IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 21, 2021, accepted June 30, 2021, date of publication July 8, 2021, date of current version July 21, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3095755

Multi-Armed Bandit Regularized Expected Improvement for Efficient Global Optimization of Expensive Computer Experiments With Low Noise

RAJITHA MEKA[1], ADEL ALAEDDINI[1], CHINONSO OVUEGBE[1], PRANAV A. BHOUNSULE[2], PEYMAN NAJAFIRAD[3], AND KAI YANG[4]
[1]Department of Mechanical Engineering, The University of Texas at San Antonio, San Antonio, TX 78249, USA
[2]Department of Mechanical and Industrial Engineering, University of Illinois, Chicago, IL 60607, USA
[3]Department of Information Systems and Cyber Security, The University of Texas at San Antonio, San Antonio, TX 78249, USA
[4]Department of Industrial and Systems Engineering, Wayne State University, Detroit, MI 48202, USA

Corresponding author: Adel Alaeddini (adel.alaeddini@utsa.edu)

This work was supported in part by the Air Force Office of Scientific Research (AFOSR) under Grant FA9550-16-1-0171.

ABSTRACT Computer experiments are widely used to mimic expensive physical processes as black-box functions. A typical challenge of expensive computer experiments is to find the set of inputs that produce the desired response. This study proposes a multi-armed bandit regularized expected improvement (BREI) method to adaptively adjust the balance between exploration and exploitation for efficient global optimization of long-running computer experiments with low noise. The BREI adds a stochastic regularization term to the objective function of the expected improvement to integrate the information of additional exploration and exploitation into the optimization process. The proposed study also develops a multi-armed bandit strategy based on Thompson sampling for adaptive optimization of the tuning parameter of the BREI based on the preexisting and newly tested points. The performance of the proposed method is validated against some of the existing methods in the literature under different levels of noise using a case study on optimization of the collision avoidance algorithm in mobile robot motion planning as well as extensive simulation studies.

INDEX TERMS Computer experiments, Gaussian process regression, expected improvement, multi-armed bandit, Thompson sampling.

I. INTRODUCTION

Computer experiments are often used to simulate the physical processes which are time consuming, costly or simply impossible to test [1]. However, for complex problems they can still be computationally expensive, and therefore, there is often the desire to limit the number of simulations performed [2], [3]. A response surface model, also known as surrogate model, provides an approximation of the underlying black-box function that describes the relationship between the input variables and the response of the computer experiments. The Gaussian process (GP) model, which can be viewed as an extension of the standard regression model, is one of the most popular non-parametric probabilistic models for estimating black-box functions [4], [5]. The GP has key advantages over most estimation methods, which includes: (1) ability to fit highly nonlinear functions with minimal risk of overfitting, and (2) built-in capability for uncertainty estimation [6], [7]. Once a GP model is fit using some tested points, the expected response at any untested point can be easily estimated using the fitted surrogate model. Comprehensive reviews of the design and analysis of computer experiments are provided in [5], [8]–[10].

Computer experiments often require optimizing the underlying expensive black-box functions. An optimization procedure should be employed to find the optimal input with as few additional tests as possible. The black-box function to be optimized (minimized) is often denoted as $f(x)$, which is assumed to be a smooth (differentiable) function of the inputs over the feasible region $\chi \subset \mathbb{R}^d$. It is commonly assumed that the observed responses of the black-box function are corrupted by some noise, $y = f(x) + \epsilon$. The only available information is the response value y after testing the function at a given input point x.

The associate editor coordinating the review of this manuscript and approving it for publication was Sunith Bandaru.

To minimize a black-box function, a space filling design such as Latin hypercube design (LHD) [11], [12], sphere packing [13], or uniform designs [14] is used to generate a small number of initial input points at which the computer experiments are tested and respective responses are collected. Next, a GP model is fit and updated after testing each new point until the optimal point is found. The selection of new test points is often guided by an acquisition function. There is a vast literature on different acquisition functions for selecting the next most promising point to test [15]–[17]. Efficient global optimization is one of the most popular algorithms that uses expected improvement (EI) acquisition function for selection of the next most informative point [18].

Several extensions of the EI algorithm have been proposed in the literature to improve its performance and also extend its application to constrained problems [19]–[22], noisy responses [23], and parallel optimization [24]–[26]. Recently, [27] provided a comprehensive review of the EI extensions designed for parallel optimization, multiobjective optimization, constrained optimization, noisy optimization, multi-fidelity optimization and high-dimensional optimization. Sequential kriging optimization (SKO) [28] is an extension of EI that augments the expected improvement acquisition function to include the effective best solution instead of the observed minimum, which might differ from the true minimum of the function. Besides EI, knowledge gradient (KG) is another popular acquisition function that revisits the risk averse assumption made in EI's derivation, wherein the decision maker is only willing to return a previously tested point as the final solution [29], [30]. Known to work well for problems with noisy functions, KG allows to return to more promising solutions, which might have not been previously tested, by maximizing the expected increase in the conditional expected solution due to sampling [31]. However, as shown in the numerical study, KG doesn't provide significant advantage over classical EI when experiments have low noise. Besides EI-based methods, there also exists a group of algorithms that focus on optimizing expensive functions [32], [33]. Algorithms that are developed based on popular sampling techniques like upper confidence bound (UCB) and Thompson sampling, are discussed in [15], [34]–[41]. A survey of different algorithms in bandit setting are presented in [42].

Most of the EI-based acquisition functions in the literature, only consider the information of the expected value of possible improvement by each candidate test point. While informative, the expected value (of possible improvement) does not fully capture the uncertainty of the stochastic improvement by each candidate test point, which can help with better adjustment of the exploration and exploitation trade-off based on the system under consideration and the response value of the points already tested.

This study proposes an acquisition function based on adaptive regularization of EI (BREI) by each candidate test point to further improve the balance between exploration and exploitation. The proposed global optimization method has two major contributions: (1) regularizing the popular expected improvement (EI) acquisition function to better incorporate the information of additional exploration and exploitation to the optimization process, and (2) creating an efficient Bandit framework for optimizing the tuning parameter of the proposed regularized expected improvement, to improve the exploration vs exploitation balance. The proposed BREI method identifies the sequence of points that quickly converge to the global optimum of expensive computer experiments. The BREI is most suitable for the applications involving expensive black-box functions with no or low noise for which the resources are limited or the cost of testing the points is very high. This include expensive computer experiments, some robotic tests as provided in the case study, etc. Generally, for the situations where the evaluations are easily obtained, having an acquisition function like BREI might not be needed. However, for situations in which each evaluation might take days/weeks to complete or costs a lot, the BRIE helps finding the global optimum of a function with as few evaluations as possible.

The rest of the paper is organized as follows. A brief description of the related works is provided in Section II. Section III explains the major components of the proposed BREI method. Section IV provides the description of the proposed algorithms. In Section V, the performance of the proposed algorithm is evaluated along with some of the existing methods using a case study and simulations under different levels of noise. Finally, the concluding remarks and future directions are provided in Section VI.

II. RELATED WORKS

This section provides a brief description of the related methods. Throughout the paper, z is used to denote the design vector of the tested (observed) points, and x is used to denote the design vector of any (either tested or untested) point. Also, n is used to denote the number of tested points, and d is used to denote the dimensionality of the input variables (X).

A. GAUSSIAN PROCESS REGRESSION

Having some tested points (training set) represented by input-output pairs $(\boldsymbol{z_i}, y_i)$, where y_i might be corrupted by some noise ϵ_i, the GP defines a prior over an unknown link function f, and gives the posterior after seeing the data [43]. More specifically, the GP regression is defined as $y_i = f(\boldsymbol{z_i}) + \epsilon_i$ for $i = 1, \ldots, n$. The functional evaluation at the untested point x is denoted as f_*. $Y = (y_1, y_2, \ldots, y_n)^T$ is the observed outputs at the tested points $Z = \{\boldsymbol{z_1}, \boldsymbol{z_2}, \ldots, \boldsymbol{z_n}\}$. According to the joint distribution of the tested outputs and untested output:

$$\begin{bmatrix} Y \\ f_* \end{bmatrix} \sim \mathcal{N}\left(0\,, \begin{bmatrix} K(Z,Z)+\sigma_n^2\, I & K(Z,x) \\ K(x,Z) & K(x,x) \end{bmatrix}\right) \tag{1}$$

where $K(Z, Z)$, $K(Z, x)$, $K(x, Z)$, $K(x, x)$ are the covariance between the tested and tested points, tested and untested points, untested and tested points, untested and untested points respectively, and $K(., .)$ is an appropriate kernel function to evaluate the covariance. This study considers the squared exponential kernel $K(\mathbf{z_i}, \mathbf{z_j}) = \sigma_f^2 \; exp(-\frac{\| z_i - z_j \|^2}{2l^2})$, where σ_f^2 denotes the signal variance, and l denotes the characteristic length scale. Let $K(Z, Z) = K_{ZZ}, K_{Zx} = K(Z, x), K_{xZ} = K(x, Z), K_{xx} = K(x, x)$, by conditional distribution:

$$E(f_*(x)) = K_{xZ}(K_{ZZ} + \sigma_n^2 I)^{-1} y \tag{2}$$

$$cov(f_*(x)) = [K_{xx} - K_{xZ}[K_{ZZ} + \sigma_n^2 I]^{-1} K_{Zx}] \tag{3}$$

For a given untested point (x), the predictive mean (μ_x) is simply $E(f_*(x))$ in Equation (2), and the predictive variance s_x^2 is a diagonal element of the covariance matrix $cov(f_*(x))$ in Equation (3).

B. BANDIT PROBLEM

The multi-armed bandit framework is commonly used to formulate the trade-off between exploration and exploitation in sequential decision making [44]. The bandit problem aims to maximize the rewards of a player who plays an arm i out of the h arms of a slot machine at each time step t over a long run. After playing one arm at each time step, the player receives a real valued stochastic reward that is independently drawn from a fixed and unknown distribution. The player selects the arm to play based on the rewards of the $t - 1$ plays. If the player myopically chooses the arm that gave the highest reward in past plays (exploitation), he/she might fail to discover the (real) best arm due to the stochasticity of the rewards. On the other hand, if the player randomly selects an arm at each time step to explore the reward of different arms (exploration), the opportunity of playing the best arm multiple times decreases along with a decrease in the total reward. The multi-armed bandit problem helps to decide the best arm to play by balancing the exploration and exploitation to maximize the total rewards of the player. Thompson sampling [45] is one of the popular (Bayesian) approaches for solving the multi-armed bandit problem. It is also known as the posterior sampling or probability matching as it selects the arm based on posterior probability to be the best arm [46], [47]. Compared to other multi-armed bandit methodologies like UCB, Thompson sampling has the ability to handle wide range of information models that go beyond observing the individual rewards alone [39].

