Noname manuscript No.
(will be inserted by the editor)

A Robust Multi-Objective Bayesian Optimization Framework Considering Input Uncertainty

Jixiang Qing · Ivo Couckuyt · Tom Dhaene

Received: date / Accepted: date

ABSTRACT

Bayesian optimization is a popular tool for optimizing time-consuming objective functions with a limited number of function evaluations. In real-life applications like engineering design, the designer often wants to take multiple objectives as well as input uncertainty into account to find a set of robust solutions. While this is an active topic in single-objective Bayesian optimization, it is less investigated in the multi-objective case. We introduce a novel Bayesian optimization framework to perform multi-objective optimization considering input uncertainty. We propose a robust Gaussian Process model to infer the Bayes risk criterion to quantify robustness, and we develop a two-stage Bayesian optimization process to search for a robust Pareto frontier, i.e., solutions that have good average performance under input uncertainty. The complete framework supports various distributions of the input uncertainty and takes full advantage of parallel computing. We demonstrate the effectiveness of the framework through numerical benchmarks.

1 Introduction

In many real-life applications, we are faced with multiple conflicting goals. For instance, tuning the topology of neural networks for accuracy as well as inference time (Fernández-Sánchez et al., 2020). A solution that is optimal for all objectives usually does not exist, and one has to compromise: identify a set of solutions that provides a trade-off among different objectives. Moreover, the calculation of the objectives sometimes requires a significant computational

J. Qing
Ghent University – imec, IDLab, Department of Information Technology (INTEC), Tech Lane – Zwijnaarde 126, 9052 Ghent, Belgium
E-mail: Jixiang.Qing@UGent.be

effort. Hence, a Multi-Objective Optimization (MOO) strategy, which is able to quickly locate the optimal trade-offs while reducing the number of expensive evaluations, is of practical interest.

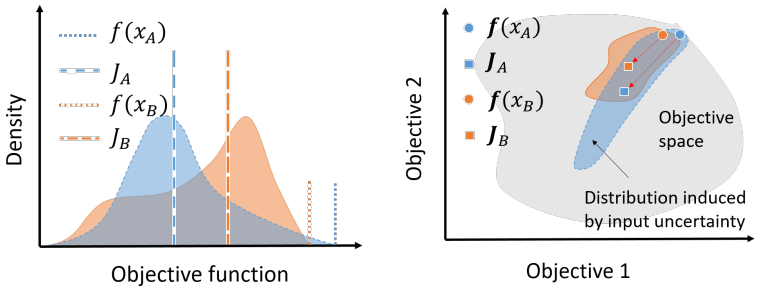
Multi-Objective Bayesian Optimization (MOBO) (e.g., (Daulton et al., 2020; Yang et al., 2019)) is a well-established efficient global optimization (EGO) (Jones et al. (1998)) technique to search for an optimal trade-off between conflicting objectives. Its useful properties, including data-efficiency and an agnostic treatment of the objective function, have made MOBO a widely applicable optimization technique, especially where the objectives are time-consuming to evaluate.

In a chaotic world full of uncertainties, it is almost impossible to implement an optimal solution exactly as defined. For instance, consider an optimal configuration of a system found by MOBO. Any manufacturing uncertainty could result in a slightly different configuration and hence result in a possible degradation of the actual performance. Among these uncertainties, we are specifically interested in considering **input uncertainty**: a common uncertainty type caused by perturbations of the input parameters, that might result in different outputs. Considering input uncertainty in MOO is important to ensure that the final implemented optimal solutions are still likely to be satisfactory. Hence, it is also of high interest in MOBO.

Limitation of current approaches Data-efficient approaches have been proposed to perform MOBO considering input uncertainty (Zhou et al. (2018); Rivier and Congedo (2018)). These approaches extend existing robust MOO methodologies with a computationally efficient surrogate model, however, the surrogate model is only utilized in a non-Bayesian way, i.e, the posterior mean is used as a point estimation, and the model refinement step has to be defined explicitly. This has usually resulted a complicated robust MOO framework. Motivated by these, we propose a lightweight robust MOO framework that deals with robustness in a principle way and still enjoys the elegance of the standard BO flow.

Contributions This paper introduces a Robust Multi-Objective Bayesian Optimization framework to pursue a set of optimal solutions that considers a pre-specified Input Uncertainty (RMOBO-IU). In order to handle the input uncertainty, we optimize a robustness measure, defined as the expectation of the objective distribution induced by the input uncertainty also known as, or Bayes risk (Beland and Nair, 2017) (see Fig. 1) (Deb and Gupta, 2005). To provide a data-efficient inference of this unobservable quantity. we construct a *Robust Gaussian Process* (R-GP) as the surrogate model of the Bayes risk, where a deterministic GP realization can be achieved through the utilization of the Sample Average Approximation (SAA) (Kleywegt et al., 2002; Balandat et al., 2019).

Note that there is a mismatch in the type of uncertainty provided by the R-GP and the uncertainty expected by a common myopic acquisition function, as the latter usually implicitly assumes that this uncertainty comes from a random variable that is directly observable. In order to mitigate this issue, we propose a two-stage approach that can handle existing acquisition functions,



(a) (b)

Fig. 1: Comparing two solutions in a maximization problem using their Bayes risk measures J_A and J_B, i.e., the expectation of the objective function under uncertainty as defined in Eq. 1 a) Single-objective: candidate $\boldsymbol{x}_B$ (orange) is superior over $\boldsymbol{x}_A$ (blue) as its Bayes risk measure J_B is higher. b) Multi-objective: candidate $\boldsymbol{x}_B$ is preferable as $\boldsymbol{J}_B$ dominates the Bayes risk $\boldsymbol{J}_A$ of candidate $\boldsymbol{x}_A$.

including myopic acquisition functions which are commonly used in MOBO. The proposed flexible RMOBO-IU framework, illustrated in Fig. 2 and detailed in Algorithm. 1, can be used with existing acquisition functions and with different input uncertainty distributions. The effectiveness of this novel method has been demonstrated on several synthetic functions.

The key contributions can be highlighted as:

1. A **Bayesian optimization taxonomy** for robust multi-objective optimization.
2. A deterministic Robust Gaussian Process (R-GP), using the efficient Sample Average Approximation (SAA) based Monte Carlo kernel expectation approximation (SAA-MC KE) to infer the Bayes risk, with a proper complexity analysis.
3. We highlight some problems when applying a myopic acquisition function with a robust Gaussian Process and present a novel nested active learning policy to alleviate these.
4. New synthetic benchmark problems for robust multi-objective Bayesian optimization under input uncertainty.

The remaining of the paper is structured as follows. First, the background and related techniques are described in section 2. The RMOBO-IU framework, including the model description, is introduced in section 3. The numerical experiments are presented in section 4. Conclusions are provided in section 5.

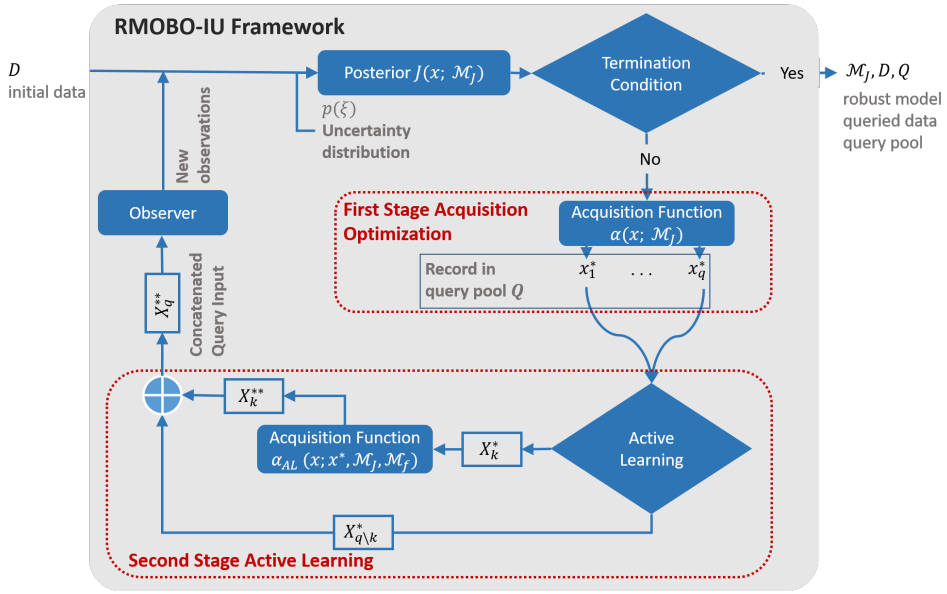


Fig. 2: RMOBO-IU flowchart: The Bayes risk J of the objectives are derived from Gaussian processes $\mathcal{M}_J$. In the first stage, an off-the-shelf acquisition function is used to select (q) query points: $\boldsymbol{X}_q^* := \{\boldsymbol{x}_1^*, ..., \boldsymbol{x}_q^*\}$. For the second stage, these points are then re-investigated by an Active Learning (AL) acquisition function if the AL process activation condition has been met. The updated query points are then fed to the expensive objective observer and augment the training data. The iteration loop continues until a termination condition has been met. Eventually, the robust model, queried data, and the query pool can be utilized to make optimal robust recommendations.

