problem, the decision-maker has a single attempt and, being risk-averse, they tend to select decisions with a lower variance while ensuring a certain expected return $Z(x)$. In contrast, the problem we explore reduces the risk of a decision through multiple attempts with additional resources. It facilitates the decision maker to explore decisions with large variability that can reach much better performance compared with conservative decisions. The second difference is that the value of E_m in our problem is determined in a more systematic manner. Once the number of attempts is given under budget or resources constraint, E_m can be decided as in (10).

We further illustrate the difference between our problem and the traditional risk-neutral and risk-averse problem with the example in Figure 1. Here, the Y-axis represents the profit function. For any x, the profit $Y(x)$ is random. The solid red curve illustrates the mean profit Y, i.e. $Z(x)$, for each decision x, while the upper and lower reddish outlines depict the mean plus and minus the standard deviation of Y at x multiplied by the constant $E_m > 0$. From a risk-neutral decision-maker's perspective, x_1 should be the optimal solution with the maximum mean. Meanwhile, x_2 is the optimal solution for a risk-averse decision-maker whose risk tolerance parameter exactly equals E_m. In our setting, since we have multiple attempts, x_3 should be considered optimal.

4 MULTI-ATTEMPT UCB ALGORITHM

Our aim is to solve (12) sequentially. For each iteration $t = 1, \dots, T$, we select a point $x_t \in D$ and perform n simulations to acquire i.i.d. outputs $y_1(x_t) := y(x_t, \eta_1(x_t)), \dots, y_n(x_t) := y(x_t, \eta_n(x_t))$. Running several simulation replications at each design point is a common practice for stochastic BO algorithms to estimate the responses more accurately (See a review in Jalali et al. (2017)).

In this section, we present an Upper-Confidence-Bound (UCB) based algorithm called Multi-Attempt UCB for the problem with multiple attempts. We can observe from Equation (12), the objective function contains not only the expectation of simulation output but its standard deviation. To construct the BO algorithm for solving (10), we thus need an additional GP model for the standard deviation $\tau(x)$. However, we notice that $\tau(x)$ is always non-negative while the conventional GP model is for the response on the real line. Hence, we first develop a proper surrogate model for $\tau(x)$ in Section 4.1 and propose the algorithm in Section 4.2.

4.1 Surrogate Model for $\tau(x)$

Since $\tau(x)$ is always positive, the conventional GP model is unsuitable for directly modeling $\tau(x)$. Especially when $\tau(x)$ is small, the posterior distribution may provide a negative estimation for $\tau(x)$, which brings difficulty in both the model interpretability and algorithm construction. Thus, to introduce the GP model