C. EXPECTED IMPROVEMENT (EI)

EI is one of the most common Bayesian optimization methods. Let, the stochastic improvement of a candidate test point x be $I_x = \max(f_{min} - y_x, 0)$, where $f_{min} = min(y_1, y_2, \ldots, y_n)$, and $y_x \sim \mathcal{N}(\mu_x, s_x^2)$ is the random variable that corresponds to the predicted response at x, with $\mu_x = E(f_*(x))$, and $s_x^2 = var(f_*(x))$. The expected value of the improvement is obtained as $E(I(x)) = E(\max(f_{min} - y_x, 0))$. The closed form solution for the EI is given as:

$$E(I_x) = (f_{min} - \mu_x)\Phi(\frac{f_{min} - \mu_x}{s_x}) + s_x \phi(\frac{f_{min} - \mu_x}{s_x}) \tag{4}$$

where $\phi(.)$ is the standard normal density function, and $\Phi(.)$ is the standard normal distribution function. EI often provides acceptable performance in reducing the number of test points for global optimization of expensive black-box functions. However, as the name implies, EI only utilizes the expected value of the random variable $I(x)$ and does not consider the uncertainty of the stochastic improvement. Section III extends the EI method by a specialized regularization term to better capture the uncertainty the stochastic improvement to boost its performance.

III. PROPOSED METHODOLOGY

In this section, first the formulation for the standard deviation of stochastic improvement by a candidate test point is derived. Next, an acquisition function (REI) is developed which uses some similar terms as the standard deviation of the stochastic improvement for selecting the next most informative test point. Finally, an adaptive strategy is presented for optimizing the tuning parameter of the proposed acquisition function (BREI).

A. STANDARD DEVIATION OF IMPROVEMENT

The fundamental definition of the standard deviation is used to derive the formulation of the standard deviation of improvement. Statistically, the uncertainty of improvement after adding each test point is defined as $\sigma(I_x) = \sqrt{E(I_x^2) - E(I_x)^2}$. After some tedious algebraic calculations, the closed form solution for $\sigma(I_x)$ is derived as:

$$\begin{aligned}\sigma(I_x) = \text{sqrt}\Big[&(f_{min} - \mu_x)^2 \Phi(\frac{f_{min} - \mu_x}{s_x}) \\ &+ 2s_x(f_{min} - \mu_x)\phi(\frac{f_{min} - \mu_x}{s_x}) \\ &- s_x^2((\frac{f_{min} - \mu_x}{s_x})\phi(\frac{f_{min} - \mu_x}{s_x}) - \Phi(\frac{f_{min} - \mu_x}{s_x})) \\ &- ((f_{min} - \mu_x)\Phi(\frac{f_{min} - \mu_x}{s_x}) + s_x\phi(\frac{f_{min} - \mu_x}{s_x}))^2\Big]\end{aligned} \tag{5}$$

B. REGULARIZED EXPECTED IMPROVEMENT

The proposed regularized expected improvement (REI) integrates the information of the expected improvement (EI) with a regularization term that utilizes some similar terms as the standard deviation of the stochastic improvement.

Adding the standard deviation of improvement, as shown in Equation 5, to the EI acquisition function as a regularization term results in small to moderate improvement in the efficiency (number of points) of black-box optimization as shown in Appendix VI-E. Supported by extensive simulation analysis, a revised version of Equation 5 is proposed to be added to the EI acquisition function as the regularization term to further improve the optimization performance:

$$REI_{Revised} = E(I_x) + \lambda\sigma^*(I_x) \tag{6}$$

where λ is a tuning parameter balancing the effect of the regularization term, and $\sigma^*(I_x)$ is calculated as:

$$\begin{aligned}\sigma^*(I_x) = \mathrm{sqrt}\Big[&(f_{min}-\mu_x)^2\Phi(\frac{f_{min}-\mu_x}{s_x})\\ &+2s_x(f_{min}-\mu_x)^{\mathbf{2}}\phi(\frac{f_{min}-\mu_x}{s_x})\\ &-s_x^2((\frac{f_{min}-\mu_x}{s_x})\phi(\frac{f_{min}-\mu_x}{s_x})-\mathbf{1})\\ &-((f_{min}-\mu_x)\Phi(\frac{f_{min}-\mu_x}{s_x})+s_x\phi(\frac{f_{min}-\mu_x}{s_x}))^2\Big]\end{aligned} \tag{7}$$

A positive λ value encourages more exploration, specially around the boundaries of the feasible region. A negative λ value encourages more exploitation around the estimated optimum of the underlying function. Also, a zero λ value reduces the proposed acquisition function to EI. Section III-C provides a detail discussion about the tuning parameter λ and its range. It also proposes a Thompson sampling approach to adaptively optimize λ based on the existing and newly added test points to the design.

It may also be worth noting that, the adjustment of standard deviation of stochastic variables, such as $\sigma^*(I_x)$ which is considered as the regularization term in the proposed REI acquisition function, has been commonly used in the statistical analysis [48], regression modeling [49], and deep learning [50].

C. ADAPTIVE OPTIMIZATION OF THE TUNING PARAMETER: EXPLORATION AND EXPLOITATION TRADE-OFF

The second term in Equation (6) adds a bias to the EI acquisition function to better adjust the balance between exploration and exploitation. The tuning parameter λ controls the level of trade-off between exploration and exploitation and therefore has a significant impact on the performance of the proposed method. Theoretically, λ can take any real value between $(-\infty, \infty)$, with positive values encouraging more exploration and negative values encouraging more exploitation. Meanwhile, to keep the bias (regularization) term small compared to the first/original term (the EI acquisition function), the tuning parameter should be set to as small value. Using simulation, setting $\lambda = -0.75$ provides an acceptable performance (in comparison to the EI) in most cases.

Meanwhile, a better strategy is to adaptively optimize the tuning parameter within a small interval at each iteration, because the optimal level of exploration and exploitation changes dynamically based on newly tested points and their observed responses. Based on extensive simulations, limiting the range of λ to $(-0.75, +0.75)$ provides the best performance.

In machine learning, tuning parameters are usually optimized using cross validation, i.e. ridge regression. However, sequential optimization of the tuning parameter at each iteration over a continuous space requires considerable computational effort. To reduce the computational complexity of the proposed method, an efficient adaptive optimization strategy is proposed based on Thompson sampling for multi-armed bandit where the candidates values of λ are treated as the arms of a slotting machine. The proposed tuning parameter optimization algorithm also utilizes the real gap information of the previously selected arms (λ values). The gap information helps to penalize for the arms that provide less improvement than expected. The optimal λ value selected by Thompson sampling at each iterations is used by the proposed BREI algorithm to select the most informative point for next evaluation:

1) IDENTIFYING THE SET OF CANDIDATE VALUES FOR THE TUNING PARAMETER

Instead of using a continuous range of possible values for the tuning parameter, a small set of candidate values, namely $\lambda_c = \{-0.75 \leq \lambda_1, \ldots, \lambda_h \leq 0.75\}$ is proposed. While there are different strategies for selecting the candidate values of the tuning parameter, a simple 7 equally distanced candidate values, $\lambda_c = \{-0.75, -0.5, -0.25, 0, 0.25, 0.5, 0.75\}$ are proposed. Such small set of candidate λ values reduce the search space without having significant negative impact.

2) SELECT THE OPTIMAL VALUE OF THE TUNING PARAMETER USING THOMPSON SAMPLING

Given the finite set of candidate values for λ, and the sequential nature of tuning parameter optimization in the proposed global optimization method, Thompson sampling is proposed to conduct a quick linear search among the candidate λ values to select the optimal value (λ^*) at each iteration. Thompson sampling treats each value in the candidate set λ_c as an arm of multi-armed bandit problem and uses the information of the tested points to select the best arm, which represents the optimal tuning parameter, to be used in each iteration.

Meanwhile, unlike classical problems of optimizing the tuning parameter using cross validation, in global optimization of expensive (computer) experiments only a small set of tested points is available, which should be used for both the training of the model and optimization of the tuning parameter. Therefore, in each iteration, just before selecting the next candidate point, the set of (already) tested points (Z) is split into two subsets, P (validation) and Q (train), such that the set P contains the first and the second minimum response points, and the set Q contains the rest of the points. The justification for using only the two lowest response points for the validation set P is to reduce the computational complexity. Reducing the size of P to only two points (lowest two responses), minimizes the computational effort for identifying whether an arm (candidate λ) is able to correctly identify the minimum point in the validation set, and is discussed in detail below.

After splitting the set Z into two subsets, P and Q, a Gaussian process is fitted with the points in set Q and test each of the arms $i \in \{1, \ldots, h\}$ for identifying the minimum response between the two points in P. Then the expected reward of selecting each arm ($R_t[i]$) is calculated as:

$$R_t[i] = \min_{x \in Q}(Y(x)) - Y(x') \tag{8}$$

where t denotes the iteration (time), which is also equivalent to the number of additional tested points added to the design, $min_{x \in Q}(Y(x))$ is the minimum response in set Q, and $Y(x')$ is the response of the selected point by arm i from set P. The expected reward in Equation (8) will always be positive as the minimum response value of the points in set P is less than the response value of any of the points in set Q. The reward approximates the gap between the correct and incorrect choice of minimum response points among the existing points. Once the expected reward of each arm is calculated, the probability of selecting each arm i is calculated as:

$$S_t[i] = \frac{R_t[i]}{\sum_{i=1}^{h} R_t[i]} \tag{9}$$

Then, the optimal arm (i_t^*) is chosen stochastically with respect to $S_t[i]$.

3) IMPROVE THOMPSON SAMPLING WITH REAL GAP INFORMATION

After observing the response value of a point tested at iteration (time) $(t-1)$, its information can be used to improve the estimated reward of the associated arm for the next iteration (t). Given the response value of the point tested at iteration $(t-1)$, the real gap information of the selected arm at iteration $(t-1)$ is calculated as $G_{i_{t-1}^*} = \min_{x \in Z_{t-1}}(Y(x)) - Y(x_t)$, where $min_{x \in Z_{t-1}}(Y(x))$ is the minimum response in set of tested points up to iteration $(t-1)$, and x_t is the response value of the point identified by BREI at $(t-1)$. The expected reward $R_t[i_{t-1}^*]$ for the last selected arm at iteration (t) can be updated to incorporate the real gap information $G_{i_{t-1}^*}$. In this paper, $R_t[i_{t-1}^*] = 0.2R_t[i_{t-1}^*] + 0.8G_{i_{t-1}^*}$ is considered to update the expected reward, which includes 20% of the current reward and 80% of real gap information from iteration $(t-1)$.