2 Preliminaries and Related Work

2.1 Preliminaries

Multi-Objective Optimization (MOO) methods search for optimal solutions considering multiple objectives simultaneously. This can be mathematically expressed as finding the optimum of a vector-valued function $\boldsymbol{f} := \{f_1, ..., f_M\}$ in a bounded design space $\mathcal{X} \subset \mathbb{R}^d$, where M represents the number of objectives. In the context of MOO, the comparison of different candidates is done through a **ranking** mechanism A_{rank}. Considering the goal of **maximizing** each objective function, a candidate $\boldsymbol{x}$ is preferable to $\boldsymbol{x}'$ if $\forall i \in M : f_i(\boldsymbol{x}) \geq f_i(\boldsymbol{x}')$ and $\exists i \in M : f_i(\boldsymbol{x}) > f_i(\boldsymbol{x}')$. This specific ranking strategy is termed as **dominance** $(\succ)$ and described as $\boldsymbol{f}(\boldsymbol{x})$ dominates $\boldsymbol{f}(\boldsymbol{x}')$: $\boldsymbol{f}(\boldsymbol{x}) \succ \boldsymbol{f}(\boldsymbol{x}')$. In MOO, the candidate $\boldsymbol{x}$ is defined as **Pareto optimal input** if $\nexists \boldsymbol{x}' \in \mathcal{X}$ such that $\boldsymbol{f}(\boldsymbol{x}') \succ \boldsymbol{f}(\boldsymbol{x})$. In this case, $\boldsymbol{f}(\boldsymbol{x})$ is defined as a **Pareto optimal point**. Given that different objectives usually conflict with

each other, MOO seeks for a **Pareto frontier** $\mathcal{F}^*$ that consists of all the objective values $\boldsymbol{f}$ of the Pareto optimal solutions in the bounded design space $\mathcal{X}$: $\mathcal{F}^* := \{\boldsymbol{f} \in \mathbb{F}_{\boldsymbol{f}} | \nexists \boldsymbol{f}_{\boldsymbol{x}'} \in \mathbb{F}_{\boldsymbol{f}} \ s.t. \ \boldsymbol{f}_{\boldsymbol{x}'} \succ \boldsymbol{f}\}$, where $\mathbb{F}_{\boldsymbol{f}} := \{\boldsymbol{f}(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{X}\}$.

In many scenarios, the vector-valued function $\boldsymbol{f}$ does not have a closed-form expression, and observing the function value may have a high computational cost. For this class of problems, it is of paramount importance to restrict the number of function queries when searching for $\mathcal{F}^*$.

Bayesian Optimization (BO) (Jones et al., 1998) is a sequential model-based approach to solving optimization problems efficiently (Shahriari et al., 2015). Starting with a few training samples $D = \{\boldsymbol{X}, F\}$, it builds a Bayesian posterior model $\mathcal{M}$ (with a Gaussian Process (GP) as a common choice (Rasmussen, 2003)), as a computationally efficient **surrogate model** of f. Given the predictive distribution from the surrogate model, an **acquisition function** can be defined as a measure of informativeness for any point $\boldsymbol{x}$ in the design space. It is hence able to search and query the most informative candidate $\{\boldsymbol{x}, f(\boldsymbol{x})\}$ to augment the dataset D and update $\mathcal{M}$ accordingly. This process of refining the posterior model and searching for optimal candidates can be conducted sequentially until a predefined stopping criterion has been met. Eventually, the final model $\mathcal{M}$ and the dataset D can be utilized for recommending optimal solutions. The same paradigm is usually referred to as **Multi-Objective Bayesian Optimization (MOBO)** when f is vector-valued, and the goal is searching for the Pareto frontier $\mathcal{F}^*$.

Input Uncertainty is a common type of uncertainty that is studied in this paper. Suppose we would like to implement a configuration $\boldsymbol{x}$. The input noise, which can be formulated as an additive noise term sampled from a distribution $\boldsymbol{\xi} \sim p(\boldsymbol{\xi})$, could result in a different implementation $\boldsymbol{x} + \boldsymbol{\xi}$ that can worsen the performance. The additive noise distribution $\boldsymbol{\xi}$ results in a distribution of possible objective function values $p(\boldsymbol{f}(\boldsymbol{x} + \boldsymbol{\xi})|\boldsymbol{\xi})$, which is refereed to as the **objective distribution**. In this research, we assume that the additive noise distribution form is known from practitioners and can be unbounded.

Optimality in Robust Multi-objective Optimization The Bayes risk measure is optimized to obtain a set of non-dominated solutions in Bayes risk space. As the robustness is measured through pairwise comparison as depicted in Fig. 1, this assumes that the practitioner is more interested in the robustness of certain candidates among the Pareto Frontier instead of the (robustness of the) whole Pareto frontier. Remark that this is usually the case for MOO where only one solution is selected a posteriori. Hence, the goal of our formulation is to search for a *robust Pareto frontier*, which we define as the Pareto frontier in the Bayes risk space: $\mathcal{F}_{\boldsymbol{J}}^* := \{\boldsymbol{J} \in \mathbb{J}_{\boldsymbol{J}} | \nexists \boldsymbol{J}_{\boldsymbol{x}'} \in \mathbb{J}_{\boldsymbol{J}} \ s.t. \ \boldsymbol{J}_{\boldsymbol{x}'} \succ \boldsymbol{J}\}$, where $\mathbb{J}_{\boldsymbol{J}} := \{\boldsymbol{J}(\boldsymbol{x}) | \boldsymbol{x} \in \mathcal{X}\}$.

2.2 Related Work

Several approaches have been proposed to link the robust MOO methodology with a GP surrogate model (Xia et al., 2014; Zhou et al., 2018; Rivier and

Congedo, 2018; Abbas et al., 2016). Xia et al. (2014) consider the worst-case robustness scenario, for which the worst objective function is extracted from the GP. Zhou et al. (2018) introduce a GP surrogate model assisted multi-objective robust optimization strategy based on Li et al. (2005), where the GP acts as an efficient surrogate and hence, as a cheap intermediary for a genetic algorithm to search for the optimum. In a more probabilistic setting, Rivier and Congedo (2018) propose an interesting bounding box-based efficient MOO framework. For each observation, a conservative bounding box is constructed based on some robustness measures approximated by MC sampling on the surrogate model, with the assumption that an extra aleatory variable can be modeled with a uniform distribution built upon the bounding box. The concept of probability of box-based Pareto dominance is utilized to compare against different aleatory variables hence different observations. Subsequently, it can search for the optimum or improve the surrogate model accuracy accordingly. Nevertheless, while equipped with a GP as a probabilistic surrogate model, the robustness measure of the above-mentioned approaches are usually extracted in a non-Bayesian way as a point estimation from the posterior mean, and the surrogate model refinement step must be defined explicitly. A more principled BO-like RMOBO framework has yet to be revealed.

3 RMOBO-IU Framework

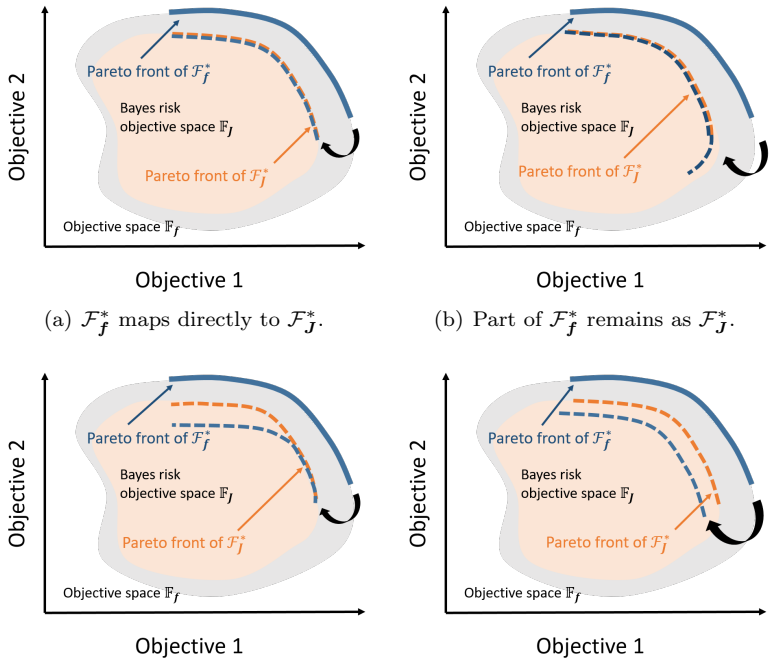
3.1 Optimizing Bayes Risk versus Optimizing the Original Objective Function

The Bayes risk is utilized as objective in the RMOBO-IU framework:

$$\begin{aligned} &\underset{\boldsymbol{x}\in\mathcal{X}\subset\mathbb{R}^d}{\text{Maximize}}\ J_1(\boldsymbol{x}), J_2(\boldsymbol{x}), ..., J_M(\boldsymbol{x}) \\ &\text{where } J(\boldsymbol{x}) = \int f(\boldsymbol{x}+\boldsymbol{\xi})p(\boldsymbol{\xi})d\boldsymbol{\xi} \end{aligned} \tag{1}$$

Given the fact that we are optimizing the Bayes risk $\boldsymbol{J}$, we use $\mathcal{F}_{\boldsymbol{J}}^*$ and $\mathcal{F}_{\boldsymbol{f}}^*$ to represent the Pareto frontier of the robust and non-robust optimization problem (i.e., optimize the original objective function $\boldsymbol{f}$), respectively. It is natural to wonder what the difference is between $\mathcal{F}_{\boldsymbol{f}}^*$ and $\mathcal{F}_{\boldsymbol{J}}^*$. Using the objective space, the difference can be categorized into four different cases (Deb and Gupta, 2005) as shown in Fig. 3. Except for the first case, the remaining cases clearly show that $\mathcal{F}_{\boldsymbol{J}}^*$ leads to more robust optimal solutions, at least for some parts of the Pareto fronts.