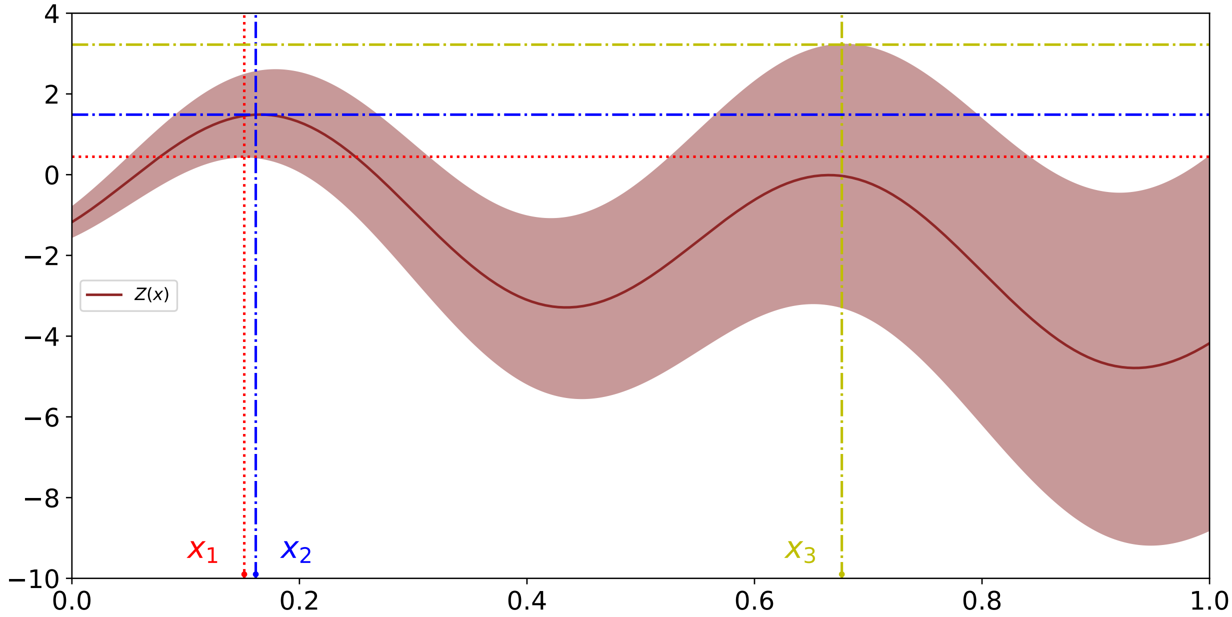


Figure 1: Illustration of the difference among risk-neutral, risk-averse ,and the proposed multiple-attempt decision-making.

to estimate $\tau(x)$, a transformation of $\tau(x)$ should be taken so that it covers the whole real line. Several research efforts have attempted to place a GP on the transformation of $\tau(x)$. Ankenman et al. (2010) place a GP distribution on the difference between variance and a constant; however, this approach has a weakness as the variance estimation may still be negative. Ng and Yin (2012) place a GP distribution on the logarithm of the variance and the obtained GP prediction can be transferred back for predicting the variance, ensuring a positive estimation. However, in the inverse transformation of the logarithm, the exponential function may magnify the uncertainty of the prediction, leading to an exponentially growing confidence interval for the variance prediction. As shown in equation (8), the length of the confidence interval is critical for bounding the regret and we find that modeling the logarithm of the variance is not adequate for the algorithm to converge.

In this work, we propose a more proper way to model the standard deviation through the logit transformation. Denote $v(s) = \log\left(\frac{s}{M-s}\right)$ as the logit transformation for $s \in (0, M)$ and $\pi(s) = \frac{M}{1+e^{-s}}$ to be its inverse transformation where M is a positive constant. In this paper, assume the standard deviation is in (0, M) and use logit transform to obtain $l(x) := v(\tau(x))$, which is in $(-\infty, \infty)$. This assumption essentially requires the standard deviation of the simulation output to be bounded in the input domain. In practice, if the exact upper bound is not known, a fairly large number can be used in the transformation. We next build a GP model for l(x). Denote $l_S(x) = v(S(x))$ for all x with observed sample variance $S^2(x)$. Thus, $l_S(x)$ is a noisy observation of $l(x)$ at the selected points x_t. We use a simple nugget-effect GP model (Yin et al. 2011) for $l(x)$. Similarly, we can also update the posterior distribution over $l(x)$, with posterior mean μ_t^l, covariance k_t^l and variance σ_t^l:

$$\begin{aligned} \mu_t^l(x) &= \mathbf{k}_t^l(x)^\top \left(\mathbf{K}_t^l + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{l}_t \\ k_t^l\left(x, x'\right) &= k^l\left(x, x'\right) - \mathbf{k}_t^l(x)^\top \left(\mathbf{K}_t^l + \sigma^2 \mathbf{I}\right)^{-1} k_t^l\left(x'\right) \\ (\sigma_t^l)^2(x) &= k_t^l(x, x), \end{aligned} \tag{13}$$

where σ^2 is the noise variance in the nugget-effect GP model, $\mathbf{k}_t^l(x) = \left[k^l(x_1,x)\dots k^l(x_{s_t},x)\right]^\top$, $\mathbf{K}_t^l$ is the positive definite kernel matrix $\left[k^l(x_i,x_j)\right]_{i,j=1,\dots,s_t}$ and $\mathbf{l}_t = \{l(x_1),\dots,l(x_{s_t})\}^\top$. The parameters in this model can be estimated through MLE (see Chapter 2 in Williams and Rasmussen (2006)).