IV. PROPOSED ALGORITHMS

Algorithm 1 illustrates the major steps of the proposed multi-armed bandit regularized expected improvement (BREI) algorithm for global optimization of expensive computer experiments.

A. GLOBAL OPTIMIZATION USING BREI

The algorithm essential input includes the set of pre-specified points (Z) generated using a space filling design such as Latin hypercube design. The outputs of the algorithm include the minimum of the function (f_{min}), and the estimated GP $(f(x))$. The algorithm begins with testing the initial set of pre-specified points and their response values (Y) (Step 1). Next, it uses GP to create the surrogate model using the tested points (Z) (Step 2). Then, the tuning parameter λ is optimized using the Algorithm 2 (Step 3.1). Having the optimal value of the tuning parameter, particle swarm optimization (PSO) is used to solve Equation (6) to identify the next best candidate test point (Step 3.2). The selected point (x^*) is then tested and moved to the set of tested points (Z) before checking the stopping criterion for initiating another iteration (Step 3.3). Here, a pre-specified number of additional test points (t_*) is considered as the stopping criterion. After each iteration, GP is used to update the fit and also the current minimum point (Step 3.4).

Algorithm 1 BREI for Global Optimization of Expensive Computer Experiments

```
Input:     Set of pre-specified points using LHD ($Z$)
Output:    Global minimum of the function ($f_{min}$)
           Estimated GP ($f(x)$)

Step 1.    Evaluate the function at pre-specified points $Z$ to
           obtain the responses
           $Y = (y_1, y_2, \ldots, y_n)$
Step 2.    Create surrogate model using the points in
           $Z$ (tested points)
           $f(x) = K_{xZ}K_{ZZ}^{-1}Y$
Step 3     Until satisfying the desired stopping criteria, i.e.
           number of additional test points ($t_*$)
    Step 3.1    Optimize the tuning parameter $\lambda$ using
                Algorithm 2
    Step 3.2    Use an optimization algorithm, i.e. PSO,
                to select $x^*$ that maximizes Equation (6)
    Step 3.3    $Z \leftarrow Z \cup x^*, n \leftarrow n + 1$
    Step 3.4    $f(x) = K_{xZ}K_{ZZ}^{-1}Y$
                $f_{min} = min(Y)$
                Go to Step 3.1
```

B. OPTIMIZATION OF THE TUNING PARAMETER

Algorithm 2 demonstrates the proposed algorithm for optimizing the tuning parameter of the BREI algorithm. The algorithm inputs include the set of candidate values for the tuning parameter (λ_c), the most current set of tested points (Z) along with their respective observed responses (Y) (including both the initial points and additional/augmented points), and the most current number of tested points (t) added to the initial design, which also shows the iteration (time) in multi-armed bandit setting. As discussed in Section III-C1, to reduce the computational complexity of the optimization algorithm, a finite set of seven candidate values is considered for the tuning parameter, $\lambda_c = -0.75, -0.5, -0.25, 0, 0.25, 0.5, 0.75$. The output of the algorithm is the optimal value of the tuning parameter (λ^*).

The algorithm begins with dividing the set of existing tested points (Z) into two subsets of P and Q, with P consisting of the two points with minimum response values, and

Q consisting of the rest of the points (Step 1). Next, a surrogate model is fitted using the points in Q (Step 2). Then, for each λ in the candidate set λ_c, the BREI algorithm is used to select the next best test point (x') from the set P (Step 3.1). Then, the reward of each arm based is calculated on the observed gap between the minimum response value of the points in Q and the response value of the function at x' (Step 3.2). When $t > 0$, the expected reward of the arm that has been used in the previous iteration $(t-1)$ (Step 3.3.1) is updated to include the real gap information of the arm ($G_{i^*_{t-1}}$) (Step 3.3.2). This simply penalizes the previously selected arm (i^*_{t-1}), if its choice of tuning parameter did not help in selecting the candidate test point that actually (further) decreased the response value of the function. Next, the selection probabilities S_t of the arms are updated based on the estimated rewards (Step 4). Finally, a stochastic policy (Step 5) is used to select the arm that provides the optimal λ^* value to be used for selecting the next best candidate test point (Step 6).

Algorithm 2 Multi-Armed Bandit Optimization of the Tuning Parameter (λ)

```
Input:       Set of $n$ tested points $(Z, Y)$,
             Current number of test points added
             to the design $(t,\ t < t_*)$,
             Set of $h$ candidate values for the
             tuning parameter $(\lambda_c = \{\lambda_1, \ldots, \lambda_h\})$
Output:      Optimal value of the tuning
             parameter $(\lambda^*)$

Step 1.      $min1 = arg\min_{x \in Z}(Y(x))$,
             $min2 = arg\min_{x \in Z-min1}(Y(x))$,
             $P = \{min1, min2\}, Q = Z - P$
Step 2.      Fit a surrogate model based on the
             points in $Q$ (Equations (2) and (3))
Step 3.      For each $\lambda_{i=1,\ldots h} \in \lambda_c$:
Step 3.1.        Use $\lambda_i$ and BREI (Equation (6))
                 to select $x'$ from $P$
Step 3.2.        Calculate the reward of each arm
                 as $R_t[i] = \min_{x \in Q}(Y(x)) - Y(x')$
Step 3.3.        If $t > 0$, for the arm selected in
                 the last iteration $(i^*_{t-1})$:
Step 3.3.1.          $G_{i^*_{t-1}} = \min_{x \in Z_{t-1}}(Y(x)) - Y(x_t)$
Step 3.3.2.          $R_t[i^*_{t-1}] = 0.2R_t[i^*_{t-1}] + 0.8G_{i^*_{t-1}}$
Step 4.      $S_t[i] = \frac{R_t[i]}{\sum_{i=1}^{h} R_t[i]}$
Step 5.      Optimal arm $i^*_t$ is selected stochastically
             with respect to $S_t[i]$
Step 6.      $\lambda^* = \lambda_c[i^*_t]$
```

V. RESULTS AND DISCUSSION

In this section, the performance of the proposed BREI method is validated along with a number of existing methods in the literature including expected improvement (EI), sequential kriging optimization (SKO), knowledge gradient (KG) and Gaussian process based UCB (GPUCB) using both a case study and simulated experiments. The justification for considering the above four algorithms for comparison is that they are among the most common and/or the best performing algorithms in the literature. To ensure a fair comparison between the BREI and the other comparing methods, 8 commonly used response surface models in the literature are considered. Each experiment is run 100 times and the average of the observed minimum response collected after each of the 100 additional points is reported. The median, 25^{th}, and 75^{th} performance percentiles are also reported in the Appendix. Furthermore, each response model is tested at different noise levels to understand the capability and limitations of the proposed method in comparison to the other methods.

The organization of this section is as follows. First, a brief discussion of each of the comparing methods and the performance metric chosen for the analysis of the results is provided. Next, the result of a case study for the weight optimization of a dynamic window approach (DWA) in obstacle avoidance algorithm for mobile robots planning is presented. Finally, the result of a simulation study based on eight nonlinear response models of 2 to 10 dimensions with different noise levels is described. In this paper, MATLAB is used for coding and GPML library [51] for optimizing the hyperparameters of the GP model.

A. COMPARING METHODS

This subsection provides a brief description of the comparing methods, except the EI which is presented earlier in Section II-C.

1) SEQUENTIAL KRIGING OPTIMIZATION (SKO)

SKO [28] selects the next test point that maximizes the acquisition function as given in Equation (10)

$$E(I(x)) = [(\mu_{x^{**}} - \mu_x)\Phi(\frac{\mu_x^{**} - \mu_x}{s_x}) + s_x\phi(\frac{\mu_x^{**} - \mu_x}{s_x})](1 - \frac{\sigma_n}{\sqrt{s_x^2 + \sigma_n^2}}) \tag{10}$$

where x^{**} is the current effective best solution. Different from the EI method that uses the best observed solution (f_{min}), SKO utilizes the current effective best solution from the utility function $u_x = -\mu_x - cs_x$, where $x \in Z$, and c is a tuning parameter generally set to 1.

2) KNOWLEDGE GRADIENT (KG)

KG [52] selects the next test point by maximizing the improvement function as given in Equation (11) [53]

$$I(x) = \min_{x\ \epsilon\ Z \cup x_{n+1}} (\mu_x) - \min_{x\ \epsilon\ Z \cup x_{n+1}} [\mu_x + \frac{cov(x, x_{n+1})}{\sqrt{s_{x_{n+1}}^2 + \sigma_n^2}} z_r] \tag{11}$$

where z_r is the standard normal variable. KG assumes the conditional mean (through the model) might be closer to the true observation rather than the functional evaluations.

3) GAUSSIAN PROCESS-UPPER CONFIDENCE BOUND (GPUCB)

The Gaussian process based UCB (GPUCB) formalizes the Gaussian process optimization as a multi-armed bandit problem and solves it using upper confidence bound (UCB) method [54]. For a maximization problem, GPUCB selects the next test point that maximizes $\mu_x + \beta^{\frac{1}{2}} s_x$, where $\beta^{\frac{1}{2}}$ is the parameter to be optimized and can be calculated as $\beta_t = 2log(\frac{t^{\frac{d}{2}+2}\pi^2}{3\delta})$ at round t and $\delta \in (0, 1)$ [15].

B. PERFORMANCE METRIC

The main objective of the proposed BREI algorithm is to efficiently find the global minimum of the computer experiments. Therefore, the observed response of the candidate test point suggested by each of the comparing methods at each iteration, namely $f_{min} = \min_{x \in Z}(y(x))$, is considered as the performance metric [28], [55], [56]. In order to achieve a high level of confidence over the results, all experiments are repeated hundred times and the average is reported. Additionally, the median, 25^{th}, and 75^{th} performance percentiles are reported in the Appendix.

