


Article

Bayesian Optimization Based on K-Optimality

Liang Yan , Xiaojun Duan *, Bowen Liu and Jin Xu

College of Liberal Arts and Sciences, National University of Defense Technology, Changsha 410000, China; yanliang@nudt.edu.cn (L.Y.); bowen_liu12@163.com (B.L.); xujin_nudt@163.com (J.X.)

* Correspondence: xjduan@nudt.edu.cn; Tel.: +86-0731-8700-1605

Received: 11 July 2018; Accepted: 8 August 2018; Published: 9 August 2018



Abstract: Bayesian optimization (BO) based on the Gaussian process (GP) surrogate model has attracted extensive attention in the field of optimization and design of experiments (DoE). It usually faces two problems: the unstable GP prediction due to the ill-conditioned Gram matrix of the kernel and the difficulty of determining the trade-off parameter between exploitation and exploration. To solve these problems, we investigate the K-optimality, aiming at minimizing the condition number. Firstly, the Sequentially Bayesian K-optimal design (SBKO) is proposed to ensure the stability of the GP prediction, where the K-optimality is given as the acquisition function. We show that the SBKO reduces the integrated posterior variance and maximizes the hyper-parameters' information gain simultaneously. Secondly, a K-optimal enhanced Bayesian Optimization (KO-BO) approach is given for the optimization problems, where the K-optimality is used to define the trade-off balance parameters which can be output automatically. Specifically, we focus our study on the K-optimal enhanced Expected Improvement algorithm (KO-EI). Numerical examples show that the SBKO generally outperforms the Monte Carlo, Latin hypercube sampling, and sequential DoE approaches by maximizing the posterior variance with the highest precision of prediction. Furthermore, the study of the optimization problem shows that the KO-EI method beats the classical EI method due to its higher convergence rate and smaller variance.

Keywords: design of experiments; K-optimal design; gaussian processes; bayesian optimization

1. Introduction

Computer simulations are widely used to reproduce the behaviour of systems [1,2] through which their performance can be estimated. Usually surrogate models are introduced to represent the physical realities which can be computationally expensive and are difficult to obtain analytical solutions for. In general, f is denoted as a response function of the real system with input $x \in \mathcal{X} \subseteq \mathbb{R}^D$ and observation $y \in \mathbb{R}$ which follows the form below:

$$y = f(x) + \epsilon. \quad (1)$$

Both x and ϵ are regarded as random parameters. Given N samples ($X = \{X_1, \dots, X_N\} \in \mathcal{X}^N, X_i \in \mathcal{X}$) and corresponding observations ($Y \in \mathbb{R}^N$), the surrogate models can be built to approximate $f(x)$ along with its statistics. The problem of proposing proper X is known as the Design of Experiments (DoE) and it was developed with various mathematical theories. Basically, DoE methods can be categorized as model-free and model-oriented.

The Monte Carlo (MC) [3,4] method is a typical model-free DoE technique and has been widely used in applications. The main advantage of MC method is its simplicity in implementation. However, it converges at a rate of $\mathcal{O}(N^{-1/2})$. As a consequence, a large N is usually needed to obtain an acceptable result and it is unsuitable for large scale high dimensional problems. A widely used way to accelerate the MC method is the quasi-MC technique [5], for example, quasi-MC based on the Sobol

set and Holton set. Another way to substitute the MC method is the Latin Hypercube Sampling (LHS) technique [6] which can generate a near-random sample from a multidimensional distribution with even probability in a pre-defined grid, which ensures the sample is representative of the real variability.

In the context of surrogate models, given a parametric or non-parametric model, we aim to estimate the corresponding parameters or hyper-parameters to achieve the most accurate model. The model-oriented DoE is obtained via some pre-specified criteria. In parameter estimation problems, a popular approach is to consider information-based criteria [7]. An A-optimal design minimizes the trace of the inverse of the Fisher information matrix (FIM) on the unknown parameters, whereas E-, T-, and D-optimal designs maximize the smallest eigenvalue, the trace, and the determinant of the FIM. In the Bayesian framework, the Markov Chain Monte Carlo (MCMC) method [8] is an adaptive DoE technique which utilizes the prior and posterior information; hence, it can focus on points with more important information. The main shortage of MCMC is that it has difficulty determining the acceptance-rejection rate, and it sometimes seems cumbersome because of the long term burn-in period.

Nowadays, more efforts have been devoted to sequential sampling strategies with non-parametric Gaussian process (GP) models [9–11]. The main idea behind those methods is to minimize the times required to call the original system which can be computationally expensive. A learning criterion should be given in prior to obtain samples sequentially. B. Echard et al. [12] have proposed the active learning reliability method which combines the Kriging and Monte Carlo simulation methods (AK-MCS) to iteratively assess the reliability in a more efficient way. Similarly, for continuous functions, Bayesian optimization (BO) [10], despite being designed to solve the optimization problem, also collects samples adaptively. The learning criterion is known as the acquisition function in the BO field. One can optimize the expected improvement (EI) or the probability improvement (PI) over the current best result or the lower/upper confidence bound (LCB/UCB) to decide the next point to be sampled. Unlike the A(E,T,D)-optimal designs which decide the DoE in one step, sequential sampling strategies utilize the information from the observations, hence producing more reliable and accurate results for our research goals.

We note that two main obstacles of the BO exist: first, the optimization of hyper-parameters and the inference of Gaussian processes may fail when the covariance in the Gram matrix of the kernel with respect to current DoE X is ill-conditioned; second, it is usually difficult to determine the trade-off parameter between exploration and exploitation, i.e., local optimization or global search. To solve the first problem, considering a similar situation where a parametric regression problem becomes unstable when the condition number of its design matrix is large, the state-of-the-art K-optimal design [13] which optimizes the condition number could be a reasonable choice. In this paper, a new BO approach is proposed with the condition number of the Gram matrix being introduced as an acquisition function, namely, the Sequentially Bayesian K-optimal design (SBKO). We show that the SBKO actually evolves towards the direction of reducing the integrated posterior variance as well as the direction of maximizing the KL divergence between the prior and posterior distributions of hyper-parameters. No extra parameter is needed to balance the exploration and exploitation, because the SBKO generally tends to fill the whole design space; hence, it is suitable for global search tasks such as approximation and prediction. To solve the second problem, the property of K-optimality can be also used to modify the trade-off parameter, based on the idea that those points leading to smaller condition numbers should be explored. We combine the K-optimality and the classical BO criterion to propose the K-optimal enhanced BO (KO-BO) method. The trade-off parameters are computed automatically according to changes in the condition number brought by the associate points. Compared with the classical BO methods, the KO-BO method is more flexible in determining the trade-off parameter, and it implicitly ensures the stability of the GP model.

The paper is organized as follows. We review Gaussian process regression in Section 2 along with the K-optimal criterion. Our main method and algorithm are in Section 3. At the beginning of Section 3, we present the corresponding acquisition function to incorporate the K-optimal design with the BO

framework, i.e., the Sequentially Bayesian K-optimal design. Secondly, we show the connections of our method with the methods that focus on minimizing the integrated posterior variance and maximizing the information gain of the inference respectively. At the end of Section 3, we propose the K-optimal enhanced Bayesian optimization algorithms to solve the optimization problem. The experimental results are presented in Section 4, and the conclusions follow in Section 5.

2. Brief Review

In this section, we briefly review the general procedure of Gaussian processes and Bayesian optimization approaches, before discussing our novel contributions in Section 3.

2.1. Gaussian Processes

Firstly, we assume that a Gaussian prior is set over function f , i.e., $f \sim \mathcal{GP}(0, k)$, where the mean function is set to be 0 and $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is the kernel function. Given DoE \mathbf{X} and the corresponding observations (Y), we have the likelihood as follows:

$$Y|\mathbf{X} \sim \mathcal{N}(0, K + \sigma^2 I). \quad (2)$$

The predictions ($f(\mathbf{x})$) at a new point ($\mathbf{x} \in \mathcal{X}$) can be sampled from the posterior estimation:

$$\begin{aligned} f(\mathbf{x})|\mathbf{x}, \mathbf{X}, Y &\sim \mathcal{N}(m(\mathbf{x}), v(\mathbf{x})), \\ m(\mathbf{x}) &= K_{\mathbf{x}}^T K_{\sigma}^{-1} Y, \\ v(\mathbf{x}) &= K_{xx} - K_{\mathbf{x}}^T K_{\sigma}^{-1} K_{\mathbf{x}}, \end{aligned} \quad (3)$$

where $K_{\sigma} = K + \sigma^2 I$ with $K = [k(X_i, X_j)]_{ij}$ denoting the $N \times N$ matrix of the covariances at all pairs of training points, and $K_{\mathbf{x}} = [k(X_i, \mathbf{x})]_i$, $K_{xx} = k(\mathbf{x}, \mathbf{x})$ are defined similarly. It is worth noting that the posterior mean estimation ($m(\mathbf{x})$) is just a combination of observations (Y), and the posterior variance is actually independent of Y —it is mainly determined by the kernel function.

