Surrogate-Assisted Bounding-Box approach applied to constrained multi-objective optimisation under uncertainty

Mickael Rivier, Pietro Marco Congedo

Surrogate-Assisted Bounding-Box Approach Applied to Constrained Multi-Objective Optimisation Under Uncertainty

M. Rivier[a,b], P.M. Congedo[a]

[a] *Inria, Centre de Mathématiques Appliquées, Ecole Polytechnique, IPP, Route de Saclay, 91128 Palaiseau, France*

[b] *ArianeGroup, Le Haillan, 33185 Cedex, France*

Abstract

This paper is devoted to tackling constrained multi-objective optimisation under uncertainty problems. A Surrogate-Assisted Bounding-Box approach (SABBa) is formulated here to deal with approximated robustness and reliability measures, which can be adaptively refined.

A Bounding-Box is defined as a multi-dimensional product of intervals, centred on the estimated objectives and constraints, that contains the true underlying values. The accuracy of these estimations can be tuned throughout the optimisation so as to reach high levels only on promising designs, which allows quick convergence toward the optimal area. In SABBa, this approach is supplemented with a Surrogate-Assisting (SA) strategy, which permits to further reduce the overall computational cost. The adaptive refinement within the Bounding-Box approach is guided by the computation of the Pareto Optimal Probability (POP) of each box.

We first assess the proposed method on several analytical uncertainty-based optimisation test-cases with respect to an *a priori* metamodel approach in terms of a probabilistic modified Hausdorff distance to the true Pareto optimal set. The method is then applied to three engineering applications: the design of two-bar truss in structural mechanics, the shape optimisation of an Organic Rankine Cycle turbine blade and the design of a thermal protection system for atmospheric reentry.

Keywords:
Multi-objective optimisation, Uncertainty-based optimisation, Error bounding boxes, Imprecise Pareto front, Surrogate-Assisting strategy

1. Introduction

Optimisation under uncertainty is of particular interest for companies nowadays, since robustness, reliability and cost optimality are critical factors for assessing the quality of a product and the efficiency of a company. In this context, two main areas are of primary interest: Robust Optimisation (RO) and Reliability-Based Design Optimisation (RBDO). Although there exist several interpretations for these concepts, very often in an optimization problem under uncertainty, statistics-based metrics appear in the objective function in the

case of robustness, whereas they appear in the constraint for reliability. One of the central issues in conducting such analysis is the computational cost of attaining a prescribed level of accuracy for these statistics. Most of the time, these optimisations are performed in coupling with numerical simulations, where run times can reach magnitudes of weeks or months. This is notably the case in the context of Direct Numerical Simulation (DNS) or Large Eddy Simulation (LES), as depicted in [1] despite running on a 4 gigaflops supercomputer. Such a computational burden prohibits the use of direct simulation-based methods such as nested Monte Carlo Simulations (MCS).

RBDO formulations usually feature a failure probability or Reliability Index (RI) within the optimisation constraint. The later is classically used in the First-Order Reliability Method (FORM), and the Performance Measure Approach (PMA) [2, 3]. Recently, history-based failure probability estimation have been proposed using quasi-optimal Importance Sampling (IS) distributions [4, 5]. Some approaches rely on local approximations of the failure probability and its gradient to perform line search optimisation [6, 7]. More generally, the idea of using a surrogate model to tackle the computational cost limitation has been extensively exploited with Neural Networks [8], Polynomial Chaos Expansion (PCE) [9, 10] and PCE-Kriging metamodels [11]. Kriging-based techniques have notably been compared with FORM in [12] and allow for a natural computation of the RI in [13]. Recently, adaptive kriging approaches are among the most successful techniques. The Stepwise Uncertainty Reduction (SUR) strategy [14] and Efficient Global Reliability Analysis (EGRA) [15, 16] allow for very parsimonious reliability estimations. Support Vector Regression (SVR) models can also be employed similarly, as proposed in [17] with parameterised Matern smoothness.

As for Robust Optimisation, several methods have been proposed, with and without surrogate models. Examples of the latter for mean performance optimisation make use of the Simultaneous Perturbation Stochastic Approximation (SPSA) [18, 19] with Stochastic Subset Optimisation (SSO) and Importance Sampling (IS) respectively. As shown in [20], surrogate models like PCE [9], and mostly kriging [21, 22, 23, 24] provide very efficient techniques for solving uncertainty-based optimisation. The properties of kriging models, also called Gaussian Processes (GP), are exploited in [25] to analytically perform mean performance optimisation and in [26] by taking heterogeneous noises into account. They are also used in the context of Taguchi optimisation through the construction of a dual metamodel in [27] and of a gradient-enhanced kriging for Taylor-based measure estimations in [28]. Finally, in [29] conditional simulations are exploited in the context of multi-objective optimisation to sample possible Pareto optima.

In the vast majority of the above methods, statistics computations are prone to some error, either coming from the numerical interpolation (*e.g.* Monte Carlo) scheme or the surrogate model approximation. Both these sources are studied and estimated in [30], and the surrogate-based estimation error is analytically tracked in [25] in the context of mean statistical measures and GP models with Gaussian kernels. In this work, we propose estimating these computation errors with a simple interval assumption and incorporating them into the optimisation strategy. This choice is motivated by the fact that most of the time, only bounds are available to quantify the error of an uncertainty propagation process [31, 32].