C. CASE STUDY: ROBOTS MOTION PLANNING

This section illustrates the results of a case study for optimization of the weight of the dynamic window approach (DWA) in obstacle avoidance algorithm for robots motion planning. The methods considered for comparison include the proposed BREI method, expected improvement (EI), sequential kriging optimization (SKO), knowledge gradient (KG) and Gaussian process based UCB (GPUCB). The DWA is a classical motion planning algorithm for mobile robots developed by [57], that outputs the optimal translational and rotational velocity commands (v, w) to navigate non-holonomic vehicles through obstacle free paths to a goal. The noise free DWA algorithm has the computational complexity of $\mathcal{O}(n)$. In general, the physical experiment is conducted using a mobile robot that has a non-holonomic mobile base as shown in Figure 1. However, as the physical experiment is expensive to evaluate, typically a simulation environment [58] is developed using MATLAB which is considered for this study. The algorithm works by discretely selecting obstacle free trajectories in a dynamic window until the goal is reached. The dynamic window is a constrained velocity search space constituting of velocities based on the kinematic limitations of the robot and admissible velocities that are reachable within the next simulation time slice. The sets of translational (v) and rotational velocity (w) pairs in the dynamic window are used in the evaluation of the objective function. The (v, w) pair selection within the search space is guided by the objective function shown in Equation (12).

$$G(v, w) = \alpha heading(v, w) + \beta dist(v, w) + \gamma vel(v, w) \tag{12}$$

The objective function includes three sub functions. The $heading(v, w)$ function measures the orientation of the robot toward the goal. Heading, tested by $180 - \theta$, increases as the target angle (θ) to goal reduces. The $dist(v, w)$ function ensures obstacle free paths by calculating the norm distance to the closest obstacle per trajectory roll out in the navigation space. The $vel(v, w)$ expression measures the forward progress of the robot. This is basically a projection of the translational velocity v of the robot, updated every time step. The sub-functions in Equation (12) are indirectly dependent on new velocity pair inputs (v, w). The velocity pair updates are control inputs that change the closest obstacle position and target angle θ which are direct variables used in calculating the heading direction and distance to obstacle components. Velocity pair updates are evaluated using kinematic equations in a constrained search space (see [57] for detailed explanation). The combination of weight parameters α, β, γ play an important role in the objective function and can generate different navigation outcomes as shown in the Figure 2 by introducing bias based on the weight value. A common alternative method for selecting weight sets α, β, γ is a simple manual tuning, guided by the navigation behavior of the robot, as adopted in [57], [59]–[61]. This is simply a trial and error method, adjusting the weights based on the navigation behavior at the end of each experiment. The BREI optimization algorithm is applied along with the other comparing methods to choose the optimal weight parameters (α, β, γ) to minimize the time taken to navigate from a starting point to the ending point with 10 fixed obstacles. First, a Latin hypercube design of 30 points (weight parameters) is created. Next, the time taken to navigate from a starting point to the ending point is evaluated based on each set of weight parameters and use them as the initial set of points. In addition to the initial points, 100 additional test points (weight parameters) are tested sequentially and the associated travel times are collected. Each test is replicated hundred times and the average is reported. The median, 25^{th}, and 75^{th} performance percentiles are also reported in the Appendix.

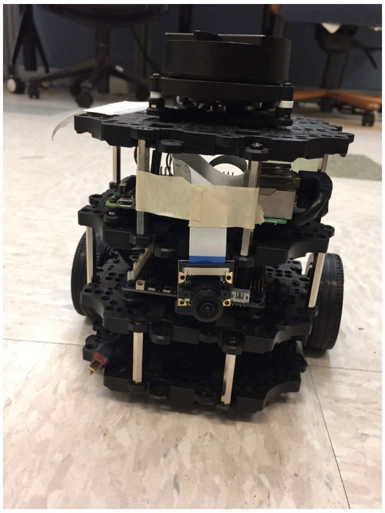


FIGURE 1. An illustration of a mobile robot.

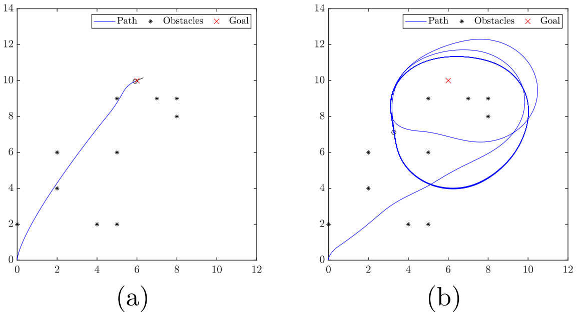


FIGURE 2. Navigation outcomes of the weight parameter combinations (a) $\alpha = 1$, $\beta = 0$, $\gamma = 0.58$, and (b) $\alpha = 0.19$, $\beta = 0.85$, $\gamma = 0.61$.

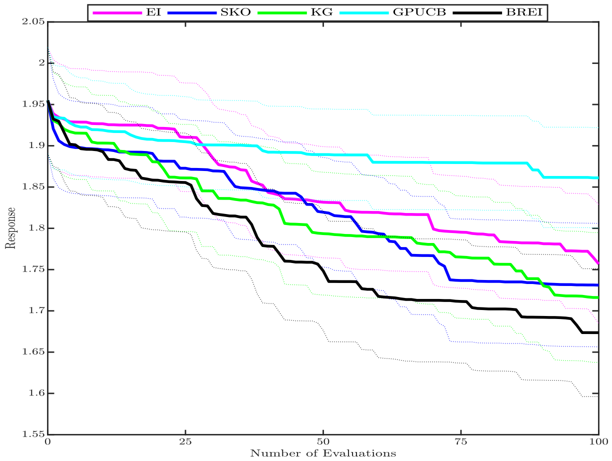


FIGURE 3. The minimum travel time of the robot averaged over 100 replicates for the initial +100 additional points. The dotted lines represent the respective 95% confidence intervals of the observed responses.

Figure 3 illustrates the mean performance of the proposed BREI acquisition function in comparison to the expected improvement (EI), sequential kriging optimization (SKO), knowledge gradient (KG) and Gaussian process based UCB (GPUCB) methods for the case study for the initial set of points as well as the 100 additional evaluation points along with the 95% confidence interval. For the initial set of test points (parameter settings) as well as the first 10 additional points, the proposed BREI acquisition function shows a similar performance to the other comparing methods. This is probably because these points ($t < 10$) are used by the comparing methods to explore the underlying function. However, the proposed BREI method shows a significant improvement over the other comparing methods after the 10^{th} additional point ($t > 10$). Also, as the number of additional points increases, the proposed acquisition function maintains and/or increases its gap over the other comparing methods.

Figure 4 complements the result of Figure 3 by providing the boxplot of the average (mean) of the observed minimum responses of the comparing methods over the 100 additional points (iterations). As shown in the Figure 4, the proposed method provides the best performance, which verifies its improvement over the other comparing methods.

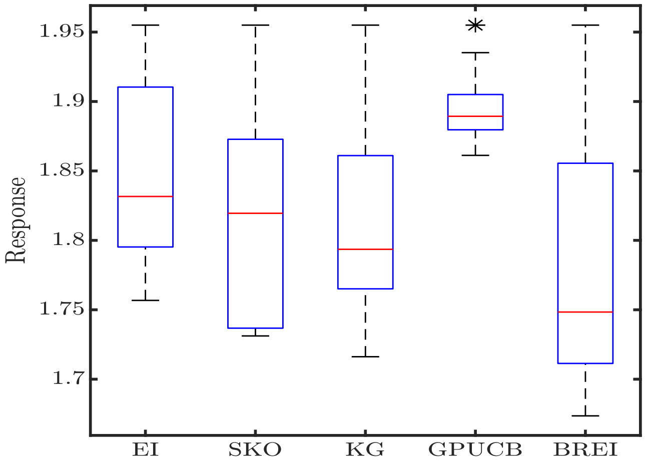


FIGURE 4. The boxplot of the averaged observed minimum response of the comparing methods over the 100 additional evaluation points for the mobile robot case study.

The results of Figures 3 and 4 are also validated using the Wilcoxon rank test for the significance of the difference between the observed minimum response of the proposed BREI method and the other comparing methods. The Wilcoxon rank test shows a p-value of 0 for all of the four pairwise comparisons between the BREI and EI, BREI and SKO, BREI and KG and BREI and GPUCB to statistically validate the significance of the improvement made by the proposed method.

D. SIMULATED EXPERIMENTS: NONLINEAR RESPONSE MODELS

This section evaluates the performance of the proposed BREI acquisition function with those of EI, SKO, KG and GPUCB over eight nonlinear response models of two, three, six and ten dimensions at different levels of noise including 0%, 1% and 5% of the mean value of the response models. These response models are presented in Figure 5. Similar to the case study, for each of the comparing methods, a Latin hypercube design of $10d$ points is created, where d represents the number of dimensions. Next, the function is evaluated at each point to create the initial set of tested points. The number of additional points, which is set to $t_* = 100$, is used as the stopping criterion.

Figure 6 illustrates the mean performance along with the 95% confidence interval of the proposed BREI acquisition function in comparison to those of the expected improvement (EI), sequential kriging optimization (SKO), knowledge gradient (KG) and Gaussian process based UCB (GPUCB) methods for different response models and different noise levels (0%,1% and 5%). The median, 25^{th}, and 75^{th} performance percentiles are also reported in the Appendix. As shown in Figure 6, when there is no noise (0% noise), the BREI method outperforms the other comparing methods over most nonlinear response models, in terms of minimum number of tests required to minimize the black-box function. For the 1% and 5% noise levels, the BREI method outperforms other

	Bounds	Response model
2.1	$x_1 = [-1.6, 2.4]$ $x_2 = [-0.8, 1.2]$	$y = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + (4x_2^2 - 4)x_2^2 + \varepsilon$
2.2	$x_i = [-5, 5]$	$y = sin^2\,\pi\left(1 + \frac{x_1 - 1}{4}\right) + \left(1 + \frac{x_1 - 1}{4}\right)^2\left[1 + 10\,sin^2\,\pi(1 + \frac{x_1 - 1}{4}\right) + 1)] + \left(\left(1 + \frac{x_2 - 1}{4}\right) - 1\right)^2 [1 + sin^2(2\pi\left(1 + \frac{x_2 - 1}{4}\right))] + \varepsilon$
3.1	$x_i = [0, 1]$	$y = 4[(x_1 - 2 + 8x_2 - 8x_2^2)^2 + (3 - 4x_2)^2 + 16\sqrt{x_3 + 1}(2x_3 - 1)^2] + \varepsilon$
3.2	$x_i = [-5, 5]$	$y = \sum_{i=1}^{3}(x_i - i)^2\ + \varepsilon$
6.1	$x_i = [-5, 5]$	$y = \sum_{i=1}^{6}\lvert x_i sinx_i + 0.1x_i\rvert\ + \varepsilon$
6.2	$x_i = [-2, 2]$	$y = \sum_{i=1}^{6} x_i^2 + 2x_{i+1}^2 - 0.3\,cos(3\pi x_i) - 0.4\,cos(4\pi x_{i+1}) + 0.7 + \varepsilon$
10.1	$x_i = [-1, 1]$	$y = \sum_{i=1}^{10} ix_i^4\ + \varepsilon$
10.2	$x_i = [-1, 1]$	$y = \sum_{i=1}^{10}\frac{x_i^2}{4000} - \prod_{i=1}^{10} cos\left(\frac{x_i}{\sqrt{i}}\right) + 1\ + \varepsilon$

FIGURE 5. Non-linear response models considered for the comparisons.

