Structural and Multidisciplinary Optimization (2025) 68:51
https://doi.org/10.1007/s00158-025-03978-0

RESEARCH PAPER

A comparative study of acquisition functions for active learning kriging in reliability-based design optimization

Todd Thompson[1,2] · **Robert McMullen**[1,2] · **Venkat Nemani**[2,5] · **Zhen Hu**[3] · **Chao Hu**[2,4,6]

Received: 25 July 2024 / Revised: 27 January 2025 / Accepted: 6 February 2025

Abstract

Many acquisition functions are available to improve active learning-based kriging models while conducting reliability-based design optimization (RBDO). A considerable challenge for computationally expensive models is deciding which acquisition function provides the greatest chance to complete the optimization with a minimum number of function evaluations. This paper presents a comprehensive comparative study of nine different acquisition functions in terms of the number of completed optimizations, total function evaluations, and repeatability. The comparative study was conducted. The comparative study was conducted on problems with varying levels of input uncertainty and surrogate uncertainty thresholds to evaluate the performance across a range of problem settings. Two well-known mathematical examples and one engineering example are employed to compare the performance of different acquisition functions. A unified metric is proposed to evaluate the overall performance of different acquisition functions for RBDO. The results of the comparative study show that: (1) The performance of the acquisition functions can be categorized into two distinct groups based on whether they include a term for the joint probability density function; the acquisition functions within a group have similar performance; and the acquisition functions that included a term for the joint probability density function had the best performance. (2) Local approximations have a higher success rate of finding the RBDO optimum than global approximations due to higher surrogate model fidelity in the optimum region. (3) This paper also explores a common typographical error in the expected feasibility function (EFF) that limits the ability of the function to explore the design space. This error decreases the effectiveness of EFF when compared to other acquisition functions.

Keywords Acquisition function · Active learning · Kriging · Reliability analysis · Design optimization

Responsible editor: Ramin Bostanabad.

✉ Chao Hu
chao.hu@uconn.edu

[1] Deere and Company, John Deere Power Systems, Cedar Falls, IA 50613, USA

[2] Department of Mechanical Engineering, Iowa State University, Ames, IA 50011, USA

[3] Department of Industrial and Manufacturing Systems Engineering, University of Michigan-Dearborn, Dearborn, MI 48128, USA

[4] Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011, USA

[5] Present Address: Corteva Agriscience, Johnston, IA 50131, USA

[6] Present Address: School of Mechanical, Aerospace, and Manufacturing Engineering, University of Connecticut, Storrs, CT 06269, USA

1 Introduction

Reliability analysis attempts to determine the probability of a system performing its intended function with the consideration of various uncertainty sources. The input variability considered in reliability analysis can be from geometry, material properties, external loading, or their combination. The reliability R (or one minus the probability of failure P_f, i.e., $1 - P_f$) can be calculated as a multi-dimensional integration of a joint probability density function (PDF) over a safe region (Wang and Wang 2014; Hu et al. 2019; Echard et al. 2011; Li et al. 2021). It takes the following form:

$$R = \Pr[G(\mathbf{x}) \leq 0] = \int \cdots \int_{\Omega_{\text{safe}}} f_{\mathbf{x}}(\mathbf{x}) d\mathbf{x}, \tag{1}$$

where $\Pr[\bullet]$ is a probability operator, $G(\mathbf{x})$ is a performance function, also called a limit-state function, typically defined based on a response of interest, $\mathbf{x}$ is a vector of continuous

Published online: 22 March 2025

random input variables, whose statistical properties can in general be fully characterized by a joint PDF $f_{\mathbf{x}}(\mathbf{x})$, and $\Omega_{\text{safe}} = \{\mathbf{x} : G(\mathbf{x}) \leq 0\}$ defines a safe region where the performance function takes a value less than or equal to zero. The boundary between the safe and failure regions $G(\mathbf{x}) = 0$ is the limit state. For notational simplicity, we use the lower case $\mathbf{x}$ (x) to denote both a random vector (random variable) and its realization.

Evaluation of the integral in Eq. (1) directly is difficult, if not impossible, for complex systems involving expensive-to-evaluate performance functions. A simple and straightforward approach to approximate this integral is to perform a Monte Carlo simulation (MCS) directly on the performance function (Echard et al. 2011). Given that, a direct MCS generates a set of N_{mcs} random input points, and the reliability approximated by the direct MCS can be calculated as the proportion of the random points falling within the safe region, expressed as:

$$\widehat{R}_{\text{mcs}} = \frac{N_{\text{safe}}}{N_{\text{mcs}}}, \tag{2}$$

where N_{safe} is the number of input points falling in the safe region Ω_{safe}, i.e., the number of the random $\mathbf{x}$ realizations that satisfy $G(\mathbf{x}) \leq 0$. If the reliability R is high (i.e., the probability of failure P_{f} is small), the direct MCS method requires many points (e.g., $N_{\text{mcs}} = 10^6$–10^7) for an accurate reliability assessment (Li et al. 2021).

Modeling complex systems (e.g., structures with complex geometries) requires high-fidelity simulation models (e.g., finite element models (FEMs)) to represent the physics of the systems. These high-fidelity models are often computationally expensive, making $G(\mathbf{x})$ in Eq. (1) time-consuming to evaluate. The time to conduct the FEM simulations would take months, if not years. Conducting a large number of FEM simulations (e.g., 10^6) would not be practical for most cases. An efficient alternative to running an MCS directly on a high-fidelity model is building a cheap-to-evaluate surrogate model (i.e., an explicit mathematical function) of the high-fidelity model and running an MCS on the surrogate. Three good survey papers on the applications of surrogate modeling to engineering design focus on performance function approximation (Jin et al. 2001), uncertainty propagation (Lee and Chen 2009), and reliability-based design optimization (RBDO) (Moustapha 2019), respectively. Kriging or Gaussian process regression is a popular non-parametric surrogate modeling technique (Rasmussen and Williams 2006). Kriging surrogate models are versatile and have many different applications. The applications include modeling fatigue of truck components (Fang et al. 2015; Mi et al. 2019), wind turbines (Hu et al. 2016; Teixeira et al. 2019), adhesive (Le Pavic et al. 2019) and welded joints (Dong et al. 2020), building fragility (Ghosh et al. 2021), and dental implant resonant frequency (Chu et al. 2019). As explained further in Sect. 2, the kriging model takes a set of initial points, called training points, from the complex simulation (including FEMs) and creates a response model given the input parameters. The kriging method has two advantages. First, the kriging model is an exact interpolation; this means that the model will exactly match the complex simulation results at the training points. Next, the kriging model can predict the response at a point and estimate the variance associated with this prediction, giving insight into the possible model error at that point (Echard et al. 2011). Once an accurate kriging model is created, that model can be used to conduct reliability analysis in the inner loop as part of the commonly used double-loop RBDO.

RBDO combines reliability analysis and design optimization to realize probabilistic engineering designs. RBDO can find an optimal design based on some objective, such as minimizing weight or volume, while meeting reliability targets based on fatigue or fracture, accounting for uncertainties in geometry, materials, and external loading (Hu et al. 2019). However, RBDO often requires a substantial number of function evaluations to estimate the reliability of each candidate system design. A kriging surrogate model can be built on a few evaluations of the real performance function, which is more computationally efficient for reliability calculations (Dubourg et al. 2011). For example, Zhao et al. (2011) replaced a FEM of a road arm with a kriging model to reduce the overall weight by ~ 10% while still meeting the crack initiation fatigue life of 5 years. Kroetz et al. (2020) use kriging models to minimize the total cost of a structure to ensure survivability under the effects of environmental-induced corrosion over time. Dilip et al. (2021) developed kriging models of pavement structures to change section thicknesses to reduce cost while still maintaining a reliability of 99%. Using kriging models made these optimizations more efficient while not losing result accuracy.

Figure 1 shows a classification of commonly used RBDO approaches involving surrogate models, also called surrogate-assisted RBDO approaches. Branch I directly models the probability of failure (P_{f}) or reliability index (β) as a function of the design inputs. Branch II performs reliability analysis on a surrogate model of the performance function. The surrogate model can be created in two different ways. The local approximation (II.1) creates a surrogate model of the performance function in a subregion of the design space. The local approximation is centered on the current optimal design point with a window size based on the expected uncertainty. The optimization routine uses the local approximation to select a more optimal design point, and then a new local window and approximation are created to continue the optimization. We focus on using kriging (Branch II.1.a) to create the local approximation, but other surrogate modeling methods (Branch II.1.b) are available,

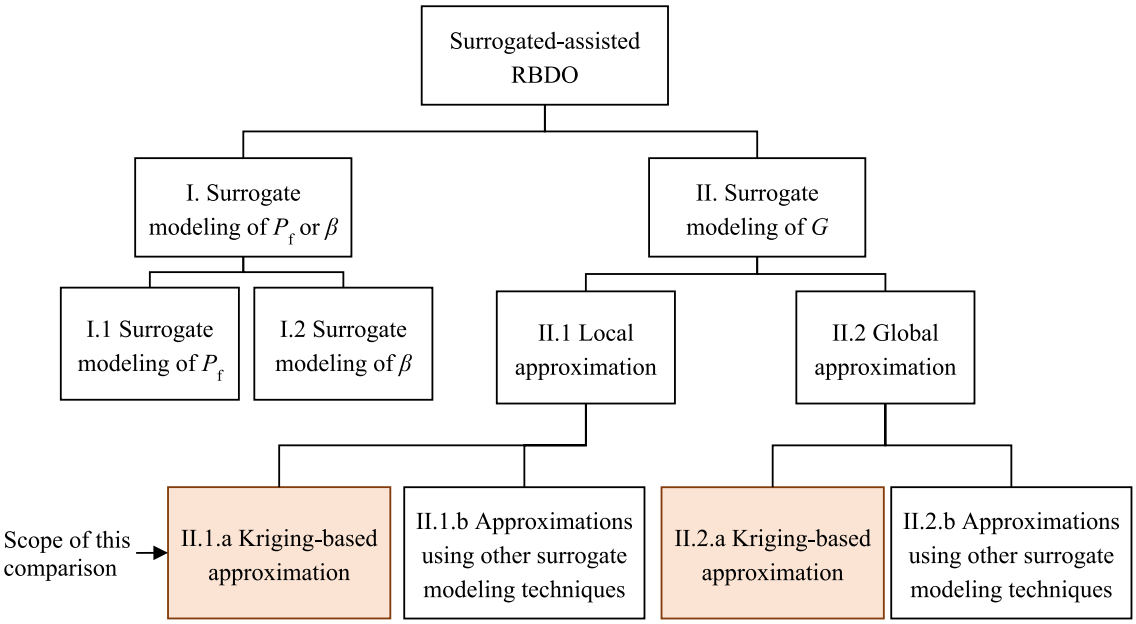


Fig. 1 A flowchart classifying surrogate-assisted RBDO approaches. Adapted from Fig. 1 in Moustapha (2019)

such as neural networks or support vector machines. The global approximation (Branch II.2) is another option for creating a surrogate model for RBDO. It starts by creating a surrogate model covering the entire design space. Fortunately, the global model does not need to be accurate at all locations in the design space but only in limited regions of interest. This observation has led to approaches called adaptive design or active learning (Teixeira et al. 2021), also used in local approximation approaches like the kriging-based ones. The active learning process uses local sampling centered around the current optimum to pick new points, often in the vicinity of a limit state and having high predictive uncertainty. The enrichment points are added to refine the global surrogate, and the optimization continues. The local approximation can be more computationally expensive than the global approximation because a new model is built for each new approximation. Reusing previous samples to build a surrogate in the current local window may somewhat alleviate this issue (Lee et al. 2011a). Conversely, the global approximation adds sample points to the existing model (Moustapha 2019). However, the global model can be inefficient due to adding initial Latin hypercube sampling points in regions of low interest.

Kriging models become more accurate as training points are added, but adding points at random, especially for complex simulations that take a long time to run, is inefficient. Adding enrichment points to kriging models at specific locations determined based on the information acquired so far efficiently improves the ability of the model to estimate the system's reliability (Moustapha 2019). Many acquisition functions have been developed to improve kriging modeling efficiency by adaptively identifying enrichment points (Wang and Wang 2014; Echard et al. 2011; Li et al. 2021; Lee et al. 2011a; Bichon et al. 2008; Lv et al. 2015; Zhang et al. 2019; Sadoughi et al. 2018; Wang et al. 2022a; Meng et al. 2019; Sun et al. 2017). While efforts have been made to account for the complex covariances among probabilistic predictions at different input points in acquisition functions (Hu and Mahadevan 2016b; Zhu and Du 2016), most acquisition functions are designed to focus solely on the predictive uncertainty readily available from the kriging prediction at a specific input point ($\mathbf{x}$). This approach is preferred for its ease of implementation and reduced computational complexity. Therefore, this paper focuses exclusively on the class of acquisition functions that rely on predictive uncertainty at individual input points ($\mathbf{x}$). Such adaptive sampling generally starts with a space-filling random sampling technique such as Latin hypercube sampling, which creates a cloud of candidate points around the design optimum. The acquisition functions then select points close to the limit state (exploiting points that are promising) to improve the ability to predict the expected reliability and regions of high uncertainty (exploration to gather more information) to improve the accuracy of kriging models (Sadoughi et al. 2018).

The challenge with many acquisition functions being available is choosing the best for a specific engineering application. Each function specifies its own stopping threshold, which makes a fair comparison difficult. Despite tremendous research efforts that led to the creation of many acquisition functions for reliability analysis, there is

a lack of systematic comparison that provides insights into the performance of different acquisition functions when used to solve RBDO problems. The survey in Moustapha (2019) compared a decent collection of surrogate-assisted RBDO approaches, yet comparing acquisition functions used in the kriging-based surrogate modeling part of RBDO was not in the scope of the survey. Our comparative study aims to fill this knowledge gap by determining which acquisition function provides acceptable accuracy while minimizing the number of function evaluations to limit the number of computationally expensive model simulations. We conducted 3,870 optimization runs across nine acquisition functions for three mathematical problems. Input standard deviation levels of 0.2, 0.3, 0.4, 0.5, and 0.6 were analyzed, along with surrogate uncertainty thresholds for the local approximation set at 0.005, 0.01, and 0.03. We also compared the performance of the local and global approximation. Finally, we extended this comparison study to an industry-relevant design application, where several dimensions of an engine oil pickup tube are optimized to minimize its weight while satisfying reliability constraints.

The contributions of this paper are as follows:

(1) The acquisition functions' performance is separated into two groups that differ by whether a term for the joint probability density function is included. The acquisition functions within a group have similar performance.
(2) Evidence that acquisition functions that include the joint probability density function perform the best.
(3) Local approximations have a higher success rate of finding the RBDO optimum than global approximations due to higher surrogate model fidelity in the optimum region.
(4) Commentary and analysis on a common typographical error for the Expected Feasibility Function (EFF) that affects how the EFF algorithm behaves.

The remainder of this paper is organized as follows. Section 2 briefly reviews the kriging model and the nine acquisition functions used in this study. Section 3 describes the methodology of the comparison. The acquisition functions are compared using three mathematical and one practical case study in Sect. 4. Section 5 provides concluding remarks.

2 Review of kriging-based reliability analysis

Kriging can create surrogate models of complex behavior with limited data points. After a kriging model is created with a set of known points, the kriging model can predict the mean $\mu_{\widehat{G}}(\mathbf{x})$ and standard deviation $\sigma_{\widehat{G}}(\mathbf{x})$ of a performance function $G(\bullet)$ at a new data point $\mathbf{x}$. Understanding the variance at a new data point helps determine where the model is accurate and where the model can be enriched (Zhao et al. 2011). Kriging models are implemented as surrogate models for complex simulation models, such as FEMs, where the computational cost of evaluating a model at every point would be time-consuming (Echard et al. 2011).

2.1 Basics of kriging

In kriging-based reliability analysis, a kriging surrogate $\widehat{G}(\mathbf{x})$ is built to approximate the input–output relationship of an unknown but observable performance function $G(\mathbf{x})$. This kriging model often consists of a trend function $\mathbf{h}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\gamma}$ and a zero-mean Gaussian process $Z(\mathbf{x})$, where $\mathbf{h}(\mathbf{x})$ is a vector of basis functions and $\boldsymbol{\gamma}$ denotes the corresponding coefficients. can be expressed as

$$\widehat{G}(\mathbf{x}) = \mathbf{h}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\gamma} + Z(\mathbf{x}), \tag{3}$$

$$Z(\mathbf{x}) \sim \mathcal{GP}\left(0, k\left(\mathbf{x}, \mathbf{x}'\right)\right), \tag{4}$$

where $Z(\mathbf{x})$ is a zero-mean Gaussian process defined by a covariance or kernel function $k\left(\mathbf{x}, \mathbf{x}'\right)$. This study employs the squared exponential kernel, which takes the following form:

$$k\left(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}\right) = \sigma_{\mathrm{s}}^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma_{\mathrm{l}}^2}\right), \tag{5}$$

where σ_{s} is the signal standard deviation and σ_{l} is the characteristic length scale, and σ_{s} and σ_{l} must be greater than zero. These values can be kept greater than zero by the unconstrained parameterization vector $\boldsymbol{\theta} = \left[\theta_1, \theta_2\right]^{\mathrm{T}}$, where $\theta_1 = \ln\left(\sigma_{\mathrm{s}}\right)$ and $\theta_2 = \ln\left(\sigma_{\mathrm{l}}\right)$ with $\ln(\bullet)$ indicating the natural logarithm (Rasmussen and Williams 2006; MathWorks 2022b). The signal standard deviation parameter σ_{s} defines the upper bound for the prior variance and covariance. It should be set to a large value when $G(\mathbf{x})$ is known to exhibit a wide vertical range (i.e., significant variation along the y-axis). The length scale parameter σ_{l} controls how quickly the correlation between points decreases as their distance in the input space increases. If $G(\mathbf{x})$ spans a small input region but is expected to have sharp changes, a small σ_{l} may be appropriate; for a smoothly varying $G(\mathbf{x})$ over a large region, σ_{l} should be set to a large value.

The squared exponential kernel in Eq. (5) employs a uniform length scale σ_{l} across all N_{d} dimensions of $\mathbf{x}$. Alternatively, a different length scale $\sigma_{\mathrm{l},i}$ can be assigned to each input dimension x_i, an approach known as automatic relevance

determination (ARD) (Neal 2012). The ARD squared exponential kernel is expressed as:

$$k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}) = \sigma_s^2 \exp\left(-\frac{1}{2} \sum_{i=1}^{N_d} \frac{(x_i - x_i')^2}{\sigma_{l,i}^2} \right), \tag{6}$$

where the N_d length scale parameters $\sigma_{l,1}$, …, σ_{l,N_d} measure the relevance of the respective input dimensions to the kriging model prediction. Given that input values are normalized along all input dimensions, if $\sigma_{l,i}$ is learned to have a very large value, the associated input dimension x_i is considered irrelevant, contributing minimally to the kriging prediction. This kernel is often referred to as the anisotropic variant of the (isotropic) squared exponential kernel in Eq. (5).

Suppose we have access to a training dataset of N_{train} input–output pairs, $\mathcal{D}_{train} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_{train}}$, where the observation at each input $\mathbf{x}_i$ matches exactly the performance function value at $\mathbf{x}_i$, i.e., $y_i \equiv G(\mathbf{x}_i)$. In other words, we can directly observe the unknown performance function G with *no measurement noise*. Then, a kriging surrogate model can be built based on these *noise-free observations* to predict the performance function at a new input, $\mathbf{x}_{new}$. This kriging surrogate prediction can be compactly written as $\widehat{G}(\mathbf{x}) \sim \mathcal{N}(\mu_{\widehat{G}}(\mathbf{x}), \sigma_{\widehat{G}}^2(\mathbf{x}))$, where $\mathcal{N}(\bullet, \bullet)$ stands for a normal distribution, and $\mu_{\widehat{G}}(\mathbf{x})$ and $\sigma_{\widehat{G}}(\mathbf{x})$ are, respectively, the posterior mean and standard deviation derived based on conditionals of a multivariate Gaussian (Rasmussen and Williams 2006). The noise-free setting is the most common in RBDO, as we can directly obtain the true output values of a simulation model (e.g., an FEM model) without any data acquisition noise. Moreover, we do not deal with simulation models with stationary or non-stationary noise, limiting the scope of our comparison to noise-free simulation models with perfectly repeatable outputs.

Although not directly relevant to our comparative study, it is worth noting that kriging models can also account for *noisy observations*. Now suppose we can only acquire noisy observations of $\widehat{G}(\mathbf{x})$, typically through a Gaussian observation model of the following form:

$$y_i = G(\mathbf{x}_i) + \varepsilon_i, \tag{7}$$

where ε_i is a zero-mean Gaussian noise associated with the ith observation, i.e., $\varepsilon_i \sim \mathcal{N}(0, \sigma_{\varepsilon_i}^2)$. Most GPR work assumes homoscedastic noise to avoid having too many hyperparameters (note that each σ_{ε_i} now becomes a hyperparameter in addition to the kernel parameters) due to a high risk of overfitting. In the homoscedastic noise setting, the noise standard deviation is fixed across all inputs, i.e., we only have one hyperparameter, σ_ε, to characterize the observation noise. The subsequent treatment for hyperparameter optimization is very simple: adding a term for the noise variance σ_ε^2 to each diagonal element of the covariance matrix for the training observations (Rasmussen and Williams 2006; MathWorks 2022a). Some recent open-source efforts went one step further by considering the observation noise varies with the data source, not the input variables, in a multi-fidelity modeling setup (Yousefpour et al. 2024). This treatment balances the risk of overfitting and the ability to model observation noise accurately.

Finally, we are not solving RBDO problems with qualitative design variables, where some or all of the design variables represent categorical or discrete choices rather than continuous quantities. Standard kriging models can be extended to accommodate categorical or integer-valued input variables in sequential Bayesian optimization (Garrido-Merchán and Hernández-Lobato 2020). Some recent work in the engineering design community attempted to apply kriging to solve design problems involving categorical inputs, such as materials design (Wang et al. 2022b), building renovation (Moustapha et al. 2022), and structural optimization (Toal 2023). These types of design problems involving qualitative inputs do not fall within the scope of this comparative study.

2.2 Acquisition functions for active learning kriging models

Active learning kriging, also called adaptive surrogate modeling or active learning, improves model accuracy and reduces model uncertainty by initializing and adaptively expanding a set of training points (Moustapha 2019). A frequent practice of active kriging is to start with a small set of initial training samples generated using Latin hypercube sampling and then augment this training set with samples sequentially and adaptively identified according to what is important to the problem at hand (reliability analysis or optimization). For a given kriging model $\widehat{G}(\mathbf{x})$, active kriging identifies a new training sample $\mathbf{x}_{new}$ for the refinement of the kriging model by solving the following optimization model

$$\mathbf{x}_{new} = \arg\max_{\mathbf{x} \in \Omega_{cand}} \{L(\mathbf{x})\} \quad \text{or} \quad \arg\min_{\mathbf{x} \in \Omega_{cand}} \{L(\mathbf{x})\}, \tag{8}$$

where Ω_{cand} is a bounded or unbounded region of the input space where a candidate sample of the random variables, $\mathbf{x}$, resides, and $L(\mathbf{x})$ is an active learning or acquisition function which is a function of $\mu_{\widehat{G}}(\mathbf{x})$ and $\sigma_{\widehat{G}}(\mathbf{x})$ of the kriging surrogate model prediction.