4.2 Upper Confidence Bound for $G(x)$

The Upper-confidence-Bound principle suggests $x_t = \arg\max_{x\in D}\mathrm{UCB}_t(x)$, where $\mathrm{UCB}_t(x)$ is some upper confidence bound of $G(x)$ at iteration t. Recall $G(x) = Z(x) + \tau(x)\cdot E_m = Z(x) + \pi(l(x))\cdot E_m$. If we derive the confidence bound for $Z(\cdot)$ and $l(\cdot)$ respectively, denoted as $[\mathrm{LCB}_t^Z(x), \mathrm{UCB}_t^Z(x)]$ and $[\mathrm{LCB}_t^l(x), \mathrm{UCB}_t^l(x)]$, at some level $1-\alpha/2$, using the union bound, we can construct the regret bound at level $1-\alpha$ for $G(x)$ as

$$\begin{aligned}\mathrm{LCB}_t(x) &= \mathrm{LCB}_t^Z(x) + E_m\pi(\mathrm{LCB}_t^l(x))\\ \mathrm{UCB}_t(x) &= \mathrm{UCB}_t^Z(x) + E_m\pi(\mathrm{UCB}_t^l(x)).\end{aligned} \tag{14}$$

The confidence bound for $Z(\cdot)$ and $l(\cdot)$ can be constructed in the following way:

$$\begin{aligned}\mathrm{LCB}_t^Z(x) &= \mu_{t-1}^Z(x) - \beta_t^{1/2}\sigma_{t-1}^Z(x)\\ \mathrm{UCB}_t^Z(x) &= \mu_{t-1}^Z(x) + \beta_t^{1/2}\sigma_{t-1}^Z(x)\end{aligned} \tag{15}$$

and

$$\begin{aligned}\mathrm{LCB}_t^l(x) &= \mu_{t-1}^l(x) - \beta_t^{1/2}\sigma_{t-1}^l(x)\\ \mathrm{UCB}_t^l(x) &= \mu_{t-1}^l(x) + \beta_t^{1/2}\sigma_{t-1}^l(x)\end{aligned} \tag{16}$$

for some parameter $\beta_t > 0$. Here, when D is finite, we recommend $\beta_t = 2\log\left(|D|t^2\pi^2/3\delta\right)$ to ensure a sublinear regret with high probability no smaller than $1-\delta$. The value of β_t can be set by referring to Srinivas et al. (2012).

Pseudocode for the Multi-Attempt UCB algorithm is provided in Algorithm 1.

Algorithm 1 Multi-Attempt UCB Algorithm

```
Fit GP for $Z(\cdot)$ and $l(\cdot)$ respectively.
**Input:** Design space $D$, GP prior for $Z(\cdot)$: $\mu_0^Z = 0, k^Z(\cdot,\cdot)$, GP prior for $l(\cdot)$: $\mu_0^l = 0, k^l(\cdot,\cdot)$
for $t \leftarrow 1,2,\dots,T$ do
    Select
        $x_t = \arg\max_{x\in D} \mathrm{UCB}_t^Z(x) + \pi(\mathrm{UCB}_t^l(x))\cdot E_m.$    (17)
    Obtain $n$ evaluations $y_i(x_t) = Z(x_t) + \eta_i(x_t), i = 1,\dots,n$.
    Update the sample mean $\bar{\mathbf{y}}_t$ and the sample variance $\mathbf{S}_t^2$.
    Perform Bayesian updates to obtain $\mu_t^Z$, $\sigma_t^Z$ and $\mu_t^l$, $\sigma_t^l$ .
end for
```

5 NUMERICAL EXPERIMENTS

In this section, we aim to demonstrate the efficiency of the Multi-Attempt UCB algorithm in addressing the optimization problem defined in Equation (12) by employing a one-dimensional (1D) test function and a two-dimensional (2D) test function.

Benchmark Method. We aim to introduce an intuitive benchmark method first. Recall that Equation (3) presents an equivalent formulation of the optimization problem in Equation (12). Given that $H(x,\Psi(x))$ represents the maximum value of $Y(x,\eta_1(x)),\dots,Y(x,\eta_m(x))$, it is necessary to have m i.i.d. evaluations of $Y(x,\eta(x))$ to acquire a single evaluation of $H(x,\Psi(x))$. The benchmark approach entails placing a GP distribution on $G(x)=E_{\Psi(x)}[H(x,\Psi(x))]$ and implementing the GP-UCB algorithm to maximize $G(x)$ over the domain $x\in$. This benchmark algorithm essentially treats $G(x)$ as a black-box function and directly applies the conventional GP-UCB algorithm. The framework for this benchmark approach is outlined as follows:

Algorithm 2 Benchmark: GP-UCB algorithm

```
Fit GP for $G$.
**Input:** Design space $D$, GP prior for $G(\cdot)$: $\mu^G=0$, $k^G(\cdot,\cdot)$.
**for** $t\leftarrow 1,2,\dots,T$ **do**
    Select
        $x_t=\arg\max_{x\in D}\mathrm{UCB}_t^G(x)=\mu_t^G(x)+\beta_t^{1/2}\sigma_t^G(x).$        (18)
    Obtain $n$ evaluations of $H(x_t,\Psi(x))$, denoted as $h_i(x_t)$, $i=1,\dots,n$. (To obtain one evaluation of $H$ at $x$, we need $m$ evaluations of $Y$ at $x$ and take their maximum.)
    Update the sample mean $\bar{\mathbf{h}}_t$ and the sample variance $\mathbf{S}^2_{H,t}$.
    Perform Bayesian updates to obtain $\mu_t^G$, $\sigma_t^G$.
**end for**
```

Test function. Initially, we illustrate the method's performance on a 1D function with the design space from $[0,1]$. Here, we define $Z(x)=2\sin(6\pi x)-3x$ and $\tau(x)=0.2+3x$. We consider $m=5$ and 8 respectively to compare the performance of Multi-Attempt UCB and the benchmark in different settings. Figure 2 illustrates $Z(x)$ and $G(x)$ for $m=5,8$. It is evident that the maximum values of $Z(x)$ and $G(x)$ occur in distinct regions.

Then we use a commonly-used 2D test function, the Ackley function, to further illustrate the advances of our proposed algorithm. Specifically, we employ a combination of two Ackley functions to construct a more complex objective function. A 2D Ackley function is defined as $\mathrm{ackley}(\mathbf{x})=-a\exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{2}x_i^2}\right)-\exp\left(\frac{1}{2}\sum_{i=1}^{2}\cos(cx_i)\right)+a+\exp(1)$. Since our objective is to solve a maximization problem, we use the negation of the Ackley function. Specifically, we set $Z(x)=-\frac{1}{2}\left[\mathrm{ackley}(x)+\mathrm{ackley}(x-(4,4))\right]$ with $a=20, b=0.2, c=2\pi$. We define $\tau(x)=0.02(|x_1-2|+|x_2-2|)+0.2(|x_1|+|x_2|)$. The design space is from $[-10,10]^2$. Figure 3 displays $Z(x)$ and $G(x)$ for $m=5,8$ respectively.

Experimental setup. During each iteration t, a design point x_t is queried and n evaluations, denoted as $\{y_i(x_t)\}_{i=1}^n$ for Multi-Attempt UCB and $\{h_i(x_t)\}_{i=1}^n$ for the benchmark method, are obtained. The cost of acquiring n evaluations of H for the benchmark is equivalent to the cost of obtaining $m\cdot n$ evaluations of Y. For both examples, we set $n=15$. To maintain the same evaluation cost for both algorithms, the total number of iterations T_{Multi} for Multi-Attempt UCB should be m times greater than the total number of iterations $T_{\mathrm{benchmark}}$ for the benchmark. Additionally, the number of initial design points for the Multi-Attempt UCB algorithm should also be m times greater than that of the benchmark. We employ Latin Hypercube Design to select the initial design points for both algorithms.

For the 1D test function, we set the total number of function evaluations of Y to be 1200, and for the 2D test function, we set it to 3000. We utilize the Matérn covariance function with $\nu=\frac{5}{2}$ to fit GP models due to its flexibility to cover different types of problems (See Chapter 4 in (Williams and Rasmussen 2006)). We conduct 50 macro-replications for both example functions and both algorithms.

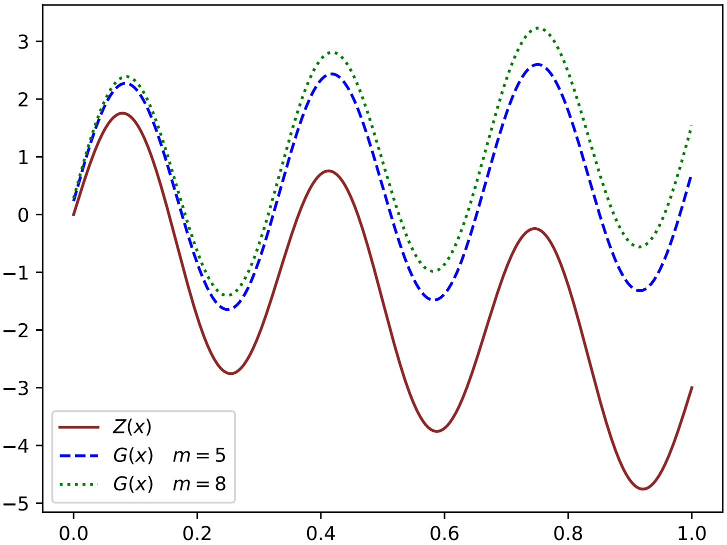


Figure 2: The 1D test function.