TABLE 1. P-values of the Wilcoxon rank test - simulated experiments.

		2.1	2.2	3.1	3.2	6.1	6.2	10.1	10.2
		BREI							
EI	0%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
SKO		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
KG		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
GPUCB		0.06	0.0	0.06	0.89	0.0	0.0	0.0	0.0
EI	1%	0.0	0.56	0.0	0.23	0.0	0.0	0.0	0.0
SKO		0.23	0.0	0.0	0.09	0.0	0.0	0.0	0.0
KG		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
GPUCB		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EI	5%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
SKO		0.0	0.19	0.0	0.0	0.0	0.0	0.0	0.0
KG		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
GPUCB		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

comparing methods for high dimensional response models, i.e. 6d and 10d, whereas for low dimensional response models i.e. 2d and 3d, the BREI performance is mixed with KG and GPUCB.

The results demonstrate the proposed BREI method provides the most competitive performance for high dimensional functions and low noise in general. Meanwhile, as the dimensionality of the functions increases (from 2d to 10d), the proposed algorithm generally increases its advantage over other methods, even for high noise levels. This is mainly due to the contribution of the proposed regularization term along with the adaptive optimization of the tuning parameter that helps improving the exploration and exploitation of the design space. However, for low dimensional functions, i.e. 2.1 and 2.2, all of the comparing methods provide a competitive performance at different levels of noise. Therefore, the proposed algorithm does not provide significant improvement over the best of existing methods, namely knowledge gradient, when the function is low dimension and the noise level is high. Figure 7 provides the boxplot of the mean performance of each of the comparing methods, over the 100 replicates of the observed response, for each of the response models across the initial set of points as well as the 100 additional points. As shown in Figure 7, for majority of cases, the proposed BREI method provides the best performance, in terms of the 1st, 2nd and 3rd quartiles, compared to the others. Also, for most cases, BREI shows a lower variance in the boxplot, which can be attributed to better exploitation of the points near the global optimum, which results in faster convergence in comparison to other methods. This is mostly because of the better adjustment of exploration and exploitation by the proposed BREI acquisition function.

Finally, Table 1 provides the result of the Wilcoxon rank test for the significance of the difference between the observed minimum response of the proposed method against the other comparing methods, where lower values show an increased probability of difference in the observed minimum response. As shown in the Table 1, the Wilcoxon rank test also signifies the improvements made by the proposed method at low noise and high dimensions, which further validates the earlier results.

To better illustrate the performance of the comparing methods, Figure 8 visualises the distribution of the selected points by each method for the popular six hump camel (SHC) function (response model 2.1) at 0% noise. The SHC function has two global minimum at $x_1 = (0.0898, -0.0898)$ and $x_2 = (-0.7127, 0.7127)$ with the corresponding response value of $y = -1.0316$. All of the comparing methods start with selecting the same set of initial points based on the LHD design. Next, they use their specialized acquisition functions, i.e. EI, SKO, KG, GPUCB and BREI, to identify the global minimum. As shown in the Figure 8, all of the comparing methods provide competitive performance by exploring the areas around the global minimum. Meanwhile, the proposed BREI algorithm,in addition to GPUCB, provide the best performance by quickly exploiting the knowledge gained from the first few additional test points (4 points) and converges to the global minimum.

Figure 9 complements the results of Figure 8 by illustrating the distribution of the selected points by each of the comparing methods for the more complex response model 2.2 at 0% noise. Similar to the preceding results, the proposed BREI acquisition function demonstrates a superior performance by reaching to the global optimum with fewer number of tests, namely 16 additional test points. As both the response models tested are of lower dimension and lower noise, GPUCB also provided best performance but it can be seen that the algorithm focused only on exploitation and did not explore at all. As it only exploited, for response model 2.1, it could

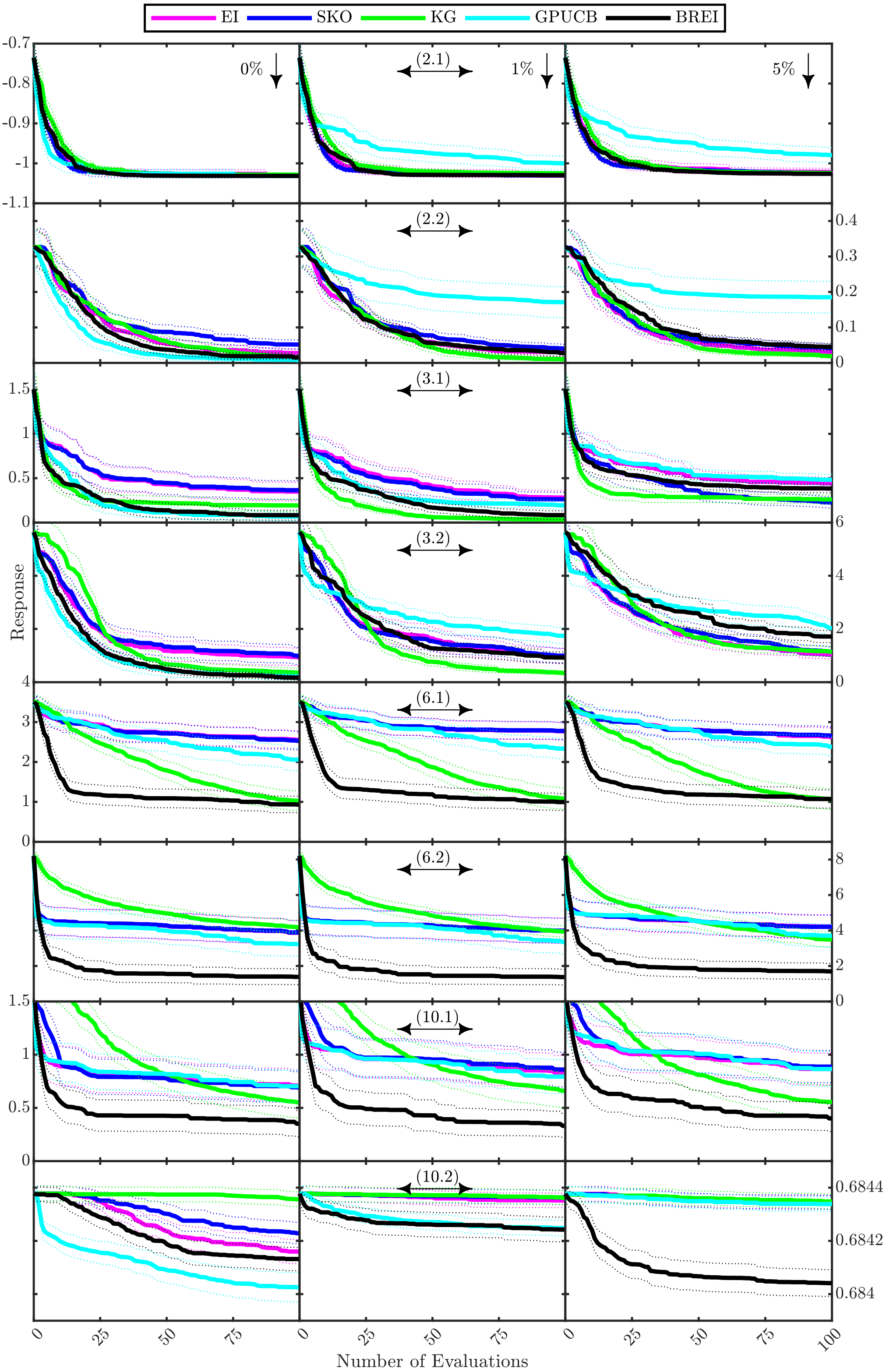


FIGURE 6. Average (mean) of the 100 replicates of the observed response values for each of the comparing methods after each additional test for the simulated experiments. The dotted lines represent the respective 95% confidence intervals of the observed minimum responses at each iteration.

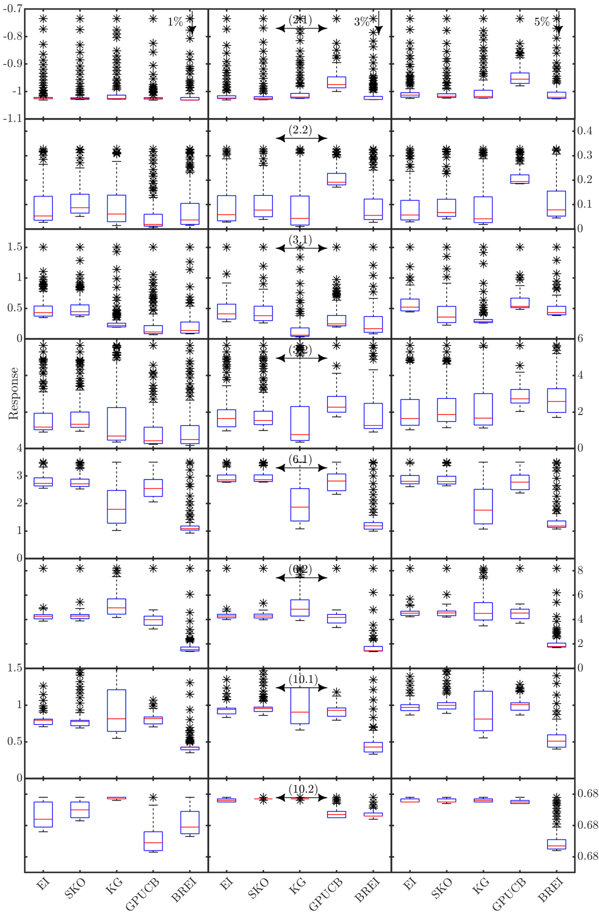


FIGURE 7. Boxplots of the average performance of the comparing methods over 100 replicates, for the initial +100 additional points for different response models (rows) and different levels of noise (columns).

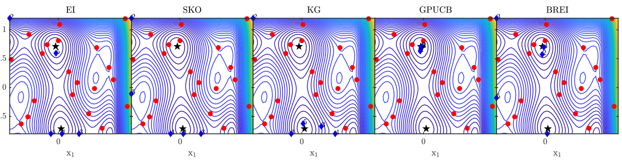


FIGURE 8. Distribution of selected points by the comparing methods for response model 2.1 at 0% noise. The global minimum is in black (star), initial points are in red (circle), and additional points are in blue (diamond).