It might be difficult to determine whether a robust Pareto front exists that is different from $\mathcal{F}_f^*$. This is not trivial to answer due to the agnostic property of the black-box function $\boldsymbol{J}$. Nevertheless, from a practitioner perspective, we define a sufficient condition based on the objective functions which helps to determine whether a distinct robust Pareto front exists:



(a) $\mathcal{F}_{\boldsymbol{f}}^*$ maps directly to $\mathcal{F}_{\boldsymbol{J}}^*$.

(b) Part of $\mathcal{F}_{\boldsymbol{f}}^*$ remains as $\mathcal{F}_{\boldsymbol{J}}^*$.

(c) Part of $\mathcal{F}_{\boldsymbol{f}}^*$ remains in $\mathcal{F}_{\boldsymbol{J}}^*$.

(d) $\mathcal{F}_{\boldsymbol{f}}^*$ is not robust.

Fig. 3: Four different cases comparing the Pareto front $\mathcal{F}_{\boldsymbol{f}}^*$ with the Pareto front $\mathcal{F}_{\boldsymbol{J}}^*$ considering input uncertainty. Images courtesy of Deb and Gupta (2005).

Proposition 1. *If* $\exists f_i \in \boldsymbol{f}, J_i \in \boldsymbol{J}$, *s.t. for* $\boldsymbol{x}_{f_i^*} := \underset{\boldsymbol{x}\in\mathcal{X}}{argmax}\ f_i(\boldsymbol{x})$, $\boldsymbol{x}_{J_i^*} := \underset{\boldsymbol{x}\in\mathcal{X}}{argmax}\ J_i(\boldsymbol{x})$. $\boldsymbol{x}_{f_i^*} \neq \boldsymbol{x}_{J_i^*}$, $\boldsymbol{x}_{J_i^*}$ *is unique and* $\boldsymbol{f}_{\boldsymbol{x}_{J_i^*}} \notin \mathcal{F}_{\boldsymbol{f}}^*$.
Then:
$\exists \boldsymbol{x}_{diff} \in \mathcal{X}$ *such that* $\boldsymbol{J}(\boldsymbol{x}_{diff}) \in \mathcal{F}_{\boldsymbol{J}}^*$ *while* $\boldsymbol{f}(\boldsymbol{x}_{diff}) \notin \mathcal{F}_{\boldsymbol{f}}^*$, *and* $\mathcal{F}_{\boldsymbol{f}}^* \neq \mathcal{F}_{\boldsymbol{J}}^*$.

Proof. Given $\boldsymbol{x}_{f_i^*} := \underset{x\in\mathcal{X}}{argmax}\ f_i(\boldsymbol{x})$, $\boldsymbol{x}_{J_i^*} := \underset{\boldsymbol{x}\in\mathcal{X}}{argmax}\ J_i(\boldsymbol{x})$, having $\boldsymbol{x}_{f_i^*} \neq \boldsymbol{x}_{J_i^*}$ means $f_i(\boldsymbol{x}_{f_i^*}) > f_i(\boldsymbol{x}_{J_i^*})$ and $J_i(\boldsymbol{x}_{f_i^*}) < J_i(\boldsymbol{x}_{J_i^*})$, according to the definition of Pareto dominance, Let $\boldsymbol{x}_{diff} := \boldsymbol{x}_{J_i^*}$, we have $\boldsymbol{J}(\boldsymbol{x}_{diff}) \in \mathcal{F}_{\boldsymbol{J}}^*$ and $\boldsymbol{f}(\boldsymbol{x}_{diff}) \notin \mathcal{F}_{\boldsymbol{f}}^*$. Meanwhile, as $\nexists \boldsymbol{x} \in \{\boldsymbol{x} \in \mathcal{X} | \boldsymbol{f}(\boldsymbol{x}) \in \mathcal{F}_{\boldsymbol{f}}^*\}$ such that the ith component of its outcome: $J_i(\boldsymbol{x}) \geq J_i(\boldsymbol{x}_{diff})$, hence $\mathcal{F}_{\boldsymbol{f}}^* \neq \mathcal{F}_{\boldsymbol{J}}^*$ and the proposition holds. □

The proposition conveys that if the objective function f_i has a different global maximum location $\boldsymbol{x}_{J_i^*}$ (for Bayes risk) which is also not Pareto optimal in the objective space, then there will be a distinct robust Pareto front.

3.2 Inference of the Bayes Risk

3.2.1 Robust Gaussian Process (R-GP)

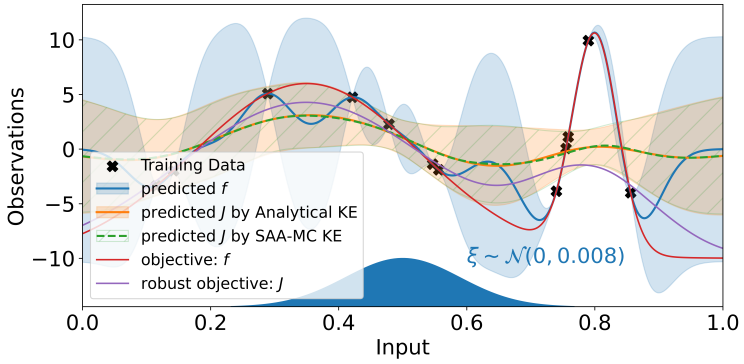


Fig. 4: 1D example of a standard GP (blue) and Robust GP (orange and green) considering a Gaussian input uncertainty distribution (with variance 0.008), illustrated at input $x = 0.5$. It can be observed that the Bayes risk favors the less risky maxima. The comparison of the posterior distribution using analytical KE (orange) and SAA-MC based KE (green) GP is also shown. The SAA-MC KE results in a differentiable approximation of the GP posterior.

Each black-box function i is assumed to be a realization of Gaussian Process. Hence, given limited training data D, the ith GP model $\mathcal{M}_{f_i}$'s posterior representing f_i at $\boldsymbol{x}$ is:

$$m_{f_i}(\boldsymbol{x}|D) = \boldsymbol{k}_{f_i}(\boldsymbol{x})^T \boldsymbol{K}_i^{-1} \boldsymbol{F}_i \tag{2}$$

$$Cov_{f_i}(\boldsymbol{x}|D) = \boldsymbol{k}_{f_i}(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}_{f_i}(\boldsymbol{x})^T \boldsymbol{K}_i^{-1} \boldsymbol{k}_{f_i}(\boldsymbol{x}') \tag{3}$$

where $\boldsymbol{F}_i$ is the training data for the ith black-box function, $\boldsymbol{K}_i$ is the kernel matrix of the ith objective.

Now consider the transformation of expectation through input uncertainty. Since the expectation in Eq. 1 is a linear operator, we can obtain a Gaussian Process for inferring Bayes risk J_i by applying linear transformation rules (Rasmussen, 2003; Papoulis and Pillai, 2002), resulting in:

$$p(J_i|D, \boldsymbol{x}) = \mathcal{N}(m_{J_i}, Cov_{J_i}) \tag{4}$$

$$m_{J_i}(\boldsymbol{x}|D) = \boldsymbol{k}_{J_i f_i}(\boldsymbol{x})^T \boldsymbol{K}_i^{-1} \boldsymbol{F}_i \tag{5}$$

$$Cov_{J_i}(\boldsymbol{x}|D) = \boldsymbol{k}_{J_i}(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}_{J_i f_i}(\boldsymbol{x})^T \boldsymbol{K}_i^{-1} \boldsymbol{k}_{f_i J_i}(\boldsymbol{x}') \tag{6}$$

where $\boldsymbol{k}_{J_i f_i}$ and $\boldsymbol{k}_{J_i}$ are defined using the following Kernel Expectation (KE):

$$\boldsymbol{k}_{J_i f_i}(\boldsymbol{x}) = \int k_{f_i}(\boldsymbol{x}+\boldsymbol{\xi})p(\boldsymbol{\xi})d\boldsymbol{\xi} \tag{7}$$

$$\boldsymbol{k}_{J_i}(\boldsymbol{x},\boldsymbol{x}') = \int\int k_{f_i}(\boldsymbol{x}+\boldsymbol{\xi},\boldsymbol{x}'+\boldsymbol{\xi}')p(\boldsymbol{\xi})p(\boldsymbol{\xi}')d\boldsymbol{\xi}d\boldsymbol{\xi}' \tag{8}$$