Like many optimization problems, the definition of the loss function (i.e., the acquisition function $L(\mathbf{x})$ in Eq. (8)) is crucial to the effectiveness of adaptive surrogate modeling/active learning in improving the prediction accuracy of $\widehat{G}(\mathbf{x})$. $L(\mathbf{x})$ needs to be defined according to the goal of the kriging surrogate modeling. For instance, in order to train

an accurate kriging surrogate model over the whole design domain Ω, an acquisition function may pick samples with the highest uncertainty as new training points (i.e., using $\sigma_{\hat{G}}(\mathbf{x})$ as $L(\mathbf{x})$). If a kriging surrogate model is trained for solving an optimization problem in the context of efficient global optimization (EGO) or Bayesian optimization (Jones et al. 1998), the expected improvement (EI) function can be used as $L(\mathbf{x})$.

This paper focuses on reliability-based design optimization using kriging surrogate models. Therefore, this section concentrates on acquisition functions designed for reliability analysis. Reliability analysis is a binary (two-class) classification problem that divides the domain Ω of random variables into safe and failure categories. The most important thing for any classification problem is accurately approximating the boundary. This boundary is called a limit state in reliability analysis. This limit state, as described at the beginning of Sect. 1, separates the safe region from the failure region. In kriging-based reliability analysis, we are only concerned about the accuracy of a surrogate model near the limit state. This primary focus implies that the kriging surrogate model needs to be refined adaptively so that the limit state can be accurately approximated. However, the accuracy of kriging prediction in other regions is not especially important. This type of surrogate model is also called a local surrogate in contrast to a global surrogate, which requires high accuracy in the whole design domain. Motivated by improving the prediction accuracy of a kriging surrogate model near the limit state, numerous acquisition functions (also known as learning functions) have been proposed in the past decade, with a limited subset discussed in Sadoughi et al. (2018). Figure 2 shows an overview of the development of various acquisition functions over the years.

Since the first acquisition function for reliability analysis was proposed in 2008 (Bichon et al. 2008), many more (15+) acquisition functions have been reported in the RBDO literature. The functions depicted in Fig. 2 are just the representative examples selected for comparison in this paper. The rest of this subsection describes some technical details of the acquisition functions used for comparison.

Before explaining the details of various acquisition functions, it is worth mentioning that the optimization problem given in Eq. (8) can be solved using global optimization algorithms, such as the COBYLA optimization algorithm used in the EGO method (Jones et al. 1998). As the dimension of $\mathbf{x}$ increases, solving Eq. (8) using a global optimizer becomes challenging. Alternatively, one can *first* generate a large number of random candidate samples of $\mathbf{x}$, for instance, via Latin hypercube sampling over a local hypercube centered around the mean point of $\mathbf{x}$ or by reusing all or a portion of the MCS samples drawn from $f_{\mathbf{x}}(\mathbf{x})$ for reliability analysis, *then* evaluate the acquisition function at these candidate points, and *finally* pick the candidate point with the maximum or minimum value of the acquisition function. This sampling-based approach for identifying new training samples is commonly used in the reliability analysis field, as evidenced by studies such as Wang and Wang (2014), Echard et al. (2011) and Toal (2023), which introduced some of the acquisition functions evaluated in this work. The primary reason is that this approach achieves a good balance between (1) computational efficiency, given that the acquisition function is typically inexpensive to evaluate, (2) numerical stability, considering that the sample locations for evaluations are almost always fixed, and (3) accuracy in identifying reasonably promising new points. For this reason, and to ensure consistency with the literature, the sampling-based approach is applied uniformly across all the acquisition functions studied in this paper.

2.2.1 Expected feasibility function (EFF)

As indicated in Fig. 2, the expected feasibility function (EFF) is the first acquisition function specifically designed for reliability analysis using a kriging surrogate model. EFF was proposed by Bichon et al. in 2008 and was inspired by the idea of

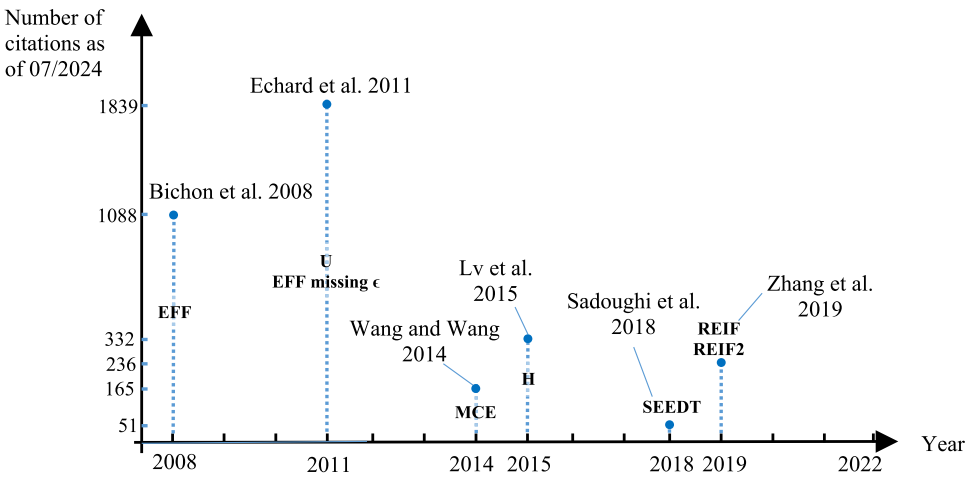


Fig. 2 Timeline overview of the development of various acquisition functions

the EI function in EGO (Jones et al. 1998). EFF quantifies the expected probability that the true value of $\widehat{G}(\mathbf{x})$ equals to the failure threshold (i.e., 0 in this paper) within a tolerance level $\epsilon(\mathbf{x}) = 2\sigma_{\widehat{G}}(\mathbf{x})$ as follows

$$EFF(\mathbf{x}) = \int_{-\epsilon}^{\epsilon} (\epsilon - |g|) f_{\widehat{G}(\mathbf{x})}(g) dg, \tag{9}$$

in which $f_{\widehat{G}(\mathbf{x})}(g)$ is the PDF of the kriging prediction $\widehat{G}(\mathbf{x})$ at $\mathbf{x}$. Maximizing *EFF* allows for the determination of the best candidate point that is close to the limit state but also has high uncertainty due to the consideration of the tolerance level $\epsilon(\mathbf{x}) = 2\sigma_{\widehat{G}}(\mathbf{x})$. Even though adjusting the tolerance level ϵ could change the level of exploration in the uncertain domain, it is recommended to set $\epsilon(\mathbf{x}) = 2\sigma_{\widehat{G}}(\mathbf{x})$ based on their case studies (Bichon et al. 2008). The close-form solution of EFF has been derived as

$$\begin{aligned} EFF(\mathbf{x}) =& \mu_{\widehat{G}}(\mathbf{x})\left[2\Phi\left(\frac{-\mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \Phi\left(\frac{-\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \Phi\left(\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right)\right] \\ & - \sigma_{\widehat{G}}(\mathbf{x})\left[2\phi\left(\frac{-\mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \phi\left(\frac{-\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \phi\left(\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right)\right] \\ & + \epsilon(\mathbf{x})\left[\Phi\left(\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \Phi\left(\frac{-\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right)\right], \end{aligned} \tag{10}$$

where $\Phi(\bullet)$ is the cumulative distribution function of the standard normal distribution, and $\phi(\bullet)$ is the PDF of the standard normal distribution. Please note that $\epsilon(\mathbf{x})$ is present before the third term.

2.2.2 EFF missing ϵ

EFF is an effective acquisition function used to benchmark newer acquisition functions. However, since the equation is long, a typographical error has been discovered in several papers (Echard et al. 2011; Lv et al. 2015; Meng et al. 2019). The ϵ in front of the third term in Eq. (10) is missing. The error was first noticed in Echard et al. (2011), and we define a variant of EFF called EFF missing ϵ as

$$\begin{aligned} \text{EFF missing}\epsilon(\mathbf{x}) =& \mu_{\widehat{G}}(\mathbf{x})\left[2\Phi\left(\frac{-\mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \Phi\left(\frac{-\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \Phi\left(\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right)\right] \\ & \sigma_{\widehat{G}}(\mathbf{x})\left[2\phi\left(\frac{-\mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \phi\left(\frac{-\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \phi\left(\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right)\right] \\ & + \left[\Phi\left(\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \Phi\left(\frac{-\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right)\right] \end{aligned} \tag{11}$$

The missing $\epsilon = 2\sigma_{\widehat{G}}(\mathbf{x})$ term changes how the EFF acquisition function behaves. Figure 3 shows the effect of omitting ϵ. For example, in Fig. 3a where $\sigma_{\widehat{G}}(\mathbf{x}) = 2$, the maximum value of EFF occurs at the limit state where $\mu_{\widehat{G}}(\mathbf{x}) = 0$. However, when ϵ is missing, the EFF maximum values occur at the edges, − 10 and 10. Figure 3b–d show that the value of EFF becomes larger as $\sigma_{\widehat{G}}(\mathbf{x})$ increases, which is desirable, as the primary intent of exploration is to identify new training points in regions of high predictive uncertainty. Conversely, with EFF missing ϵ the maximum occurs in regions of low uncertainty close to $\sigma_{\widehat{G}}(\mathbf{x}) = 0$. When the ϵ term is missing from EFF, the acquisition function has limited exploration of the design space, and new training points are placed close to the limit state even if the kriging model is well defined in that region. We evaluate EFF missing ϵ to determine how this function performs compared to the original EFF. To stay consistent with how EFF is calculated, we maximize EFF missing ϵ to determine the best candidate points to improve model performance.

2.2.3 *U* function

Following the idea of EFF, the *U* function was proposed by Echard et al. in their method called AK-MCS (Echard et al. 2011). The *U* function identifies the sample with the highest probability of misclassification as the new training sample. The rationale is that samples near the limit state have the highest probability of misclassification. Since the kriging prediction at any arbitrary $\mathbf{x}$, $\widehat{G}(\mathbf{x})$, follows a normal distribution, the probability of incorrect classification can be computed as (Hu and Mahadevan 2016a)

$$P(\mathbf{x}) = \begin{cases} \Pr\left[\widehat{G}(\mathbf{x}) \le 0\right] = \Phi\left(\frac{0-\mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) = 1 - \Phi\left(\frac{\mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right), \textit{if } \mu_{\widehat{G}}(\mathbf{x}) > 0, \\ \Pr\left[\widehat{G}(\mathbf{x}) > 0\right] = 1 - \Phi\left(\frac{0-\mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) = \Phi\left(\frac{\mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right), \textit{otherwise}. \end{cases} \tag{12}$$

The above equation can be simplified as

$$P(\mathbf{x}) = \Phi\left(-\frac{|\mu_{\widehat{G}}(\mathbf{x})|}{\sigma_{\widehat{G}}(\mathbf{x})}\right) \tag{13}$$

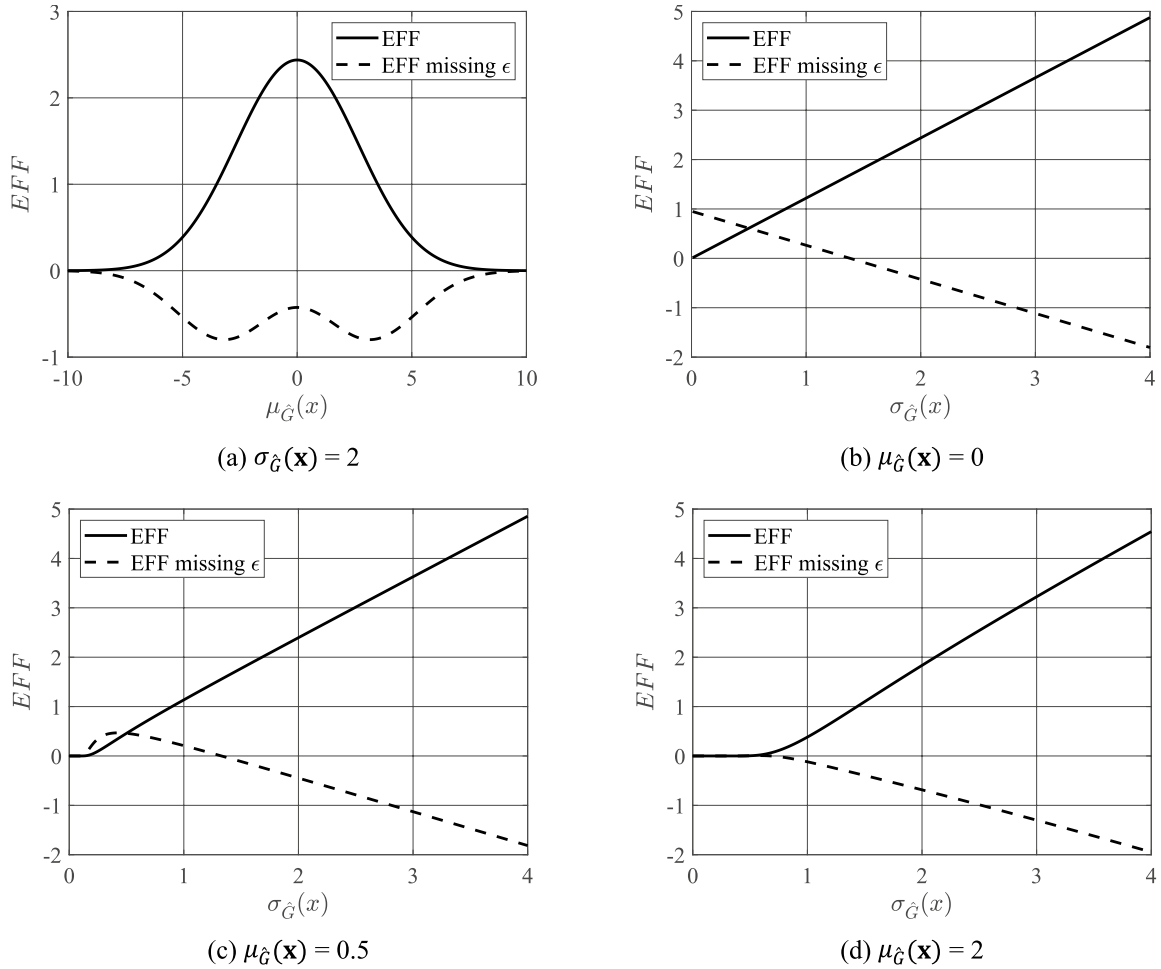
Since we want to identify the sample with the highest probability of classification error, the optimization model given in Eq. (8) for this acquisition function becomes

$$\mathbf{x}_{\text{new}} = \arg\max_{\mathbf{x}\in\Omega} \Phi\left(-\frac{|\mu_{\widehat{G}}(\mathbf{x})|}{\sigma_{\widehat{G}}(\mathbf{x})}\right) = \arg\min_{\mathbf{x}\in\Omega} \frac{|\mu_{\widehat{G}}(\mathbf{x})|}{\sigma_{\widehat{G}}(\mathbf{x})}. \tag{14}$$

Based on Eq. (14), the *U* function is defined as (Echard et al. 2011)

$$U(\mathbf{x}) = \frac{|\mu_{\widehat{G}}(\mathbf{x})|}{\sigma_{\widehat{G}}(\mathbf{x})} \tag{15}$$

As shown in Eq. (15), the *U* function has the same effect as that given in Eq. (13), but without computing the probability. By minimizing the *U* value, a new training sample can be chosen to improve the prediction accuracy of the



(a) $\sigma_{\hat{G}}(\mathbf{x}) = 2$ (b) $\mu_{\hat{G}}(\mathbf{x}) = 0$

(c) $\mu_{\hat{G}}(\mathbf{x}) = 0.5$ (d) $\mu_{\hat{G}}(\mathbf{x}) = 2$

Fig. 3 Comparison of EFF and EEF while not multiplying by ϵ on the 3rd term: **a** vary $\widehat{G}(\mathbf{x})$ while keeping $\sigma_{\widehat{G}}(\mathbf{x}) = 2$, **b–d** vary $\sigma_{\widehat{G}}(\mathbf{x})$ while holding $\widehat{G}(\mathbf{x})$ constant

kriging surrogate model near the limit state. The minimum is chosen because candidate samples with small values of $\mu_{\widehat{G}}(\mathbf{x})$ are close to the limit state that is important for calculating the reliability, and those with large values of $\sigma_{\widehat{G}}(\mathbf{x})$ are in regions of high uncertainty. Due to the simplicity of the expression, this acquisition function has gained much attention in the past decade. It has become the most cited acquisition function for reliability analysis in the literature, as shown in Fig. 2.

2.2.4 Maximum confidence enhancement (MCE)

The maximum confidence enhancement (MCE) function is another novel extension of the U function described above (Wang and Wang 2014). In MCE, the confidence level (CL) estimates the probability that the sign of $\mu_{\widehat{G}}(\mathbf{x})$ is correct and is given by

$$CL(\mathbf{x}) = \Phi\left(\frac{|\mu_{\widehat{G}}(\mathbf{x})|}{\sigma_{\widehat{G}}(\mathbf{x})}\right) \tag{16}$$

The reliability calculation is improved at the limit state if there is a high probability the sign of the kriging function is correct since the limit state is the boundary where $\mu_{\widehat{G}}(\mathbf{x}) = 0$. Based on the defined CL, the MCE acquisition function is defined as

$$MCE(\mathbf{x}) = (1 - \mathrm{CL}(\mathbf{x}))f_{\mathbf{x}}(\mathbf{x})\sigma_{\widehat{G}}(\mathbf{x}) = \Phi\left(-\frac{|\mu_{\widehat{G}}(\mathbf{x})|}{\sigma_{\widehat{G}}(\mathbf{x})}\right)f_{\mathbf{x}}(\mathbf{x})\sigma_{\widehat{G}}(\mathbf{x}), \tag{17}$$

where $f_{\mathbf{x}}(\mathbf{x})$ denotes the PDF value of a candidate sample point $\mathbf{x}$. Multiplication of $(1 - CL(\mathbf{x}))$ and $f_{\mathbf{x}}(\mathbf{x})$ indicates how likely it is that point will occur and multiplying by $\sigma_{\widehat{G}}(\mathbf{x})$ identifies points with high uncertainty. Maximizing EI determines the best candidate point to improve model performance.

Comparing Eqs. (17) with (15) shows that the main difference between the original U function and the MCE function is that the MCE multiplied two additional terms (i.e., the joint PDF $f_{\mathbf{x}}(\mathbf{x})$ and the standard deviation of the prediction $\sigma_{\widehat{G}}(\mathbf{x})$) to the U function. In order to investigate which added term caused the difference between MCE and U functions, in this paper, we also studied a modified MCE function by including only the joint PDF to the original U function. The modified MCE function is explained as follows.

2.2.5 Modified maximum confidence enhancement (MMCE)

As mentioned above, we modified the MCE function to observe the effect of increasing the emphasis on points near the limit state. The term $\sigma_{\widehat{G}}(\mathbf{x})$ is removed from Eq. (15) for the modified MCE (MMCE) to limit exploration. The MMCE function is given by

$$MMCE(\mathbf{x}) = (1 - CL(\mathbf{x}))f_{\mathbf{x}}(\mathbf{x}) = \Phi\left(-\frac{|\mu_{\widehat{G}}(\mathbf{x})|}{\sigma_{\widehat{G}}(\mathbf{x})}\right)f_{\mathbf{x}}(\mathbf{x}). \quad (18)$$

Maximizing MMCE determines the best candidate point to improve model performance. The definition of the MMCE function allows us to separate effects of different terms in the comparative study to investigate which term has the major impact on the refinement of the kriging surrogate model.

2.2.6 *H* function

The learning function H is based on the concept of information entropy (Lv et al. 2015). Entropy indicates the amount of disorder in the data. When the level of disorder is higher, the uncertainty in the prediction is higher. The H function is similar to the EFF given in Eq. (9), but is different in the computed quantity. Instead of computing the expected probability that the response is close to the limit state, the H function quantifies the information entropy of the response for a given tolerance level $\epsilon(\mathbf{x}) = 2\sigma_{\widehat{G}}(\mathbf{x})$ as follows

$$H(\mathbf{x}) = \left|-\int_{-\epsilon}^{\epsilon} \ln\left(f_{\widehat{G}(\mathbf{x})}(g)\right)f_{\widehat{G}(\mathbf{x})}(g)\mathrm{d}g\right|. \quad (19)$$

The close-form solution of the above equation has been derived as (Lv et al. 2015)

$$H(\mathbf{x}) = \left|\ln\left(\sqrt{2\pi}\sigma_{\widehat{G}}(\mathbf{x}) + \frac{1}{2}\right)\left[\Phi\left(\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) - \Phi\left(\frac{-\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right)\right]\right.$$
$$\left. - \left[\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{2}\phi\left(\frac{\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right) + \frac{\epsilon(\mathbf{x}) + \mu_{\widehat{G}}(\mathbf{x})}{2}\phi\left(\frac{-\epsilon(\mathbf{x}) - \mu_{\widehat{G}}(\mathbf{x})}{\sigma_{\widehat{G}}(\mathbf{x})}\right)\right]\right|. \quad (20)$$

Maximizing $H(\mathbf{x})$ determines the best candidate to improve model performance. Comparing the above-reviewed acquisition function, it can be seen that the above acquisition functions can be considered as a variant of either the EFF function or the U function. For instance, $H(\mathbf{x})$ can be classified into the EFF family, and the MCE, the MMCE, and the U function can be classified into one family since they all focus on classification errors. In addition to the variants of EFF and U functions, some other acquisition functions which are quite different from the EFF or U functions have been proposed in recent years. Two representative examples are given below.

2.2.7 Sequential exploration–exploitation with dynamic trade-off (SEEDT)

All previous acquisition functions assign fixed weights on the parameters $\mu_{\widehat{G}}(\mathbf{x})$ and $\sigma_{\widehat{G}}(\mathbf{x})$ to account for exploration (regions with high uncertainty) and exploitation (regions close to the limit state). Sadoughi et al. (2018) proposed an acquisition function, SEEDT, that dynamically adjusts these weights to reduce the error in the active kriging model efficiently. Let α_{t} denote an exploration–exploitation trade-off coefficient that measures the uncertainty in the limit-state prediction. Mathematically, this coefficient can be expressed as (Sadoughi et al. 2018)

$$\alpha_{\mathrm{t}} \approx \exp\left(\frac{1}{N_{\mathrm{mcs}}}\sum_{i=1}^{N_{\mathrm{mcs}}} I_{\Omega_{\mathrm{ls}}^{\mathrm{prob}}}(\mathbf{x}_i)\right). \quad (21)$$

Here, $\Omega_{\mathrm{ls}}^{\mathrm{prob}}$ is the probable region of the limit state in the input space, defined as a set of all $\mathbf{x}$ realizations where the prediction interval of G at a given confidence level includes the limit-state value (i.e., 0). The indicator function $I_{\Omega_{\mathrm{ls}}^{\mathrm{prob}}}(\mathbf{x}_i)$ takes the value of 1 if $\mathbf{x}_i \in \Omega_{\mathrm{ls}}^{\mathrm{prob}}$ and 0 otherwise. N_{mcs} denotes the number of MCS samples drawn from the probability distribution of $\mathbf{x}$, $f_{\mathbf{x}}(\mathbf{x})$, to numerically estimate the probability of a random realization of $\mathbf{x}$ falling within $\Omega_{\mathrm{ls}}^{\mathrm{prob}}$. The logic is that as the kriging model produces less uncertain (and typically more accurate) predictions of the limit state, $\Omega_{\mathrm{ls}}^{\mathrm{prob}}$ tends to shrink, resulting in a smaller value of α_{t}. The SEEDT acquisition function utilizes α_{t} to dynamically weigh exploration and exploitation and takes the following form:

$$\mathrm{SEEDT}(\mathbf{x}) = f_{\mathbf{x}}(\mathbf{x})\frac{\alpha_{\mathrm{t}}^2\sigma_{\widehat{G}}(\mathbf{x})}{\sqrt{1+\alpha_{\mathrm{t}}^2}}\exp\left[-\frac{\mu_{\widehat{G}}(\mathbf{x})^2}{2\sigma_{\widehat{G}}(\mathbf{x})\sqrt{1+\alpha_{\mathrm{t}}^2}}\right], \quad (22)$$

Maximizing the SEEDT acquisition function, here denoted as SEEDT($\mathbf{x}$), determines the best candidate point

to improve the kriging model performance in predicting the limit state in regions of importance with respect to $f_{\mathbf{x}}(\mathbf{x})$.