2.2. Bayesian Optimization

There are two main aspects in Bayesian optimization. Firstly, the prior assumption about the surrogate model, i.e., Gaussian processes in this paper discussed in previous subsection, must be selected. Secondly, an acquisition function must be constructed based on the model posterior, which can be used to sample the “best” point sequentially. We denote the acquisition function as $a_{ac}(\mathbf{x}) : \mathcal{X} \rightarrow \mathbb{R}^+$, and then the next entry of the expensive original system is determined by an optimization problem, for instance, $\mathbf{x}_{\text{next}} = \arg \max_{\mathbf{x} \in \mathcal{X}} a_{ac}(\mathbf{x})$. In general, the shape of the acquisition function depends on the previous learning results, i.e., the mean and variance of the GP prediction. As mentioned in the introduction, several popular acquisition functions exist. The best value is denoted as $\{x_{\text{best}}^t = \arg \min_{x \in X^t} f(x), y_{\text{best}}^t = f(x_{\text{best}}^t)\}$, and X^t is the DoE at iteration t . Thus, we have the PI, EI, and LCB acquisition functions as follows:

Probability Improvement (PI): The idea of the PI method is to maximize the probability of improving the current best value. Under the GP assumption, it has the following form:

$$a_{\text{PI}}(\mathbf{x}) = P[m(\mathbf{x}) < y_{\text{best}}^t] = \Phi(\gamma(\mathbf{x})), \quad \gamma(\mathbf{x}) = \frac{y_{\text{best}}^t - m(\mathbf{x})}{v(\mathbf{x})}, \quad (4)$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution.

Expected Improvement (EI): Alternatively, we can choose to maximize the expected improvement over the best current value. The explicit mathematical expression is given as follows:

$$a_{\text{EI}}(\mathbf{x}) = \mathbb{E}[(y_{\text{best}}^t - m(\mathbf{x}))^+]. \quad (5)$$

Lower Confidence Bound (LCB): The LCB criterion aims to minimize the regrets over the course of their optimization, and it has the following form:

$$a_{\text{LCB}}(\mathbf{x}) = -m(\mathbf{x}) + \xi_t v(\mathbf{x}), \quad (6)$$

where ξ_t is a constant to trade off the exploration and exploitation.

2.3. K-optimal Design

The K-optimal design is based on the idea of finding a specific set of support points which results in the smallest condition number of the information matrix. The p -th order polynomial regression model is investigated, and a theoretical symmetry DoE in the space $[-1, 1]$ was given in the original paper of Ye [13], where the boundary is usually included. Sándor Baran [14] extended the K-optimal to the correlated processes, i.e., Ornstein–Uhlenbeck processes, in his research. The simulation results in reference [14] show the superiority of restricted K-optimal designs for large covariance parameter values. So, the K-optimal design has potential application in deriving stable and accurate approximations. We embedded the K-optimality into the Bayesian optimization framework, where a sequentially K-optimal design was sampled iteratively. The main methodology and corresponding discussions are given in the next section.

3. Methodology

We restate that our main goal was to choose an optimal design from the predefined input domain which is appropriate for inferring the model in the Bayesian framework. The Gaussian processes was chosen as the model, while the K-optimality was taken into consideration. As reviewed in Section 2, the performance of the Gaussian processes was generally controlled by the covariance functions, i.e., kernels, which are continuous, positive semi-definite functions. It is notable that an inverse term of K_σ exists in Equation (3). When the collected samples are close enough, it will lead to potential failure to calculate K_σ^{-1} as well as the inference of the Gaussian processes, although a nugget term $\sigma^2 I$ was added.

In this work, we focused on an experimental design that ensured the correctness and accurateness of Bayesian inference. If the condition number of K in Equation (3) is bounded by a relative small constant, then the inference of Gaussian processes can be always achieved. The Sequentially Bayesian K-optimal design (SBKO) was then proposed which is straightforward and simple to present. Like the classical BO methods, the acquisition function is given as $a_K = \kappa(K_\sigma(\mathbf{x}; \theta))$, where $\kappa(\cdot)$ stands for the condition number and $K_\sigma(\mathbf{x}; \theta)$ is the updated covariance matrix, with θ being the hyper-parameters of the kernel function. The term θ can be omitted in the following sections without causing any misunderstanding. Hence, $K_\sigma(\mathbf{x})$ is defined as follows, and the next point (\mathbf{x}_{next}) is sampled by solving the optimization problem:

$$\begin{aligned} \mathbf{x}_{\text{next}} &= \arg \min_{\mathbf{x} \in \mathcal{X}} a_K(\mathbf{x}), \\ K_\sigma(\mathbf{x}) &= \begin{bmatrix} K_\sigma & K_x \\ K_x^T & K_{xx} + \sigma^2 \end{bmatrix}. \end{aligned} \quad (7)$$

There are two main concerns about the minimization of $a_K(\mathbf{x})$. On one hand, the condition number and its optimization problem are not convex. Hence, non-smooth algorithms, such as the Dividing RECTangles (DIRECT) algorithm [15] or the genetic algorithm, are used to solve Equation (7). A few works in the literature focused on optimizing the condition number under certain conditions. P. Maréchal and J. J. Ye investigated the optimization of condition number over a compact convex subset of the cone of symmetric positive semi-definite $n \times n$ matrices in 2009 [16], while X. J. Chen, R. S. Womersley, and J. J. Ye investigated the minimization of the condition number of a Gram matrix

of the polynomial regression model in 2011 [17]. Both of the works introduced the idea of the Clarke generalized gradient which can accelerate the optimization process.

On the other hand, the hyper-parameters θ control the value of $K_\sigma(\mathbf{x})$; hence, one can consider the MLE (Maximum Likelihood Estimation) or MAP (Maximum A Posterior) of θ . Note that the data are sampled sequentially, which implicitly implies that the MLE (MAP) of θ satisfies the criterion with current samples, and it usually does not hold when a new point is added. Instead of using the point estimate of θ , one can consider the general technique [18,19] of integrating the acquisition function $\bar{a}_K(\mathbf{x})$ over the posterior distribution:

$$\bar{a}_K(\mathbf{x}) = \int a_K(\mathbf{x})p(\theta|\mathbf{X}, Y)d\theta, \tag{8}$$

where $p(\theta|\mathbf{X}, Y) \propto p(\theta)p(Y|\mathbf{X}, \theta)$ is the posterior distribution with the DoE (\mathbf{X}), observations (Y) and prior distribution of the hyper-parameters ($p(\theta)$). The expectation in Equation (8) generally accounts for uncertainty in the hyper-parameters or the average level of $a_K(\mathbf{x})$. $\bar{a}_K(\mathbf{x})$ can be approximated by the MC estimate, where the samples of θ from the posterior distribution can be obtained by the MCMC procedure. In this work, the efficient slice sampling approach proposed by I. Murray [20] was introduced to obtain samples of θ from the posterior distribution.

In fact, minimizing the condition number has more significance than generating stable inference for the GP model. In the next subsections, we show that minimizing $a_K(\mathbf{x})$ has a close connection with the prediction uncertainty as well as the information gain.