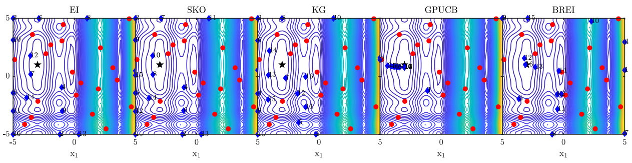


FIGURE 9. Distribution of selected points by the comparing methods for response model 2.2 at 0% noise. The global minimum is in black (star), initial points are in red (circle), and additional points are in blue (diamond).

not investigate any points around the other global minimum; whereas BREI was able to search around both the global minimums.

E. COMPUTATIONAL COMPLEXITY

The acquisition function of the proposed BREI acquisition function as shown in Equation (6) has three components: (1) the expected improvement component $E(I_x)$, (2) the regularization term, $\sigma^*(I_x)$, and (3) the tuning parameter λ. According to [62] the computational complexity of $E(I_x)$ for optimization of expensive black-box functions is $\mathcal{O}(n^3)$. The regularization term of BREI acquisition function ($\sigma(I_x)$) has similar terms as expected improvement component and therefore can be calculated with the same computational complexity. For the tuning parameter λ, different from the traditional cross validation methods where the tested points are divided randomly into subsets resulting in fitting the surrogate model multiple times, the proposed Thompson sampling based approach fits the surrogate model only once using points in Q which takes $\mathcal{O}(n^3)$ (the computational complexity of adding gap information, etc. is negligible). Consequently, the computational complexity of the proposed algorithm is $\mathcal{O}(n^3)$.

VI. CONCLUSION

In this paper, a novel acquisition function based on the multi-armed bandit regularized expected improvement (BREI) is proposed for efficient global optimization of expensive computer experiments with low noise. The proposed method extends the expected improvement by adaptive regularization based on each candidate point. A Thompson sampling algorithm is also proposed under multi-armed bandit setting to adaptively optimize the tuning parameter of the proposed BREI acquisition function to balance the exploration and exploitation based on the previously tested points. Using a case study in robot motion planning and several nonlinear response models of 2 to 10 dimensions with different levels of noise, the performance of the proposed acquisition function is studied in comparison to some of the most popular methods in the literature including Expected Improvement (EI), Sequential Kriging Optimization (SKO), knowledge gradient (KG) and Gaussian process based UCB (GPUCB). Several statistics including the average (mean), median, 25^{th}, and 75^{th} performance percentiles are considered for performance analysis in terms of the predicted global minimum under low level of noise. The proposed method demonstrates a competitive performance to the existing methods in the literature for both the case study as well as the simulated experiments across different statistics. It also shows improvement over the existing methods for higher dimensions. For instance, for the case study on robot motion planning, the performance achieved by the proposed method by 50^{th} evaluation is on par with the performance of the best of other comparing methods after 75^{th} evaluation. The proposed algorithm also has a computational complexity of $\mathcal{O}(n^3)$. In many applications involving expensive black-box functions with low or no noise, such as long running computer codes, where the resources are limited, or the cost of testing the points is very high. This framework reduces the number of expensive test points in global optimization by adaptively choosing the most informative regions to explore and exploit. For the future work, the proposed method will be extended to multi-armed bandit setting.

APPENDIX

A. COMPARISON OF DIFFERENT SCENARIOS FOR SELECTING THE TUNING PARAMETER

This section compares different λ selection scenarios for the proposed BREI method. The comparing scenarios include: (1) fixed λ values instead of the proposed Thompson sampling based tuning parameter optimization algorithm, (2) annealing λ value from $+0.75$ t0 -0.75 for every 15 iterations, (3) the proposed Thompson sampling based tuning parameter optimization algorithm without including the real gap information, and (4) the proposed Thompson sampling based tuning parameter optimization algorithm with the gap information. For the tuning parameter optimization algorithm, when the real gap information is not used, Step 3.3 of Algorithm 2 is ignored. As significant improvement among methods was shown for high dimensional functions, one of 6 and one 10 dimensional functions are used for comparing different scenarios. The number of replications is also limited to ten.

The proposed Thompson sampling-based algorithm adaptively optimizes the λ value using the knowledge of previous iteration. Having the λ fixed, the constant value that worked well for one response model might not work for another response model. In general, -0.75 value for λ consistently worked well for many problems. For the response models shown in the Figure 10, although fixed -0.75 and -0.5 lambda values have provided competitive performance, it can be seen that their performance is not as consistent as the proposed strategy (Thompson sampling with gap information). From all the response models provided in the Figure 10, it can be seen that the proposed method outperforms the other scenarios using around 20 iterations. Comparing with the scenario of not including the real gap information, it can be seen that including the gap information provides similar or improved performance. Additionally, including the gap information does not degrade the performance for at least the response models tested. Comparing with the scenario of annealing the λ, it can be seen that the proposed strategy (Thompson sampling with gap information) significantly performs better as the former does not have the ability to adaptively change based on the updated surrogate model and knowledge of real gap information.

B. ANALYSIS OF THE PATTERNS OF THE TUNING PARAMETER VALUES OVER ITERATIONS

Figure 11 illustrates the selected values of the tuning parameter λ of the proposed BREI method as a function of additional points $(1, 2, \ldots, 100)$ for four different replicates of the case study. As shown in the figure, while for some replicates the λ values show some level of consistency or convergence, in general, the plot looks inconsistent. This may be due to selecting the λ values based on a stochastic policy using the expected reward of each arm. The patterns of λ value for the simulated functions also show the same behavior.

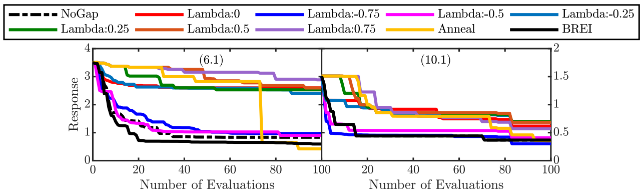


FIGURE 10. Comparisons of different scenarios for selecting the tuning parameter over a 6 dimensional (6.1) and a 10 dimensional (10.1) functions based on 10 replicates.

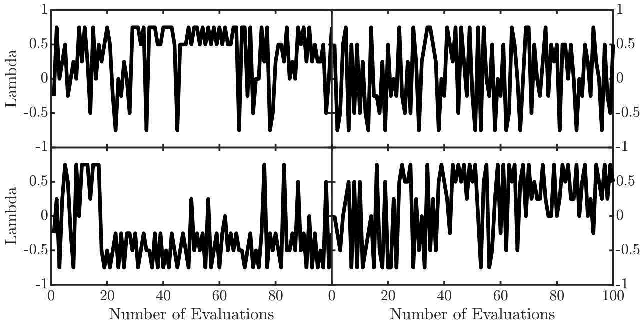


FIGURE 11. Selected values of the tuning parameter λ as a function of iteration for four different replicates of the case study.

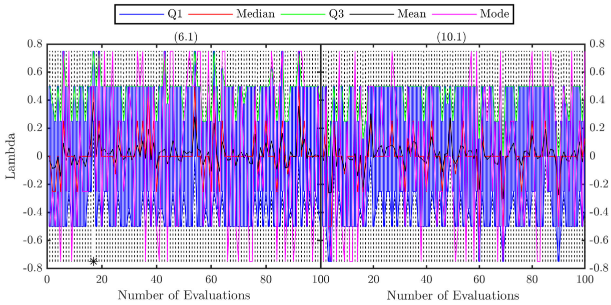
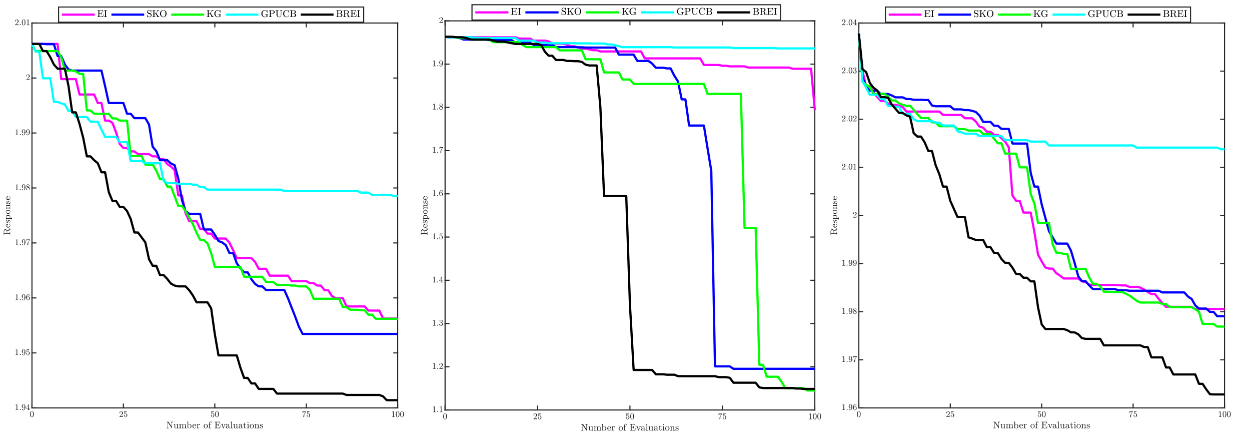


FIGURE 12. Boxplots of the selected lambda values as a function of evaluation points over the 100 replicates for the 6.1 (left) and 10.1 (right). The plots include the 1st and 3rd quartiles, mean, median and mode.

Figure 12 complements the results with the boxplot of the λ values as function of additional evaluation points over the 100 replicates for the 6.1 (6D: left plot) and 10.1 (10D: righ plot) functions in the simulation study. As shown in the Figure 12, the boxplots do not show any meaningful pattern or consistency/convergence for the λ values. The boxplots of the case study and other simulated functions also show the same behaviour.