2.2.8 REIF and REIF2

The reliability-based expected improvement functions REIF and REIF2 are acquisition functions based on the folded-normal distribution for structural reliability analysis (Zhang et al. 2019). The purpose of the REIF functions is to emphasize the conditions when $\left|\widehat{G}(\mathbf{x})\right|$ is close to zero to emphasize reliability. $\beta(\mathbf{x}) = \mu_{\widehat{G}}(\mathbf{x})/\sigma_{\widehat{G}}(\mathbf{x})$ and w is a constant parameter to adjust the relative weights of $\mu_{\widehat{G}}(\mathbf{x})$ and $\sigma_{\widehat{G}}(\mathbf{x})$. The value of w was set to 2 to stay consistent with the original paper. REIF2 is similar to REIF, but REIF2 multiplies each term by the PDF value $f_{\mathbf{x}}(\mathbf{x})$ of candidate point **x.** Maximizing REIF(**x**) or REIF2(**x**) determines the best candidate point to improve model performance.

$$\mathrm{REIF}(\mathbf{x}) = \mu_{\widehat{G}}(\mathbf{x})[1 - 2\Phi(\beta(\mathbf{x}))] + \sigma_{\widehat{G}}(\mathbf{x})\left[w - \sqrt{\frac{2}{\pi}}e^{-\frac{1}{2}\beta^2(\mathbf{x})}\right], \tag{23}$$

$$\mathrm{REIF2}(\mathbf{x}) = \mu_{\widehat{G}}(\mathbf{x}) \cdot f_{\mathbf{x}}(\mathbf{x}) \cdot [1 - 2\Phi(\beta(\mathbf{x}))] + \sigma_{\widehat{G}}(\mathbf{x}) \cdot f_{\mathbf{x}}(\mathbf{x}) \cdot \left[w - \sqrt{\frac{2}{\pi}}e^{-\frac{1}{2}\beta^2(\mathbf{x})}\right]. \tag{24}$$

As mentioned above, numerous acquisition functions have been proposed in recent years for kriging-based reliability analysis and RBDO. The above-reviewed acquisition functions are some representative examples and will be compared in this paper. Another reason that these acquisition functions are chosen for comparison is that they are quite easy to implement due to the close-form expressions. The authors recognize that many more methods are available to evaluate in the literature, including Wang et al. (2022b) and Moustapha et al. (2022). Although these approaches may produce comparable or better performance, they are treated out of scope for this comparative study due to time and resource constraints.

3 RBDO

Deterministic design optimization (DDO) finds an optimal design based on minimizing or maximizing the value of a performance function while respecting specified design constraints. However, DDO does not include information on the stochastic variability of design parameters or loading inputs. Conversely, RBDO accounts for the variability found in a system and determines an optimum that meets the design reliability targets. Sources of system variability include dimensional tolerances, material properties, and external loading conditions. In a traditional engineering design DDO, the objective is typically to minimize the weight or volume of a specific part to reduce part cost or improve the entire system's performance, such as increasing fuel economy. RBDO also finds a design optimum but verifies that the design meets the expected reliability goals to ensure the system performs dependably. For example, RBDO may add a small amount of material, increasing the weight to account for the input variability, making the part more robust, and increasing the part reliability from ~ 50% to a target reliability level (e.g., 99.87%). Evidence can be found in Fig. 22, discussed in the RBDO study on an engine oil pickup tube in Sect. 6. The results demonstrate that the optimization process eliminated abrupt changes at the corner, creating a smooth transition in stiffness that reduced the maximum stress values. This improvement illustrates the potential effect of RBDO in enhancing part robustness by mitigating stress concentrations.

Active kriging for RBDO is historically an extension of active kriging for traditional (global) optimization. The main difference is in the definition of an acquisition function, the maximization of which yields the next point for sampling. The essence of any acquisition function is the trade-off between exploration and exploitation, summarized in Table 1. When looking for a new point to sample next, both groups of acquisition functions explore regions in the design space where the predictive uncertainty of the kriging model is high. This exploration is done by maximizing $\sigma_{\widehat{f}}$ (standard optimization) or $\sigma_{\widehat{G}}$ (RBDO). Note that traditional optimization using active kriging deals with the problem of finding a global minimizer of an implicit, often computationally expensive objective function $f(\bullet)$. The difference between acquisition functions for traditional optimization and RBDO lies in how exploitation is defined. Acquisition functions for

Table 1 Criteria for exploration and exploitation in different fields (adapted from Sadoughi et al. 2018)

	Machine learning	Optimization	Reliability analysis
Exploration	Gather more information	Maximize $\sigma_{\widehat{f}}$	Maximize $\sigma_{\widehat{G}}$
Exploitation	Sample close to promising points	Maximize $\mu_{\widehat{f}}$	Minimize $\left\|\mu_{\widehat{G}}\right\|$ and maximize $f_{\mathbf{x}}$

traditional optimization take large values where the kriging model predicts low objective values. However, acquisition functions for RBDO typically minimize the value of $|\mu_{\widehat{G}}|$.

3.1 RBDO formulation

The formulation for a typical RBDO problem is given by (Hu et al. 2019):

$$\begin{aligned} &\text{minimize} f(\mathbf{d}) \\ &\text{subject to } R(\mathbf{d}) = \Pr\left[G_j(\mathbf{x},\boldsymbol{\theta};\mathbf{d}) \le 0\right] \ge \Phi\left(\beta^{\mathrm{t}}\right) = R^{\mathrm{t}}, \\ &\quad j = 1,\ldots,N_{\mathrm{c}},\ d_i^{\mathrm{L}} \le d_i \le d_i^{\mathrm{U}},\ i = 1,\ldots,N_{\mathrm{d}}. \end{aligned} \tag{25}$$

In the above, $f(\mathbf{d})$ is the objective function, such as the cost or weight of a candidate design characterized by the mean design vector $\mathbf{d} = \left(d_1,\ldots,d_{N_{\mathrm{d}}}\right)^{\mathrm{T}}$, where N_{d} is the number of design variables; the random design vector $\mathbf{x} = (x_1,\ldots,x_{N_{\mathrm{d}}})^{\mathrm{T}}$ has mean values that form $\mathbf{d}$, i.e., $\mathbf{d} = \boldsymbol{\mu}(\mathbf{x})$; the random parameter vector $\boldsymbol{\theta} = (\theta_1,\ldots,\theta_{N_{\mathrm{r}}})^{\mathrm{T}}$ has N_{r} random parameter variables that cannot be varied during optimization; $R(\mathbf{d})$ is the reliability level of the candidate design $\mathbf{d}$; G_j is the performance function of the jth design constraint, for $j = 1,\ldots,N_{\mathrm{c}}$, where N_{c} is the number of constraints; R^{t} is the target reliability level, corresponding to a target reliability index β^{t}; and d_i^{L} and d_i^{U} are the lower and upper bounds on the mean design variable d_i, for $i = 1,\ldots,N_{\mathrm{d}}$.

3.2 Local versus global approximation

There are two different approaches to creating a surrogate model for kriging-based RBDO. Figure 4 provides a schematic diagram comparing the local vs. global approximation. The *local approximation* defines a region around the current design rather than the entire bounded design space and builds kriging models to work well in this smaller region (Moustapha 2019). This region is also called the local window, and its center and sometimes size change dynamically with the design point throughout the optimization process. If we consider a normally distributed random variable x_i for $i = 1,\ldots,N_{\mathrm{d}}$, we can define a two-sided interval for this dimension as $[\mu_{x_i} - c_{\mathrm{local}}\Phi^{-1}\left(\beta^{\mathrm{t}}\right)\sigma_{x_i}, \mu_{x_i} + c_{\mathrm{local}}\Phi^{-1}\left(\beta^{\mathrm{t}}\right)\sigma_{x_i}]$, where c_{local} is a local window coefficient used to control the size of the neighborhood for the local approximation and $\Phi^{-1}(\bullet)$ is the inverse cumulative distribution function of the standard normal distribution. Now, the local window can be obtained as a tensor product of the intervals defined for the N_{d} design variables, expressed as

$$\Omega_{\mathrm{local}} = \prod_{i=1}^{N_d}[\mu_{x_i} - k\Phi^{-1}\left(\beta^{t}\right)\sigma_{x_i}, \mu_{x_i} + k\Phi^{-1}\left(\beta^{t}\right)\sigma_{x_i}], \tag{26}$$

Under the assumption that each design variable follows a normal distribution. The local window in Eq. (26) is a hypercube in the N_{d}-dimensional random input space. This smaller hypercube region allows fewer points to provide an acceptable fit for the kriging model. Latin hypercube sampling is

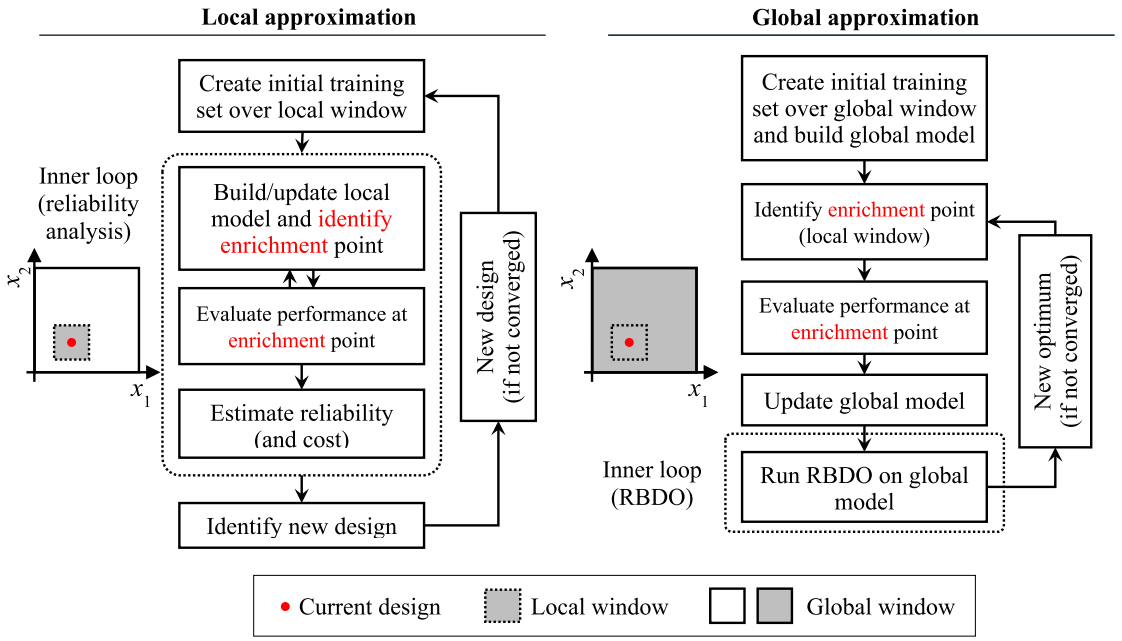


Fig. 4 A schematic diagram comparing the local approximation and global approximation approaches to kriging-based RBDO. At each design iteration, the local/global kriging model is built based on samples within the local/global window filled with a gray color

conducted over this region (here, Ω_{cand} in Eq. (8) is defined as Ω_{local} in Eq. (26)) to produce random candidate samples so that the acquisition functions can choose enrichment points to improve the model. For non-normally distributed design variables, candidate samples are first generated in the independent standard normal space (commonly referred to as the U-space) within a hypercube similar to the one defined in Eq. (26), but with means set to 0 and standard deviations set to 1. These candidate samples are then transformed to the **x**-space using the Rosenblatt transformation (Rosenblatt 1952).

In this comparative study, c_{local} was set to 1.3 to expand the local window, and a direct benefit is a reduction in extrapolation, i.e., fewer evaluations of the surrogate model on points outside the existing training sample distribution. This coefficient value was empirically determined by observing results in mathematical examples to balance accuracy and efficiency. It is also consistent with earlier kriging-based RBDO studies that attempted to expand local windows, such as the 1.2, 1.3, and 1.5 values used in Lee et al. (2011b). Ideally, c_{local} should be adaptively adjusted according to the nonlinearity of the limit state ($G = 0$) in the local window, which may vary from one design to the next during optimization (Lee et al. 2011b; Chen et al. 2014). However, dynamically updating c_{local} based on the local nonlinearity of the limit state is beyond the scope of this comparative study. Moreover, setting a fixed value of c_{local} over throughout the optimization process ensures fairness and facilitates a consistent apples-to-apples comparison across all acquisition functions. Additionally, some theoretical evidence shows that hyperspherical local windows yield improved efficiency over hypercubical windows, especially when the input dimensions are high (Lee et al. 2011b).

Once the kriging model is updated, the reliability and cost are calculated. If an optimum is not achieved, a new design is identified to create a new local approximation. A new hypercube is created around the new design. Then, the acquisition function identifies new enrichment points in the local window to improve the kriging model. A new design is again identified, and this process continues until the optimization converges. A local optimum, started at the direct design optimum, is identified as one that meets the reliability constraints, which is not necessarily the global optimum. The local approximation method can be used to reduce optimization time since the whole design space is not evaluated.

The *global approximation* builds a single kriging surrogate over the entire design space, often as a bounded hypercube with user-defined lower and upper bounds. This region can be large if many design variables exist and the bounds are wide. Unlike the local approximation, where a local surrogate model is only valid for the current design, the global approximation uses a single global surrogate to estimate the reliability of multiple designs in the bounded design space. Given the large design space, a kriging model can have difficulty fitting a single surrogate model that produces accurate predictions in all regions of interest in the design space. To improve the prediction accuracy of a global surrogate, the global approximation chooses new enrichment points within a local window around the current design. The enrichment points are added to the global training set to create a new global model. RBDO is conducted on the new global model, and if the optimization does not converge, a new local window is created to identify more enrichment points to improve the global surrogate model. This process continues until the optimization converges.

4 Methodology

Section 4.1 starts by reviewing the procedure to evaluate acquisition function performance. Then, Sect. 4.2 discusses the uncertainty indicator of the limit-state prediction, which is used as a stopping criterion. Section 4.3 details the procedure for evaluating the different acquisition functions. Section 4.4 provides details of the procedure implementation in MATLAB to aid RBDO practitioners. Section 4.5 elaborates on comparison criteria.

4.1 Procedure setup

Figure 5 depicts the analysis procedure used to compare the different acquisition functions. The process starts by conducting a DDO. Once the optimal point is found from the DDO, the initial kriging model is built using the Latin hypercube sampling with $N_d \times 4$ points where N_d equals the number of design variables. The acquisition function then evaluates ten thousand candidate points from an MCS to find the next best point to enrich the kriging model. A universal stopping criterion determines when to stop improving the kriging model. Once the model improvement is halted, the kriging model is used to perform RBDO. If the RBDO does not meet the reliability requirements, the new optimum point is used to create a new local kriging approximation. The process continues until the required reliability targets are met.

4.2 Uncertainty indicator of the limit-state prediction as a stopping criterion for local approximation

A challenge with comparing the different acquisition functions is that each may have its own stopping criterion. We attempt to reduce variability by applying the same stopping criteria to each acquisition function. The limit state divides

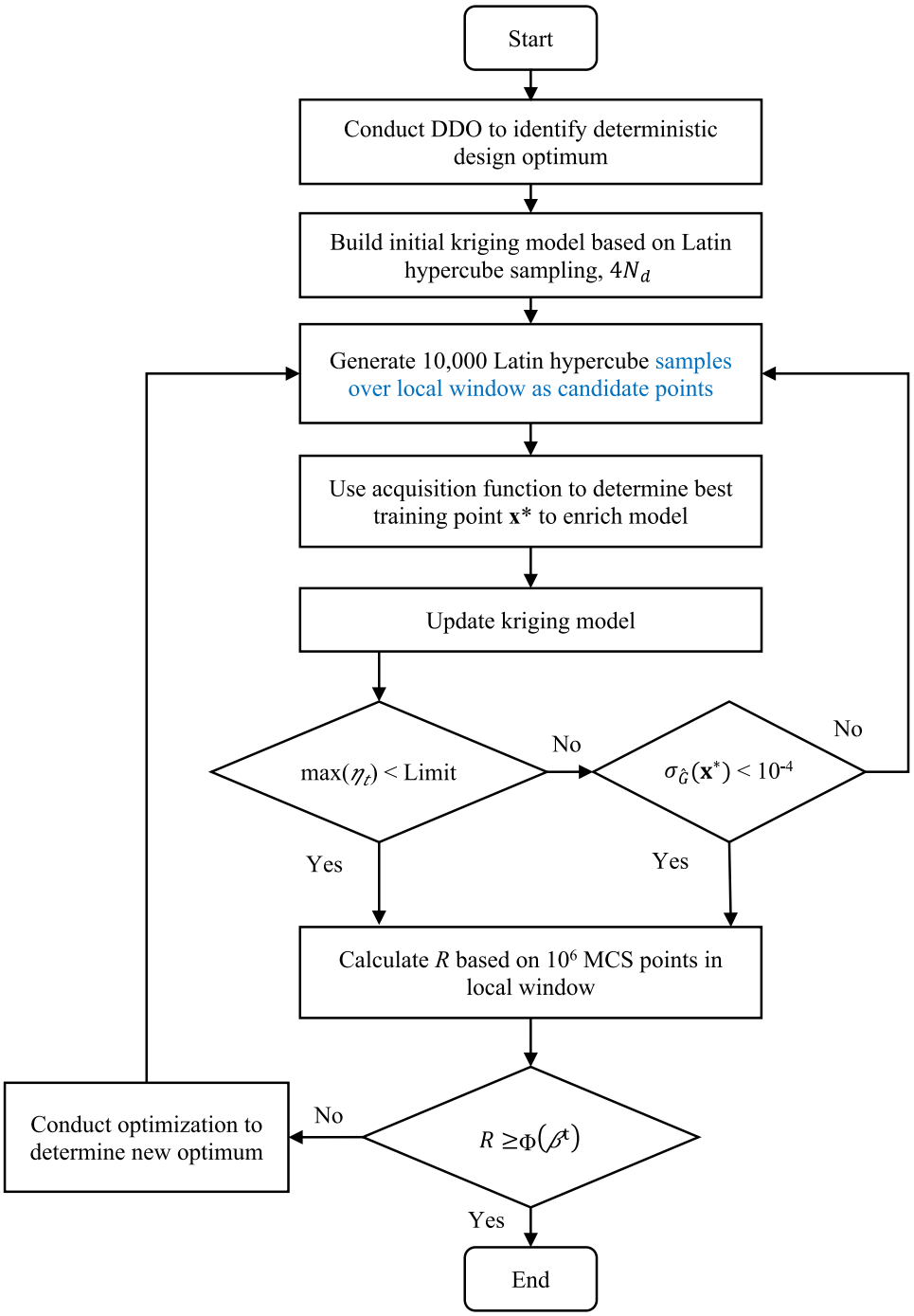


Fig. 5 Process for comparing different acquisition functions

the input ($\mathbf{x}$) space into acceptable (safe) and unacceptable (failure) regions. We use an uncertainty indicator η to quantify the uncertainty in the limit-state prediction (Sadoughi et al. 2018) and use that uncertainty measure as one of the stopping criteria. Once the value of η dropped below a predefined threshold η^{thr}, the point enrichment process stops.

$$\eta \approx \frac{1}{N_{\text{mcs}}} \sum_{i=1}^{N_{\text{mcs}}} I_{\Omega_{\text{ls}}^{\text{prob}}}(\mathbf{x}_i). \tag{27}$$

In the above, $\Omega_{\text{ls}}^{\text{prob}}$ is the probable region of the limit state as defined below Eq. (21) in Sect. 2.2.7. The above uncertainty indicator measures the overall uncertainty of the local kriging surrogate in predicting the limit state, weighted by the likelihood of a random realization of $\mathbf{x}$ falling within these regions.

We observed in experiments that the limit-state uncertainty criterion (i.e., $\eta < \eta^{\text{thr}}$) was not met at times, and the acquisition process continued to suggest new points;

however, these suggested points had the same $\mathbf{x}_{\text{new}}$ values. Adding these same points caused the covariance matrix not to be positive definite, leading to errors in computing the Cholesky decomposition. Fortunately, the kriging model provides not only the mean prediction $\mu_{\widehat{G}}(\mathbf{x})$ of the performance function at a candidate sample point $\mathbf{x}$ but also the predictive uncertainty $\sigma_{\widehat{G}}(\mathbf{x})$. We address the issue of adding the same $\mathbf{x}_{\text{new}}$ values by stopping the kriging model refinement process when the value of $\sigma_{\widehat{G}}(\mathbf{x}_{\text{new}})$ falls below 10^{-4}. Below this threshold, the surrogate model is considered sufficiently well defined within the local window, and additional enrichment is deemed unnecessary.

We acknowledge that many other stopping criteria can be used to determine when to stop model refinement. Many studies have defined a threshold directly on the acquisition function, comparing its maximum/minimum value (i.e., the value at $\mathbf{x}_{\text{new}}$) to this threshold as the stopping criterion. For example, the AK-MCS paper studied two stopping criteria: one for the EFF function and the other for the U function (Echard et al. 2011). In these criteria, the maximum value of the EFF function or the minimum value of the U function is compared against a predefined threshold. If the maximum value of EFF falls below the threshold or the minimum value of U exceeds the threshold, the refinement process stops. The work introducing MCE took a slightly different approach by defining a measure to quantify the average confidence in classifying a random realization of $\mathbf{x}$ into the safe or failure region (Wang and Wang 2014). Here, the stopping criterion involved comparing the value of this overall confidence measure to a predefined threshold and stopping model refinement when reliability analysis reached sufficient confidence. This stopping criterion shares a conceptual similarity with the uncertainty indicator criterion used in this comparative study (Sadoughi et al. 2018), as both criteria look at some measure of the average uncertainty or confidence in reliability analysis. It would be interesting to compare these types of stopping criteria with those in Echard et al. (2011), where some worst-case value (maximum or minimum), not an average value, is compared to a threshold. To keep our comparative study focused and manageable, we leave such comparisons for future research.

Additionally, our study used the same stopping criterion for all acquisition functions to keep the comparison fair and consistent. However, we believe some acquisition functions might work better with stopping criteria specifically tailored to their strengths or goals. For example, a separate comparative study could be conducted where the stopping criterion for each acquisition function compares its worst-case value (maximum or minimum) to a threshold customized for that function. Such stopping criteria would be similar to those used for the EFF and U functions in the AK-MCS paper (i.e., Echard et al. 2011). A difficult aspect would be introducing mechanisms that ensure consistency across acquisition functions in maintaining a minimum level of accuracy in reliability analysis where model refinement stops. Such comparative studies represent an interesting direction to explore further, and we also leave these studies for future work.

4.3 Size of failure domain as a stopping criterion for global approximation

The local window for the global approximation is continually changing. This requires changing the stopping criteria because the uncertainty around the limit state will be continually changing as well. For the global approximation, we use the change in size of the failure domain as the stopping criterion for kriging model enrichment (UQLab 2022). When the size of the failure domain for the current iteration N_{f}^{i} changes less than δ^{thr} from the previous iteration N_{f}^{i-1}, the enrichment process stops. This stopping criterion can be expressed as

$$\frac{\left|N_{\text{f}}^{i} - N_{\text{f}}^{i-1}\right|}{N_{\text{f}}^{i-1}} < \delta^{\text{thr}}. \tag{28}$$

We use a value of $\delta^{\text{thr}} = 0.01$ as the stopping threshold for the global approximation. This level was chosen to be similar to the local threshold level of $\eta^{\text{thr}} = 0.01$.