We refer to the derived GP that infers the robustness measure as the *robust GP*. For some kernels and uncertainty distributions $p(\boldsymbol{\xi})$, an analytical expression exists for the KE. One of the most well-known analytical KE is the squared exponential kernel under Gaussian input uncertainty (Dallaire et al., 2009), see Fig. 4 (orange posterior mean and uncertainty interval). Unfortunately, for more generic cases, an analytical expression is non-trivial to obtain. In this case, one can defer to Monte Carlo (MC) approximations[1]:

$$m_{J_i}(\boldsymbol{x}|D) \approx \frac{1}{N}\sum_{j=1}^{N}\left[k_{f_i}(\boldsymbol{x}+\boldsymbol{\xi}_j)\right]\boldsymbol{K}_i^{-1}\boldsymbol{F}_i \tag{9}$$

$$Cov_{J_i}(\boldsymbol{x}|D) \approx \frac{1}{N}\sum_{j=1}^{N}\left[k_{f_i}(\boldsymbol{x}+\boldsymbol{\xi}_j,\boldsymbol{x}'+\boldsymbol{\xi}'_j) - k_{f_i}(\boldsymbol{x}+\boldsymbol{\xi}_j,\boldsymbol{x}')\boldsymbol{K}_i^{-1}k_{f_i}(\boldsymbol{x},\boldsymbol{x}'+\boldsymbol{\xi}'_j)\right] \tag{10}$$

While the common approach is to redraw samples $\boldsymbol{\xi}$ for every evaluation point $\boldsymbol{x}$ to obtain the posterior predictive distribution, we apply the sample average approximation (Kleywegt et al., 2002; Balandat et al., 2019) through the MC based kernel expectation (SAA-MC KE). This is illustrated in Fig. 4 (green posterior mean and uncertainty interval). Given a differentiable kernel, by holding MC samples fixed: $E = \{\boldsymbol{\xi}^1, ..., \boldsymbol{\xi}^N\}$ for KE, we are able to provide a deterministic and differentiable approximation of the posterior distribution, which is easily utilizable by off-the-shelf acquisition functions. Furthermore, we can still use gradient-based optimizers for optimizing the acquisition function.

3.2.2 Inference Complexity

We derive the computation complexity of inferencing the Bayes risk J with respect to the test sample size n_{test}, as well as the memory consumption[2] in Table. 1. Fortunately, the main extra computation effort only affects the inference stage instead of the model training stage. The latter is usually regarded as the main bottleneck of GPs. For common GP implementations, the introduction of MC samples increases the complexity N times, i.e., it grows linear with the number of MC samples. We propose to parallelize the computation through N MC samples and so, we trade of the time increment against memory consumption.

[1] To improve the numerical stability, we leverage the methodology of Higham (1988) with a nugget term to search for the nearest positive definite matrices for Eq. 10 when a full covariance posterior matrix is needed.

[2] We report the single storage component that can possibly take the maximum memory, and we do not consider the memory consumption for the original kernel matrix storage as it is not correlated with n_{test}.

Table 1: Inference complexity of a standard GP and R-GP, where n_{tr} is the training sample size and n_{test} is the test sample size. N is the MC sample size for the kernel expectation.

	standard GP	R-GP
Computation Complexity		
Not Full-Cov Inference	$n_{tr}n_{test}$	$N \cdot n_{tr}n_{test}$
Full-Cov Inference	$n_{tr}n_{test}^2$	$N \cdot n_{tr}n_{test}^2$
Memory Consumption (Parallized)		
Not Full-Cov Inference	$max(n_{test}n_{tr})$	$N \cdot max(n_{test}n_{tr})$
Full-Cov Inference	$max(n_{test}^2, n_{test}n_{tr})$	$N \cdot max(n_{test}^2, n_{test}n_{tr})$

3.3 Two-Stage Acquisition Function Optimization Process

3.3.1 First Stage: Acquisition Optimization

As the R-GP provides a (multivariate) normal posterior distribution $p(\boldsymbol{J}|\boldsymbol{x}, D)$, it is convenient to utilize existing (multi-objective) acquisition functions to search for the Pareto frontier $\mathcal{F}_{\boldsymbol{J}}^*$. We use common myopic acquisition functions for MOBO (e.g., Expected Hypervolume Improvement (EHVI) (Yang et al., 2019), Parallel Expected Hypervolume Improvemet (qEHVI) (Daulton et al., 2020, 2021) and Expected Hypervolume Probability of Improvement (EHPI) (Yang et al., 2019; Couckuyt et al., 2014)), with a brief remark below.

Recall that many acquisition functions can be written in the following form (Wilson et al., 2018):

$$\alpha(\boldsymbol{X}_q; \psi, D) = \int_{\boldsymbol{J}_{\boldsymbol{X}_q}} \ell(\boldsymbol{J}_{\boldsymbol{X}_q}; \psi) p(\boldsymbol{J}_{\boldsymbol{X}_q}; m_J, Cov_J) d\boldsymbol{J}_{\boldsymbol{X}_q} \tag{11}$$

where ℓ denotes the utility function (using the acquisition function parameter ψ), $\boldsymbol{X}_q := \{\boldsymbol{x}_1, ..., \boldsymbol{x}_q\}$ represents a batch of q input candidates. For myopic acquisition functions, ψ can be defined as the *current best Pareto frontier* inferred using the R-GPs: $\psi := A_{rank}(\boldsymbol{J}_D|\mathcal{M}_{\boldsymbol{J}}, D)$, and results in the following expression:

$$\begin{aligned}
\alpha(\boldsymbol{X}_q; \psi, D) = &\int_{\boldsymbol{J}_D} \int_{\boldsymbol{J}_{\boldsymbol{X}_q}} \ell(\boldsymbol{J}_{\boldsymbol{X}_q}; A_{rank}(\boldsymbol{J}_D|\mathcal{M}_{\boldsymbol{J}}, D)) \\
&p(\boldsymbol{J}_{\boldsymbol{X}_q}; m_{\boldsymbol{J}_{\boldsymbol{X}_q}}, Cov_{\boldsymbol{J}_{\boldsymbol{X}_q}}) p(\boldsymbol{J}_D; m_{\boldsymbol{J}_D}, Cov_{\boldsymbol{J}_D}) d\boldsymbol{J}_{\boldsymbol{X}_q} d\boldsymbol{J}_D \\
\approx &\int_{\boldsymbol{J}_{\boldsymbol{X}_q}} \ell(\boldsymbol{J}_{\boldsymbol{X}_q}; A_{rank}(\overline{\boldsymbol{J}_D}|\mathcal{M}_{\boldsymbol{J}}, D)) p(\boldsymbol{J}_{\boldsymbol{X}_q}; m_{\boldsymbol{J}_{\boldsymbol{X}_q}}, Cov_{\boldsymbol{J}_{\boldsymbol{X}_q}}) d\boldsymbol{J}_{\boldsymbol{X}_q}
\end{aligned} \tag{12}$$

The operator A_{rank} is the non-dominated sorting operation. We note that the extracted current best Pareto frontier is also a distribution due to the fact that the Bayes risk $\boldsymbol{J}$ is not observable. We could simplify the problem by making use of the posterior mean of the R-GP: $A_{rank}(\overline{\boldsymbol{J}_D}|\mathcal{M}_{\boldsymbol{J}}, D)$ as an approximation to avoid the integration of Pareto frontier distribution (Gramacy and Lee, 2010), resulting in the last line of Eq. 12. Nevertheless, the distribution of the Pareto frontier can also be considered, for instance, by leveraging MC sampling (Daulton et al., 2021). Finally, we remark the last line of Eq. 12 can be analytically calculated exactly for EHVI, EHPI, and approximately calculated by qEHVI acquisition functions.

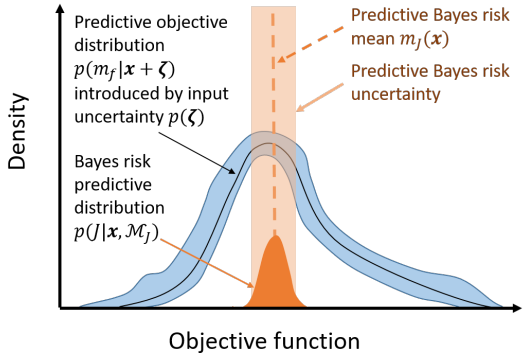


Fig. 5: Illustration of the non-observable property of the posterior distribution of R-GP at input location $\boldsymbol{x}$. The R-GP prediction of the Bayes risk's uncertainty of is illustrated as the orange shaded area. The blue shaded area represents the uncertainty of the predictive objective distribution which comes from the model $\mathcal{M}_f$. This implies since we do not have direct observation $\{\boldsymbol{x}, \boldsymbol{J}(\boldsymbol{x})\}$, only when the blue shaded area is reduced by sampling the Bayes risk predictive uncertainty will be reduced to zero.