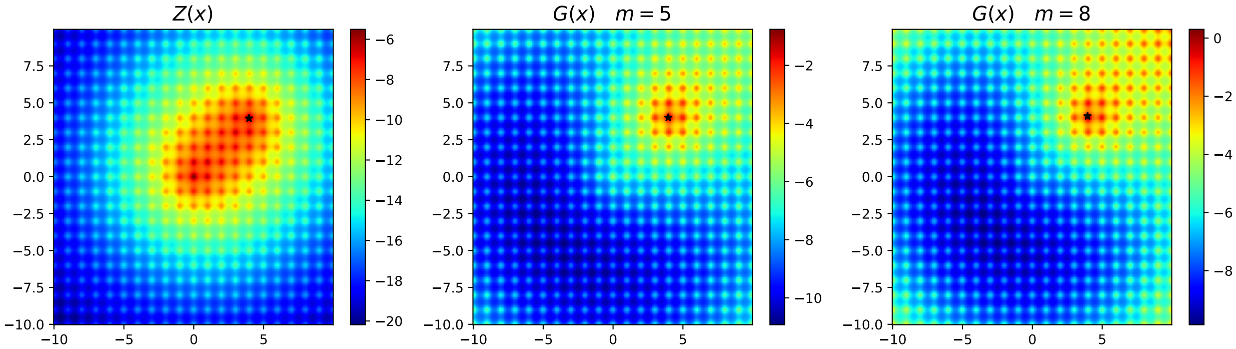
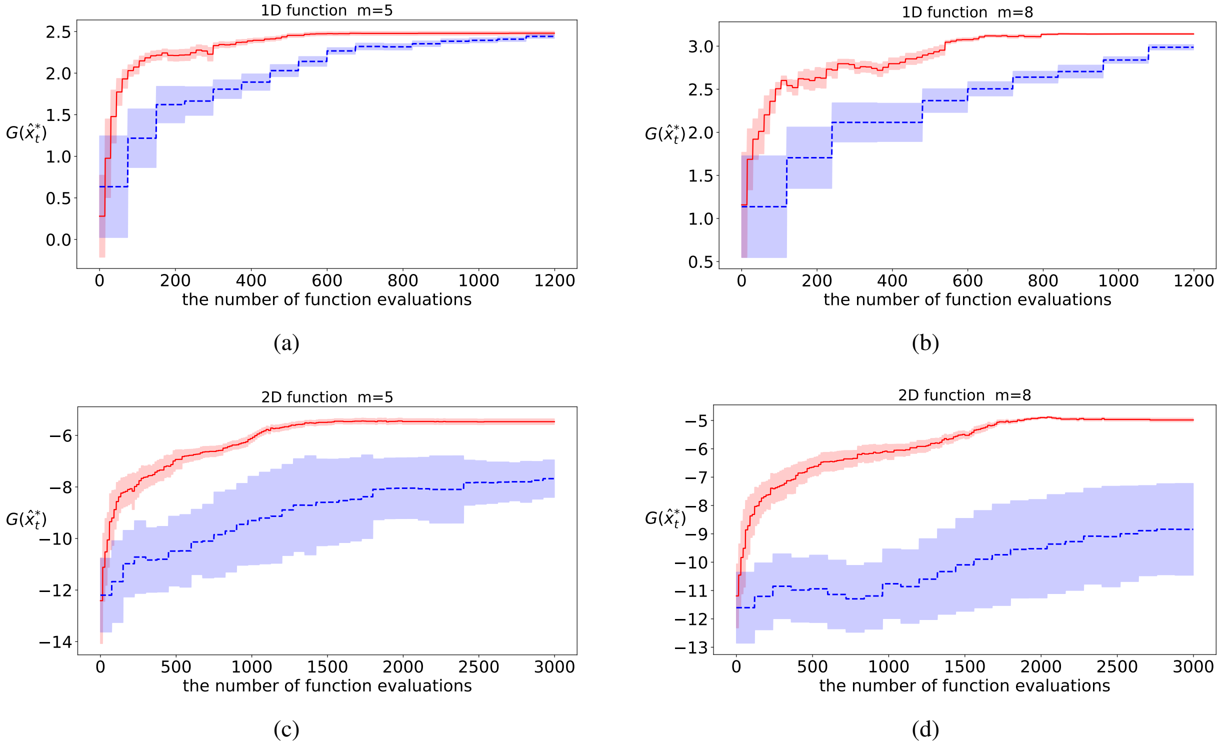