4.4 MATLAB implementation

MATLAB 2022a was used to implement the comparative study. MATLAB offers the flexibility to program multiple acquisition functions while having efficient functions for optimization and fitting the kriging (referred to as Gaussian process regression in MATLAB) models. This section describes the optimization and kriging functions.

4.4.1 Optimization parameters

The optimization function in MATLAB called fmincon finds the minimum of a constrained nonlinear multivariable function. The function fmincon has been used in many RBDO studies (UQLab 2022; Chaudhuri et al. 2020; Kusano et al. 2014;

Table 2 Parameter options for fmincon

Parameter	Value
Algorithm	Interior-point
ConstraintTolerance	1e−6
MaxFunEvals	500
OptimalityTolerance	1e−6
Optimizer	fmincon
StepTolerance	1e−6

Pearce et al. 2020). The options implemented in fmincon are listed in Table 2. We implemented the interior-point method because we found it less likely to converge to a local minimum at higher standard deviation values in case studies 1 and 2 (see Sects. 5.1 and 5.2) when compared to sequential quadratic programming (sqp). This way, we could provide the most consistent comparison between acquisition functions. However, sqp does offer the possibility of faster and more accurate results. The sqp algorithm is preferred if the standard deviations on the input parameters are small (MathWorks 2022c).

4.4.2 Parameters of kriging (Gaussian process regression)

The kriging function in MATLAB called fitrgp fits a kriging (Gaussian process regression) model to a training dataset. The parameter options for fitrgp are listed in Table 3. The MATLAB function fitrgp was compared to the Python function "scikit-learn.GaussianProcessRegressor." For the noise-free setting, which is common in RBDO, the two functions produce similar results, as shown in Figs. 24 and 25 in Appendix 1. MATLAB's fitrgp was chosen because it is considered a thoroughly tested function for industrial applications. The kernel function performance of "squaredexponential" (see the functional form of this isotropic kernel in Eq. (5)) was compared to "ardsquaredexponential" (see the functional form of this anisotropic kernel in Eq. (8)). In Appendix 2, the acquisition functions of U and EFF were compared for local and global approximation. The "squaredexponential" kernel optimization converged at the same rate or quicker for the local approximation (Figs. 26 and 27). The "ardsquaredexponential" kernel performed better for the global approximation (Figs. 28 and 29). The "ardsquaredexponential" kernel performed better on the larger global design space because it is important to consider the importance/smoothness of each dimension separately. For the local approximation, the design space is much smaller, and the dimensional importance of the kernel is less critical. The squared exponential kernel option was chosen because it was the most robust at minimizing Cholesky errors during implementation. The SigmaLowerBound parameter of fitrgp sets the lower bound on the noise standard deviation for fitting kriging models to noisy observations. This parameter was set to a value of 1e−5 to keep the model noise level low, ensuring consistency with the noise-free setting of RBDO. Additionally, this choice increases optimization efficiency by preventing the selection of the same enrichment point in consecutive iterations of sequential sampling. However, it should be noted that MATLAB limits the noise standard deviation parameter Sigma to be larger than SigmaLowerBound by 1e−6 when the parameter ConstantSigma is set to true. This limits the noise floor of fitrgp to 1e−6 (MathWorks 2022d).

Table 3 Parameter options for fitrgp

Parameter	Value
Basis	None
ConstantSigma	True
FitMethod	exact
Gaussian Model	fitrgp
KernelFunction	squaredexponential
KernelParameters	Varied
PredictMethod	Exact
Sigma	1e−5 + 1e−6
SigmaLowerBound	1e−5
Standardize	true

4.5 Comparison criteria

We defined several performance metrics to determine which acquisition functions performed the best. These performance metrics were (1) the total number of optimizations completed (denoted as *OptsComplete*), (2) the total number of function evaluations (*FE*), and (3) an overall metric to combine the multiple factors (*Overall*). The overall metric was defined as the multiplication of the average number of function evaluations ($\overline{FE}$) by the standard deviation of the number of function evaluations ($\sigma(FE)$) for ten replications, divided by the total number of completed optimizations ($\sum OptsComplete$). A lower overall value indicates the optimizer and acquisition functions were more computationally efficient (low $\overline{FE}$) and robust to the randomness in the generation of initial training data (low $\sigma(FE)$ and high $\sum OptsComplete$) when finding the optimum at the desired reliability target. Since Case Study 3 required a high number of function evaluations, each case study needed to be normalized so that Case Study 3 did not dominate the overall results. Each case study was grouped by the uncertainty threshold level.

The majority of optimizations that were unsuccessful converged to an infeasible point, with the following MATLAB error message: "fmincon stopped because the size of the current step is less than the value of the step size tolerance, but constraints are not satisfied to within the value of the constraint tolerance." This error occurred when fmincon would converge on a local minimum that did not meet the reliability requirements such as 0.9987. For example, the reliability of the three constraints for Case Study 1 would be (0.9829, 0.9898, 0.9981).

Taking Case Study 1 as an example (details are given in the subsequent section), the overall metric for this case study with $\eta^{\text{thr}} = 0.01$ is summarized in Table 4. The values in Table 4 are normalized in Table 5 by dividing the overall metric for each combination of acquisition function and standard deviation by the minimum value for that group. We then calculate a cumulative score for both the local and global approximation by summing all the normalized overall values for each acquisition function by approximation type.

Table 4 Overall metric for case study #1 $\eta^{\text{thr}} = 0.01$ (minimum highlighted)

σ	EFF	EFF missing ϵ	U	MCE	MMCE	H	SEEDT	REIF	REIF2
0.2	35.1	21.9	24.0	19.8	12.8	23.3	19.0	34.1	17.9
0.3	37.4	36.8	20.2	12.9	27.1	44.6	12.4	42.6	27.5
0.4	25.7	28.9	38.9	17.6	19.2	29.4	16.2	36.7	23.6
0.5	52.7	31.1	61.9	25.3	73.1	46.2	41.7	34.7	36.4
0.6	47.7	31.9	52.5	30.6	47.6	61.8	116.6	49.4	59.0

Table 5 Normalized overall metric for case study #1 $\eta^{\text{thr}} = 0.01$ (minimum highlighted)

σ	EFF	EFF missing ϵ	U	MCE	MMCE	H	SEEDT	REIF	REIF2
0.2	2.8	1.8	1.9	1.6	1.0	1.9	1.5	2.8	1.4
0.3	3.0	3.0	1.6	1.0	2.2	3.6	1.0	3.4	2.2
0.4	2.1	2.3	3.1	1.4	1.6	2.4	1.3	3.0	1.9
0.5	4.3	2.5	5.0	2.0	5.9	3.7	3.4	2.8	2.9
0.6	3.9	2.6	4.2	2.5	3.8	5.0	9.4	4.0	4.8

$$Overall = \frac{\overline{FE} * \sigma(FE)}{\sum OptsComplete}. \tag{29}$$

5 Case studies

In this section, we employ three mathematical and one practical case study to evaluate the performance of the nine acquisition functions described in Sect. 2.2. Case Study 1 is a classic example of RBDO evaluations with a two-parameter linear cost function and three mildly nonlinear constraints. Case Study 2 is a more nonlinear version of Case Study 1. It has a nonlinear two-parameter cost function, and the second constraint is highly nonlinear, challenging the optimizer and model fitting. Case Study 3 is another classical case study for crashworthiness with seven design variables and ten constraints. Finally, Case Study 4 is an example of an engineering application.

5.1 Case study 1: 2D RBDO benchmark

The first case study is a mathematical problem widely used in RBDO literature that contains three performance functions described by the following equations and depicted in Fig. 6 (Shan and Wang 2008; Kim and Song 2021; Moustapha et al. 2016; Yang et al. 2022). The DDO was started at $\mathbf{d}_0 = (5,5)$ and required 42 (14×3) function evaluations to reach the minimum value of 5.177 at (3.114, 2.063). Figure 6 shows the progress from starting point $\mathbf{d}_0$ to the DDO optimum and then to the RBDO optimum. Both x_1 and x_2 follow a normal distribution.

We compare the acquisition functions with different standard deviation levels on the input design parameters and

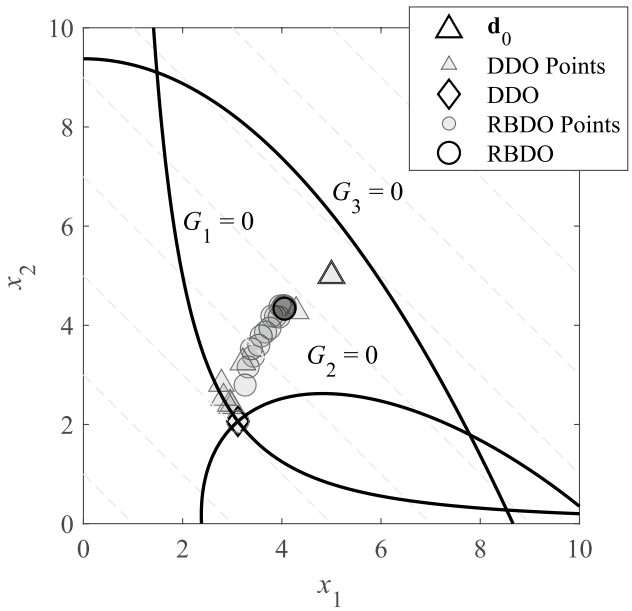


Fig. 6 Case Study 1—optimization progression

stopping criteria. Changing these parameters helps determine the robustness of the acquisition functions. We test the surrogate uncertainty threshold at three levels: 0.005, 0.01, and 0.03. We want to determine how the uncertainty indicator stopping level affects the number of function evaluations. Does the level of 0.005 provide more accurate results with the trade-off of increased function evaluations, while the level of 0.03 provides a less accurate model with fewer function evaluations?

We also compare different design input standard deviation levels to test how the acquisition functions behave with higher uncertainties. When dealing with RBDO, many papers use standard deviation values in the range of 0.2 to

0.3 (Lee and Chen 2009; Yousefpour et al. 2024; Garrido-Merchán and Hernández-Lobato 2020). We increase the standard deviation up to 0.6 by evaluating values of 0.2, 0.3, 0.4, 0.5, and 0.6. The higher sigma values stress the optimizer and acquisition functions more to evaluate robustness.

Each combination of acquisition function, uncertainty indicator, and input standard deviation level is repeated ten times to understand the robustness of the optimizer and the variability in the total number of function evaluations. The random seed for each repetition numbered one through ten was controlled so that the initial training points did not influence how the acquisition functions performed. For example, all replications labeled #1 started with the same initial Latin hypercube training points.

Case study 1: RBDO problem formulation

$$\underset{\mathbf{d}}{\text{minimize}}\ f(\mathbf{d}) = d_1 + d_2$$

$$\text{subject to } R = \Pr\left[G_i(\mathbf{x};\mathbf{d}) \le 0\right] \ge R^{\text{t}} = 0.9987, i = 1-3 \quad 0 \le d_1 \le 10,\ 0 \le d_2 \le 10$$

$$G_1(\mathbf{x}) = 1 - \frac{x_1^2 x_2}{20}$$

$$G_2(\mathbf{x}) = 1 - \frac{\left(x_1 + x_2 - 5\right)^2}{30} + \frac{\left(x_1 - x_2 - 12\right)^2}{120}$$

$$G_3(\mathbf{x}) = 1 - \frac{80}{x_1^2 + 8x_2 + 5}$$

The difference in training point behavior for EFF and EFF missing ϵ is shown in Fig. 7 for the same initial training points. When the ϵ term is missing from the equation, the training points are concentrated on regions close to the limit states, limiting the exploration of the design space by grouping many points in a similar location.

5.1.1 Case Study 1: local approximation results

Table 6 shows that 98.67% (1332 out of 1350) optimizations were completed successfully. All optimizations that did not reach the reliability target of 0.9987 had an error computing the Cholesky matrix during the optimization process. The error occurred because the matrix input into the Cholesky algorithm was not positive definite or positive semi-definite. The values that caused the errors were approaching the state of numerical round-off error. The minimum eigenvalues for the runs that produced errors ranged from − 6.4e−10 to 1.17e−11. For consistency, we did not attempt to correct the round-off error by adding a small value to the diagonal of the input matrix. *U* and REIF2 completed all optimizations successfully. SEEDT had the lowest success rate at 96% completing 144 optimizations.

Table 7 compares the first-order reliability method (FORM) (Hu et al. 2019) to the average value of the successful active kriging optimization for different input standard deviations. The average value of all acquisition functions was used to condense the table since the results were similar. Full results per acquisition function for Case Studies 1 and 2 for $\sigma = 0.6$ are available in Appendix 3. The FORM results and active kriging results were comparable for all sigma input levels.

The average number of function evaluations is plotted as bars in Fig. 8, where the standard deviation is represented by the black line around the average number of function evaluations. The number of function evaluations increases as the input standard deviation is increased. The high number of function evaluations occurs because the higher input standard deviation is more challenging for the optimizer because the data are more scattered. The bars with the * above them indicate the lowest number of average function evaluations plus one standard deviation. MMCE and SEEDT had the best performance for input standard deviations of 0.2 to 0.4.

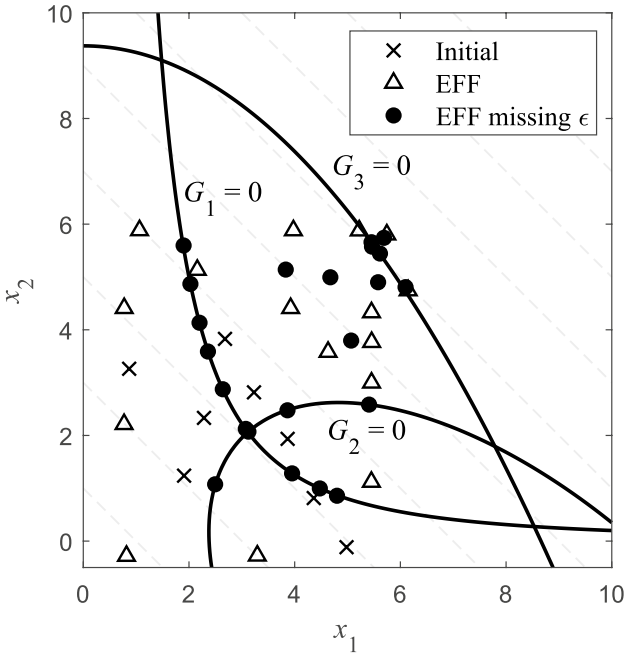


Fig. 7 Comparison of training points for EFF and EFF missing ϵ

Table 6 Completed optimizations for Case Study 1 local approximation (10 repetitions)

Uncertainty	σ	EFF	EFF missing ϵ	U	MCE	MMCE	H	SEEDT	REIF	REIF2
0.030	0.2	10	10	10	10	10	10	10	10	10
	0.3	10	10	10	10	10	10	10	10	10
	0.4	10	10	10	10	10	10	10	10	10
	0.5	10	10	10	10	10	10	10	10	10
	0.6	10	9	10	10	9	10	9	10	10
0.010	0.2	10	10	10	10	10	10	10	10	10
	0.3	10	10	10	10	10	10	10	10	10
	0.4	10	10	10	10	10	10	10	10	10
	0.5	10	10	10	10	9	10	9	10	10
	0.6	10	9	10	10	10	9	9	10	9
0.005	0.2	10	10	10	10	10	10	10	10	10
	0.3	10	10	10	10	10	10	10	10	10
	0.4	10	10	10	10	10	10	10	10	10
	0.5	9	9	10	10	10	10	8	10	10
	0.6	10	9	10	10	8	10	9	9	10

Table 7 Comparison of FORM and local active kriging for Case Study 1

σ	Function	$f(\mathbf{d})$	d_1	d_2	$\Pr[G_1(\mathbf{x}) \le 0]$	$\Pr[G_2(\mathbf{x}) \le 0]$	$\Pr[G_3(\mathbf{x}) \le 0]$
0.2	FORM	6.1963	3.2959	2.9004	0.9987	0.9987	0.9999
	AK	6.2020	3.3051	2.8969	0.9987	0.9987	1.0000
0.3	FORM	6.7318	3.4409	3.2909	0.9987	0.9987	0.9998
	AK	6.7365	3.4576	3.2790	0.9987	0.9987	1.0000
0.4	FORM	7.2766	3.6117	3.6649	0.9987	0.9987	0.9996
	AK	7.2794	3.6353	3.6441	0.9987	0.9987	1.0000
0.5	FORM	7.8265	3.8028	4.0237	0.9987	0.9987	0.9992
	AK	7.8250	3.8334	3.9916	0.9987	0.9987	1.0000
0.6	FORM	8.3787	4.0101	4.3687	0.9987	0.9987	0.9988
	AK	8.3741	4.0473	4.3269	0.9987	0.9987	0.9993

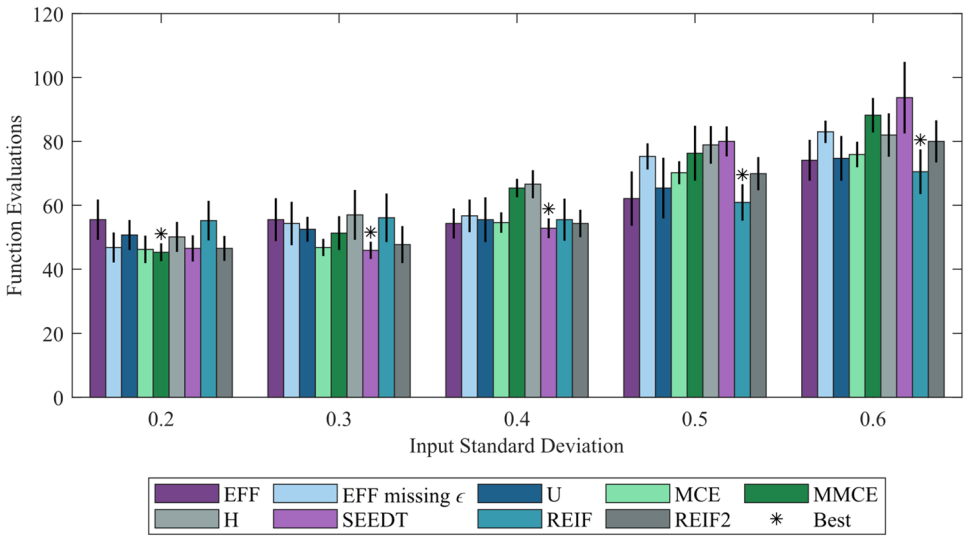


Fig. 8 Average number of function evaluations for surrogate accuracy threshold = 0.01 for Case Study 1

REIF performed the best for input standard deviations of 0.5 and 0.6. Next, Fig. 9 shows the trend of the number of function evaluations based on the surrogate model accuracy threshold. The number of function evaluations does not change significantly for the lower input standard deviation levels of 0.2 and 0.3. The number of function evaluations increases more at higher input standard deviation levels. For example, for the level of 0.6, the average number for

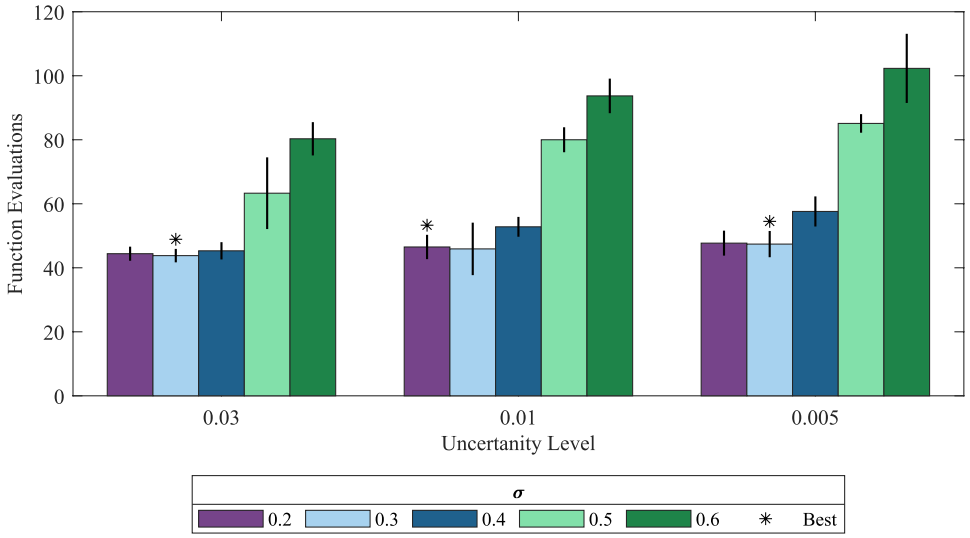


Fig. 9 Number of function evaluations for SEEDT based on surrogate uncertainty threshold for Case Study 1

function evaluations increases from 80.3 at 0.03 to 93.7 at 0.01 to 102.3 at 0.005. This change could be significant if the model is computationally expensive, like a complex FEM.

Additionally, we conducted parametric studies on the effects of the local window coefficient c_{local} on the accuracy and efficiency of reliability analysis, which would directly influence the accuracy of RBDO in locating a design optimum and the efficiency of this optimization process. These parametric studies were conducted on the mathematical constraints in the first two case studies, and the results can be found in Tables. 19 and 20 under Appendix 4. A few important observations can be made from these two tables.

- *First*, too small of a c_{local} value (e.g., 0.5 or 0.7 for Case Study 1 and 0.5, 0.7. or 1.0 for Case Study 2) led to large reliability analysis errors in some or all constraints.
- *Second*, a larger c_{local} value means that the kriging surrogate model needs to be built over a larger local window, necessitating more function evaluations to meet the stopping criteria (Sect. 4.2) and generally results in greater accuracy, especially for the limit states that are more nonlinear in the local window (e.g., G_1 and G_2 in Table 20). However, beyond a certain point—such as when $c_{\text{local}} \geq 1.0$ for $G_1 - G_3$ in Case Study 1 and G_3 in Case Study 2, or $c_{\text{local}} \geq 1.3$ for the more nonlinear cases G_1 and G_2 in Case Study 2—the improvement in accuracy diminishes, offering minimal additional benefit.
- *Third*, based on the second observation, the local window size should ideally be dynamically adjusted according to the limit state's nonlinearity, as discussed in Sect. 3.2 and suggested in earlier kriging-based RBDO studies such as Lee et al. (2011b) and Chen et al. (2014).

5.1.2 Case Study 1: global approximation results

The analysis was repeated for the global approximation and is presented in Table 8. Comparing the global approximation to the local approximation for the threshold level of 0.01, we observe that the global approximation completed 115 fewer optimizations (329) than the local approximation (444). Due to the reduced fidelity in the region of interest, the optimization using the global approximation often stops before satisfying the constraints due to the step size falling below a tolerance limit. Similar to the local approximation, the acquisition functions containing a term for the joint PDF completed the most optimizations.