3.1. Connection to Optimization of the Integrated Posterior Variance

The prediction uncertainty is given as the posterior variance ($v(\mathbf{x})$) in Equation (3). We chose to integrate the posterior variance into the input domain instead of the approximation itself; the integration accounts for every point in the whole domain, and it also quantifies the uncertainty which provides the quality of the approximation. We let the input space (\mathcal{X}) be a first-countable space equipped with a strictly Borel measure (μ), and represented $k(\mathbf{x}, \mathbf{x}')$ as a convergent series according to Mercer’s theorem [21]:

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &= \sum \lambda_i \psi_i(\mathbf{x})\psi_i(\mathbf{x}'), \\ \text{s.t. } \int |k(\mathbf{x}, \mathbf{x})|d\mu &< \infty, \quad \sum \lambda_i^2 < \infty, \\ \iint k(\mathbf{x}, \mathbf{x}')g(\mathbf{x})g(\mathbf{x}')d\mu &\geq 0, \quad \forall g \in L_2(\mathcal{X}), \end{aligned} \tag{9}$$

where $\{\psi_i(\mathbf{x}), i \geq 1\}$ forms an orthonormal basis of $L_2(\mathcal{X})$. Then, the next sample was obtained by minimizing its corresponding integrated posterior variance (IPV), i.e.,

$$\begin{aligned}
 \mathbf{x}_{\text{next}} &= \arg \min_{\mathbf{x} \in \mathcal{X}} \int v(\mathbf{x}') d\mu(\mathbf{x}') \\
 &= \arg \min_{\mathbf{x} \in \mathcal{X}} \int \left(K_{\mathbf{x}'\mathbf{x}'} - \begin{bmatrix} K_{\mathbf{x}'}^T & K_{\mathbf{x}\mathbf{x}'} \end{bmatrix} \begin{bmatrix} K_{\sigma} & K_{\mathbf{x}} \\ K_{\mathbf{x}}^T & K_{\mathbf{x}\mathbf{x}} + \sigma^2 \end{bmatrix}^{-1} \begin{bmatrix} K_{\mathbf{x}'} \\ K_{\mathbf{x}\mathbf{x}'} \end{bmatrix} \right) d\mu(\mathbf{x}') \\
 &= \arg \max_{\mathbf{x} \in \mathcal{X}} \int \begin{bmatrix} K_{\mathbf{x}'}^T & K_{\mathbf{x}\mathbf{x}'} \end{bmatrix} \begin{bmatrix} K_{\sigma} & K_{\mathbf{x}} \\ K_{\mathbf{x}}^T & K_{\mathbf{x}\mathbf{x}} + \sigma^2 \end{bmatrix}^{-1} \begin{bmatrix} K_{\mathbf{x}'} \\ K_{\mathbf{x}\mathbf{x}'} \end{bmatrix} d\mu(\mathbf{x}') \\
 &= \arg \max_{\mathbf{x} \in \mathcal{X}} \sum \lambda_i^2 \begin{bmatrix} \boldsymbol{\psi}_i^T & \boldsymbol{\psi}_i(\mathbf{x}) \end{bmatrix} \begin{bmatrix} K_{\sigma} & K_{\mathbf{x}} \\ K_{\mathbf{x}}^T & K_{\mathbf{x}\mathbf{x}} + \sigma^2 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{\psi}_i \\ \boldsymbol{\psi}_i(\mathbf{x}) \end{bmatrix} \tag{10} \\
 &= \arg \max_{\mathbf{x} \in \mathcal{X}} \sum \lambda_i^2 \boldsymbol{\alpha}^T \left(\begin{bmatrix} K_{\sigma}^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{1}{v(\mathbf{x}) + \sigma^2} \begin{bmatrix} K_{\sigma}^{-1} K_{\mathbf{x}} K_{\mathbf{x}}^T K_{\sigma}^{-1} & -K_{\sigma}^{-1} K_{\mathbf{x}} \\ -K_{\mathbf{x}}^T K_{\sigma}^{-1} & 1 \end{bmatrix} \right) \boldsymbol{\alpha} \\
 &= \arg \max_{\mathbf{x} \in \mathcal{X}} \sum \lambda_i^2 \boldsymbol{\alpha}^T \left(\begin{bmatrix} K_{\sigma}^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{\boldsymbol{\beta}_{\sigma} \boldsymbol{\beta}_{\sigma}^T}{v(\mathbf{x}) + \sigma^2} \right) \boldsymbol{\alpha} \\
 &= \arg \max_{\mathbf{x} \in \mathcal{X}} \frac{1}{v(\mathbf{x}) + \sigma^2} \sum (\lambda_i \boldsymbol{\alpha}^T \boldsymbol{\beta}_{\sigma})^2,
 \end{aligned}$$

where $\boldsymbol{\alpha} = [\boldsymbol{\psi}_i^T, \boldsymbol{\psi}_i(\mathbf{x})]^T$, $\boldsymbol{\beta}_{\sigma} = [K_{\sigma}^{-1} K_{\mathbf{x}}, -1]^T$, $\boldsymbol{\psi}_i = [\boldsymbol{\psi}_i(X_1), \dots, \boldsymbol{\psi}_i(X_N)]^T$. The fourth equation was obtained by the orthonormality of $\boldsymbol{\psi}_i(\mathbf{x})$, and we assumed that the hyper-parameters were fixed for simplicity. The last term is not easy to calculate; however, we investigated its upper bound which reflects the maximum reduction in the IPV:

$$\begin{aligned}
 \frac{1}{v(\mathbf{x}) + \sigma^2} \sum (\lambda_i \boldsymbol{\alpha}^T \boldsymbol{\beta}_{\sigma})^2 &= \frac{1}{v(\mathbf{x}) + \sigma^2} \sum | \langle \lambda_i \boldsymbol{\alpha}, \boldsymbol{\beta}_{\sigma} \rangle |^2 \\
 &\leq \frac{1}{v(\mathbf{x}) + \sigma^2} \sum \langle \lambda_i \boldsymbol{\alpha}, \lambda_i \boldsymbol{\alpha} \rangle \langle \boldsymbol{\beta}_{\sigma}, \boldsymbol{\beta}_{\sigma} \rangle \\
 &\leq \frac{\lambda_{\max} \boldsymbol{\beta}_{\sigma}^T \boldsymbol{\beta}_{\sigma}}{v(\mathbf{x}) + \sigma^2} \sum \lambda_i \boldsymbol{\alpha}^T \boldsymbol{\alpha} \tag{11} \\
 &= \lambda_{\max} \frac{\boldsymbol{\beta}_{\sigma}^T \boldsymbol{\beta}_{\sigma}}{v(\mathbf{x}) + \sigma^2} \text{tr}(K_{\sigma}(\mathbf{x})) \triangleq \text{IPV}_{\text{upper}},
 \end{aligned}$$

where λ_{\max} is the maximum $\{\lambda_i\}$ and $\text{tr}(\cdot)$ represents the trace of a matrix. The first inequality was derived by the Cauchy–Schwarz inequality, while the last equality was given with the help of Equation (9). Suppose the isotropic kernel functions, for example, the isotropic squared exponential covariance function or the isotropic Matérn covariance function, are used in the Gaussian process model, then $K_{\mathbf{x}\mathbf{x}}$ is an invariant, as well as the term $\text{tr}(K_{\sigma}(\mathbf{x}))$.

If we recall the SBKO criterion demonstrated in Equation (7), we have the following results:

$$\begin{aligned}
 a_{\text{K}}(\mathbf{x}) &= \kappa(K_{\sigma}(\mathbf{x})) = \kappa(K_{\sigma}(\mathbf{x})^{-1}) \\
 &= \kappa \left(\begin{bmatrix} K_{\sigma}^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{\boldsymbol{\beta}_{\sigma} \boldsymbol{\beta}_{\sigma}^T}{v(\mathbf{x}) + \sigma^2} \right). \tag{12}
 \end{aligned}$$

We let $s(\mathbf{x})_1 \geq s(\mathbf{x})_2 \geq \dots \geq s(\mathbf{x})_{N+1}$ be the singular values of $K_{\sigma}(\mathbf{x})$, while $s_1 \geq s_2 \geq \dots \geq s_N$ were those of K_{σ} . Note that we have the Cauchy’s interlacing theorem, which states that

$$s(\mathbf{x})_1 \geq s_1 \geq s(\mathbf{x})_2 \geq s_2 \geq \dots \geq s_N \geq s(\mathbf{x})_{N+1}. \tag{13}$$

Hence, it was derived that

$$\begin{aligned}
 a_K(x) &= \frac{1}{s(x)_{N+1}} \left/ \left[\text{tr} \left(\begin{bmatrix} K_\sigma^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{\beta_\sigma \beta_\sigma^T}{v(x) + \sigma^2} \right) - \sum_{i=2}^{N+1} \frac{1}{s(x)_i} \right] \right. \\
 &\geq \frac{1}{s_N} \left/ \left[\frac{\beta_\sigma^T \beta_\sigma}{v(x) + \sigma^2} + \sum_{i=1}^N \frac{1}{s_i} - \sum_{i=2}^{N+1} \frac{1}{s(x)_i} \right] \right. \\
 &\geq \frac{1}{s_N} \left/ \left[\frac{\beta_\sigma^T \beta_\sigma}{v(x) + \sigma^2} \right] \right. .
 \end{aligned} \tag{14}$$