3.3.2 Second Stage: Active Learning for Reducing Uncertainty

While we can already use the acquisition function to search for the Pareto front $\mathcal{F}^*_{\boldsymbol{J}}$, we note that there is an inconsistency between the R-GP's inference $p(\boldsymbol{J}|\boldsymbol{x}, D)$ and what the acquisition functions mentioned above expects. More specifically, as illustrated in Fig. 5, the posterior variance of the R-GP posterior aggregates uncertainty coming from the input uncertainty and the model approximations $\mathcal{M}_{\boldsymbol{f}}$. This means the inferred Bayes risk $\boldsymbol{J}|\boldsymbol{x}, D$ could still be uncertain (i.e., the posterior variance of $p(\boldsymbol{J}|\boldsymbol{x}, D)$ doesn't vanish to zero) at $\boldsymbol{x}$ even if the model has already included data at $\boldsymbol{x}$. Nevertheless, the myopic acquisition function, which build on the assumption that its predictive

quantity $p(\boldsymbol{J}|\boldsymbol{x}, D)$ to be directly observable (Iwazaki et al., 2021; Fröhlich et al., 2020), cannot handle this inconsistency intrinsically.

This results in two possible issues when applying of standard BO. First, as the input uncertainty could result in a design that is outside the bounded design space, the inference variance of the Bayes risk cannot be lowered to zero when restricting sampling only inside the design space. This results in the acquisition function adding duplicate samples at boundary locations. Secondly, common acquisition functions will waste resources on the same sample within the design space in a futile effort to reduce uncertainty, resulting in another duplication issue, which can also impose numerical instabilities to the model.

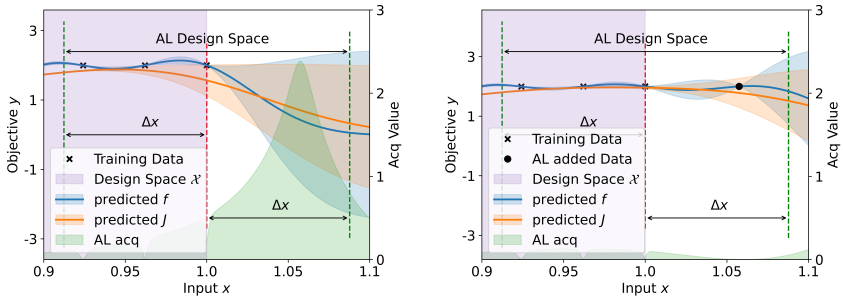
While not explicitly discussed in most of the existing research, we remark that these issues generically exist in single-objective robust BO when performing optimization on the Bayes risk. In order to resolve these issues, we propose an AL policy. We introduce an information-theoretic-based active learning acquisition function. As illustrated in Fig. 6, its intuitive interpretation is that we want to maximally reduce the uncertainty of the predictive distribution of $\boldsymbol{J}|D$ at candidate $\boldsymbol{x}^*$. Instead of directly sampling at $\boldsymbol{x}^*$, we seek the candidate that can maximally reduce its uncertainty, which is quantified by differential entropy.

$$\alpha_{AL} = \mathbb{H}[\boldsymbol{J}(\boldsymbol{x}^*|D)] - \mathbb{E}_{\boldsymbol{f}(\boldsymbol{x})}\mathbb{H}[\boldsymbol{J}(\boldsymbol{x}^*|D, \{\boldsymbol{x}, \boldsymbol{f}(\boldsymbol{x})\})] \tag{13}$$

where the expectation is taken through all possible $\boldsymbol{f}(\boldsymbol{x})$ described by the GP posterior. Given the assumption that we fixed the GP model $\mathcal{M}$'s hyperparameters during the acquisition optimization, the variance of $\boldsymbol{J}(\boldsymbol{x}^*|D, \{\boldsymbol{x}, \boldsymbol{f}(\boldsymbol{x})\})$ is independent of $\boldsymbol{f}(\boldsymbol{x})$, and hence it is sensible to avoid the expensive computation of the one dimensional integration by only making use of the posterior mean of $\boldsymbol{f}(\boldsymbol{x})$:

$$\begin{aligned} \alpha_{AL} &\approx \mathbb{H}[\boldsymbol{J}(\boldsymbol{x}^*|D)] - \mathbb{H}[\boldsymbol{J}(\boldsymbol{x}^*|D, \{\boldsymbol{x}, \overline{\boldsymbol{f}}(\boldsymbol{x})\})] \\ &= \frac{1}{2}\log\frac{\prod_{i=1}^{M}\mathbb{V}_{J_i}(\boldsymbol{x}^*|D)}{\prod_{i=1}^{M}\mathbb{V}_{J_i}(\boldsymbol{x}^*|D, \{\boldsymbol{x}, \overline{\boldsymbol{f}}(\boldsymbol{x})\})} \end{aligned} \tag{14}$$

where $\mathbb{V}_{J_i}$ represents the variance of ith J. As AL brings extra computational complexity, it is sensible to only use it when at least one of the following conditions is met: (i). when BO has resulted in sampling at the design space boundary, which can be defined as: $min_{vec}(\boldsymbol{x}^* - \boldsymbol{B}_{\mathcal{X}_l}) < \epsilon$ **or** $min_{vec}(\boldsymbol{B}_{\mathcal{X}_u} - \boldsymbol{x}*) < \epsilon$, where $\boldsymbol{B}_{\mathcal{X}_l}, \boldsymbol{B}_{\mathcal{X}_u}$ represents the lowest and largest point coordinates that can define the design space, ϵ is a small non-negative threshold, min_{vec} is the coordinate-wise minimum operator, (ii). when BO has resulted in duplicate sampling in the design space: $min||\boldsymbol{X} - \boldsymbol{x}^*|| < \epsilon$. Assuming the acquisition function has resulted in sampling $\boldsymbol{x}^*$, we propose to perform the AL optimization step within the bounded space $\mathcal{B} : [\boldsymbol{x}^* - \Delta\boldsymbol{x}, \boldsymbol{x}^* + \Delta\boldsymbol{x}]$, where $\Delta\boldsymbol{x}$ is a hyperparameter (illustrated in Fig. 6) that needs to be specified upfront. For bounded input uncertainty distributions, this can be intuitively specified as the distribution boundary; for the unbounded input uncertainty distribution



(a) Before the AL process (b) After the AL process

Fig. 6: Illustration of the *boundary issue* and the *duplication issue* in robust optimization. The predictive uncertainty of Bayes risk at $x = 1$ (i.e., $\mathbb{V}(J(x = 1))$) cannot be lowered to zero even by sampling exactly at this location. The active learning (green) acquisition function is proposed to resolve the *boundary issue*. The design space boundary is illustrated as the red vertical dashed line. The AL process can hence result in sampling outside the original design space in order to reduce the uncertainty at the design space boundary.

like Gaussian distribution, a distance between the mean and 97.5 percentage of the marginal distribution can be chosen as $\boldsymbol{\Delta x}$.

3.4 Framework Outline

The complete RMOBO-IU approach is presented in Algorithm. 1. The main paradigm is similar to a standard BO flow. Starting with a limited amount of data, the R-GP is constructed, and the Bayes risks are inferred. The first stage acquisition optimization (line 4) is conducted to search for the robust Pareto optimal points. Next, in the second phase, AL process (line 7-13) is utilized as needed to pursue better sampling candidates. Once the optimization has stopped, the Pareto front $\mathcal{F}_{\boldsymbol{J}}^*$ can be extracted based on the final models (out-of-sample) or on the sampled points (in-sample). We also note that this framework can be used for single objective robust BO if the objective number $M = 1$, and the ranking operation in Eq. 12 is defined as $A_{rank} := max(\cdot)$.

4 Numerical Investigation

There are relatively few benchmark functions in literature for robust multi-objective optimization. Therefore, we construct some new synthetic functions for benchmarking RMOBO-IU. They are listed in Table 2 and detailed in appendix A, and used with various input uncertainty distributions. We note that these new synthetic functions cover all 4 cases that we have discussed

Algorithm 1: Robust Multi-Objective Bayesian Optimization considering Input Uncertainty (RMOBO-IU)