The limit-state predictions are plotted in Fig. 10 to provide insight into how the acquisition functions EFF and EFF missing ϵ perform. The local approximation of EFF (Fig. 10a) shows that EFF explores the design space while EFF missing ϵ (Fig. 10b) concentrates the points

Table 8 Completed optimizations for Case Study 1 global approximation (10 repetitions)

σ	EFF	EFF missing ϵ	U	MCE	MMCE	H	SEEDT	REIF	REIF2
0.2	7	9	8	10	10	7	10	7	9
0.3	5	7	4	10	9	6	10	5	8
0.4	6	6	5	9	9	7	10	6	7
0.5	4	7	8	7	9	4	9	5	9
0.6	4	8	6	9	8	2	9	6	9

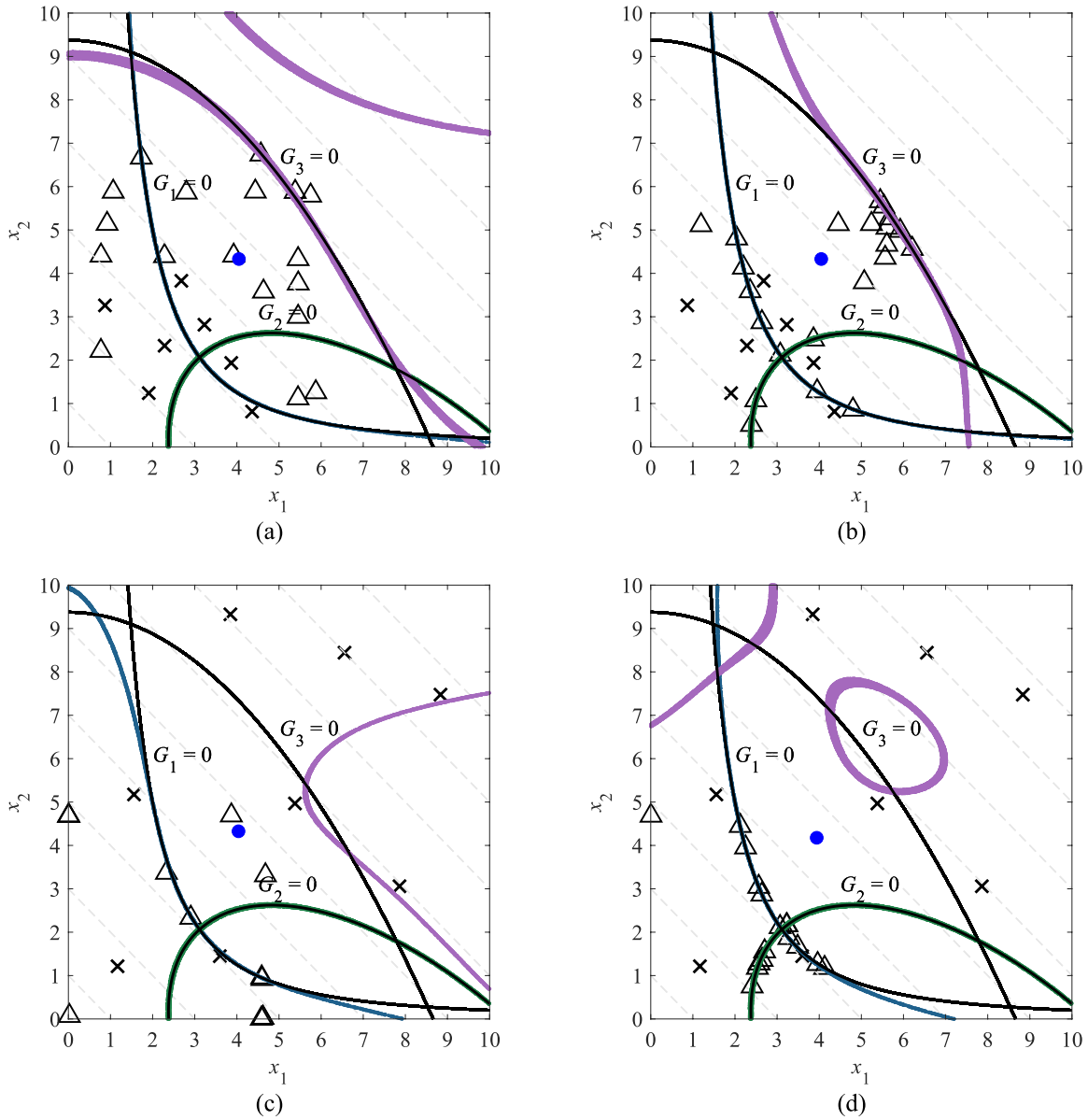


Fig. 10 Case Study 1 Limit-State Approximation of G1 (blue), G2 (green), and G3 (purple) for **a** EFF Local, **b** EFF missing ϵ Local, **c** EFF Global, and **d** EFF missing ϵ Global. • current design optimum, △ training points, × initial training points. Refinement of the kriging models for these methods is visualized in the Supplementary Material files named "Animation for Fig. 10a.mp4," "Animation for Fig. 10b.mp4," "Animation for Fig. 10c.mp4," and "Animation for Fig. 10d.mp4," respectively. (Color figure online)

in proximity to the optimum limiting exploration reducing the accuracy of G_3. The global approximations of EFF (Fig. 10c) and EFF missing ϵ further reinforces this behavior. The global approximation of EFF explores the whole design space placing several points on the boundaries of the design space. EFF missing ϵ (Fig. 10d) again concentrates the points closer to the optimum.

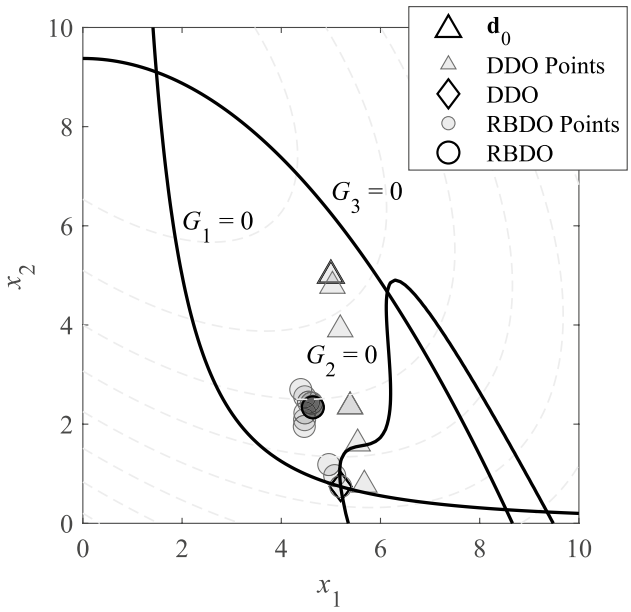


Fig. 11 Case Study 2—optimization example

5.2 Case Study 2: 2D example with nonlinearity

The second case study considers a nonlinear system with a change in the constraint for $G_2(\mathbf{x})$ that is described by the following equations and shown in Fig. 11 (Hu et al. 2019; Youn et al. 2005). The second case study is a more difficult problem to solve than Case Study 1 due to the added nonlinearity and DDO optimum located close to the lower bound. This DDO was started at $\mathbf{d}_0 = (5,5)$ and required 117 ($= 39 \times 3$) function evaluations to reach the minimum value of -1.191 at (5.197, 0.741). Both x_1 and x_2 follow a normal distribution.

Table 9 Completed optimizations for Case Study 2 local approximation (10 repetitions)

Uncertainty	σ	EFF	EFF missing ϵ	U	MCE	MMCE	H	SEEDT	REIF	REIF2
0.030	0.2	10	10	10	10	10	10	10	10	10
	0.3	10	8	10	10	10	10	10	10	10
	0.4	10	8	9	10	9	10	10	10	10
	0.5	10	7	10	10	9	10	10	9	10
	0.6	10	7	10	10	10	8	10	9	9
0.010	0.2	10	9	10	10	10	10	10	10	10
	0.3	10	8	9	10	10	10	10	10	10
	0.4	10	10	9	10	9	10	10	10	10
	0.5	10	6	10	10	9	10	10	9	10
	0.6	9	8	9	9	10	10	9	9	9
0.005	0.2	10	9	10	10	10	9	10	10	10
	0.3	10	8	9	10	10	10	10	10	10
	0.4	10	10	8	9	9	10	10	10	10
	0.5	10	6	10	10	9	10	10	9	10
	0.6	10	6	10	9	10	9	9	9	10

Table 10 Comparison of FORM and local active kriging for Case Study 2

σ	Function	$f(\mathbf{d})$	d_1	d_2	$\Pr[G_1(\mathbf{x})\leq 0]$	$\Pr[G_2(\mathbf{x})\leq 0]$	$\Pr[G_3(\mathbf{x})\leq 0]$
0.2	FORM	-2.0275	4.7983	1.2964	0.97725	0.97725	0.99774
	AK	-2.0257	4.8153	1.2989	0.97791	0.97812	1.00000
0.3	FORM	-1.9526	4.9681	1.4455	0.97725	0.97725	0.99571
	AK	-1.9076	4.7339	1.5525	0.97725	0.97726	1.00000
0.4	FORM	-1.8762	5.0939	1.6125	0.97725	0.97725	0.99314
	AK	-1.7942	4.6986	1.8019	0.97725	0.97725	1.00000
0.5	FORM	-1.7811	5.0782	1.8304	0.97398	0.95781	0.99131
	AK	-1.6866	4.6930	2.0478	0.97727	0.97727	1.00000
0.6	FORM	-1.6732	4.9723	2.0837	0.97725	0.97264	0.98856
	AK	-1.5719	4.6651	2.3228	0.97733	0.97731	0.99998

Case Study 2: RBDO problem formulation

$$\underset{\mathbf{d}}{\text{minimize}}\, f(\mathbf{d}) = -\frac{\left(d_1 + d_2 - 10\right)^2}{30} - \frac{\left(d_1 - d_2 + 10\right)^2}{120}$$

$$\text{subject to } \Pr\left[G_i(\mathbf{x};\mathbf{d}) \le 0\right] \ge R^{\mathrm{t}} = 0.97725, \quad i = 1-3 \quad 0 \le d_1 \le 10,\ 0 \le d_2 \le 10$$

$$G_1(\mathbf{x}) = 1 - \frac{x_1^2 x_2}{20}$$

$$G_2(\mathbf{x}) = -1 - \left(z_1 - 6\right)^2 + \left(z_1 - 6\right)^3 - 0.6 \times \left(z_1 - 6\right)^4 + z_2$$

$$G_3(\mathbf{x}) = 1 - \frac{80}{x_1^2 + 8x_2 + 5}$$

where

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0.9063 & 0.4226 \\ 0.4226 & -0.9063 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

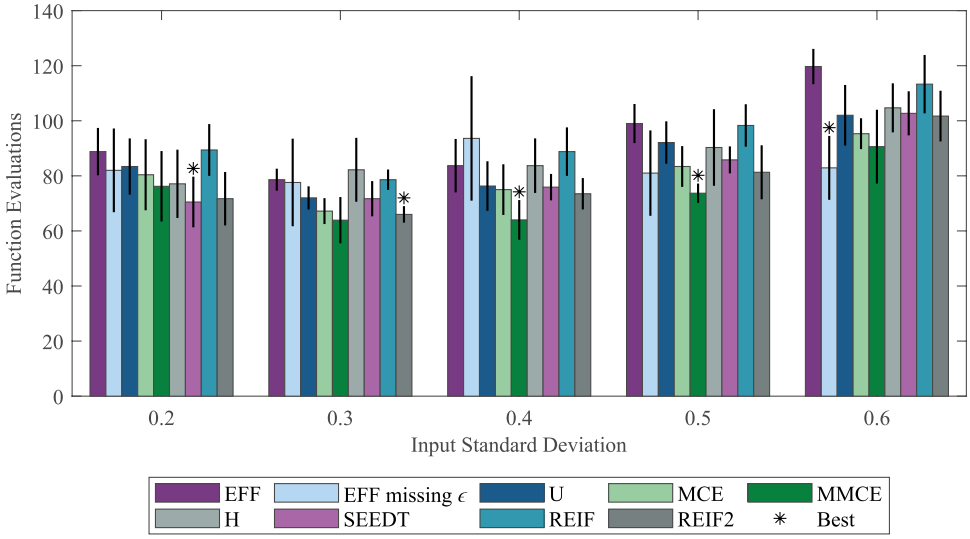


Fig. 12 Number of function evaluations for surrogate uncertainty threshold = 0.01 for Case Study 2

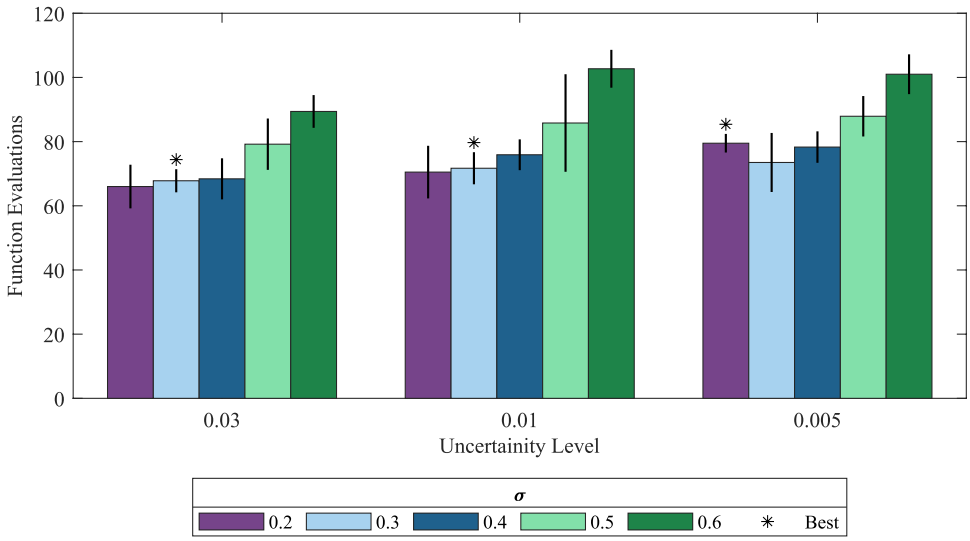


Fig. 13 Number of function evaluations for SEEDT based on surrogate uncertainty threshold for Case Study 2

Table 11 Comparison of FORM and local active kriging for Case Study 2

σ	EFF	EFF missing ϵ	U	MCE	MMCE	H	SEEDT	REIF	REIF2
0.2	8	2	6	10	7	6	9	8	9
0.3	4	2	6	6	8	7	8	5	7
0.4	5	2	6	7	5	7	7	6	7
0.5	4	2	6	8	6	5	6	6	9
0.6	6	5	6	9	3	4	7	6	7

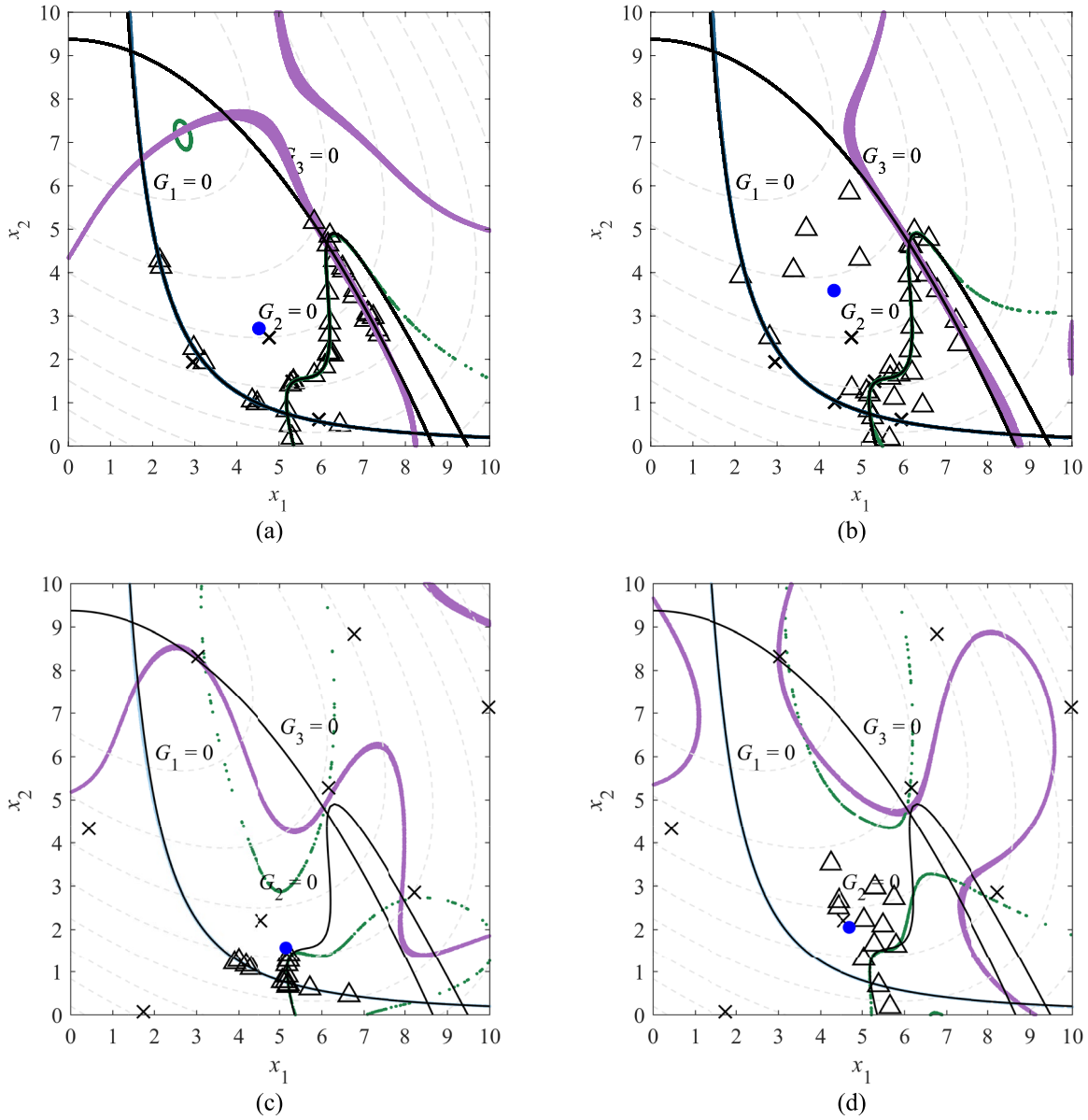


Fig. 14 Case Study 2 limit-state approximation of G1 (blue), G2 (green), and G3 (purple) for **a** EFF missing ϵ local, **b** MCE local, **c** EFF missing ϵ Global, and **d** MCE Global. • current design optimum, △ training points, × initial training points. Refinement of the kriging models for these methods is visualized in the Supplementary Material files named "Animation for Fig. 14a.mp4," "Animation for Fig. 14b.mp4," "Animation for Fig. 14c.mp4," and "Animation for Fig. 14d.mp4," respectively. (Color figure online)

5.2.1 Case Study 2: local approximation results

Table 9 shows that 95.48% (1289 out of 1350) optimizations were completed successfully for Case Study 2. EFF completed 149 out of 150 optimizations successfully. Conversely, EFF missing ϵ had the lowest success rate at 80% completing 120 out of 150 optimizations. FORM (Table 10) performed better for the case of 0.2. FORM and active kriging were the most accurate at finding the target reliability for the input standard deviation of 0.3 and 0.4. Active kriging outperformed FORM for higher input standard deviations, finding a lower minimum value while meeting the target reliability goals. SEEDT (0.2), REIF (0.3), MMCE (0.4, 0.5), and EFF missing ϵ had the lowest number of function evaluations when the optimizations were completed. When considering the standard deviation, shown in Fig. 12, there was no clear trend between the average number of function evaluations and input standard deviation for Case Study 2. Figure 13 again shows that as the surrogate uncertainty threshold decreases, the number of function evaluations increases, similar to Case Study 1.

5.2.2 Case Study 2: global approximation results

The analysis was repeated for the global approximation and is presented in Table 11. Comparing the global approximation to the local approximation for the threshold level of 0.01, we observe that the global approximation completed 65 fewer optimizations (275) than the local approximation (340). The performance of EFF missing ϵ had the biggest reduction in performance. This is due to EFF missing ϵ not exploring the whole design space.

For Case Study 2, Fig. 14 shows the limit-state predictions for EFF missing ϵ and MCE to illustrate why EFF missing ϵ completed fewer optimizations. The EFF missing ϵ local approximation (Fig. 14a) has difficulty exploring the design space. EFF missing ϵ concentrates the training points on the limit-state boundary in regions of low uncertainty, not allowing the optimizer to find a design point to satisfy the constraints. Conversely, the MCE local approximation (Fig. 14b) balances points near the limit states with points of high uncertainty to accurately define the limit states in the region of interest. The global approximations of EFF missing ϵ (Fig. 14c) and MCE (Fig. 14d) do not define the limit states as well compared to the local approximation. The global approximation has lower fidelity in the region around the design optimum due to the kriging model attempting to fit a model over the entire design space. For example, MCE global (Fig. 14d) does not approximate the second limit state, $G_2 = 0$, as well as MCE Local (Fig. 14b).