Similarly, we have

$$\begin{aligned}
 a_K(x) &= \left[\text{tr} \left(\begin{bmatrix} K_\sigma^{-1} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} + \frac{\beta_\sigma \beta_\sigma^T}{v(x) + \sigma^2} \right) - \sum_{i=1}^N \frac{1}{s(x)_i} \right] \left/ \frac{1}{s(x)_1} \right. \\
 &\geq s_1 \left[\frac{\beta_\sigma^T \beta_\sigma}{v(x) + \sigma^2} + \sum_{i=1}^N \frac{1}{s_i} - \sum_{i=1}^N \frac{1}{s(x)_i} \right] \\
 &\geq s_1 \left[\frac{\beta_\sigma^T \beta_\sigma}{v(x) + \sigma^2} \right] .
 \end{aligned} \tag{15}$$

According to Equations (14) and (15), we have the boundaries of $\beta_\sigma^T \beta_\sigma v(x) + \sigma^2$ as follows:

$$\frac{\beta_\sigma^T \beta_\sigma}{v(x) + \sigma^2} \in \left[\frac{1}{s_N a_K(x)}, \frac{a_K(x)}{s_1} \right] . \tag{16}$$

By considering Equations (11) and (16) together, we obtained the lower bound of IPV_{upper} as

$$IPV_{upper} = \lambda_{\max} \frac{\beta_\sigma^T \beta_\sigma}{v(x) + \sigma^2} \text{tr}(K_\sigma(x)) \geq \frac{\lambda_{\max} \text{tr}(K_\sigma(x))}{s_N a_K(x)} . \tag{17}$$

The lower bound of IPV_{upper} is inversely proportional to $a_K(x)$, so the new sample x that minimizes the condition number also maximizes the reduction of the IPV.

3.2. Connection to Optimization of the KL-Divergence

Equation (7) presents a simple way to incorporate the K-optimal design and BO framework. Such a procedure ensures the success of Bayesian inference; however, it is notable that the covariance matrix (K) alone does not reflect how well the new sample supports the inference of model. We used Kullback–Leibler (KL) divergence [22] from the posterior to prior as a metric to illustrate the performance of the new sample, as follows:

$$a_{KL}(x) = -D_{KL}(p(\theta|\mathbf{X}, Y) || p(\theta|\mathbf{X}, Y, x, y)) = - \int p(\theta|\mathbf{X}, Y) \log \frac{p(\theta|\mathbf{X}, Y)}{p(\theta|\mathbf{X}, Y, x, y)} d\theta, \tag{18}$$

where $p(\theta|\mathbf{X}, Y, x, y)$ is the posterior distribution given DoE $\{\mathbf{X}, x\}$ and a new point is sampled such that $x_{\text{next}} = \arg \max_{x \in \mathcal{X}} a_{KL}(x)$. Unlike the entropy search acquisition function which maximizes the expected reduction in the negative differential entropy ($H[p(x_{\text{best}}|\mathbf{X}, Y)]$) *w.r.t* the current best location (x_{best}), Equation (18) aims to reduce the uncertainty of the hyper-parameters, i.e., the uncertainty of the inference. We chose the inclusive direction of the KL divergence since we had $p(\theta|\mathbf{X}, Y)$ known as the prior at each step, and the KL-divergence explicitly quantified the additional information captured in $p(\theta|\mathbf{X}, Y, x, y)$ relative to the previous distribution, where a larger negative KL divergence reflects a greater information gain about θ upon the possible new design ($\{x, y\}$).

We note that the new observation (y) cannot be attained before being actually sampled at the point, so the prediction $m(x)$ in Equation (3) is introduced to substitute the unknown y . However, $m(x)$ has

high uncertainty at some points; hence, Equation (18) becomes unsuitable for inference. An analogue technique is taking the expectation over the prediction which is presented as follows:

$$\bar{a}_{\text{KL}}(\mathbf{x}) = - \int p(\boldsymbol{\theta}|\mathbf{X}, Y) \left(\int p(f|\mathbf{X}, Y, \mathbf{x}, \boldsymbol{\theta}) \log \frac{p(\boldsymbol{\theta}|\mathbf{X}, Y)}{p(\boldsymbol{\theta}|\mathbf{X}, Y, \mathbf{x}, f)} df \right) d\boldsymbol{\theta}. \tag{19}$$

The above acquisition function was introduced by Kim et al. [23], where $\bar{a}_{\text{KL}}(\mathbf{x})$ is interpreted as the mutual information [24] between the parameter variables $\boldsymbol{\theta}$ and the predictive observation $f(\mathbf{x})$ (which is also a random variable given \mathbf{x}) conditional upon candidate design \mathbf{x} , i.e., $\bar{a}_{\text{KL}}(\mathbf{x}) = I(\boldsymbol{\theta}; f|\mathbf{x})$. Then, the next sample is obtained according to the criterion $\mathbf{x}_{\text{next}} = \arg \max_{\mathbf{x} \in \mathcal{X}} \bar{a}_{\text{KL}}(\mathbf{x})$, i.e.,

$$\begin{aligned} \mathbf{x}_{\text{next}} &= \arg \max_{\mathbf{x} \in \mathcal{X}} - \int p(\boldsymbol{\theta}|Y) \left(\int p(f|Y, \boldsymbol{\theta}) \left(\log p(\boldsymbol{\theta}|Y) - \log p(\boldsymbol{\theta}|Y, f) \right) df \right) d\boldsymbol{\theta} \\ &= \arg \max_{\mathbf{x} \in \mathcal{X}} \int p(\boldsymbol{\theta}|Y) \left(\int p(f|Y, \boldsymbol{\theta}) \left(\log p(f|Y, \boldsymbol{\theta}) - \log p(f|Y) \right) df \right) d\boldsymbol{\theta} \\ &= \arg \max_{\mathbf{x} \in \mathcal{X}} H \left[\mathbb{E}_{\boldsymbol{\theta}|Y} (p(f|Y, \boldsymbol{\theta})) \right] - \mathbb{E}_{\boldsymbol{\theta}|Y} [H(p(f|Y, \boldsymbol{\theta}))] \\ &= \arg \max_{\mathbf{x} \in \mathcal{X}} c \mathbb{E}_{\boldsymbol{\theta}|Y} [H(p(f|Y, \boldsymbol{\theta}))] \end{aligned} \tag{20}$$

where \mathbf{x}, \mathbf{X} are omitted for simplicity, and $H(\cdot)$ represents the differential entropy. The second equation is derived from the fact that $p(\boldsymbol{\theta}|Y)$ does not depend on \mathbf{x} . Notice that $H(\cdot)$ is a concave function; hence, we have the last equation with a constant $c > 0$. Now that $H(\mathcal{N}(\mu, \sigma^2)) = 1/2 \log(2\pi e \sigma^2)$, which is a strictly monotonically increasing function on σ^2 , given Equation (3), we can rewrite Equation (20) as follows:

$$\mathbf{x}_{\text{next}} = \arg \max_{\mathbf{x} \in \mathcal{X}} \mathbb{E}_{\boldsymbol{\theta}|Y} [v(\mathbf{x})]. \tag{21}$$

The right-hand side of Equation (21) is the average uncertainty of prediction over all possible parameters (models). Specifically, we investigated $v(\mathbf{x})$ only with fixed hyper-parameters ($\boldsymbol{\theta}$) for simplicity. Using Equation (16), we considered the lower bound of $a_{\text{K}}(\mathbf{x})$ as follows:

$$a_{\text{K}}(\mathbf{x}) \geq s_1 \frac{\boldsymbol{\beta}_\sigma^T \boldsymbol{\beta}_\sigma}{v(\mathbf{x}) + \sigma^2} \geq \frac{s_1}{K_{\text{xx}} + \sigma^2}. \tag{22}$$

The above lower bound is an invariant if the isotropic kernel function is introduced. Since $v(\mathbf{x}) \leq K_{\text{xx}}$ (see Equation (3)), it is likely to be reached when $v(\mathbf{x})$ is maximized. Hence, the minimization of $a_{\text{K}}(\mathbf{x})$ tends to optimize the KL-divergence between the prior and posterior distributions.

3.3. K-Optimal Enhanced Bayesian Optimization

Compared with the classical BO methods which aim to solve the optimization problems, the *optimization* process in the previous method focuses on the condition number of $K_\sigma(\mathbf{x})$. Actually, the DoE generated by our method tend to be scattered throughout the whole design space (the K-optimal designs are called support points in the original paper); hence, they are suitable for the global prediction behaviour of the Gaussian process model. Based on the previous discussion, the idea of K-optimality can be used to refine the classical BO methods. In this work, we focused our research on comparison with the EI criterion, which generally outperforms the PI criterion and is simpler than the LCB criterion.