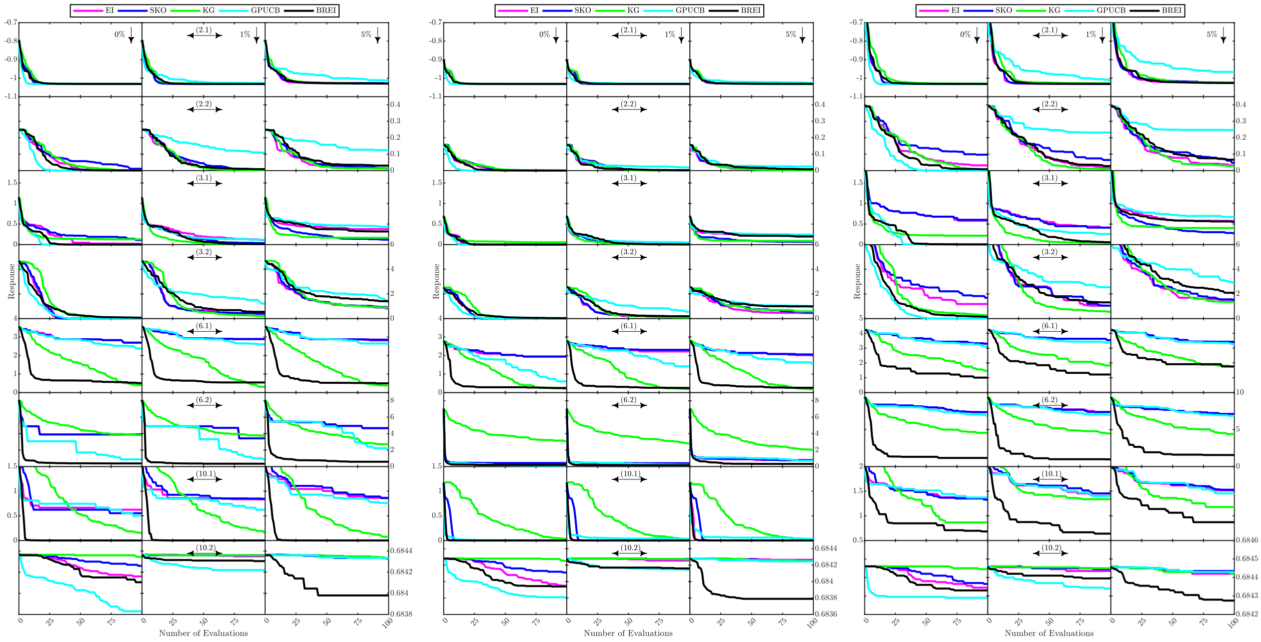
C. ADDITIONAL PERFORMANCE PLOTS OF THE COMPARING METHODS FOR THE CASE STUDY AND SIMULATED EXPERIMENTS

Figures 13 and 14 illustrate the median, first quartile (Q1), and third (Q3) quartiles trends of the observed responses of the comparing methods over the 100 replicates at the initial



(a) Median (b) Q1 (c) Q3

FIGURE 13. Case study: Median, 25th(Q1) and 75th(Q3) percentiles of the 100 replicates of the observed response values for each of the comparing methods after each additional test.



(a) Median (b) Q1 (c) Q3

FIGURE 14. Simulated experiments: Median, 25th(Q1) and 75th(Q3) percentiles of the 100 replicates of the observed response values for each of the comparing methods after each additional test at different noise levels.

set of points as well as the 100 additional points for the case study and the simulated experiments.

D. COMPUTATIONAL TIME

Figure 15 shows the computational time for teh comparing methods (in seconds) for some 2, 3, 6 and 10 dimensional response models used in this study. Specifically, 2.1, 3.1, 6.1 and 10.1 response models are considered for plotting. For lower dimension models, the computational time taken by the BREI is similar to the EI. However, as the number of dimensions increases the computational time of teh BREI surpasses the EI with a linear slope. This is probably due to the increase in the size of Z used to fit the surrogate model in Algorithm 2. From the analysis, out of all comparing methods, KG requires highest computational time.

E. COMPARISON AMONG EI, REGULARIZED EI WITH CLASSIC STANDARD DEVIATION, AND THE PROPOSED BREI

Figure 16 illustrates the mean performance of EI, BREI with classic standard deviation (BREI Original) and the proposed BREI for higher dimension functions i.e. 6.1, 6.2, 10.1 and 10.2. BREI Original uses $REI_{Original} = E(I_x) + \lambda\sigma(I_x)$, while BREI uses Equation 6 to select the next test point. As seen

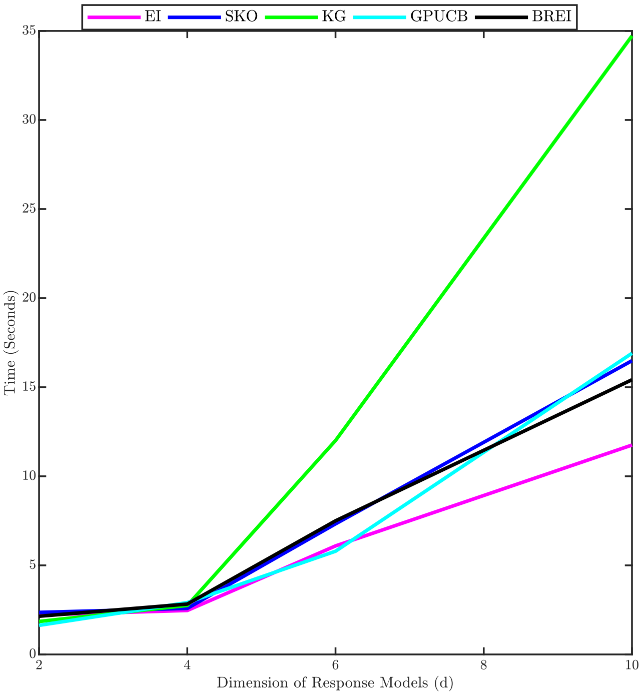


FIGURE 15. Computational times of the comparing methods as a function of dimensions based on a 2d, 3d, 6d and 10d response models.

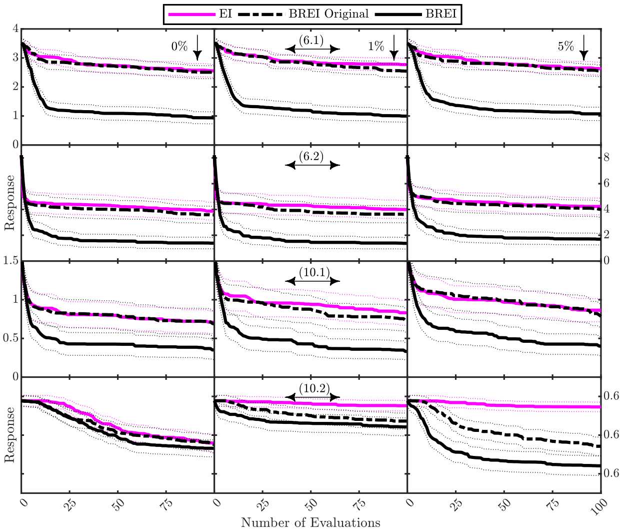


FIGURE 16. Comparison of EI, BREI original and BREI for higher dimension functions - averaged over 100 replicates of the observed responses after each additional test. The dotted lines represent the 95% confidence intervals of the observed responses.

in Figure 16, although BREI original provides superior performance compared to EI, the proposed BREI acquisition function with $\sigma^*(I_x)$ provides the best performance.

REFERENCES

[1] K.-T. Fang, R. Li, and A. Sudjianto, *Design and Modeling for Computer Experiments*. Boca Raton, FL, USA: CRC Press, 2005.

[2] X. He, R. Tuo, and C. F. J. Wu, "Optimization of multi-fidelity computer experiments via the EQIE criterion," *Technometrics*, vol. 59, no. 1, pp. 58–68, Jan. 2017.

[3] B. Zhang, D. A. Cole, and R. B. Gramacy, "Distance-distributed design for Gaussian process surrogates," *Technometrics*, vol. 63, no. 1, pp. 1–13, 2019.

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

[5] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, "Design and analysis of computer experiments," *Stat. Sci.*, vol. 4, no. 4, pp. 409–423, 1989.

[6] J. Kocijan, R. Murray-Smith, C. E. Rasmussen, and A. Girard, "Gaussian process model based predictive control," in *Proc. Amer. Control Conf.*, vol. 3, 2004, pp. 2214–2219.

[7] H. (Heidi) Xia, Y. Ding, and J. Wang, "Gaussian process method for form error assessment using coordinate measurements," *IIE Trans.*, vol. 40, no. 10, pp. 931–946, Aug. 2008.

[8] T. J. Santner, B. J. Williams, W. Notz, and B. J. Williams, *The Design and Analysis of Computer Experiments*, vol. 1. New York, NY, USA: Springer, 2003.

[9] X. Deng, Y. Hung, and C. D. Lin, "Design and analysis of computer experiments," in *Handbook of Research on Applied Cybernetics and Systems Science*. Hershey, PA, USA: IGI Global, 2017, pp. 264–279.

[10] T. J. Santner, B. J. Williams, and W. I. Notz, "Space-filling designs for computer experiments," in *The Design and Analysis of Computer Experiments*. New York, NY, USA: Springer, 2018, pp. 145–200.

[11] M. D. Mckay, R. J. Beckman, and W. J. Conover, "Comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[12] L. Pronzato and W. G. Müller, "Design of computer experiments: Space filling and beyond," *Statist. Comput.*, vol. 22, no. 3, pp. 681–701, May 2012.

[13] M. E. Johnson, L. M. Moore, and D. Ylvisaker, "Minimax and maximin distance designs," *J. Stat. Planning Inference*, vol. 26, no. 2, pp. 131–148, 1990.

[14] K.-T. Fang, D. K. J. Lin, P. Winker, and Y. Zhang, "Uniform design: Theory and application," *Technometrics*, vol. 42, no. 3, pp. 237–248, 2000.

[15] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," 2010, *arXiv:1012.2599*. [Online]. Available: http://arxiv.org/abs/1012.2599

[16] J. Theiler and B. G. Zimmer, "Selecting the selector: Comparison of update rules for discrete global optimization," *Stat. Anal. Data Mining*, vol. 10, no. 4, pp. 211–229, 2017.

[17] H. Wang, B. van Stein, M. Emmerich, and T. Back, "A new acquisition function for Bayesian optimization based on the moment-generating function," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2017, pp. 507–512.

[18] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Global Optim.*, vol. 13, no. 4, pp. 455–492, 1998.

[19] M. J. Sasena, "Flexibility and efficiency enhancements for constrained global design optimization with kriging approximations," Ph.D. dissertation, Dept. Mech. Eng., Univ. Michigan Ann Arbor, MI, USA, 2002.

[20] M. Schonlau, W. J. Welch, and D. R. Jones, "Global versus local search in constrained optimization of computer models," *Lect. Notes-Monograph Ser.*, vol. 34, pp. 11–25, Jan. 1998.

[21] R. B. Gramacy, G. A. Gray, S. Le Digabel, H. K. H. Lee, P. Ranjan, G. Wells, and S. M. Wild, "Modeling an augmented lagrangian for blackbox constrained optimization," *Technometrics*, vol. 58, no. 1, pp. 1–11, Jan. 2016.