Input: max iter: N_{iter}, design space : $\mathcal{X}$, training data $D = \{\boldsymbol{X}, \boldsymbol{Y}\}$, query pool: $Q = \{\}$, design space boundary stack: $\boldsymbol{B}_{\mathcal{X}} = \{\boldsymbol{B}_{\mathcal{X}_l}, \boldsymbol{B}_{\mathcal{X}_u}\}$, minimum distance threshold: ϵ, Δx ;
for *iter* $:= 1$ *to* N_{iter} **do**
  construct model based on D: $\mathcal{M}_{\boldsymbol{J}} : \{J_1 \sim \mathcal{GP}'_1, ..., J_M \sim \mathcal{GP}'_M\}$
  $\boldsymbol{X}_q^* = \underset{\boldsymbol{x} \in \mathcal{X}}{argmax}\ \alpha(\boldsymbol{X}_q, \psi, \mathcal{M}_{\boldsymbol{J}})$
  Augment query pool: $Q = \{Q \cup \boldsymbol{X}_q^*\}$
  initialize AL and BO pool: $X_k^{**} = \{\}$, $X_{q\setminus k}^* = \{\}$
  for $j := 1$ *to* q **do**
    if $min||\boldsymbol{X} - \boldsymbol{x}_j^*|| < \epsilon$ ***or*** $min_{vec}(\boldsymbol{x}_j^* - \boldsymbol{B}_{\mathcal{X}_l}) < \epsilon$ ***or*** $min_{vec}(\boldsymbol{B}_{\mathcal{X}_u} - \boldsymbol{x}_j^*) < \epsilon$ **then**
      $\boldsymbol{x}_j^{**} = arg \underset{\boldsymbol{x} \in [\boldsymbol{x}_j^* - \boldsymbol{\Delta_x}, \boldsymbol{x}_j^* + \boldsymbol{\Delta_x}]}{max}\ \alpha_{\mathrm{AL}}(\boldsymbol{x}, \boldsymbol{x}_j^*, \mathcal{M}_{\boldsymbol{J}}, \mathcal{M}_{\boldsymbol{f}})$
      $\boldsymbol{X}_k^{**} = \boldsymbol{X}_k^{**} \cup \boldsymbol{x}_j^{**}$
    else
      $\boldsymbol{X}_{q\setminus k}^* = \boldsymbol{X}_{q\setminus k}^* \cup \boldsymbol{x}_j^*$
    end
    Concatenate: $\boldsymbol{X}_q^{**} = \boldsymbol{X}_k^{**} \cup \boldsymbol{X}_{q\setminus k}^*$
  end
  Query observations and augment training data: $D = \{D \cup \{X_q^{**}, \boldsymbol{f}(X_q^{**})\}\}$
end
Concatenate optimal candidates : $\boldsymbol{X}_{cand} = \{\boldsymbol{x} \in \mathcal{X} : \boldsymbol{x} \in \boldsymbol{X} \cup Q\}$
Output ranking on model inferred optimal candidates: $A_{rank}(\mathcal{M}_{\boldsymbol{J}}(\boldsymbol{X}_{cand}))$, robust model: $\mathcal{M}_{\boldsymbol{J}}$

Table 2: Bi-objective benchmark function settings (see also appendix A), where $t(\cdot), \mathcal{N}(\cdot), Tr\mathcal{N}(\cdot), \mathrm{U}(\cdot)$ represents the student-t, normal, truncated normal and uniform distribution respectviely.

Function	ξ distribution	Input Dimension	Problem Type (Fig, 3)	AL design space $\Delta\boldsymbol{x}$
VLMOP2	$t(200, 0, 0.01^2)$	2	C.1	[0.0166, 0.0166]
SinLinForrester	$\mathcal{N}(0, 0.05^2)$	1	C.2	[0.098, 0.098]
MDTP2	$Tr\mathcal{N}([0,0], [0.02^2, 0.04^2]$ $[-0.05, -0.05]$, [0.05, 0.05])	2	C.3	[0.05, 0.05]
MDTP3	U([-2e-2, -0.1], [2e-2, 0.1])	2	C.4	[0.02, 0.1]
BraninGMM	U([-0.2, -0.2], [0.2, 0.2]	2	C.4	[0.02, 0.02]

in Section. 3.1. We employ the squared exponential kernel with a Maximum A Posterior (MAP) strategy, driven by the L-BFGS-B optimizer. We follow the same strategy of Fröhlich et al. (2020) by specifying a log-normal prior on

the lengthscales, and 2000 MC samples are used for approximating the kernel expectation.

The code is implemented using the Trieste library (Berkeley et al., 2021), and we test the RMOBO-IU framework using two popular acquisition functions for MOBO, i.e., EHVI and qEHVI. We start each benchmark with $5d$ initial data points uniformly generated in the design space, where d is the problem dimensionality. The experiments are conducted on a server with Intel(R) Xeon(R) CPUs E5-2640 v4 @ 2.40GHz, and each synthetic problem is repeated 30 times for robustness.

The performance is evaluated using the Averaged Hausdorff Distances (AVD) based indicator (Eq. 45 of Schutze et al. (2012)) in the scaled objective space[3] as the performance metric with $p = 2$. The reference Pareto frontier $\boldsymbol{F}^*$ is generated using an exhaustive NSGAII (Deb et al., 2002) search with population size 60. We compare RMOBO-IU (using an in-sample (IS) strategy) with standard MOBO, as well as a non-Bayesian MOO strategy. In the latter we use the NSGAII evolutionary algorithm (EA) based on a one-shot learned standard GPs as an Out-of-Sample (OS) strategy, which we refer to as the EA-GP-OS method. [4]

The AVD's convergence histories of different acquisition functions[5] and strategies are depicted in Fig. 7, and the final recommended Pareto fronts are shown in Fig. 8, more experiment detailes are sent to appendix B. According to the results, it can be observed that for the VLMOP2 problem (case 1), its robust Pareto frontier is similar to its original Pareto frontier and that has led to similar convergence properties of the AVD measure. For the other cases, RMOBO-IU converges to the robust Pareto frontier while the non-robust MOO identifies of course non-robust solutions. We also note that in general a faster convergence speed in terms of (batch) iterations can be observed by utilizing batch acquisition functions.

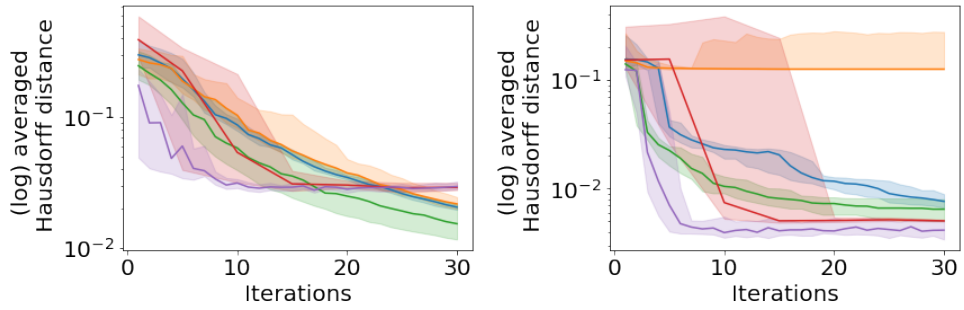
We also provide out-of-sample recommendations based on the R-GP for our RMOBO method to compare with EA-GP-OS, which we denote as RMOBO-EHVI-OS [6]. We note that RMOBO-EHVI-OS has in general an improved performance over EA-GP-OS, while the EA-GP-OS method is more robust for BraninGMM. Overall, while the out-of-sample strategies demonstrate better results than in-sample strategies on some benchmarks, their performance are not consistent across all problems. The worse performance can be shown especially on MDTP2 and MDTP3, where we deduce that if the problem is more difficult for an accurate surrogate model, the out-of-sample recommendation can have outliers of the Pareto frontier leading to a worse AVD score.

[3] When calculating the AVD metric, we scale the objective space to $[0, 1]^M$ based on the real Pareto front. This scaling aims to reduce bias from AVD if the magnitude between the objectives differ significantly.

[4] The Bayes risk of the EA-GP-OS method is calculated using 2000 Monte Carlo samples on the GP posterior mean. For NSGAII we use a population size of 20 and 200 generations.

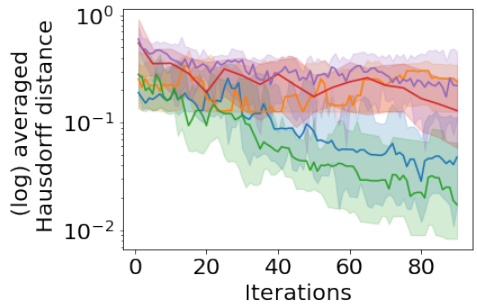
[5] The qEHVI acquisition function has batch size $q = 2$.

[6] We use the same NSGAII settings as used in EA-GP-OS.

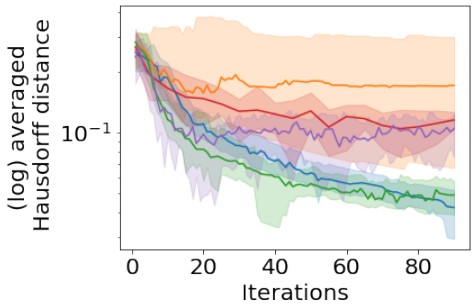


(a) VLMOP2

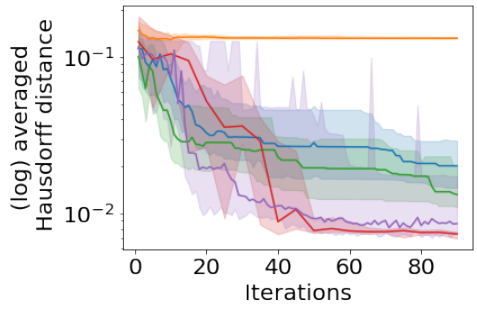
(b) SinLinForrester Function



(c) MDTP2 Function



(d) MDTP3 Function



(e) BraninGMM

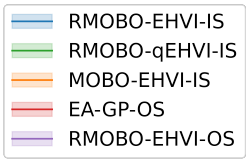
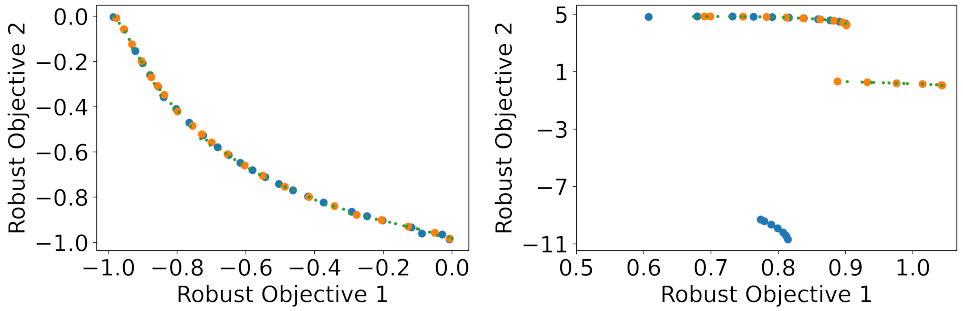
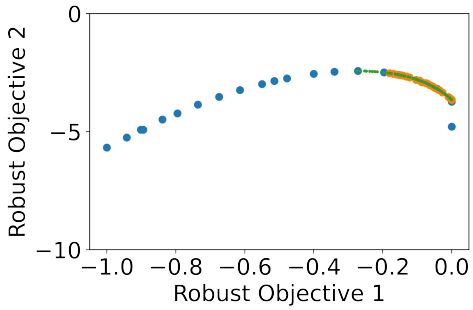


Fig. 7: Synthetic benchmark results for the AVD score with respect to the number of iterations. The median across 30 experiments is represented as a line and the 25/75th percentiles are reported as the shaded area.