The K-optimal was introduced to enhance the performance of Bayesian optimization for the following reason. It is well-known that balancing the trade-off between exploiting (where the prediction is expected to be high) and exploring (where the prediction uncertainty is high) is a key problem in the BO framework. For instance, an additional parameter, ζ , is introduced for the EI algorithm, where $m(\mathbf{x})$ is replaced by $m(\mathbf{x}) + \zeta$ in both Equations (4) and (5). The value of ζ determines the

range of exploration, i.e., the anticipated improvement is likely to be greater than ξ . The choice of ξ is an open problem for researchers, and there is no universal rule to determine the optimal value of ξ . An unsuitable ξ for the EI algorithm sometimes leads to the local optimum, whose information will be strengthened as the data number increases. Notice that since the K-optimality naturally forces the samples to spread sparsely in the design space, it may be an alternative way to perform exploration.

The natural way of introducing the K-optimality to the classical BO framework is to take account of the criteria together, where we tend to choose the one that leads to a smaller condition number when several points have comparable performances in terms of the EI criterion. Given two points x, x' and corresponding classical acquisition function $a_{cl}(\cdot)$, as well as $a_K(\cdot)$ defined in Equation (7), we have to decide which point should be sampled for four different situations considering the acquisition function and K-optimality simultaneously, which is illustrated in Table 1.

Table 1. Four situations considering the acquisition function and K-optimality simultaneously.

	$a_{cl}(x) < a_{cl}(x')$	$a_{cl}(x) \geq a_{cl}(x')$
$a_K(x) < a_K(x')$	Not decided	x
$a_K(x) \geq a_K(x')$	x'	Not decided

The above table shows that there two situations exist where the sample strategy remains unclear to us when combining the classical BO criteria and the K-optimality directly; hence, a new method to balance the two factors is needed. Since we aimed to solve the optimal problems in the Bayesian framework, the classical BO criteria were regarded as the main factors that indicate the direction of the next sample, while K-optimality was used to tune the strength of exploration. Basically, we have stronger belief in the point that improves the optimization results while maintaining the validity of the inference.

We used the condition number κ as the indicator of the strength of exploration. In this work, $\kappa(K_\sigma(x))$ was used to show the goodness of the point for the next Bayesian inference; thus, the exploration was based on the following idea: if the next point to be sampled leads to a large condition number, then we should consider extending the exploration range. We considered the analytic expression of the EI acquisition function as follows:

$$\begin{aligned}
 a_{EI}(x) &= (y_{best}^t - m(x) - \xi)\Phi(Z) + v(x)\phi(Z) \\
 Z &= \frac{y_{best}^t - m(x) - \xi}{v(x)},
 \end{aligned}
 \tag{23}$$

where $\phi(\cdot)$ denotes the probability density function of the standard normal distribution. We then investigated how ξ affects the value of $a_{EI}(x)$ by calculating the derivative $\partial a_{EI}(x) / \partial \xi$:

$$\begin{aligned}
 \frac{\partial a_{EI}(x)}{\partial \xi} &= -\Phi(Z) + (y_{best}^t - m(x) - \xi) \frac{\partial \Phi(Z)}{\partial \xi} + v(x) \frac{\partial \phi(Z)}{\partial \xi} \\
 &= -\Phi(Z) - Z\phi(Z) + Z\phi(Z) = -\Phi(Z) < 0.
 \end{aligned}
 \tag{24}$$

Hence, $a_{EI}(x)$ is a monotonically decreasing function on ξ . Since we aimed to enlarge the utility of the point which leads to better inference (smaller condition number), the simplest way was to replace ξ with κ . However, note that the condition number κ is always greater than 1, and usually, it is a relative large number, so firstly, we normalized κ from $[1, \infty)$ to $(0, 1)$ with the help of the following function:

$$\xi(\kappa) = \frac{\log \kappa}{\log \kappa + c \log \kappa_T},
 \tag{25}$$

where κ_T is the threshold of the condition number (say, greater than 1000 as a rule of thumb), and c is a constant that controls the shape of $\xi(\kappa)$, as illustrated in Figure 1. For example, let $\kappa_T = 1000$ and

$\zeta(\kappa_T) = 0.8$; then, we have $c = 0.25$ displayed as the blue line in Figure 1. Actually, c determines the exploration strength *w.r.t* κ_T . A smaller c leads to a larger $\zeta(\kappa)$, and we are less likely to trust the point that results in κ_T . On the other hand, the smaller the κ_T is, the fewer the points we can accept in practice. Compared with the classical EI algorithm, $\zeta(\kappa)$ is more flexible because it automatically updates its value.

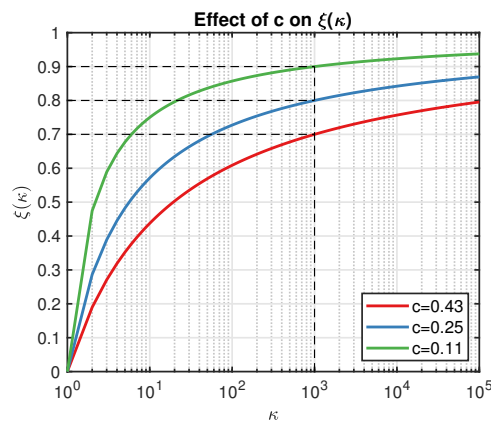


Figure 1. Demonstration of the effect of c on the shape of $S(\kappa)$.

Several interesting features for the above methodology exist. Firstly, if a point is far away from the current exploitation region but it may result in better inference for the model, then the probability to keep it as the next sample still exists. Secondly, if a point can improve the current best value, however it may be derived from a false inference, then we are likely to dump the point by shrinking its utility. We demonstrate these properties in Section 4.2.

4. Experimental Results

The main theories and methodologies of this work are tested in this section. The first subsection demonstrates the Sequentially Bayesian K-optimal design for approximation problems, while the second one focuses on the comparison of the K-optimal enhanced Bayesian optimization problems.

4.1. Sequentially Bayesian K-Optimal Design for Prediction Problem

We proposed a simple acquisition function which is used to sequentially generate a DoE which ensures the validity of Bayesian inference. We consider three examples to demonstrate our SBKO method. Firstly, we implement our method on the one-dimensional Viana function [25] and the two-dimensional Branin function [26], along with comparison to alternative sampling methods. An application with the Borehole function model [27] is presented thereafter.

All of the following experiments were implemented with the Matérn 5/2 kernel, and there were three different DoE methodologies adopted for prediction comparison: the Monte-Carlo sampling strategy, the LHS method [6], and the sequential experimental design based on maximizing the posterior variance (MPV). Four measures were introduced to evaluate the performance of each method, namely, the leave-one-out cross validation error (LOO-CV), the integrated posterior variance (IPV), the root mean squared error (RMSE), and the condition number (CN). They were computed by the following formulas:

$$\begin{aligned}
 \text{LOO-CV} &= \frac{1}{N} \sum_{i=1}^N \left(m^{-i}(X_i) - Y_i \right)^2 \\
 \text{IPV} &= \int v(x) d\mu(x) \\
 \text{RMSE} &= \frac{1}{N'} \sum_{i=1}^{N'} \left(m(X_i^t) - Y_i^t \right)^2 \\
 \text{CN} &= \kappa(K_\sigma),
 \end{aligned}
 \tag{26}$$

where $m^{-i}(\cdot), m(\cdot)$ represents the prediction of the GP model given $\{X, Y\}$ except $\{X_i, Y_i\}$, and $\{X, Y\}$ respectively, while $\{X_i^t, Y_i^t, i = 1, \dots, N'\}$ was the test data set, $v(x)$ and K_σ was defined with Equation (3). The LOO-CV reflected the expected level of fit of the Gaussian process model, while the IPV estimated the overall uncertainty of prediction, and the RMSE measured the average difference between the real response and the prediction. Additionally, the CN, which we care about most in this work, showed us the robustness of Bayesian inference. Furthermore, noted that 10,000 points of independent test data were introduced to calculate the RMSE. Although, this is generally impossible for practical problems, we applied it for research purposes.