Figure 3: The 2D test function: the global optimal points are marked with black stars.

Performance. The performances for both algorithms over the two test functions are illustrated in Figure 4. The horizontal axis indicates the number of function evaluations for Y, and the vertical axis shows the average value of $G(\hat{x}_t^*)$ over the 50 macro-replications with the shaded region representing the confidence intervals. For each macro-replication, $\hat{x}_t^*$ represents the selected point exhibiting the highest empirical value of $G(x)$ up to iteration t, i.e., the best point found by the algorithm so far. As $G(\hat{x}_t^*)$ is updated only at the end of each iteration and multiple evaluations of Y are conducted in each iteration, the performance of $G(\hat{x}_t^*)$ over the number of evaluations looks like a step function. Notice that each replication of Multi-Attempt UCB costs n evaluations of Y while that of the benchmark algorithm costs mn evaluations of Y (to obtain n observations of H). Hence, the step length of the benchmark is also m times greater than that of Multi-Attempt UCB.

We see from Figure 4 that in all the test scenarios, the target function value identified by the Multi-Attempt UCB algorithm surpasses that of the benchmark method. We also observe that the performance curve of Multi-Attempt UCB exhibits a notably steeper ascent compared to the benchmark, particularly when the number of function evaluations is limited. The shaded region demonstrates the enhanced stability of the Multi-Attempt UCB algorithm compared to the benchmark. If we compare the results of 4(a) and 4(b) for the 1D function, we can see that as m increases, the differences between the two algorithms get larger. When m is large, we need more replications of Y to obtain one observation of H, and thus treating

H as a pure black box, as the benchmark algorithm does, becomes less efficient. Similar observations can be obtained when comparing the two 2D cases. Consequently, in scenarios with high evaluation costs and a relatively small number of function evaluations, the proposed Multi-Attempt UCB algorithm can achieve better results than the vanilla GP-UCB algorithm. The efficiency of our algorithm is due to our reformulation (12) of the original multi-attempt problem (2), which allows us to transform a black-box problem (3) into a grey-box problem with the structure information to facilitate the optimization.



(a)

(b)

(c)

(d)

Figure 4: The performance comparison between Multi-Attempt UCB and the benchmark algorithm.

6 DISCUSSION AND FUTURE EXTENSIONS

In this paper, we first propose the multi-attempt simulation optimization problem (2). We then reformulate the problem under the assumption that the simulation outputs are normally distributed and decompose the term inside the expectation from (2) into its expectation $Z(x)$ and the randomness term $\xi(x)$. Subsequently, we develop a GP-based surrogate model to handle $Z(\cdot)$ and $\xi(\cdot)$ separately and propose the Multi-Attempt UCB algorithm for efficiently solving such optimization problems.

Several future research directions can be considered. First, a theoretical analysis of the Multi-Attempt UCB algorithm's performance, including aspects such as cumulative regret, is worth exploring. Second, our current setting focuses on the scenario when one recommended action is conducted multiple times on real systems. A more challenging case is when *m* different actions can be selected. Investigating the performance of simulations with multiple designs merits further exploration. Moreover, the current reformulation is for Gaussian distributed stochastic outputs. Examining non-Gaussian distributed stochastic outputs would be of interest. Finally, in this paper, we adopt expectation in (2) as the performance measure. We can also consider replacing it with a risk measure for more risk-averse decision-makers.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation (NNSF) of China under Grant 72101106 and was supported in part by the Shenzhen Science and Technology Program under Grant RCBS20210609103119020.

REFERENCES

Ankenman, B., B. L. Nelson, and J. Staum. 2010. "Stochastic Kriging for Simulation Metamodeling". *Operations Research* 58(2):371–382.