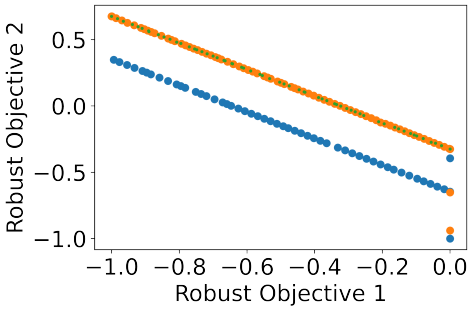


(a) VLMOP2

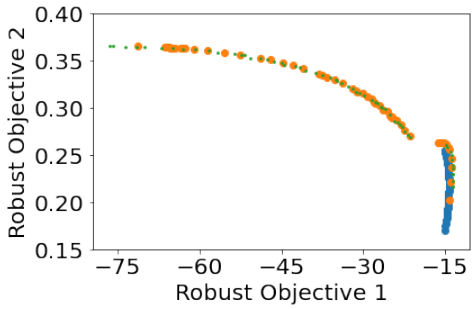
(b) SinLinForrester Function



(c) MDTP2 Function



(d) MDTP3 Function



(e) BraninGMM

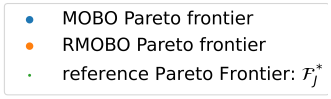


Fig. 8: Pareto front comparison of RMOBO and MOBO which doesn't inherently consider input uncertainty. With the consideration of input uncertainty, the Pareto front of MOBO (orange) is less optimal than RMOBO (blue) based on in-sample recommendations.

Hence, we recommend to keep using in-sample strategies as a more robust choice.

5 Conclusion

We presented RMOBO-IU: an approach for robust multi-objective optimization within the Bayesian optimization framework which considers input uncertainty.

We optimize for Bayes risk, which is efficiently inferred using a robust Gaussian Process. The robust Gaussian Process is integrated in a two-stage Bayesian optimization process to search for the robust Pareto front. The effectiveness of the RMOBO-IU framework has been demonstrated on various new benchmark functions with promising results.

Future research will focus on several aspects: the SAA-MC-based kernel expectation still relies on sampling in the input space, which restricts its usage for a higher number of input dimensions. A more scalable approach is needed. Moreover, Bayesian versions of other robustness measures will also be investigated.

Acknowledgements This research received funding from the Flemish Government (AI Research Program) and Chinese Scholarship Council under grant number 201906290032.

Data availability Statement The code for reproducing the experiments for the current study are available from the corresponding author on reasonable request.

References

Abbas AT, Aly M, Hamza K (2016) Multiobjective optimization under uncertainty in advanced abrasive machining processes via a fuzzy-evolutionary approach. Journal of Manufacturing Science and Engineering 138(7)

Balandat M, Karrer B, Jiang DR, Daulton S, Letham B, Wilson AG, Bakshy E (2019) Botorch: A framework for efficient monte-carlo bayesian optimization. arXiv preprint arXiv:191006403

Beland JJ, Nair PB (2017) Bayesian optimization under uncertainty. In: NIPS BayesOpt 2017 workshop

Berkeley J, Moss HB, Artemev A, Pascual-Diaz S, Granta U, Stojic H, Couckuyt I, Qing J, Satrio L, Picheny V (2021) Trieste. URL `https://github.com/secondmind-labs/trieste`

Couckuyt I, Deschrijver D, Dhaene T (2014) Fast calculation of multiobjective probability of improvement and expected improvement criteria for pareto optimization. Journal of Global Optimization 60(3):575–594

Dallaire P, Besse C, Chaib-Draa B (2009) Learning gaussian process models from uncertain data. In: International Conference on Neural Information Processing, Springer, pp 433–440

Daulton S, Balandat M, Bakshy E (2020) Differentiable expected hypervolume improvement for parallel multi-objective bayesian optimization. arXiv preprint arXiv:200605078

Daulton S, Balandat M, Bakshy E (2021) Parallel bayesian optimization of multiple noisy objectives with expected hypervolume improvement. arXiv preprint arXiv:210508195

Deb K, Gupta H (2005) Searching for robust pareto-optimal solutions in multi-objective optimization. In: International conference on evolutionary multi-criterion optimization, Springer, pp 150–164

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE transactions on evolutionary computation 6(2):182–197

Fernández-Sánchez D, Garrido-Merchán EC, Hernández-Lobato D (2020) Improved max-value entropy search for multi-objective bayesian optimization with constraints. arXiv preprint arXiv:201101150

Fonseca CM, Fleming PJ (1995) Multiobjective genetic algorithms made easy: selection sharing and mating restriction. In: First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, IET, pp 45–52

Forrester A, Sobester A, Keane A (2008) Engineering design via surrogate modelling: a practical guide. John Wiley & Sons

Fröhlich LP, Klenske ED, Vinogradska J, Daniel C, Zeilinger MN (2020) Noisy-input entropy search for efficient robust bayesian optimization. arXiv preprint arXiv:200202820

Gramacy RB, Lee HKH (2010) Optimization under unknown constraints. 1004.4027

Higham NJ (1988) Computing a nearest symmetric positive semidefinite matrix. Linear algebra and its applications 103:103–118

Iwazaki S, Inatsu Y, Takeuchi I (2021) Mean-variance analysis in bayesian optimization under uncertainty. In: International Conference on Artificial Intelligence and Statistics, PMLR, pp 973–981

Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. Journal of Global optimization 13(4):455–492

Kleywegt AJ, Shapiro A, Homem-de Mello T (2002) The sample average approximation method for stochastic discrete optimization. SIAM Journal on Optimization 12(2):479–502

Li M, Azarm S, Boyars A (2005) A New Deterministic Approach Using Sensitivity Region Measures for Multi-Objective Robust and Feasibility Robust Design Optimization. Journal of Mechanical Design 128(4):874–883, DOI 10.1115/1.2202884, URL https://doi.org/10.1115/1.2202884, https://asmedigitalcollection.asme.org/mechanicaldesign/article-pdf/128/4/874/5923754/874_1.pdf

Papoulis A, Pillai S (2002) Probability, Random Variables, and Stochastic Processes. McGraw-Hill series in electrical engineering: Communications and signal processing, McGraw-Hill, URL https://books.google.be/books?id=g6eUoWOlcQMC

Picheny V, Wagner T, Ginsbourger D (2013) A benchmark of kriging-based infill criteria for noisy optimization. Structural and Multidisciplinary Optimization 48(3):607–626

Rasmussen CE (2003) Gaussian processes in machine learning. In: Summer school on machine learning, Springer, pp 63–71

Rivier M, Congedo PM (2018) Surrogate-assisted bounding-box approach applied to constrained multi-objective optimisation under uncertainty. PhD thesis, Inria Saclay Ile de France

Schutze O, Esquivel X, Lara A, Coello CAC (2012) Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization. IEEE Transactions on Evolutionary Computation 16(4):504–522

Shahriari B, Swersky K, Wang Z, Adams RP, De Freitas N (2015) Taking the human out of the loop: A review of bayesian optimization. Proceedings of the IEEE 104(1):148–175

Wilson JT, Hutter F, Deisenroth MP (2018) Maximizing acquisition functions for bayesian optimization. arXiv preprint arXiv:180510196

Xia B, Ren Z, Koh CS (2014) Utilizing kriging surrogate models for multi-objective robust optimization of electromagnetic devices. IEEE transactions on magnetics 50(2):693–696

Yang K, Emmerich M, Deutz A, Bäck T (2019) Efficient computation of expected hypervolume improvement using box decomposition algorithms. Journal of Global Optimization 75(1):3–34

Zhou Q, Jiang P, Huang X, Zhang F, Zhou T (2018) A multi-objective robust optimization approach based on gaussian process model. Structural and Multidisciplinary Optimization 57(1):213–233

A Synthetic Functions

We provide a detailed description of the synthetic functions that we have utilized for numerical benchmarking, with a math formulation in Table. 3. We note that the inverse of these synthetic functions is used to perform MOO for maximization.
VLMOP2 (Fonseca and Fleming, 1995) A bi-objective synthetic problem, where each objective function has only one global optima within the design space.
MDTP2 A modified version of Deb and Gupta (2005)'s test problem 2.
SinLinForrester A bi-objective problem with SineLiner (Fröhlich et al., 2020) function and Forrester function Forrester et al. (2008).
MDTP3 A modified version of Deb and Gupta (2005)'s test problem 3.
BraninGMM A bi-objective problem with Branin function (Picheny et al., 2013) and Gaussian Mixture Model (Fröhlich et al., 2020), the input uncertainty is taken from (Beland and Nair, 2017).