The Gaussian process models were constructed with the *gpml* toolbox [28] by Carl C. Rasmussen, and the optimizations of condition number in the SBKO were performed with the DIRECT algorithm [15] of the NLOPT library [29]. The main results were as follows:

Example 1. Viana function [25]

$$y = \frac{10 \cos(2x) + 15 - 5x + x^2}{50} + \varepsilon, \quad x \sim \mathcal{U}(-3, 3), \quad \varepsilon \sim \mathcal{N}(0, 0.01^2).
 \tag{27}$$

The number of all DoEs for Example 1 was set as 7, and specifically, a randomly sampled point was given as the initial experimental design for the SBKO and the sequential MPV design. Each of the four methods was replicated 100 times. Table 2 presents the means and standard deviations of the LOO-CV, IPV, RMSE, and CN based on 4 DoEs. It is clear that the SBKO can always lead to a smaller condition number. On the other hand, considering the LOO-CV/IPV/RMSE, the SBKO also showed the best performance with the smallest standard deviation. This means that the SBKO has the most stable performance for repeatable simulations.

Table 2. Means and standard deviations of the leave-one-out cross validation error (LOO-CV), the integrated posterior variance (IPV), the root mean squared error (RMSE) and the condition number (CN) based on 4 DoEs in Example 1, where the number in the parentheses is the standard deviation and the bold numbers represent the best outcomes.

	LOO-CV	IPV	RMSE	CN
MC	0.5528 (0.3874)	0.0778 (0.1100)	0.1644 (0.1734)	1.2651×10^5 (8.4871×10^5)
LHS	0.7557 (0.2217)	0.0566 (0.1032)	0.0837 (0.1354)	3.2924 (0.9513)
MPV	0.6242 (0.1916)	0.0104 (0.0044)	0.0254 (0.0078)	2.5248 (0.5519)
BKO	0.6771 (0.1711)	0.0093 (0.0025)	0.0241 (0.0072)	2.2601 (0.4658)

Example 2. Branin function [26]

$$\begin{aligned}
 y &= \left(15x_2 - \frac{5.1}{4\pi^2} (15x_1 - 5)^2 + \frac{5}{\pi} (15x_1 - 5) - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) \cos(15x_1 - 5) + 10 + \varepsilon, \\
 x_i &\sim \mathcal{U}(0, 1), \quad i = 1, 2, \quad \varepsilon \sim \mathcal{N}(0, 0.01^2).
 \end{aligned}
 \tag{28}$$

We ran the experiments with similar setups for Example 2, only changing the number of DoEs to 15. The means and standard deviation of the LOO-CV, IPV, RMSE, and CN derived by 100 independent

simulations are given in Table 3. It is clear that the SBKO generally outperforms the other three DoEs. The SBKO design leads to the smallest condition number; it also has the potential ability to lower the IPV, as discussed in Section 3.2. Because the MPV focuses on the point with maximum posterior variance, the experimental design tends to distribute sparsely in the whole domain which improves its global accuracy. We note that the MPV and the SBKO have comparable performances, and the reason that the SBKO is generally slightly better may be that the Bayesian inference with the SBKO is more robust than the MPV.

Table 3. Means and standard deviations of the LOO-CV, IPV, RMSE, and CN based on 4 DoEs in Example 2 where the number in the parentheses is the standard deviation. Bold numbers represent the best outcomes.

	LOO-CV	IPV	RMSE	CN
MC	0.1832 (0.1416)	300.0422 (219.0879)	23.0570 (8.8187)	7.4361×10^4 (2.2117×10^5)
LHS	0.3032 (0.2120)	216.2945 (223.1399)	20.6949 (9.0519)	1.5387×10^3 (645.5456)
MPV	0.1825 (0.1806)	109.4216 (103.9399)	12.8602 (2.6683)	195.8292 (27.0753)
BKO	0.1553 (0.1114)	97.4508 (42.3682)	12.2822 (2.8295)	113.3958 (20.5164)

Example 3. Borehole function [27]

The Borehole function models the flow of water through a borehole drilled from the ground surface through two aquifers. Although it is an eight-dimensional problem, it can be evaluated very fast; hence, it is commonly used test model. The explicit expression is

$$y = \frac{2\pi T_u(H_u - H_l)}{\ln(r/r_w) \left(1 + \frac{2LT_u}{\ln(r/r_w)r_w^2 K_w} + \frac{T_u}{T_l}\right)} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, 2^2), \tag{29}$$

and the input variables and their distributions are given in Table 4 as follows:

Table 4. The input variables of the Borehole function and their usual distributions.

	Input	Distribution	Unit
r_w	radius of borehole	$\mathcal{U}(0.05, 0.15)$	m
r	radius of influence	$\mathcal{U}(100, 50,000)$	m
T_u	transmissivity of upper aquifer	$\mathcal{U}(63,070, 115,600)$	m ² /yr
H_u	potentiometric head of upper aquifer	$\mathcal{U}(990, 1110)$	m
T_l	transmissivity of lower aquifer	$\mathcal{U}(63.1, 116)$	m ² /yr
H_l	potentiometric head of lower aquifer	$\mathcal{U}(700, 820)$	m
L	length of borehole	$\mathcal{U}(1120, 1680)$	m
K_w	hydraulic conductivity of borehole	$\mathcal{U}(9855, 12,045)$	m/yr

The number of the DoEs was set as 100 for the Borehole function. The comparison of the four different DoEs is given in Table 5. Obviously, the SBKO design still possesses the smallest condition number, however interestingly the LHS design generally outperforms the others *w.r.t* the LOO-CV, IPV, and RMSE. Note that the condition numbers of the four designs are approximately equal to 1, and the reason for this is that the 100 samples are located extremely sparsely in the eight-dimensional space. Since the four designs all lead to valid Bayesian inference, this limits the potential advantage of our SBKO method, such as in the Examples 1 and 2. We discussed that the SBKO design usually include points on the boundary, hence it does not distribute as evenly as the LHS design which may be the main reason that it performs a little worse than the LHS design.

Table 5. Means and standard deviations of the LOO-CV, IPV, RMSE, and CN based on 4 DoEs in Example 3, where the numbers in parentheses are the standard deviations and the bold numbers are the best results.

	LOO-CV	IPV	RMSE	CN
MC	0.0024 (0.0012)	8.5170 (1.0624)	2.0212 (0.3855)	$1 + 1.0808 \times 10^{-6}$ (2.9431×10^{-7})
LHS	0.0024 (0.0010)	8.0039 (0.7151)	1.9091 (0.2877)	$1 + 1.0808 \times 10^{-6}$ (2.9431×10^{-7})
MPV	0.0033 (0.0028)	11.2356 (2.8431)	2.7163 (0.5043)	$1 + 6.3612 \times 10^{-7}$ (2.6469×10^{-7})
BKO	0.0026 (0.0013)	8.8565 (0.4842)	2.2822 (0.2895)	$1 + 5.6624 \times 10^{-8}$ (1.0439×10^{-7})

The above three examples illustrate the potential usage range of the SBKO method. Basically, the SBKO outperforms the other DoEs if the required sample number leads to a compact set in the input domain where the Bayesian inference has high probability of failing. However, when we have the knowledge that those samples form a sparse set, the classical LHS design could be an option. This also reflects a potential application of the SBKO method for the high dimensional problem if we have to deal with non-linear constraints and a non-convex region where the LHS design would be inadequate.

4.2. K-Optimal Enhanced Bayesian Optimization Problem

In this subsection, we demonstrate the K-optimal enhanced Bayesian optimal algorithm. As discussed in Section 3.3, $\zeta(\kappa)$ is the indicator of the strength of exploration. We compared the classical EI algorithm (with $\zeta = 0.01$ suggested by Lizotte) and our K-optimal enhanced EI (KO-EI) algorithm. Our experiments consisted of two parts. Firstly, we illustrated the capability of exploration of the two methods. Secondly, the comparison of the convergence rate was demonstrated. The Viana function and Branin function were used as the benchmark test functions again, and the Gaussian process model was equipped with the Matérn 5/2 kernel too. We also investigated the BO algorithms on a logistic regression classification task on the MNIST data in the last experiment. The main results are as follows.

Example 4. Comparison of Exploration

We started our experiments with an extreme special case of the Viana function, Equation (27). We let $[X_0, y_0] = [-2.6594, 0.8303]$ be the initial experimental design, and then the EI and KO-EI were implemented for 6 iterations. The hyper-parameters were optimized via the maximum likelihood criterion before sampling the next point at each iteration. The corresponding values of prediction and acquisition functions are displayed in Figure 2.