[22] M. A. Gelbart, "Constrained Bayesian optimization and applications," Ph.D. dissertation, Dept. Biophys., Harvard Univ., Cambridge, MA, USA, 2015.

[23] V. Picheny, D. Ginsbourger, Y. Richet, and G. Caplin, "Quantile-based optimization of noisy computer experiments with tunable precision," *Technometrics*, vol. 55, no. 1, pp. 2–13, Feb. 2013.

[24] D. R. Jones, "A taxonomy of global optimization methods based on response surfaces," *J. Global Optim.*, vol. 21, no. 4, pp. 345–383, 2001.

[25] D. Zhan, J. Qian, and Y. Cheng, "Balancing global and local search in parallel efficient global optimization algorithms," *J. Global Optim.*, vol. 67, no. 4, pp. 873–892, Apr. 2017.

[26] F. Viana, R. Haftka, and L. Watson, "Why not run the efficient global optimization algorithm with multiple surrogates?" in *Proc. 51st AIAA/ASME/ASCE/AHS/ASC Struct., Struct. Dyn., Mater. Conf. 18th AIAA/ASME/AHS Adapt. Struct. Conf. 12th*, Apr. 2010, p. 3090.

[27] D. Zhan and H. Xing, "Expected improvement for expensive optimization: A review," *J. Global Optim.*, vol. 78, no. 3, pp. 507–544, Nov. 2020.

[28] D. Huang, T. T. Allen, W. I. Notz, and N. Zeng, "Global optimization of stochastic black-box systems via sequential kriging meta-models," *J. Global Optim.*, vol. 34, no. 3, pp. 441–466, Mar. 2006.

[29] S. S. Gupta and K. J. Miescke, "Bayesian look ahead one-stage sampling allocations for selection of the best population," *J. Stat. Planning Inference*, vol. 54, no. 2, pp. 229–244, Sep. 1996.

[30] P. I. Frazier, W. B. Powell, and S. Dayanik, "A knowledge-gradient policy for sequential information collection," *SIAM J. Control Optim.*, vol. 47, no. 5, pp. 2410–2439, Jan. 2008.

[31] P. I. Frazier, "A tutorial on Bayesian optimization," 2018, *arXiv:1807.02811*. [Online]. Available: http://arxiv.org/abs/1807.02811

[32] X. Cai, H. Qiu, L. Gao, P. Yang, and X. Shao, "A multi-point sampling method based on kriging for global optimization," *Struct. Multidisciplinary Optim.*, vol. 56, no. 1, pp. 71–88, Jul. 2017.

[33] X. Cai, H. Qiu, L. Gao, X. Li, and X. Shao, "A hybrid global optimization method based on multiple metamodels," *Eng. Comput.*, vol. 35, no. 1, pp. 71–90, Mar. 2018.

[34] A. Krause and C. S. Ong, "Contextual Gaussian process bandit optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2447–2455.

[35] O. Cappé, A. Garivier, O.-A. Maillard, R. Munos, and G. Stoltz, "Kullback–Leibler upper confidence bounds for optimal sequential allocation," *Ann. Statist.*, vol. 41, no. 3, pp. 1516–1541, 2013.

[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*, vol. 58, no. 5, pp. 3250–3265, May 2012.

[37] D. Russo and B. Van Roy, "Learning to optimize via posterior sampling," *Math. Oper. Res.*, vol. 39, no. 4, pp. 1221–1243, Nov. 2014.

[38] D. Russo and B. Van Roy, "An information-theoretic analysis of thompson sampling," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2442–2471, 2015.

[39] A. Gopalan, S. Mannor, and Y. Mansour, "Thompson sampling for complex online problems," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 100–108.

[40] A. S. Bedi, D. Peddireddy, V. Aggarwal, and A. Koppel, "Efficient large-scale Gaussian process bandits by believing only informative actions," in *Proc. Learn. Dyn. Control*, 2020, pp. 924–934.

[41] D. Calandriello, L. Carratino, A. Lazaric, M. Valko, and L. Rosasco, "Near-linear time Gaussian process optimization with adaptive batching and resparsification," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1295–1305.

[42] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and non-stochastic multi-armed bandit problems," *Found. Trends Mach. Learn.*, vol. 5, no. 1, pp. 1–122, 2012.

[43] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.

[44] H. Robbins, "Some aspects of the sequential design of experiments," *Bull. Amer. Math Soc.*, vol. 58, no. 5, pp. 527–535, 1952.

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

[46] N. Gupta, O.-C. Granmo, and A. Agrawala, "Thompson sampling for dynamic multi-armed bandits," in *Proc. 10th Int. Conf. Mach. Learn. Appl. Workshops*, Dec. 2011.

[47] D. J. Russo, B. Van Roy, A. Kazerouni, I. Osband, and Z. Wen, "A tutorial on thompson sampling," *Found. Trends Mach. Learn.*, vol. 11, no. 1, pp. 1–96, 2018.

[48] J. Gurland and R. C. Tripathi, "A simple approximation for unbiased estimation of the standard deviation," *Amer. Statistician*, vol. 25, no. 4, pp. 30–32, Oct. 1971.

[49] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*. Hoboken, NJ, USA: Wiley, 2021.

[50] S. Krishna Kumar, "On weight initialization in deep neural networks," 2017, *arXiv:1704.08863*. [Online]. Available: http://arxiv.org/abs/1704.08863

[51] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (GPML) toolbox," *J. Mach. Learn. Res.*, vol. 11, pp. 3011–3015, Nov. 2010.

[52] P. I. Frazier and J. Wang, "Bayesian optimization for materials design," in *Proc. Inf. Sci. Mater. Discovery Design*. Cham, Switzerland: Springer, 2016, pp. 45–75.

[53] H. Wang, J. Yuan, and S. H. Ng, "Gaussian process based optimization algorithms with input uncertainty," *IISE Trans.*, vol. 52, pp. 1–17, Apr. 2019.

[54] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," 2009, *arXiv:0912.3995*. [Online]. Available: http://arxiv.org/abs/0912.3995

[55] R. G. Regis, "Stochastic radial basis function algorithms for large-scale optimization involving expensive black-box objective and constraint functions," *Comput. Oper. Res.*, vol. 38, no. 5, pp. 837–853, May 2011.

[56] J. Müller and C. A. Shoemaker, "Influence of ensemble surrogate models and sampling strategy on the solution quality of algorithms for computationally expensive black-box global optimization problems," *J. Global Optim.*, vol. 60, no. 2, pp. 123–144, Oct. 2014.

[57] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Autom. Mag.*, vol. 4, no. 1, pp. 23–33, Mar. 1997.

[58] A. Sakai, D. Ingram, J. Dinius, K. Chawla, A. Raffin, and A. Paques, "PythonRobotics: A Python code collection of robotics algorithms," 2018, *arXiv:1808.10703*. [Online]. Available: http://arxiv.org/abs/1808.10703

[59] O. Brock and O. Khatib, "High-speed navigation using the global dynamic window approach," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 1, May 1999, pp. 341–346.

[60] C.-C. Chou, F.-L. Lian, and C.-C. Wang, "Characterizing indoor environment for robot navigation using velocity space approach with region analysis and look-ahead verification," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 2, pp. 442–451, Feb. 2011.

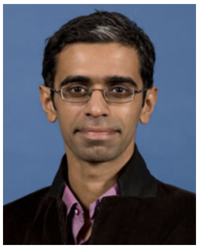
[61] Y. Chai and V. Hassani, "Hybrid collision avoidance with moving obstacles," *IFAC-PapersOnLine*, vol. 52, no. 21, pp. 302–307, 2019.

[62] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proc. IEEE*, vol. 104, no. 1, pp. 148–175, Jan. 2016.

RAJITHA MEKA received the bachelor's degree in mechanical engineering from Acharya Nagarjuna University, India, in 2010, and the master's degree in industrial engineering from the University of Houston, USA, in 2016. She is currently pursuing the Ph.D. degree in mechanical engineering with The University of Texas at San Antonio, USA. She worked with Hyundai Motors, India, from 2010 to 2014. Her research interests include design of experiments, non-parametric regression, and data analytics.

ADEL ALAEDDINI is an Associate Professor of Mechanical Engineering at University of Texas at San Antonio. He obtained his Ph.D. in Industrial and Systems Engineering from Wayne State University. His main research interests include statistical learning in systems modeling, control and optimization in health care, manufacturing, and energy. He has contributed to over 40 peer-reviewed publications in journals such as *IISE Transactions, Production and Operations Management* (POMS), and *Information Sciences*.

CHINONSO OVUEGBE received the bachelor's and master's degrees in mechanical engineering from The University of Texas at San Antonio (UTSA), USA. He worked as a Research Assistant with the Robotics and Motion Laboratory, University of Illinois at Chicago. He is currently working with the Advanced Data Engineering Laboratory, UTSA. His research interests include black-box optimization and navigation in mobile robotics.



PRANAV A. BHOUNSULE received the B.E. degree from Goa University, in 2004, the M.Tech. degree from the Indian Institute of Technology Madras, in 2006, and the Ph.D. degree from Cornell University, in 2012. He is currently an Assistant Professor with the Department of Mechanical and Industrial Engineering, University of Illinois at Chicago. His research interests include model-based and learning-based control methods for legged and humanoid robots. He received the Best Paper Awards in biological inspired robotics at the Climbing and Walking Robots Conference, in 2012, and the ASME Computers and Information in Engineering Conference, in 2019.

PEYMAN NAJAFIRAD (PAUL RAD) received the Ph.D. degree in electrical and computer engineering on cyber analytics from The University of Texas at San Antonio (UTSA). He is currently an Associate Professor of information systems and cyber security with UTSA. He holds 15 U.S. patents on cyber infrastructure, cloud computing, and big data analytics with over 300 product citations by top fortune 500 leading technology companies such as Amazon, Microsoft, IBM, Cisco, Amazon Technologies, HP, and VMware. He has advised over 200 companies on cloud computing and data analytics with over 50 keynote presentations. He serves on the Advisory Board for several startups, the high performance cloud Group Chair at the Cloud Advisory Council (CAC), an OpenStack Foundation Member, the number one Open Source Cloud Software, a San Antonio Tech Bloc Founding Member, and a Children's Hospital of San Antonio Foundation Board Member.

KAI YANG received the M.S. and Ph.D. degrees from the University of Michigan, Ann Arbor. He is currently a Professor with the Department of Industrial and Systems Engineering, and the Co-Director of the Health Informatics and Engineering Group, Wayne State University. He is an expert in the field of quality and reliability engineering, data analytics, and health informatics. He is a fellow of the Institute of Industrial and Systems Engineering.