Astudillo, R., and P. I. Frazier. 2021. "Thinking Inside the Box: A Tutorial on Grey-Box Bayesian Optimization". In *Proceedings of the 2021 Winter Simulation Conference*, edited by S. Kim, B. Feng, K. Smith, S. Masoud, and Z. Zheng, 1–15. Phoenix, AZ: Institute of Electrical and Electronics Engineers, Inc.

Chiu, C.-H., and T.-M. Choi. 2016. "Supply Chain Risk Analysis with Mean-Variance Models: A Technical Review". *Annals of Operations Research* 240(2):489–507.

Frazier, P. I. 2018. "A Tutorial on Bayesian Optimization". *arXiv preprint arXiv:1807.02811*. https://arxiv.org/abs/1807.02811, accessed 5th August.

Frazier, P. I., W. B. Powell, and S. Dayanik. 2008. "A Knowledge-Gradient Policy for Sequential Information Collection". *SIAM Journal on Control and Optimization* 47(5):2410–2439.

Hurlbert, S. H. 1984. "Pseudoreplication and the Design of Ecological Field Experiments". *Ecological monographs* 54(2):187–211.

Jalali, H., I. Van Nieuwenhuyse, and V. Picheny. 2017. "Comparison of Kriging-Based Algorithms for Simulation Optimization with Heterogeneous Noise". *European Journal of Operational Research* 261(1):279–301.

Jones, D. R., M. Schonlau, and W. J. Welch. 1998. "Efficient Global Optimization of Expensive Black-Box Functions". *Journal of Global optimization* 13(4):455.

Lau, H.-S. 1980. "The Newsboy Problem under Alternative Optimization Objectives". *Journal of the Operational Research Society* 31(6):525–535.

Ng, S. H., and J. Yin. 2012. "Bayesian kriging Analysis and Design for Stochastic Simulations". *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 22(3):1–26.

Quinn, G. P., and M. J. Keough. 2002. *Experimental Design and Data Analysis for Biologists*. Cambridge university press.

Shahriari, B., K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas. 2015. "Taking the Human Out of the Loop: A Review of Bayesian Optimization". *Proceedings of the IEEE* 104(1):148–175.

Simaan, Y. 1997. "Estimation Risk in Portfolio Selection: The Mean Variance Model versus the Mean Absolute Deviation Model". *Management science* 43(10):1437–1446.

Srinivas, N., A. Krause, S. M. Kakade, and M. W. Seeger. 2012. "Information-theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting". *IEEE transactions on information theory* 58(5):3250–3265.

Wahba, G. 1990. *Spline Models for Observational Data*. SIAM.

Williams, C. K., and C. E. Rasmussen. 2006. *Gaussian Processes for Machine Learning*, Volume 2. MIT press Cambridge, MA.

Yin, J., S. Ng, and K. Ng. 2011. "Kriging Metamodel with Modified Nugget-Effect: The Heteroscedastic Variance Case". *Computers & Industrial Engineering* 61(3):760–777.

AUTHOR BIOGRAPHIES

JINGJUN MEN is a research assistant in the School of Business, Southern University of Science and Technology. He received his B.Sci. degree in Statistics from University of Science and Technology of China. His research interests include simulation optimization and robust optimization. His email address is 0219amen@gmail.com.

ZHIHAO LIU is a research assistant in the School of Business, Southern University of Science and Technology. He received his B.Eng. degree in Computer Science from Nanchang University and his M.Eng. in Material Engineering from Southern University of Science and Technology. His research interests include Bayesian optimization and machine learning. His email address is 11930201@mail.sustech.edu.cn.

HAOWEI WANG is currently a research scientist at Rice-Rick Digitalization PTE. Ltd. Before joining Rice-Rick, he received the B.Eng. degree in industrial engineering from Nanjing University in 2016 and the Ph.D. degree in industrial and systems engineering from National University of Singapore(NUS) in 2021. His research interest includes simulation optimization, Bayesian optimization under uncertainties. His email is haowei_wang@ricerick.com.

SONGHAO WANG is an assistant professor in the School of Business, Southern University of Science and Technology. He received his B.Sci. degree in mechanical engineering from University of Science and Technology of China in 2015 and his Ph.D. degree in industrial and systems engineering from NUS in 2020. His research interests include simulation optimization, statistical modeling and machine learning. His email address is wangsh2021@sustech.edu.cn.