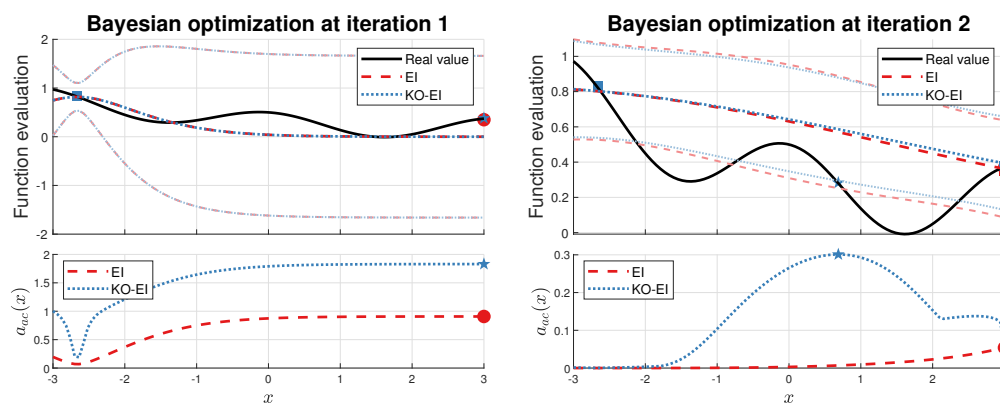


Figure 2. Cont.

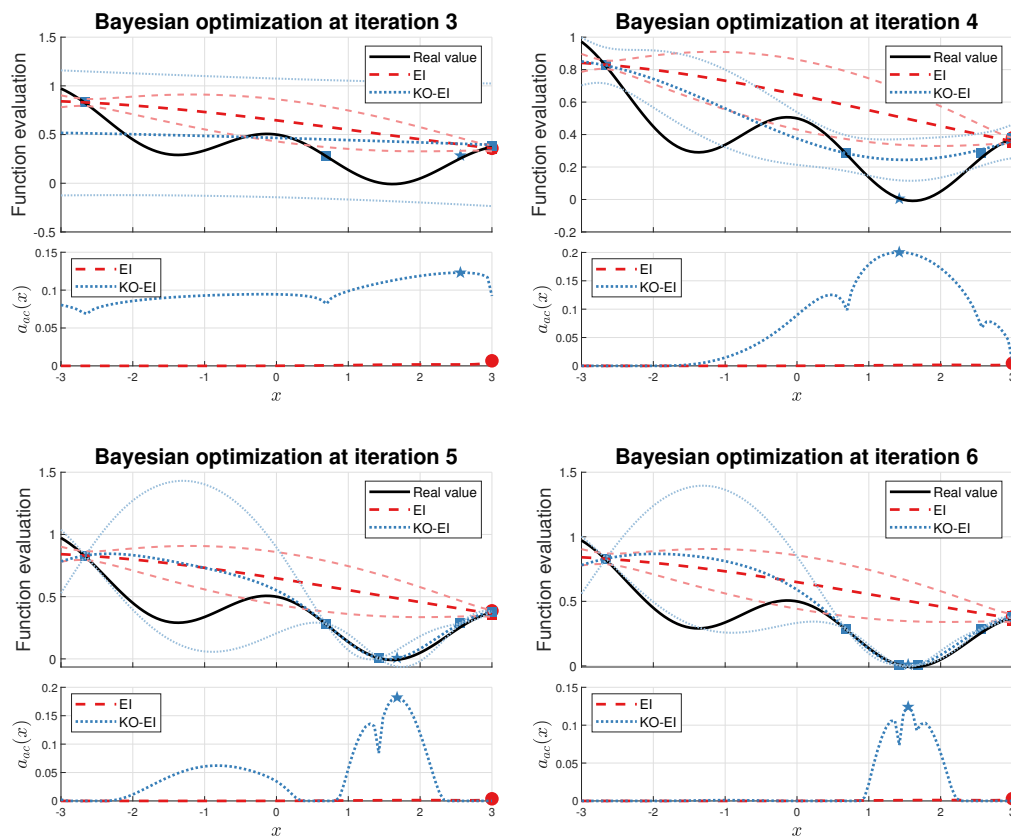


Figure 2. Comparisons of the expected improvement (EI) and K-optimal enhanced EI (KO-EI) algorithms. The above subfigure of each figure illustrates the predictions (darker line) and corresponding standard deviations (lighter line), and the one below demonstrates the values of acquisition functions. The red and blue squares represent the current samples for the EI and KO-EI algorithm respectively. The red dots are the best points to be sampled according to the EI criterion, while the blue pentagrams were collected *w.r.t* the KO-EI algorithm.

It is clear that the point to maximize the EI acquisition is trapped at $x = 3$. As the iteration continues, it strengthens the information that the corresponding Gaussian process model has ‘accurate’ approximation and the optimal value has already obtained, which is obviously a false conclusion. On the contrary, although the KO-EI sampled $X_1 = 3$ at iteration 1, it ensures that we are able to jump out the trap to reach $X_2 = 0.7008$. As discussed at the end of Section 3.3, although the point 0.7008 is far away from the current exploitation region at around $x = 3$, we can still extract it as the next sample. Furthermore, $x = 3$ leads to a failure in Bayesian inference, so it is eliminated from the candidate set for the next sample.

Example 5. Comparison of Convergence Rate

In this part, we started a new experiment to investigate the convergence rate of the two algorithms with the Viana and Branin function, Equation (28). We repeated the EI and KO-EI optimization 100 times. The numbers of initial design for the Viana function and the Branin function were set to be 1 and 5, respectively, while the maximum numbers of samples were set to be 20 and 50, respectively. The results are summarized and displayed in Figure 3. Figure 3a shows that the KO-EI convergence is faster than the EI criterion for the Viana function, and generally, the KO-EI is more stable than the EI algorithm, because the standard deviation of minimal observed objective is always smaller. When applying the two algorithms with the Branin function, the two methods have similar convergence

rates and the KO-EI criterion is slightly better. However, we note that the KO-EI has a smaller standard deviation of the minimal observed objective again; hence, it can lead to a more stable result.

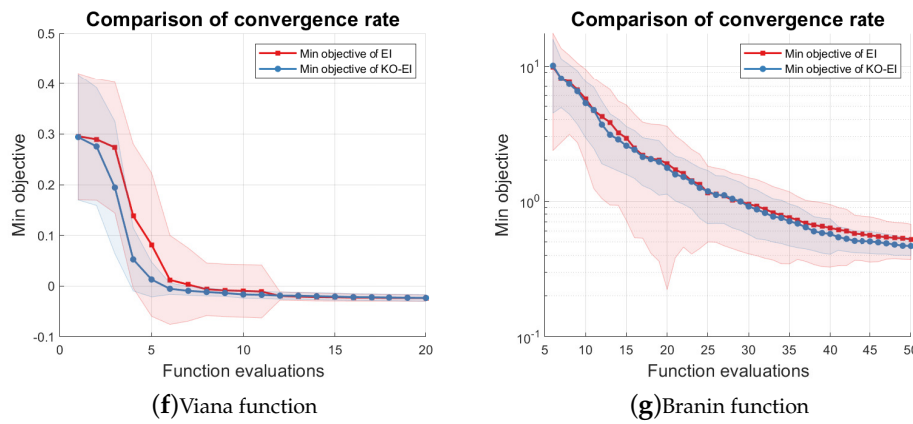
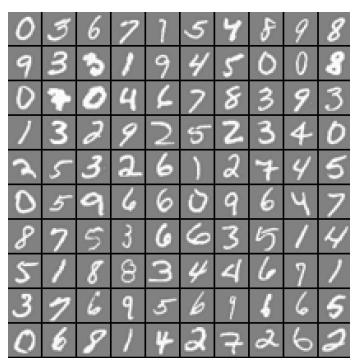


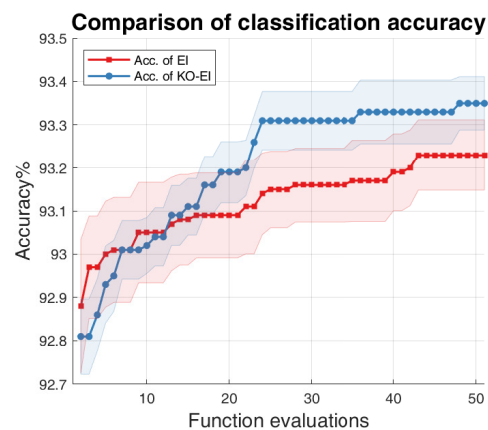
Figure 3. Comparison of convergence rate between the EI and KO-EI algorithms. The solid dash line represents the mean value of the minimal observed objective, while the transparent region represents the standard deviation *w.r.t* 100 independent simulations. (a) Viana function; (b) Branin function.