5.3 Case Study 3: crashworthiness example

The mathematical example for crashworthiness of vehicle side impact is the higher-dimensional RBDO benchmark (Youn et al. 2004; Gu et al. 2001). This case study typically consists of nine design parameters concerning vehicle design and two random parameters that account for test variability. This example minimizes the weight of the vehicle while improving safety performance. Since we are conducting a comparative study, design variable x_{10} (height of barrier) and variable x_{11} (height of impact) were set to a mean value of zero. We did not want to add uncertainty to the comparison by treating them as random variables. The optimization function fmincon requires continuous variables, but the material properties for mild steel (0.192 GPa) and high-strength steel (0.345 GPa) are discrete variables. Based on the optimum from two previous papers (i.e., Youn et al. 2004; Gu et al. 2001), the parameter d_8 for mild steel was set to 0.192, and d_9 for high-strength steel was set to 0.345. The problem includes ten constraints that are represented by the following equations. Table 12 summarizes the initial starting point, lower bound (*lb*), upper bound (*ub*), and standard deviation. All variables have a normal distribution. The DDO required 12 iterations to reach an optimum of

Table 12 Optimization inputs (units: d_1–d_7, d_{10}-d_{11} in mm, d_8 and d_9 in GPa)

	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}
d_0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.345	0.192	0	0
Lb	0.5	0.5	0.5	0.5	0.5	0.5	0.5				
ub	1.5	1.5	1.5	1.5	1.5	1.5	1.5				
σ	0.03	0.03	0.03	0.03	0.03	0.03	0.03				

Table 13 Completed optimizations for Case Study #3 local approximation (10 repetitions)

Uncertainty	EFF	EFF miss ϵ	U	MCE	MMCE	H	SEEDT	REIF	REIF2
0.030	10	10	10	10	10	10	10	10	10
0.010	10	10	10	10	10	10	10	10	10
0.005	10	10	10	10	10	10	10	10	10

Table 14 Comparison of FORM and active kriging for Case Study 3

η^{thr}	Threshold	Weight	x_1	x_2	x_3	x_4	x_5	x_6	x_7
FORM		24.3604	0.5000	1.3255	0.5000	1.2905	0.5950	1.5000	0.5000
AK-MCS	0.03	24.3588	0.5000	1.3250	0.5000	1.2908	0.5953	1.5000	0.5000
	0.01	24.3583	0.5000	1.3250	0.5000	1.2907	0.5953	1.5000	0.5000
	0.005	24.3582	0.5000	1.3250	0.5000	1.2907	0.5952	1.5000	0.5000

η^{thr}	Threshold	$\Pr[G_1 \le 0]$	$\Pr[G_2 \le 0]$	$\Pr[G_3 \le 0]$	$\Pr[G_4 \le 0]$	$\Pr[G_5 \le 0]$	$\Pr[G_6 \le 0]$	$\Pr[G_7 \le 0]$	$\Pr[G_8 \le 0]$	$\Pr[G_9 \le 0]$	$\Pr[G_{10} \le 0]$
FORM		0.9997	1.0000	1.0000	0.9987	0.9990	0.9989	0.9988	0.9987	0.9998	0.9987
AK-MCS	0.03	1.0000	1.0000	1.0000	0.9987	1.0000	1.0000	1.0000	0.9987	1.0000	0.9987
	0.01	1.0000	1.0000	1.0000	0.9987	1.0000	1.0000	1.0000	0.9987	1.0000	0.9987
	0.005	1.0000	1.0000	1.0000	0.9987	1.0000	1.0000	1.0000	0.9987	1.0000	0.9987

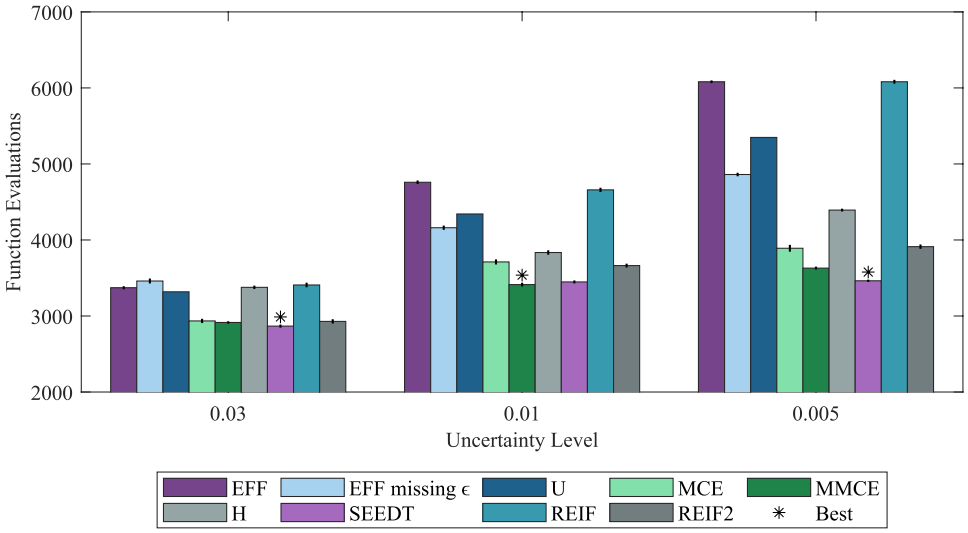


Fig. 15 Number of function evaluations for surrogate uncertainty threshold = 0.01 for Case Study 3

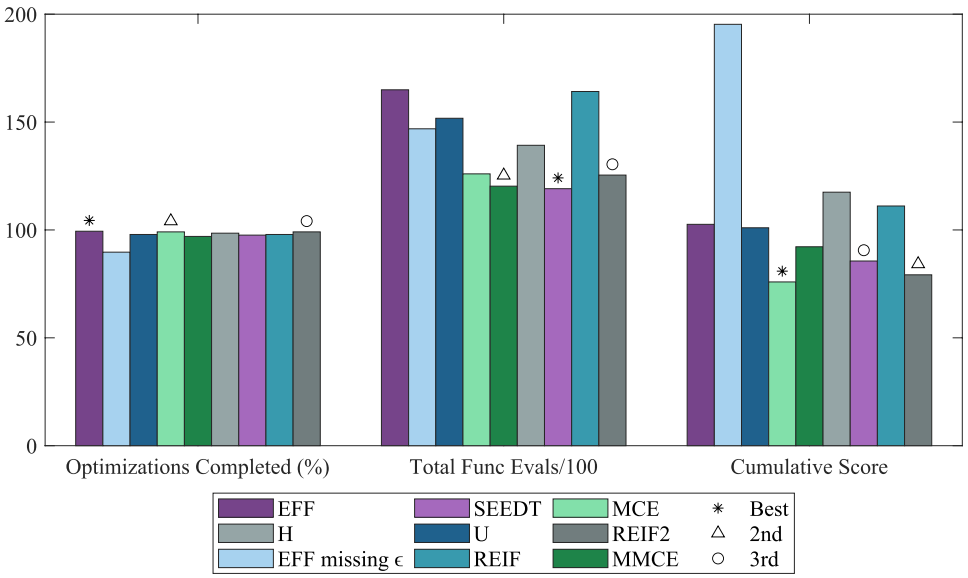


Fig. 16 Local approximation overall summary—cumulative score (lower = better)

Table 15 Cumulative score summary (lower = better)

Acq. function	Score	Global approximation	Score
MCE	75.9	MCE	20.1
REIF2	79.2	SEEDT	21.1
SEEDT	85.6	REIF2	21.2
MMCE	92.2	MMCE	36.6
AK-MCS	101.0	U	40.6
EFF	102.6	H	47.0
REIF	111.1	REIF	50.9
H	117.5	EFF	56.8
EFF miss ϵ	195.3	EFF missing ϵ	79.6

$\mathbf{d}_{\mathrm{DDO}} = (0.5\ 1.2259\ 0.5\ 1.2071\ 0.5\ 1.4947\ 0.5)$ with a minimum weight of 23.1922 kg.

$$\min(\text{weight}) = 1.98 + 4.90x_1 + 6.67x_2 + 6.98x_3 + 4.01x_4 + 1.78x_5 + 2.73x_7$$

$$\text{s.t.} \Pr\left[\text{Load}_{\text{Abdomen}} \le 1.0\ \text{kN}\right] \ge R^{\text{t}} = 0.9987$$

$$\Pr\left\{\text{Deflection}_{\text{rib}_{\text{u}},\text{rib}_{\text{m}},\text{rib}_{\text{l}}} \le 32\ \text{mm}\right\} \ge R^{\text{t}} = 0.9987$$

$$\Pr\left\{\text{VC}_{\text{upper, mid, lower}} \le 0.32\ \text{mls}\right\} \ge R^{\text{t}} = 0.9987$$

$$\Pr\left\{\text{Force}_{\text{pubic}} \le 4.0\ \text{kN}\right\} \ge R^{t} = 0.9987$$

$$\Pr\left\{\text{Velocity}_{\text{B-pillar}} \le 9.9\ \text{mm/ms}\right\} \ge R^{\text{t}} = 0.9987$$

$$\Pr\left\{\text{Velocity}_{\text{door}} \le 15.7\ \text{mm/ms}\right\} \ge R^{\text{t}} = 0.9987$$

$$\text{Load}_{\text{Abdomen}} = 1.16 - 0.3717x_2x_4 - 0.00931x_2x_{10} - 0.484x_3x_9 + 0.01343x_6x_{10}$$

$$\text{Deflection}_{\text{rib}_{\text{u}}} = 28.98 + 3.818x_3 - 4.2x_1x_2 + 0.0207x_5x_{10} + 6.63x_6x_9 - 7.7x_7x_8 + 0.32x_9x_{10}$$

$$\text{Deflection}_{\text{rib}_{\text{m}}} = 33.86 + 2.95x_3 + 0.1792x_{10} - 5.057x_1x_2 - 11.0x_2x_8 - 0.0215x_5x_{10} - 9.98x_7x_8 + 22.0x_8x_9$$

$$\text{Deflection}_{\text{rib}_{\text{l}}} = 46.36 - 9.9x_2 - 12.9x_1x_8 + 0.1107x_3x_{10}$$

$$\text{VC}_{\text{upper}} = 0.261 - 0.0159x_1x_2 - 0.188x_1x_8 - 0.019x_2x_7 + 0.0144x_3x_5 + 0.0008757x_5x_{10} + 0.08045x_6x_9 + 0.08045x_6x_9 + 0.00139x_8x_{11} + 0.00001575x_{10}x_{11}$$

$$\text{VC}_{\text{middle}} = 0.214 + 0.00817x_5 - 0.131x_1x_8 - 0.0704x_1x_9 + 0.03099x_2x_6 - 0.018x_2x_7 + 0.0208x_3x_8 + 0.121x_3x_9 - 0.00364x_5x_6 + 0.0007715x_5x_{10} - 0.0005354x_6x_{10} + 0.00121x_8x_{11}$$

$$\text{VC}_{\text{lower}} = 0.74 - 0.61x_2 - 0.163x_3x_8 + 0.001234x_3x_{10} - 0.166x_7x_9 + 0.227x_2^2$$

$$\text{Force}_{\text{pubic}} = 4.72 - 0.5x_4 - 0.19x_2x_3 - 0.0122x_4x_{10} + 0.009325x_6x_{10} + 0.000191x_{11}^2$$

$$\text{Velocity}_{\text{B-pillar}} = 10.58 - 0.674x_1x_2 - 1.95x_2x_8 + 0.02054x_3x_{10} - 0.0198x_4x_{10} + 0.028x_6x_{10}$$

$$\text{Velocity}_{\text{door}} = 16.45 - 0.489x_3x_7 - 0.843x_5x_6 + 0.0432x_9x_{10} - 0.0556x_9x_{11} - 0.000786x_{11}^2$$

Table 13 demonstrates that 100% (270 out of 270) optimizations were completed successfully for Case Study 3. FORM and active kriging also show similar results (Table 14). SEEDT and MMCE had the lowest numbers of function evaluations for the different surrogate uncertainty threshold levels. With the higher number of function evaluations, a trend appeared in Fig. 15 that the acquisition functions that included a term for the joint probability density function had a lower number of function evaluations. The four lowest function evaluations for all threshold levels were MCE, MMCE, REIF2, and SEEDT. The acquisition functions with the highest number of function evaluations, *U*, EFF, and REIF, that do not contain a term for the joint probability density function had on average, 2,100 more function evaluations.

5.4 Overall performance for the mathematical examples

The total number of optimizations completed, function evaluations, and overall performance (Eq. (29)) are plotted in Fig. 16 for the local approximation. All acquisition functions, except EFF missing ϵ, completed a similar number of optimizations. Missing ϵ on the third term of the EFF equation limited the exploration of the design space causing many points to be grouped in a smaller region, degrading optimization performance. The acquisition functions that included the term for the joint probability density function had the lowest number of function evaluations. The number of function evaluations was driven by Case Study 3, the higher-dimensional problem, which had many more function evaluations than Case Study 1 and 2. The cumulative score (Table 15) also shows that the acquisition functions that include the joint probability density function performed best. MCE and REIF2 had the lowest overall scores because they had the best combination of (1) requiring the fewest function evaluations and completing the most optimizations. MMCE and SEEDT scored slightly higher due to completing fewer optimizations.

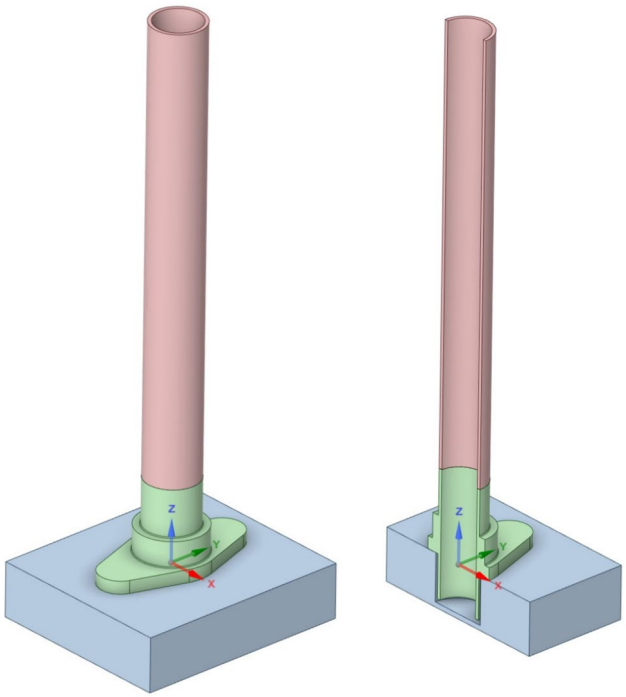


Fig. 17 Model of simplified engine oil pickup tube

MMCE completed 320, and SEEDT completed 322 out of 330 optimizations. The next group U, EFF, H, and REIF, scored higher due to more function evaluations. Finally, EFF miss ϵ scored the highest because it completed the fewest optimizations at 89.7% (296 out of 330) due to the limited design space exploration.

6 Case Study 4: design optimization of an engine oil pickup tube

The acquisition functions were tested on an FEM model of an oil pickup tube for an internal combustion engine to understand performance in an engineering application. This tube transports oil from the oil pan sump to the oil pump. The oil pickup tube is a challenging application because it is a thin-walled cantilevered tube that has considerable space constraints to avoid other engine components. The pickup tube needs to be reliable because if the pickup tube fatigues and fails to transfer the oil, the engine will not be properly lubricated.

An oil pickup tube has a complex geometry to have the inlet as close to the bottom of the engine oil pan as possible to meet off-level lubrication requirements. Since this study

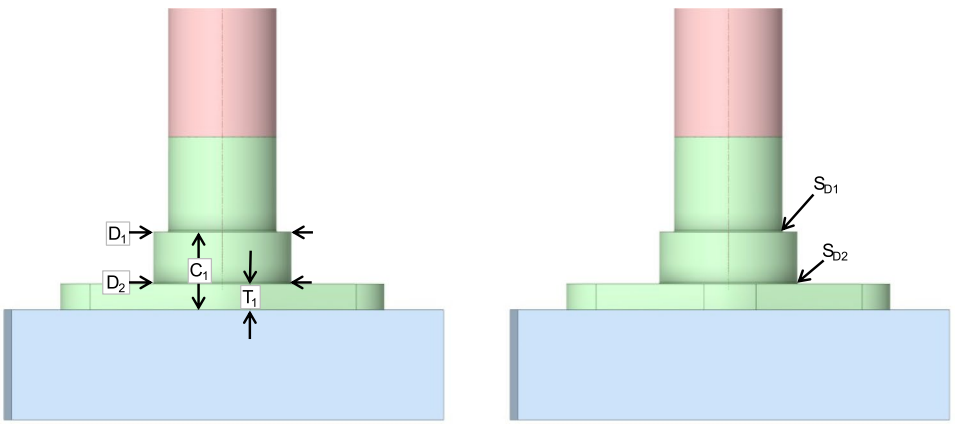


Fig. 18 Summary of design variables evaluated for FEM study

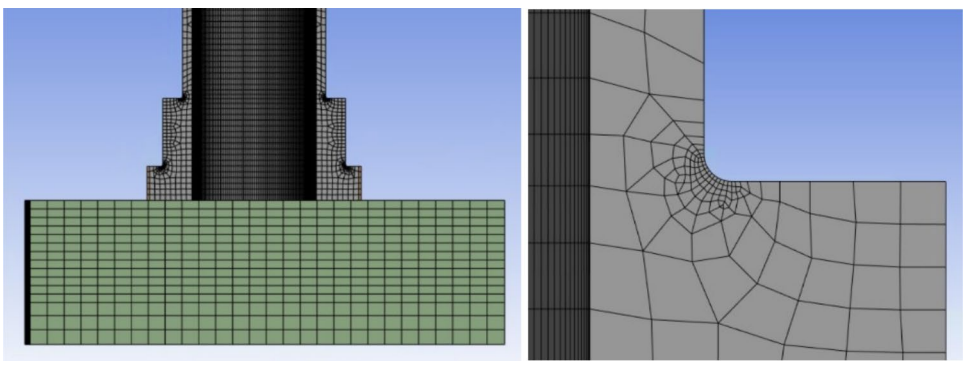


Fig. 19 FEM mesh of oil pickup tube

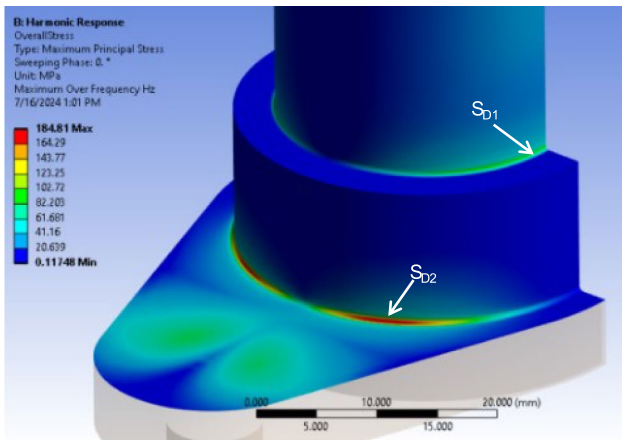


Fig. 20 Example of high stress locations on oil pickup tube

Table 16 Design Variables for Engineering Example

Variable	μ (mm)	σ (mm)	lb (mm)	ub (mm)
D_1	16.2	0.1	13.2	17
D_2	16.2	0.1	13.2	17
T_1	6	0.09	3	17
C_1	18	0.3	4	35

focuses on optimizing the flange that connects the tube to the oil pump to reduce the chance of a fatigue crack, the oil pickup tube was simplified to be a straight, thin-walled cantilevered tube. This allowed the FEM to take advantage of symmetry in the *XZ* plane to decrease solving time. The bolts were also removed and replaced with a cylinder to bond to the simulated oil pump base to eliminate the solver time required to add bolt pretension. The simplified oil pickup tube is represented in Fig. 17.

The original design was placed on an electrodynamic shaker to determine the natural frequency and acceleration needed to match the displacements experienced during engine operation. The natural frequency of the pickup tube on the shaker was found to be 200 Hz with a displacement of 0.9 mm pk–pk. The straight tube length in the model was modified to 250 mm to match the 200 Hz natural frequency. The base acceleration was set to 40,593 mm/s^2 with a 1% damping ratio to match the measured displacements on the shaker of 0.9 mm pk–pk. Base acceleration was provided along the *X*-axis.

Due to constraints of other engine components, the only part of the pickup tube that could be optimized was the connection flange. The design parameters are the attachment flange thickness (T_1), the height of the collar to braze the tube to the base (C_1), the collar top diameter (D_1), and the bottom collar diameter (D_2). The design goals were to lower the stress in the top diameter (S_{D1}) and bottom diameter (S_{D2}) to be lower than 85 MPa, which is the fatigue endurance limit for 1010 steel. The parameters are depicted in Fig. 18. The natural frequency of the tube is required to be higher than 150 Hz. For a 6-cylinder inline engine with a maximum engine speed of 2600 RPM, the maximum combustion firing frequency for 3rd order is (2600 RPM/60) × 3 = 130 Hz. Setting the minimum natural frequency to 150 Hz provides a factor of safety so that slight changes during the manufacturing process do not cause the natural frequency of the tube to be lower than 130 Hz and be excited by the 3rd engine order. All dimensions are in mm.

Table 17 Optimization summary for oil pickup tube example

Quantity of Interest	Initial Design	DDO	RBDO with Acquisition Function								
			EFF	EFF miss ϵ*	U	MCE*	MMCE	H	SEEDT	REIF	REIF2
D_1 (mm)	16.2	13.20	13.50	13.75	13.45	13.40	13.47	13.59	13.51	13.57	13.45
D_2 (mm)	16.2	14.93	15.52	15.52	15.80	15.33	15.58	15.75	15.65	15.55	15.43
T (mm)	6	9.40	9.61	9.79	9.38	9.58	9.58	9.42	9.51	9.53	9.64
C (mm)	18	20.68	20.24	20.14	20.20	20.62	20.29	20.28	20.22	20.32	20.49
Mass (kg)	1.0264	1.0368	1.0394	1.0408	1.0387	1.0389	1.0394	1.0391	1.0392	1.0392	1.0395
S_{D1} (MPa)	93.6	69.90	79.35	82.23	80.17	75.65	79.17	81.11	80.18	79.14	77.90
S_{D2} (MPa)	184.8	85.08	78.53	77.24	79.53	80.50	78.48	80.07	78.96	79.93	78.91
*Mode*1 (Hz)	200.7	229.48	231.67	232.67	231.07	231.25	231.62	231.29	231.40	231.35	231.73
S_{D1} Rel		0.9997	1.0000	0.9982	0.9998	1.0000	1.0000	1.0000	1.0000	1.0000	0.9997
S_{D2} Rel		0.2990	0.9999	1.0000	0.9992	0.9982	1.0000	1.0000	0.9998	0.9993	0.9998
*Mode*1 Rel		1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Func Evals			191	116	146	128	181	161	148	112	128

*Optimization converged to an infeasible point

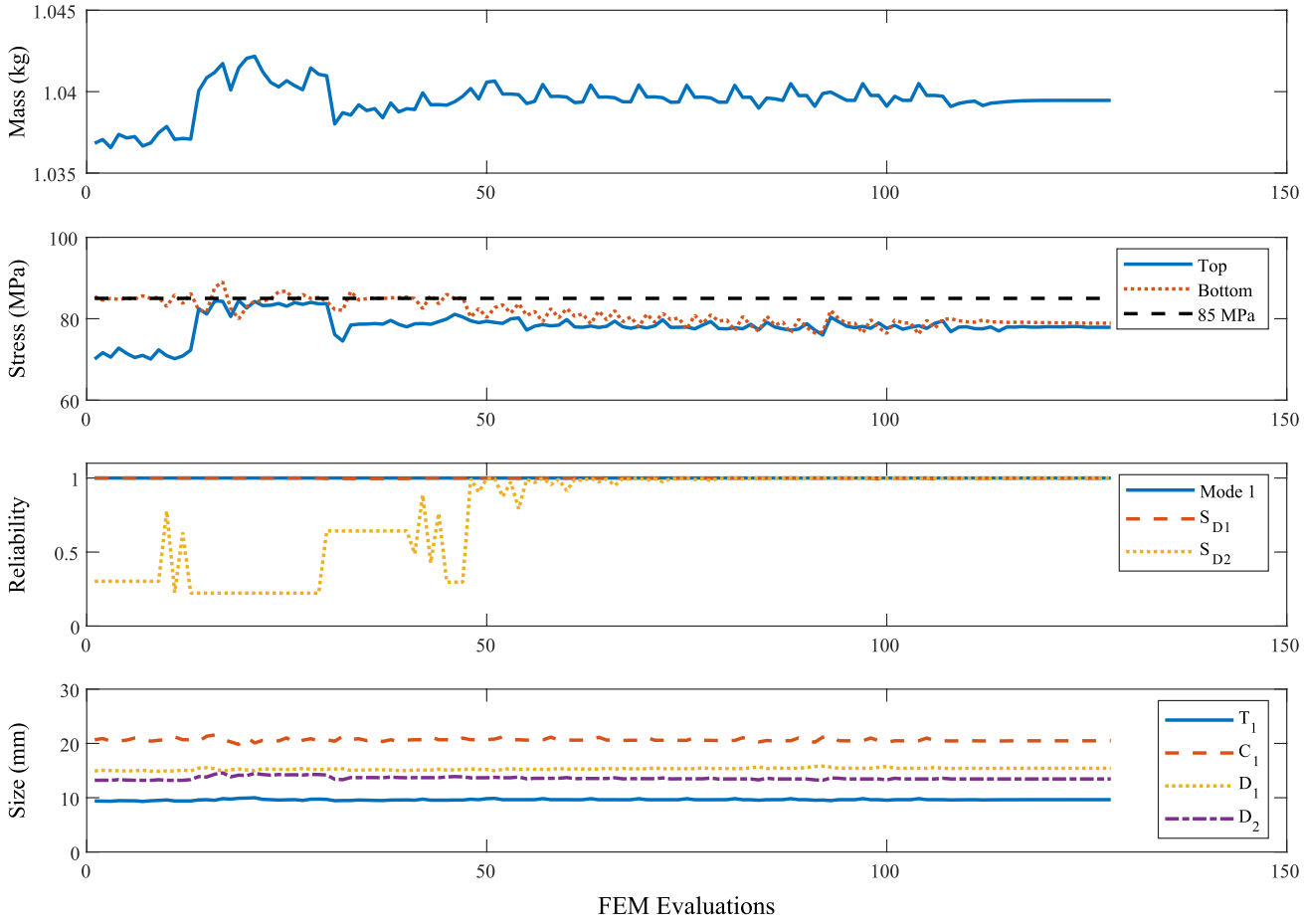


Fig. 21 Optimization summary for REIF2

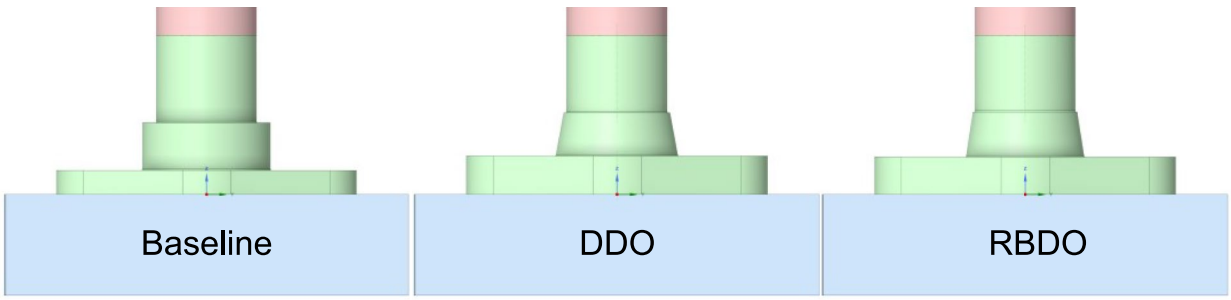


Fig. 22 Optimization results for REIF2

Engineering Example RBDO Problem Formulation

$$\text{minimize} f(\mathbf{d}) = \text{Weight}_{\text{tube}}$$

$$\text{subject to } R = \Pr\left[G_i(\mathbf{x};\mathbf{d}) \le 0\right] \ge R^{\text{t}} = 0.9987, i = 1-3$$