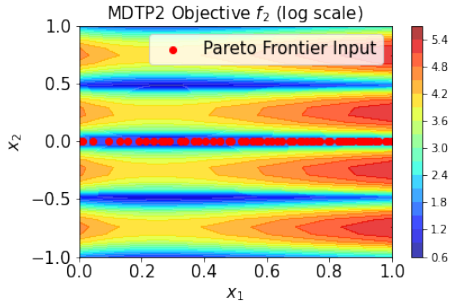
B Experiment Details

In this section we demonstrate the experimental details, more specifically, we demonstrate the final query points of RMOBO-IU on the synthetic problem. The samples that RMOBO-IU investigated is illustrated in Fig. 10, once the AL process has been activated, the pending data is not the same as query data and the difference has been noted with the arrows. The one dimensional SinLinForrester function is omitted for its simplicity. It can be observed that RMOBO-IU is searching for locating at the robust Pareto frontier. Meanwhile, the AL optimization helps to alleviate the duplication and boundary issue in all the synthetic problems.

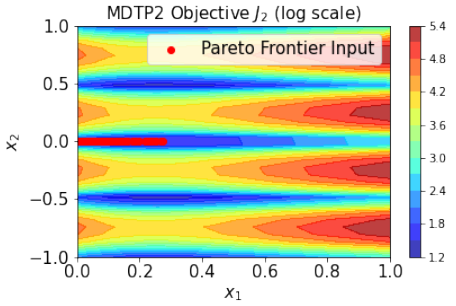
Table 3: Bi-objective benchmark functions settings

Function	**Design Space**	Function Expression
SinLinForrester	$[0,\ 1]$	$y_1 = sin(5\pi x^2) + 0.5x$ $y_2 = (6x-2)^2 sin(12x-4)$
VLMOP2	$[-2, 2]^2$	$y_1 = 1 - \exp(-\Sigma_{i=1}^{2}(x_i - \frac{1}{\sqrt{2}})^2)$ $y_2 = 1 - \exp(-\Sigma_{i=1}^{2}(x_i + \frac{1}{\sqrt{2}})^2)$
MDTP2	$[0, 1] \times [-1, 1]$	$y_1 = x_1$ $y_2 = (1 - x_1^2) + (10 + x_2^2 - 10cos(4\pi x_2))\cdot$ $(\frac{1}{0.2+x_1} + 10x_1^2)$
MDTP3	$[0, 1]^2$	$y_1 = x_1$ $y_2 = 1 - 0.9\ e^{(-\frac{x_2-0.8}{0.1})^2}$ $-1.3\ e^{(-\frac{x_2-0.3}{0.03})^2}$
BraninGMM	$[0, 1]^2$	$y_1 = \frac{1}{51.95}[(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6)^2$ $+(10 - \frac{10}{8\pi}cos(x_1)) - 44.81]$ $y_2 = \sum_{j=1}^{3} p(z=j)p(x\|z=j)$ where : $p(z=1) = 0.04\pi, x\|z=1 \sim \mathcal{N}([0.2, 0.2], 0.2^2\delta)$ $p(z=2) = 0.014\pi, x\|z=2 \sim \mathcal{N}([0.8, 0.2], 0.1^2\delta)$ $p(z=3) = 0.014\pi, x\|z=3 \sim \mathcal{N}([0.5, 0.7], 0.1^2\delta)$ where δ represents Kronecker delta.

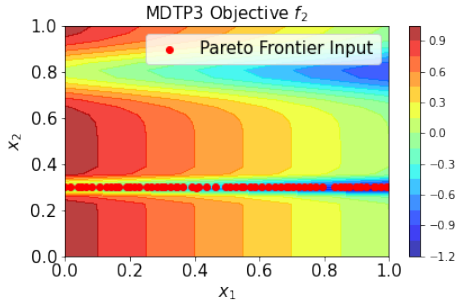
We illustrated part of the objective functions as well as their robust contour parts in Fig. 9, where the reference Pareto optimal points input are also illustrated in the figure.



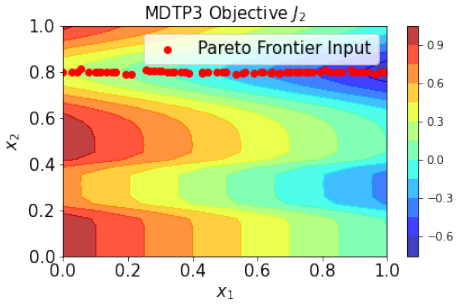
(a) MDTP2 Objective 2 original objective



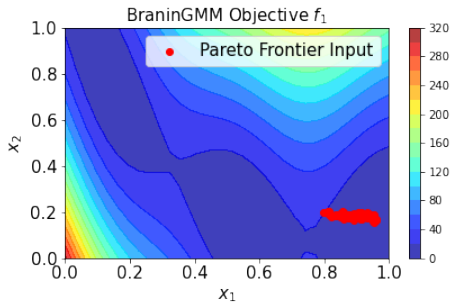
(b) MDTP2 Objective 2 robust objective



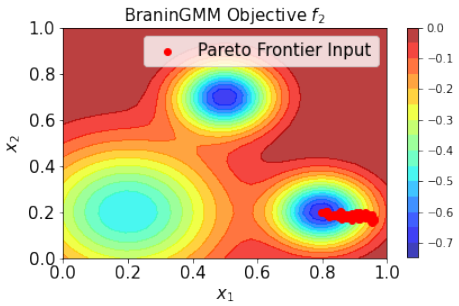
(c) MDTP3 Objective 2 original objective



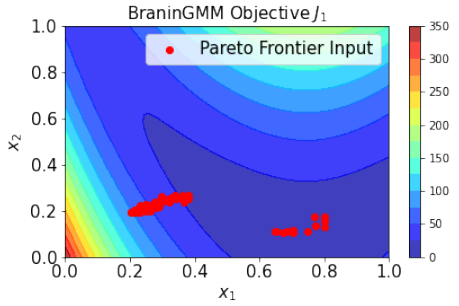
(d) MDTP3 Objective 2 robust objective



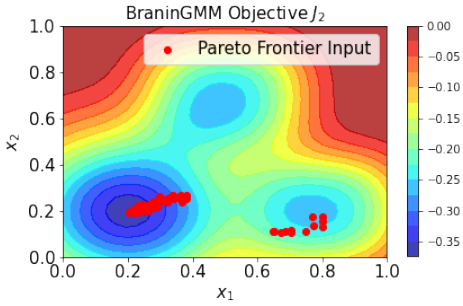
(e) BraninGMM Objective 1 original objective



(f) BraninGMM Objective 2 original objective

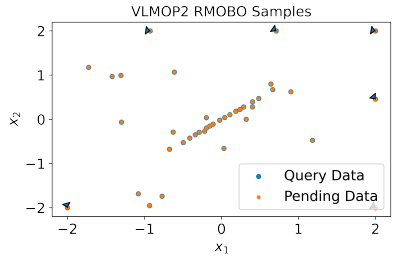


(g) BraninGMM Objective 2 robust objective

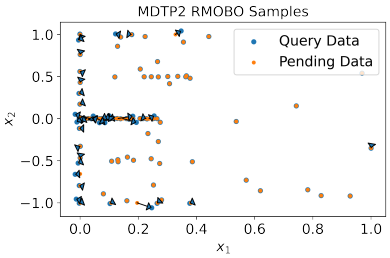


(h) BraninGMM Objective 2 robust objective

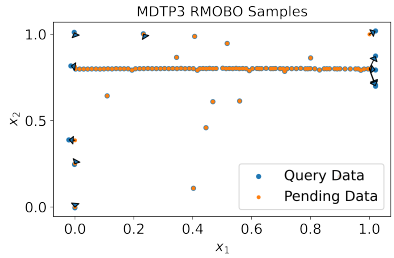
Fig. 9: Comparison of non-robust and robust objective functions, the inverse of which are used for maximization in numerical experiments. The corresponding Pareto frontier input is also illustrated in the figure, which has been obtained from NSGAII.



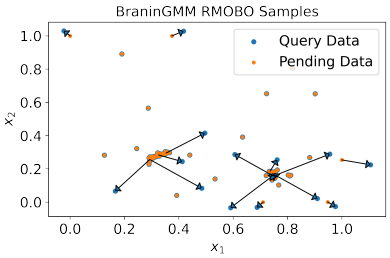
(a) VLMOP2 RMOBO-IU Input Samples



(b) MDTP2 RMOBO-IU Input Samples



(c) MDTP3 RMOBO-IU Input Samples



(d) BraninGMM RMOBO-IU Input Samples

Fig. 10: Illustration of RMOBO-IU sample in input space (based on EHVI experiment).