Example 6. Application on the Logistic Regression Classification Task

We used the EI and KO-EI algorithms to optimize the hyper-parameters of the logistic regression algorithms to maximize the classification accuracy. The experiment was implemented with the MNIST data, which was also investigated in reference [18]. The hyper-parameters of the logistic regression classification algorithm were the L^2 regularization parameter λ , between 0 and 1; the constants σ, ρ of the Wolfe–Powell conditions with $\sigma \in [0, 0.5]$ and $\rho \in [0, 1]$, respectively; the batch size from 10 to 100; the number of learning iterations from 20 to 200; and the learning slope ratio between 5 and 15. We compared the classification accuracies for 50 independent simulations for 50 EI/KO-EI runs, and the results are given in Figure 4. Note that the y-axis of Figure 4b is the percentage of classification accuracy times 100. It is obvious that the KO-EI performs better after several rounds of optimization, and it also leads to smaller standard deviations.



(a) Demonstration of MNIST data



(b) Comparison of the classification accuracy

Figure 4. Comparison of accuracy between the EI and KO-EI algorithms on the logistic regression classification on MNIST data. (a) Demonstration of the MNIST data; (b) The solid dash line represents the mean value of the classification accuracies while the transparent region represents the standard deviation *w.r.t* 50 independent simulations.

5. Conclusions

This paper examined the combination of the K-optimal design and the Bayesian optimization framework. In order to ensure the validity of Gaussian process inference, we introduced the condition number of the Gram matrix of the kernel as the acquisition function to propose the Sequentially Bayesian K-optimal design (SBKO). The SBKO is suitable for global tasks, such as approximation and prediction. On the other hand, the property of K-optimality was also used with the classical BO methods, namely the KO-BO method, in this research. The trade-off parameters were updated automatically based on the idea that points leading to smaller condition numbers should be explored. Numerical investigations on the approximation problem results showed that the SBKO generally outperforms the MC, LHS, and the MPV when the samples are compact in the input domain. Examples on the optimization problem showed that the K-optimal enhanced expected improvement (KO-EI) can deal with extreme cases where the EI criterion is trapped in a local maximum very well. Further experiments showed that the KO-EI convergences faster than the EI algorithm; however, it is much more stable.

Although the K-optimality performed well in our experiments, we also note that its calculation and optimization could still be a burden because it is not convex and there is no explicit expression of its gradient. Future work could focus on the approximation methods of the condition number, such as the Clarke generalized gradient [16,17], hence accelerating corresponding computation. We could also investigate the usage of our method to deal with non-convex constraints and input domains. An analysis of the theoretical boundaries of the KO-BO algorithms would be of great interest too.

Author Contributions: L.Y. proposed the original idea, implemented the experiments in the work and wrote the paper. X.D. contributed to the theoretical analysis and simulation designs. B.L. and J.X. partially undertook the writing and simulation work. All authors read and approved the manuscript.

Funding: This research received no external funding.

Acknowledgments: This work is supported by the program for New Century Excellent Talents in University, State Education Ministry in China (No. NCET 10-0893) And National Science Foundation of China (No. 11771450, No. 61573367).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gould, H.; Tobochnik, J.; Christian, W. *An Introduction to Computer Simulation Methods*; Addison-Wesley: New York, NY, USA, 1988; Volume 1.
2. Allen, M.P.; Wilson, M.R. Computer simulation of liquid crystals. *J. Comput. Aided Mol. Des.* **1989**, *3*, 335–353. [[CrossRef](#)] [[PubMed](#)]
3. Binder, K. Introduction: Theory and “technical” aspects of Monte Carlo simulations. In *Monte Carlo Methods in Statistical Physics*; Springer: Berlin, Germany, 1986; pp. 1–45.
4. Hurtado, J.; Barbat, A.H. Monte Carlo techniques in computational stochastic mechanics. *Arch. Comput. Methods Eng.* **1998**, *5*, 3. [[CrossRef](#)]
5. Cafilisch, R.E. Monte carlo and quasi-monte carlo methods. *Acta Numer.* **1998**, *7*, 1–49. [[CrossRef](#)]
6. Helton, J.C.; Davis, F.J. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliab. Eng. Syst. Saf.* **2003**, *81*, 23–69. [[CrossRef](#)]
7. Papalambros, P.Y.; Wilde, D.J. *Principles of Optimal Design: Modeling and Computation*; Cambridge University Press: Cambridge, UK, 2000.
8. Andrieu, C.; De Freitas, N.; Doucet, A.; Jordan, M.I. An introduction to MCMC for machine learning. *Mach. Learn.* **2003**, *50*, 5–43. [[CrossRef](#)]
9. Rasmussen, C.E. Gaussian processes in machine learning. In *Advanced Lectures on Machine Learning*; Springer: Berlin, Germany, 2004; pp. 63–71.
10. Brochu, E.; Cora, V.M.; De Freitas, N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv* **2010**, arXiv:1012.2599.

11. Busby, D. Hierarchical adaptive experimental design for Gaussian process emulators. *Reliab. Eng. Syst. Saf.* **2009**, *94*, 1183–1193. [[CrossRef](#)]
12. Echard, B.; Gayton, N.; Lemaire, M. AK-MCS: An active learning reliability method combining Kriging and Monte Carlo simulation. *Struct. Saf.* **2011**, *33*, 145–154. [[CrossRef](#)]
13. Ye, J.J.; Zhou, J. Minimizing the condition number to construct design points for polynomial regression models. *SIAM J. Optim.* **2013**, *23*, 666–686. [[CrossRef](#)]
14. Baran, S. K-optimal designs for parameters of shifted Ornstein-Uhlenbeck processes and sheets. *J. Stat. Plan. Inference* **2017**, *186*, 28–41. [[CrossRef](#)]
15. Jones, D.R.; Perttunen, C.D.; Stuckman, B.E. Lipschitzian optimization without the Lipschitz constant. *J. Optim. Theory Appl.* **1993**, *79*, 157–181. [[CrossRef](#)]
16. Maréchal, P.; Ye, J.J. Optimizing condition numbers. *SIAM J. Optim.* **2009**, *20*, 935–947. [[CrossRef](#)]
17. Chen, X.; Womersley, R.S.; Ye, J.J. Minimizing the condition number of a Gram matrix. *SIAM J. Optim.* **2011**, *21*, 127–148. [[CrossRef](#)]
18. Snoek, J.; Larochelle, H.; Adams, R.P. Practical bayesian optimization of machine learning algorithms. *Adv. Neural Inf. Process. Syst.* **2012**, *2*, 2951–2959.
19. Preuss, R.; von Toussaint, U. Sequential Batch Design for Gaussian Processes Employing Marginalization. *Entropy* **2017**, *19*, 84. [[CrossRef](#)]
20. Murray, I.; Adams, R.P. Slice sampling covariance hyperparameters of latent Gaussian models. *Adv. Neural Inf. Process. Syst.* **2010**, *2*, 1732–1740.
21. Aronszajn, N. Theory of reproducing kernels. *Trans. Am. Math. Soc.* **1950**, *68*, 337–404. [[CrossRef](#)]
22. Kullback, S. *Information Theory and Statistics*; Courier Corporation: New York, NY, USA, 1997.
23. Kim, W.; Pitt, M.A.; Lu, Z.L.; Steyvers, M.; Myung, J.I. A hierarchical adaptive approach to optimal experimental design. *Neural Comput.* **2014**, *26*, 2465–2492. [[CrossRef](#)] [[PubMed](#)]
24. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*; John Wiley & Sons: Hoboken, NJ, USA, 2012.
25. Ben Salem, M.; Roustant, O.; Gamboa, F.; Tomaso, L. Universal prediction distribution for surrogate models. *SIAM/ASA J. Uncertain. Quantif.* **2017**, *5*, 1086–1109. [[CrossRef](#)]
26. Picheny, V.; Wagner, T.; Ginsbourger, D. A benchmark of kriging-based infill criteria for noisy optimization. *Struct. Multidiscip. Optim.* **2013**, *48*, 607–626. [[CrossRef](#)]
27. Xiong, S.; Qian, P.Z.; Wu, C.J. Sequential design and analysis of high-accuracy and low-accuracy computer codes. *Technometrics* **2013**, *55*, 37–46. [[CrossRef](#)]
28. Rasmussen, C.; Williams, C. GPML: Matlab Implementation of Gaussian Process Regression and Classification, 2007. Software. Available online: <http://www.GaussianProcess.org/gpml/code> (accessed on 25 October 2015).
29. Johnson, S.G. The NLOpt Nonlinear-Optimization Package, 2014. Software. Available online: <http://ab-initio.mit.edu/nlopt> (accessed on 13 August 2016).



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).