$$G_1(\mathbf{x}) = S_{\text{D1}}(\text{MPa}) - 85$$

$$G_2(\mathbf{x}) = S_{\text{D2}}(\text{MPa}) - 85$$

$$G_3(\mathbf{x}) = 150 - \text{Mode 1(Hz)}$$

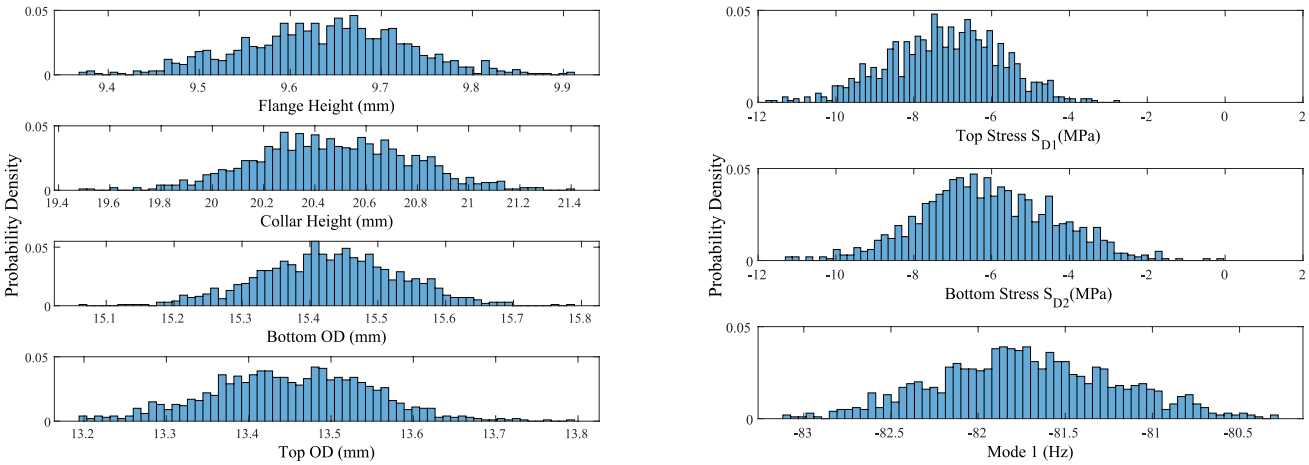


Fig. 23 Empirical probability density functions of the design variables (left column) and the two stresses and natural frequency (right column) from the 1000-point direct MCS on the RBDO optimum for the REIF2 acquisition function

$$3 \le C_1 \le 17, 4 \le T_1 \le 35, 13.2 \le D_1 \le 17, 13.2 \le D_2 \le 17$$

The Harmonic Response module in ANSYS Mechanical 2023 was used as the finite element solver. The pickup tube was meshed with 84,176 Hex20 and 720 Wedge15 elements (Fig. 19) and is made of 1010 steel with Young's Modulus $E = 200{,}000$ MPa and Poisson's ratio $\upsilon = 0.3$. A small radius (0.4 mm) was added to the top (SD1) and bottom (SD2) diameter transitions to avoid the stress singularity from a reentrant corner in the FEM. A finer mesh was added to the transitions to guarantee the stress values were based on the part geometry and not the mesh (Fig. 20). Depending on the design, the maximum stress in the tube would occur in either the top or bottom diameter. When interfacing fmincon from MATLAB to ANSYS, the fmincon option "FinDiffRelStep" was set to [0.02, 0.025, 0.01, 0.01] for [T_1, C_1, D_1, D_2]. This ensured fmincon took a large enough step to allow the mesh to capture the changes in geometry.

The design variables were modeled as truncated normal distributions. The lb and ub values (Table 16) used as the lower and upper bounds in the RBDO formulation were used as the limits for the truncated normal distributions, so that the FEM could be correctly meshed. A DDO was conducted to find the optimum design point to start the RBDO. Two of the acquisition functions (MCE and EFF missing ϵ) converged to an infeasible design point because the reliability for the stress constraints was less than 0.9987. However, the stresses were still much improved compared to the baseline design. Table 17 summarizes the optimization results of DDO and kriging-based RBDO with the nine acquisition functions for this engineering example. As shown in this table, the U function found the design with the lowest mass, REIF had the lowest number of FEM evaluations, and REIF2 had the lowest stress and highest natural frequency. EFF had the highest number of function evaluations at 191. Figure 21 shows the optimization summary for REIF2, and Fig. 22 depicts how the design changed from the Baseline to DDO to RBDO.

In addition, the results in Table 17 show that REIF stops with the lowest number of evaluations and REIF2 has the most conservative solution. Unfortunately, offering a theoretical explanation for these observations is challenging due to the complex numerical procedures involved. This is also why this paper conducted this empirical comparative study to compare various acquisition functions, providing "numerical experiment results" that might shed light on their performance. We speculate that these phenomena are caused by the randomness inherent in the numerical implementation. Despite starting with the same initial design, variations in gradients during design optimization can occur due to randomness in surrogate modeling and uncertainty in sampling-based reliability analysis. This uncertainty is why we conducted 10 repetitions for the mathematical examples to account for uncertainty in method performance evaluation. Unfortunately, repetitions were not feasible for this real-world application due to the high computational costs required. As indicated in Table 17, the results for REIF and REIF2 are very close to each other. It is quite possible that

the fewer evaluations required by REIF are due to numerical uncertainty.

Finally, a 1000-point MCS was conducted around the REIF2 RBDO optimum to verify the surrogate model accurately represented the expected reliability. All combinations had stresses below 85 MPa and natural frequencies greater than 150 Hz. Figure 23 shows the empirical probability density functions of the four design variables and the two stresses and natural frequency in the limit-state functions.

7 Conclusion

This paper evaluated nine different acquisition functions for active learning kriging in RBDO. Three mathematical examples were used to evaluate the performance for different uncertainty and standard deviation levels for global and local approximations. Finally, an engineering example was evaluated to provide insights for practitioners. Based on this comparative study, we have the following major findings:

- The family of acquisition functions (MCE, REIF2, SEEDT, and MMCE), which includes a term for the joint probability density function, performed the best in our comparison. This is due to the lower number of overall function evaluations needed to find the reliability-based design optimum. MCE and REIF2 performed the best in this group because of the higher number of optimization runs they completed.
- The next family of functions (U, EFF, REIF, and H) had similar performance.
- The acquisition function that had the lowest performance is EFF missing ϵ. This function, which has been found in several research papers, has the ϵ variable missing on the third term in the equation. This may be due to the Echard et al. (2011) for U being the most referenced active kriging paper we found. Missing the ϵ term limits the ability of this function to explore the design space, leading to fewer successful optimizations to be completed.

It is therefore recommended that practitioners using active learning kriging focus on using REIF2, MCE, SEEDT, or MMCE to limit the total number of function evaluations and save processing time. Although MCE did not converge for the engineering example, its reliability of 0.9982 is close enough to the target of 0.9987 for practical purposes, given the underlying variability in finite element modeling. REIF2, MCE, SEEDT, and MMCE are recommended based on the robustness observed from the mathematical examples. It is difficult to say any one of these four acquisition functions consistently and significantly outperforms the others. EFF missing ϵ should be avoided in cases with higher uncertainty requiring extensive design space exploration.

Appendix 1: Results of kriging regression models on 1D example ($y = \sin(0.9x)$) built by MATLAB fitrgp and Python "scikit-learn. GaussianProcessRegressor"

See Figs. 24 and 25.

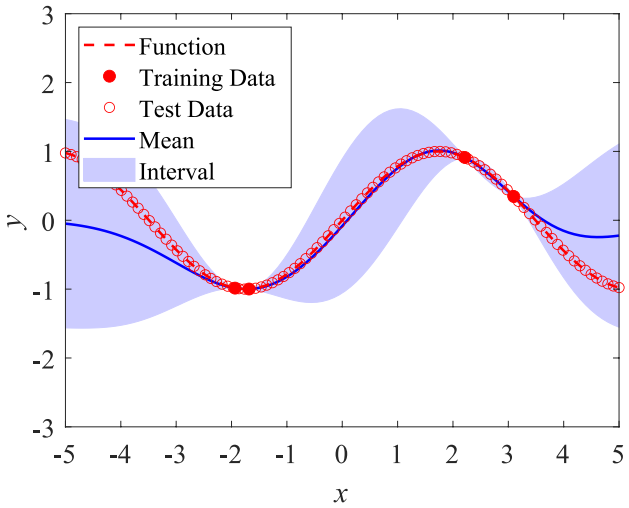


Fig. 24 MATLAB fitrgp results for $\sin(0.9x)$

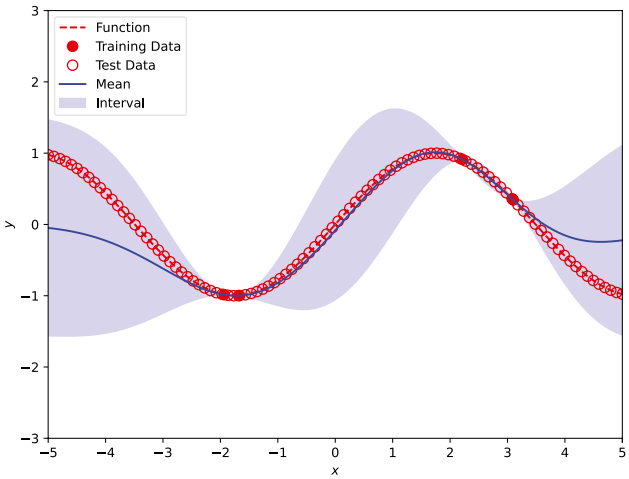


Fig. 25 Python "scikit-learn.GaussianProcessRegressor" for $\sin(0.9x)$

Appendix 2: Comparison of convergence performance of two types of kernels for kriging: "squaredexponential" and "ardsquaredexponential"

See Figs. 26, 27, 28, and 29.

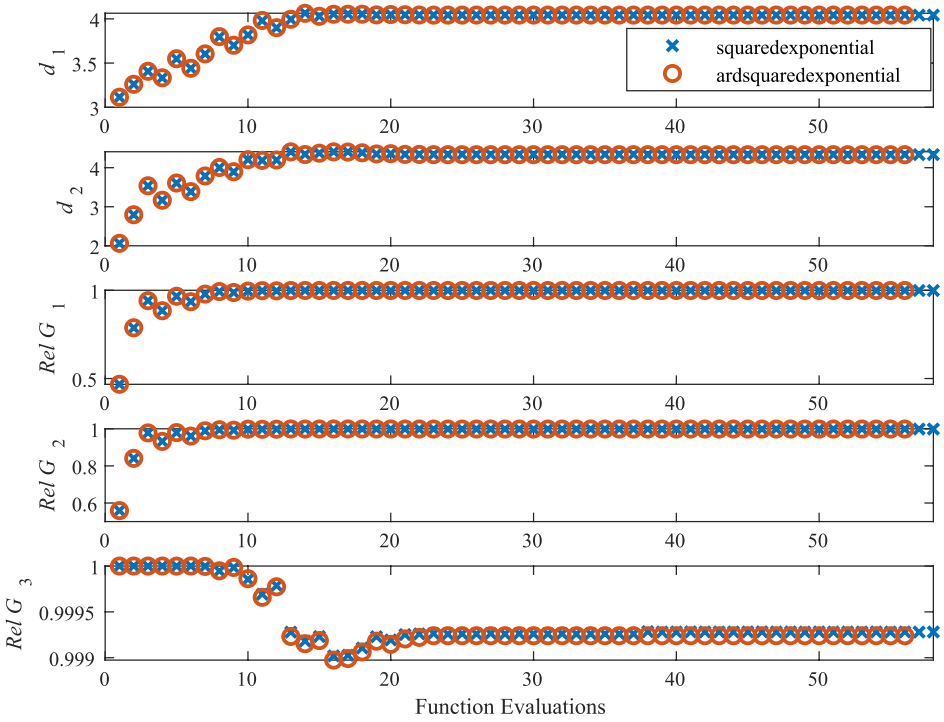
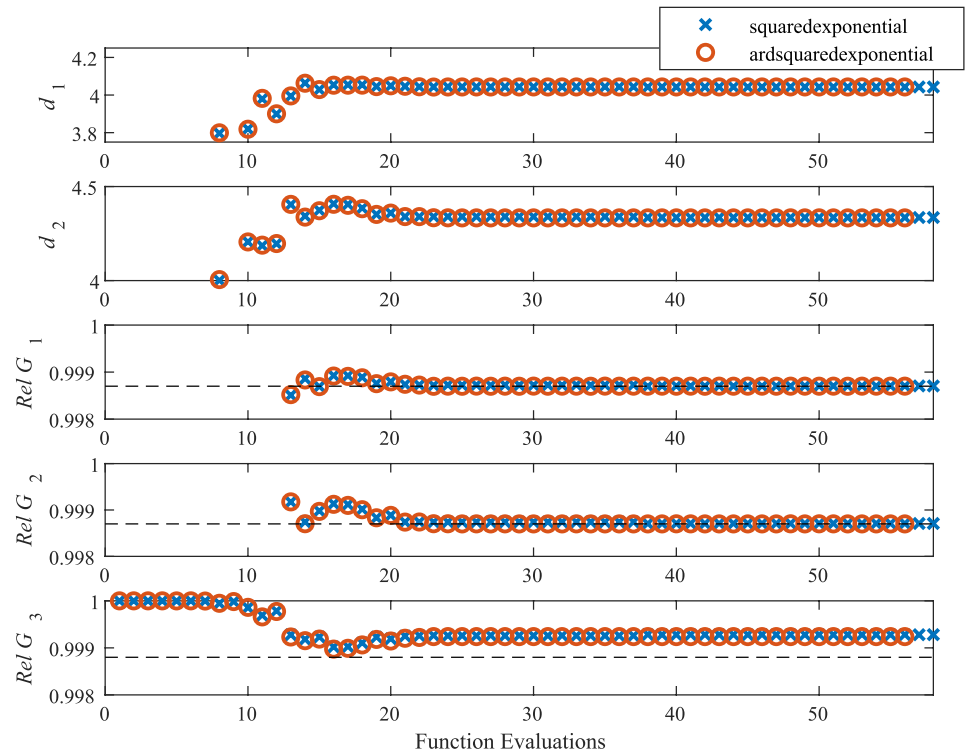


Fig. 26 Comparison of "squaredexponential" and "ardsquaredexponential" for local approximation using U function



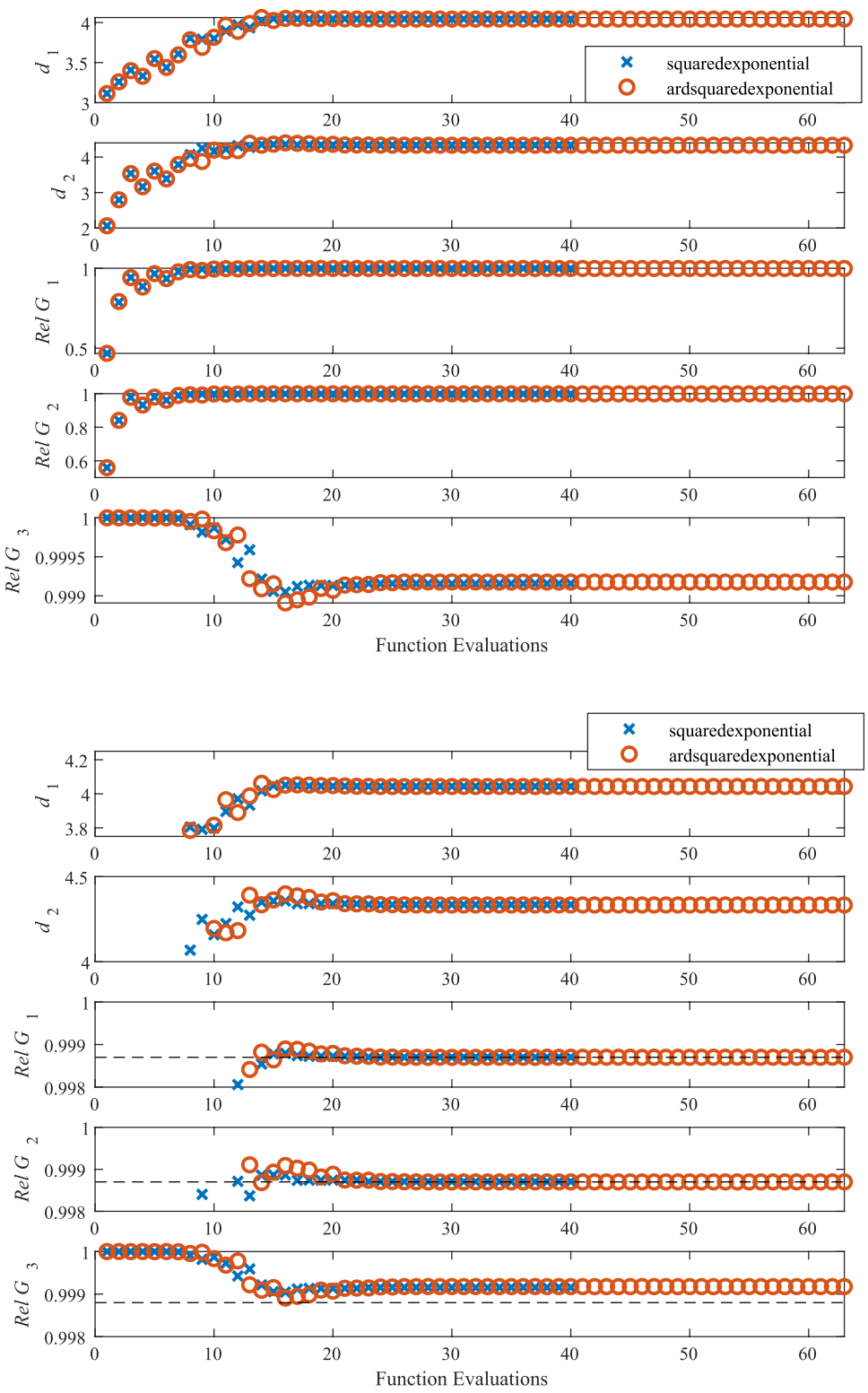
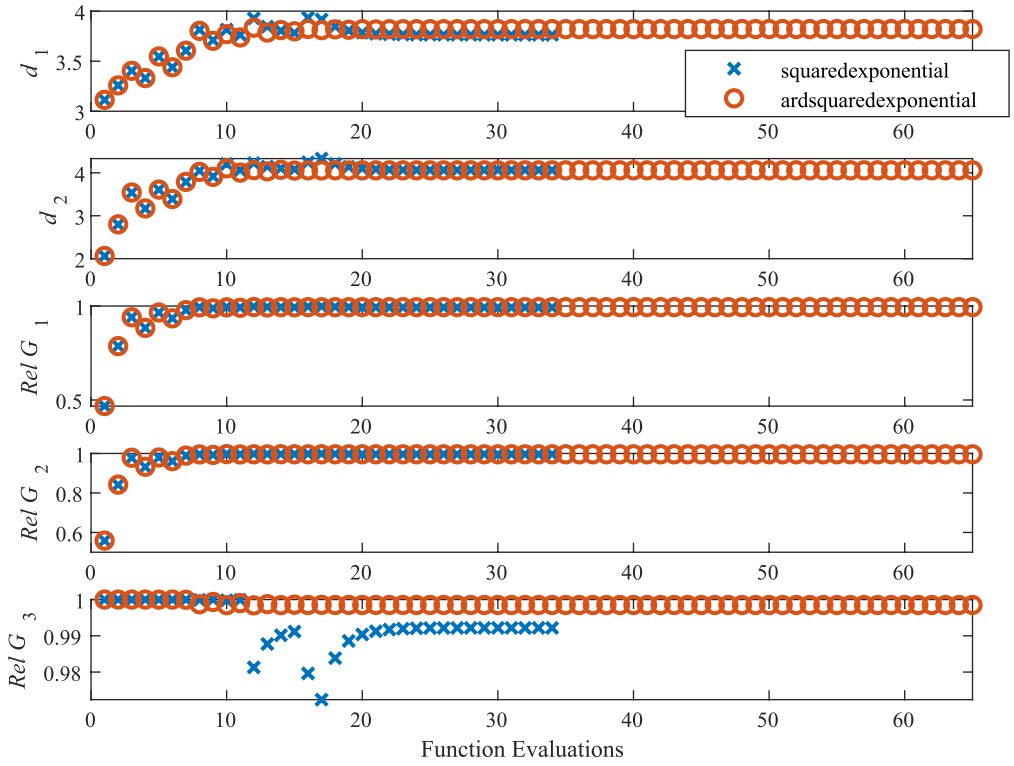


Fig. 27 Comparison of "squaredexponential" and "ardsquaredexponential" for local approximation using EFF



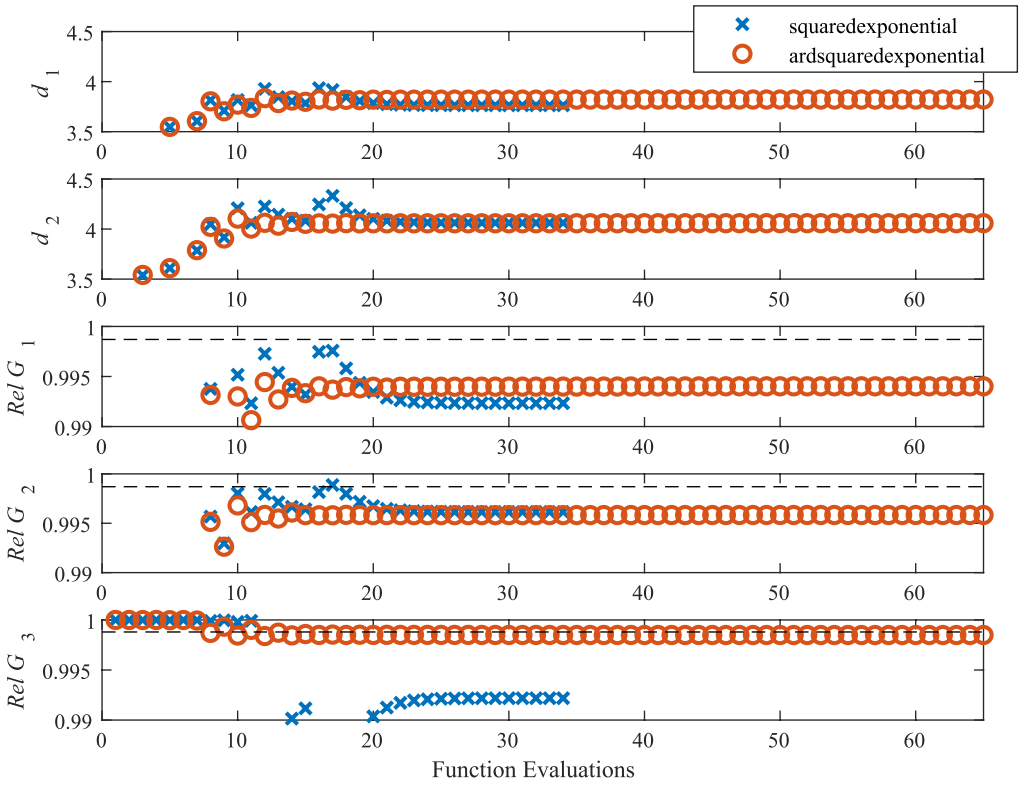


Fig. 28 Comparison of "squaredexponential" and "ardsquaredexponential" for global approximation using *U* function

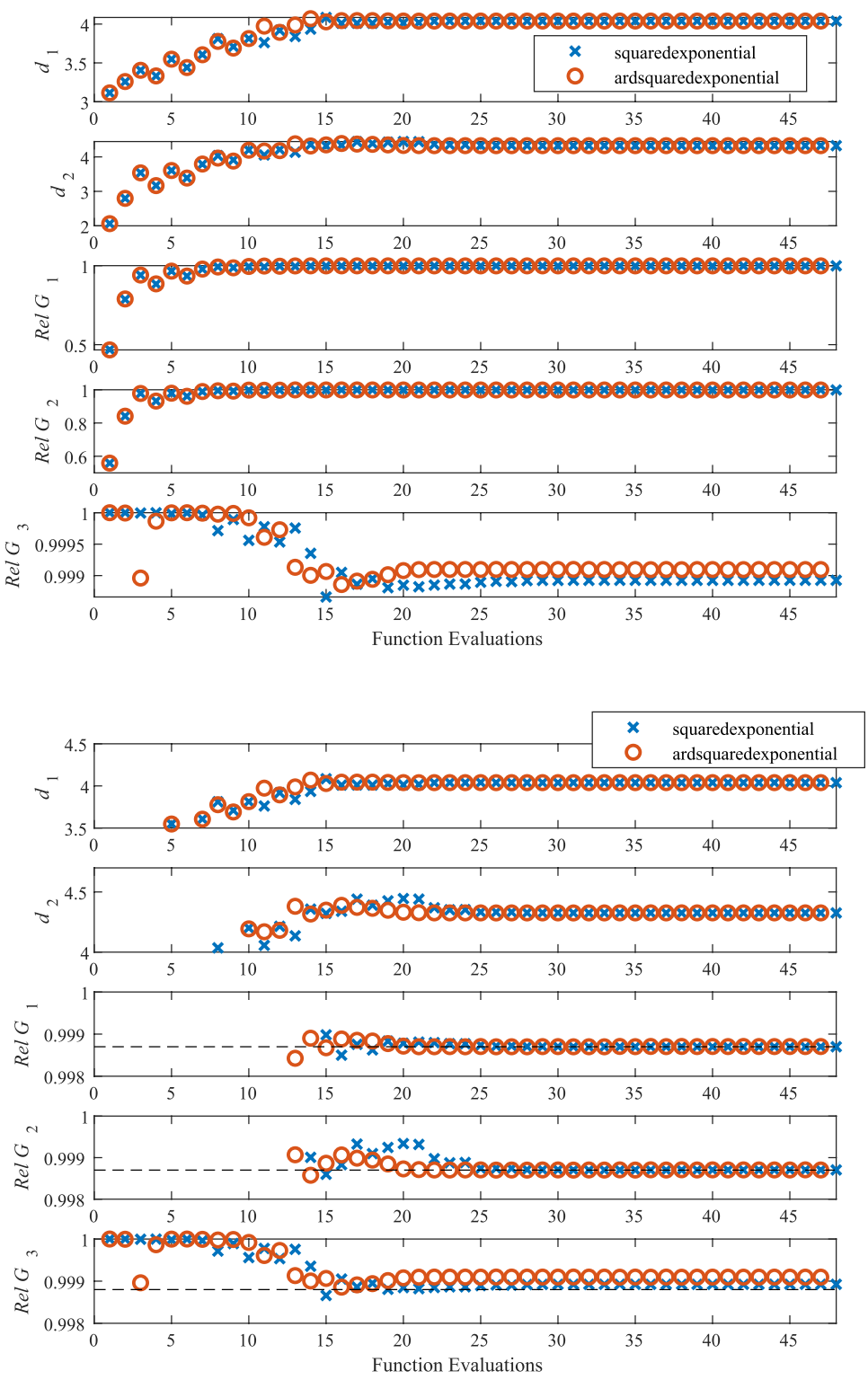


Fig. 29 Comparison of "squaredexponential" and "ardsquaredexponential" for global approximation using EFF

Appendix 3: FORM results and active kriging results for $\sigma = 0.6$

See Tables 18 and 19.

Table 18 FORM results for Case Study #1

$\sigma = 0.6$	Function	Obj	d_1 Opt	d_2 Opt	Rel 1	Rel 2	Rel 3
FORM	AMV	8.3787	4.0101	4.3687	0.9987	0.9987	0.9988
	HL_RF	8.3787	4.0101	4.3687	0.9987	0.9987	0.9993
0.03	U	8.3728	4.0469	4.3259	0.9987	0.9987	0.9993
	EFF	8.3772	4.0484	4.3288	0.9987	0.9987	0.9992
	MCE	8.3730	4.0469	4.3261	0.9987	0.9987	0.9993
	SEEDT	8.3723	4.0465	4.3259	0.9987	0.9987	0.9993
	EFF missing ϵ	8.3763	4.0474	4.3289	0.9987	0.9987	0.9993
	MMCE	8.3746	4.0472	4.3274	0.9987	0.9987	0.9993
	H	8.3731	4.0470	4.3261	0.9987	0.9987	0.9993
	REIF	8.3776	4.0461	4.3315	0.9987	0.9987	0.9992
	REIF2	8.3729	4.0470	4.3259	0.9987	0.9987	0.9993
0.01	U	8.3728	4.0469	4.3259	0.9987	0.9987	0.9993
	EFF	8.3737	4.0476	4.3261	0.9987	0.9987	0.9993
	MCE	8.3744	4.0471	4.3273	0.9987	0.9987	0.9993
	SEEDT	8.3724	4.0474	4.3249	0.9987	0.9987	0.9993
	EFF missing ϵ	8.3750	4.0469	4.3281	0.9987	0.9987	0.9993
	MMCE	8.3737	4.0471	4.3266	0.9987	0.9987	0.9993
	H	8.3745	4.0472	4.3272	0.9987	0.9987	0.9993
	REIF	8.3763	4.0477	4.3286	0.9987	0.9987	0.9993
	REIF2	8.3745	4.0477	4.3268	0.9987	0.9987	0.9993
0.005	U	8.3728	4.0470	4.3258	0.9987	0.9987	0.9993
	EFF	8.3765	4.0478	4.3287	0.9987	0.9987	0.9993
	MCE	8.3762	4.0479	4.3283	0.9987	0.9987	0.9993
	SEEDT	8.3766	4.0480	4.3286	0.9987	0.9987	0.9993
	EFF missing ϵ	8.3740	4.0468	4.3271	0.9987	0.9987	0.9993
	MMCE	8.3735	4.0477	4.3258	0.9987	0.9987	0.9993
	H	8.3729	4.0470	4.3259	0.9987	0.9987	0.9993
	REIF	8.3731	4.0470	4.3261	0.9987	0.9987	0.9993
	REIF2	8.3757	4.0477	4.3280	0.9987	0.9987	0.9993

Table 19 FORM results for Case Study 2

$\sigma=0.6$	Function	Obj	d_1 Opt	d_2 Opt	Rel 1	Rel 2	Rel 3
FORM	AMV	− 1.6732	4.972	2.084	0.9773	0.9726	0.9886
	HL_RF	− 1.5111	4.378	2.474	0.9527	0.9773	1.0000
0.03	U	− 1.5594	4.645	2.353	0.9782	0.9789	1.0000
	EFF	− 1.5699	4.656	2.327	0.9773	0.9773	1.0000
	MCE	− 1.5726	4.666	2.321	0.9773	0.9773	1.0000
	SEEDT	− 1.5719	4.663	2.323	0.9773	0.9773	1.0000
	EFF missing ϵ	− 1.5329	4.587	2.418	0.9782	0.9814	0.9991
	MMCE	− 1.5766	4.681	2.311	0.9773	0.9773	1.0000
	H	− 1.5669	4.665	2.335	0.9781	0.9775	1.0000
	REIF	− 1.5707	4.659	2.325	0.9773	0.9773	1.0000
	REIF2	− 1.5710	4.661	2.325	0.9773	0.9773	1.0000
0.01	U	− 1.5733	4.668	2.319	0.9773	0.9773	1.0000
	EFF	− 1.5717	4.663	2.323	0.9773	0.9773	1.0000
	MCE	− 1.5719	4.663	2.323	0.9773	0.9773	1.0000
	SEEDT	− 1.5718	4.663	2.323	0.9773	0.9773	1.0000
	EFF missing ϵ	− 1.5712	4.665	2.324	0.9774	0.9774	1.0000
	MMCE	− 1.5714	4.673	2.324	0.9777	0.9776	1.0000
	H	− 1.5723	4.665	2.322	0.9773	0.9773	1.0000
	REIF	− 1.5716	4.662	2.323	0.9773	0.9773	1.0000
	REIF2	− 1.5719	4.664	2.323	0.9773	0.9773	1.0000
0.005	U	− 1.5719	4.663	2.323	0.9773	0.9773	1.0000
	EFF	− 1.5721	4.664	2.322	0.9773	0.9773	1.0000
	MCE	− 1.5719	4.663	2.323	0.9773	0.9773	1.0000
	SEEDT	− 1.5718	4.663	2.323	0.9773	0.9773	1.0000
	EFF missing ϵ	− 1.5694	4.660	2.329	0.9775	0.9774	1.0000
	MMCE	− 1.5735	4.673	2.319	0.9774	0.9775	1.0000
	H	− 1.5727	4.666	2.321	0.9773	0.9773	1.0000
	REIF	− 1.5719	4.663	2.323	0.9773	0.9773	1.0000
	REIF2	− 1.5719	4.663	2.323	0.9773	0.9773	1.0000

Appendix 4: Results of parametric studies on the effects of c_{local}

See Tables 20 and 21.

Table 20 Reliability analysis results at the RBDO design optimum ($d_1 = 4.043$, $d_2 = 4.335$) under six values of c_{local} for the mathematical constraints in Case Study #1

Method	c_{local}	Function evaluations per constraint	Metric	G_1	G_2	G_3
MCS	N.A	1E+06	Reliability	0.99885	0.99879	0.99930
Kriging	0.5	8	Reliability	0.99881	0.99891	0.99955
			Absolute error	0.00006	0.00012	0.00026
	0.7	8	Reliability	0.99880	0.99887	0.99952
			Absolute error	0.00005	0.00008	0.00022
	1.0	13	Reliability	0.99870	0.99871	0.99927
			Absolute error	0.00005	0.00008	0.00002
	1.3	14	Reliability	0.99870	0.99871	0.99928
			Absolute error	0.00005	0.00008	0.00002
	1.6	15	Reliability	0.99871	0.99871	0.99931
			Absolute error	0.00005	0.00008	0.00001
	1.8	16	Reliability	0.99871	0.99871	0.99932
			Absolute error	0.00005	0.00008	0.00003

The acquisition function used to obtain the design optimum and conduct this parametric study was AK-MCS

Table 21 Reliability analysis results at the RBDO design optimum ($d_1 = 4.351$, $d_2 = 3.598$) under six values of c_{local} for the mathematical constraints in Case Study #2

Method	c_{local}	Function evaluations per constraint	Metric	G_1	G_2	G_3
MCS	N.A	1E+06	Reliability	0.99870	0.998711	0.99973
Kriging	0.5	8	Reliability	0.99984	0.997037	0.99997
			Absolute error	0.00114	0.00167	0.00024
	0.7	11	Reliability	0.99856	0.999638	0.99972
			Absolute error	0.00015	0.00093	0.00002
	1.0	17	Reliability	0.99873	0.999552	0.99975
			Absolute error	0.00003	0.00084	0.00001
	1.3	22	Reliability	0.99873	0.998767	0.99974
			Absolute error	0.00003	0.00006	0.00001
	1.5	26	Reliability	0.99873	0.998685	0.99974
			Absolute error	0.00003	0.00003	0.00001
	1.8	27	Reliability	0.99874	0.998977	0.99975
			Absolute error	0.00003	0.00027	0.00001

The acquisition function used to obtain the design optimum and conduct this parametric study was AK-MCS

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00158-025-03978-0.

Author contributions Todd Thompson: Conceptualization, methodology, software, validation, and writing—original draft. Robert McMullen: methodology, software, and writing—review and editing. Venkat Nemani: methodology and writing—review and editing. Zhen Hu: Methodology, software, and writing—review and editing. Chao Hu: Conceptualization, methodology, software, writing—review and editing, and supervision.

Funding John Deere Power Systems provided all funding for this work.

Data availability Data for the initial sample points and the sequential samples selected by all acquisition functions in Case Studies 1–3 will be made available upon reasonable request. Data for the engine pickup tube design (Case Study 4) cannot be shared due to confidentiality restrictions.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Replication of results The code for replicating the results in Case Study 1 is publicly available on GitHub: https://github.com/mcmullrj/RBDO.

References

Bichon BJ, Eldred MS, Swiler LP, Mahadevan S, McFarland JM (2008) Efficient global reliability analysis for nonlinear implicit performance functions. AIAA J 46(10):2459–2468

Chaudhuri A, Kramer B, Willcox KE (2020) Information reuse for importance sampling in reliability-based design optimization. Reliab Eng Syst Saf 201:106853

Chen Z, Qiu H, Gao L, Li X, Li P (2014) A local adaptive sampling method for reliability-based design optimization using Kriging model. Struct Multidisc Optim 49:401–416

Chu L, Shi J, de Cursi ES (2019) Kriging surrogate model for resonance frequency analysis of dental implants by a Latin hypercube-based finite element method. Appl Bionics Biomech. https://doi.org/10.1155/2019/3768695

Dilip DM, Sivakumar Babu GL (2021) Reliability-based design optimization of flexible pavements using kriging models. J Transp Eng Part B 147(3):04021046–47

Dong Y, Teixeira AP, Guedes Soares C (2020) Application of adaptive surrogate models in time-variant fatigue reliability assessment of welded joints with surface cracks. Reliab Eng Syst Saf 195:106730–31

Dubourg V, Sudret B, Bourinet JM (2011) Reliability-based design optimization using kriging surrogates and subset simulation. Struct Multidisc Optim 44:673–690. https://doi.org/10.1007/s00158-011-0653-8

Echard B, Gayton N, Lemaire M (2011) AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. Struct Saf 33(2):145–154

Fang J, Gao Y, Sun G, Chengmin Xu, Li Q (2015) Multiobjective robust design optimization of fatigue life for a truck cab. Reliab Eng Syst Saf 135:1–8

Garrido-Merchán EC, Hernández-Lobato D (2020) Dealing with categorical and integer-valued variables in Bayesian optimization with Gaussian processes. Neurocomputing 380:20–35

Ghosh S, Roy A, Chakraborty S (2021) Kriging metamodeling-based monte carlo simulation for improved seismic fragility analysis of structures. J Earthq Eng 25(7):1316–1336

Gu L, Yang RJ, Tho C-H, Makowskit M, Faruquet O, Li Y (2001) Optimisation and robustness for crashworthiness of side impact. Int J Veh Des 26(4):348–360

Hu Z, Mahadevan S (2016a) A single-loop kriging surrogate modeling for time-dependent reliability analysis. J Mech Des 138(6):061406

Hu Z, Mahadevan S (2016b) Global sensitivity analysis-enhanced surrogate (GSAS) modeling for reliability analysis. Struct Multidisc Optim 53(3):501–521

Hu W, Choi KK, Cho H (2016) Reliability-based design optimization of wind turbine blades for fatigue life under dynamic wind load

uncertainty. Struct Multidisc Optim 54:953–970. https://doi.org/10.1007/s00158-016-1462-x
Hu C, Youn BD, Wang P (2019) Engineering design under uncertainty and health prognostics. Springer, Cham
Jin R, Chen W, Simpson TW (2001) Comparative studies of metamodelling techniques under multiple modelling criteria. Struct Multidisc Optim 23(1):1–13
Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. J Glob Optim 13(4):455–492
Kim J, Song J (2021) Quantile surrogates and sensitivity by adaptive gaussian process for efficient reliability-based design optimization. Mech Syst Signal Process 161:107962
Kroetz HM, Moustapha M, Beck AT, Sudret B (2020) A two-level kriging-based approach with active learning for solving time-variant risk optimization problems. Reliab Eng Syst Saf 203:107033
Kusano I, Baldomir A, Jurado JÁ, Hernández S (2014) Reliability based design optimization of long-span bridges considering flutter. J Wind Eng Ind Aerodyn 135:149–162
Le Pavic J, Stamoulis G, Bonnemains T, Da Silva D, Thévenet D (2019) Fast failure prediction of adhesively bonded structures using a coupled stress-energetic failure criterion. Fatigue Fract Eng Mater Struct 42(3):627–639
Lee SH, Chen W (2009) A comparative study of uncertainty propagation methods for black-box-type problems. Struct Multidisc Optim 37(3):239–253
Lee I, Choi KK, Zhao L (2011a) Sampling-based RBDO using the stochastic sensitivity analysis and Dynamic Kriging method. Struct Multidisc Optim 44:299–317. https://doi.org/10.1007/s00158-011-0659-2
Lee I, Choi KK, Zhao L (2011b) Sampling-based RBDO using the stochastic sensitivity analysis and dynamic Kriging method. Struct Multidisc Optim 44:299–317
Li M, Shen S, Barzegar V, Mohammadkazem S, Hu C, Laflamme S (2021) Kriging-based reliability analysis considering predictive uncertainty reduction. Struct Multidisc Optim 63:2721–2737. https://doi.org/10.1007/s00158-020-02831-w
Lv Z, Zhenzhou Lu, Wang P (2015) A new learning function for Kriging and its applications to solve reliability problems in engineering. Comput Math Appl 70:1182–1197
MathWorks (2022a) Gaussian process regression models. MathWorks. https://www.mathworks.com/help/stats/gaussian-process-regression-models.html. Accessed 5 June 2022
MathWorks (2022b) Kernel (covariance) function options. MathWorks. https://www.mathworks.com/help/stats/kernel-covariance-function-options.html. Accessed 5 June 2022
MathWorks (2022c) fmincon. MathWorks. https://www.mathworks.com/help/optim/ug/fmincon.html. Accessed 18 June 2022
MathWorks (2022d) fitrgp. MathWorks. https://www.mathworks.com/help/stats/fitrgp.html. Accessed 18 June 2022
Meng Z, Zhang Z, Zhang D, Yang D (2019) An active learning method combining Kriging and accelerated chaotic single loop approach (AK-ACSLA) for reliability-based design optimization. Comput Methods Appl Mech Eng 357:112570
Mi C, Li W, Xiao X, Jian H, Zhengqi Gu, Berto F (2019) Lifetime assessment and optimization of a welded A-type frame in a mining truck considering uncertainties of material properties and structural geometry and load. Appl Sci 9(5):918
Moustapha M, Sudret B (2019) Surrogate-assisted reliability-based design optimization: a survey and a unified modular framework. Struct Multidisc Optim 60:2157–2176. https://doi.org/10.1007/s00158-019-02290-y
Moustapha M, Sudret B, Bourinet JM, Guillaume B (2016) Quantile-based optimization under uncertainties using adaptive Kriging surrogate models. Struct Multidisc Optim 54:1403–1421. https://doi.org/10.1007/s00158-016-1504-4
Moustapha M, Galimshina A, Habert G, Sudret B (2022) Multi-objective robust optimization using adaptive surrogate models for problems with mixed continuous-categorical parameters. Struct Multidisc Optim 65(12):357
Neal RM (2012) Bayesian learning for neural networks, vol 118. Springer, New York
Pearce R, Ireland P, Romero E (2020) Thermal matching using Gaussian process regression. Proc Inst Mech Eng Part G J Aerosp Eng 234(6):1172–1180
Rasmussen CE, Williams CKI (2006) Gaussian processes for machine learning. In: Rasmussen CE, Williams CKI (eds) Adaptive computation and machine learning. MIT, Cambridge
Rosenblatt M (1952) Remarks on a multivariate transformation. Ann Math Stat 23(3):470–472
Sadoughi MK, Hu C, MacKenzie CA, Eshghi AT, Lee S (2018) Sequential exploration-exploitation with dynamic trade-off for efficient reliability analysis of complex engineered systems. Struct Multidisc Optim 57(1):235–250
Shan S, Wang G (2008) Reliable design space and complete single-loop reliability-based design optimization. Reliab Eng Syst Saf 93(8):1218–1230
Sun Z, Wang J, Li R, Tong C (2017) LIF: a new Kriging based learning function and its application to structural reliability analysis. Reliab Eng Syst Saf 157:152–165
Teixeira R, Nogal M, O'Connor A, Nichols J, Dumas A (2019) Stress-cycle fatigue design with Kriging applied to offshore wind turbines. Int J Fatigue 125:454–467
Teixeira R, Nogal M, O'Connor A (2021) Adaptive approaches in metamodel-based reliability analysis: a review. Struct Saf 89:102019
Toal DJ (2023) Applications of multi-fidelity multi-output Kriging to engineering design optimization. Struct Multidisc Optim 66(6):125
UQLab (2022) UQLab user manual reliability-based design optimization. UQLab. https://uqftp.ethz.ch/uqlab_doc_pdf/2.0.0/UserManual_RBDO.pdf. Accessed 28 June 2022
Wang Z, Wang P (2014) A maximum confidence enhancement based sequential sampling scheme for simulation-based design. J Mech des 136(2):021006
Wang J, Guoji Xu, Li Y, Kareem A (2022a) AKSE: a novel adaptive Kriging method combining sampling region scheme and error-based stopping criterion for structural reliability analysis. Reliab Eng Syst Saf 219:108214
Wang L, Yerramilli S, Iyer A, Apley D, Zhu P, Chen W (2022b) Scalable Gaussian processes for data-driven design using big data with categorical factors. J Mech Des 144(2):021703
Yang M, Zhang D, Wang F, Han Xu (2022) Efficient local adaptive Kriging approximation method with single-loop strategy for reliability-based design optimization. Comput Methods Appl Mech Eng 390:114462
Youn BD, Choi KK, Liu Du (2005) Enriched performance measure approach for reliability-based design optimization. AIAA J 43(4):874–884
Youn BD, Choi KK, Yang R-J, Gu L (2004) Reliability-based design optimization for crashworthiness of vehicle side impact. Struct Multidisc Optim 26(3):272–283
Yousefpour A, Foumani ZZ, Shishehbor M, Mora C, Bostanabad R (2024) GP+: a Python library for kernel-based learning via Gaussian processes. Adv Eng Softw 195:103686
Zhang X, Wang L, Sørensen JD (2019) REIF: a novel active-learning function toward adaptive Kriging surrogate models for structural reliability analysis. Reliab Eng Syst Saf 185:440–454
Zhao L, Choi KK, Lee I (2011) Metamodeling method using dynamic kriging for design optimization. AIAA J 49(9):2034–2046

Zhu Z, Du X (2016) Reliability analysis with Monte Carlo simulation and dependent Kriging predictions. J Mech Des 138(12):121403