To this extent, the Bounding-Box approach [33, 34, 35] gives an efficient framework for performing multi-objective optimisation on cost functions affected with interval uncertainty. A histogram-based approach has even been proposed in [36] to extend Bounding-Boxes to non-parametric distribution shapes. An adaptive strategy has been proposed in [37] to tackle refinable interval uncertainty in the objective functions with tunable accuracy. Parsimony is notably obtained thanks to a Surrogate-Assisting strategy, and the overall strategy is referred to as the Surrogate-Assisted Bounding-Box approach (SABBa) framework.

The main contribution of the present paper is twofold. It first proposes suitable surrogate-based error estimations of the statistical measures to be used in SABBa for Robust Optimisation. It also extends SABBa to constrained problems with interval uncertainty on the constraint functions, allowing to solve both robust and reliability-based optimisation problems in a common framework.

We depict a general measure-based optimisation under uncertainty problem in Section 2, with some classical robustness and reliability measures. Then, we introduce the Bounding-Boxes and Pareto Optimal Probability (POP) in Section 3. We notably extend the existing concepts from [33, 34, 35, 37] to constrained optimisation problems. The global algorithm is depicted in Section 4 alongside computational insights for measure estimation. Three variants of SABBa are quantitatively compared in Section 5 on analytical test-case and the most efficient approach is applied on two engineering applications in Section 6. Some conclusions and perspectives are drawn in Section 7.

2. Problem formulation

The objective of this section is to formulate a general expression for an Optimisation under Uncertainty problem including both Robust and Reliability-Based Optimisation (RO and RBDO).

Without loss of generality, a deterministic constrained multi-objective optimisation problem can be described as the following minimisation problem:

$$\begin{aligned} \text{minimise:} \quad & \boldsymbol{f}(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{1}$$

where the m_1 objective functions are collected in a vector $\boldsymbol{f} \in \mathbb{R}^{m_1}$, the m_2 constraint functions in $\boldsymbol{g} \in \mathbb{R}^{m_2}$ and $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n$ are the n design variables. Note that $\boldsymbol{g}(\boldsymbol{x}) \leq \boldsymbol{0}$ stands for $\forall i, g_i(\boldsymbol{x}) \leq 0$.

The computation of $\boldsymbol{f}$ and $\boldsymbol{g}$ can also depend on a multitude of uncontrollable parameters (*e.g.* environmental, material, geometrical, ...), here denoted by $\boldsymbol{\xi} \in \Xi$. The most straightforward setting to deal with these uncertain parameters is to consider them aleatoric. In this context, $\boldsymbol{\xi}$, $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi})$ are random variables and Equation (1) is not suitable anymore to represent the problem.

Here, we tackle optimisation under uncertainty problems in a measure-based setting, where specific statistics of the random outputs $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi})$ are optimised. We denote

these new objectives and constraints as robustness and reliability measures, written $\boldsymbol{\rho_f}$ and $\boldsymbol{\rho_g}$ respectively. An uncertainty-based optimisation problem can then be formulated as follows:

$$\begin{aligned} \text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}), \\ \text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) \leq \mathbf{0}, \\ \text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}, \end{aligned}$$

with $\boldsymbol{\rho_f} \in \mathbb{R}^{m_1}$ and $\boldsymbol{\rho_g} \in \mathbb{R}^{m_2}$.

Remark. For conciseness, $\boldsymbol{f}$ and $\boldsymbol{g}$ are gathered in a vector $\boldsymbol{q} \in \mathbb{R}^{m_1+m_2}$. Namely,

$$\forall (\boldsymbol{x}, \boldsymbol{\xi}) \in \mathcal{X} \times \Xi, \ \boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi}) = \begin{pmatrix} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}) \\ \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi}) \end{pmatrix}. \tag{2}$$

Similarly, $\boldsymbol{\rho_f}$ and $\boldsymbol{\rho_g}$ are gathered in a single vector $\boldsymbol{\rho} \in \mathbb{R}^{m_1+m_2}$,

$$\forall \boldsymbol{x} \in \mathcal{X}, \ \boldsymbol{\rho}(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{\rho_f}(\boldsymbol{x}) \\ \boldsymbol{\rho_g}(\boldsymbol{x}) \end{pmatrix}. \tag{3}$$

Many formulations are possible for $\boldsymbol{\rho}$. The following are tackled in this paper:

$$\begin{aligned} \text{Expectation:} \quad & \boldsymbol{\rho}(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})], \\ \text{Variance:} \quad & \boldsymbol{\rho}(\boldsymbol{x}) = \mathbb{V}_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})], \\ \text{Worst case:} \quad & \boldsymbol{\rho}(\boldsymbol{x}) = \max_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})], \\ \text{Quantile:} \quad & \boldsymbol{\rho}(\boldsymbol{x}) = \mathrm{q}^{p}_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})], \ p \in [0, 1]. \end{aligned}$$

Note that any other measure could also be treated similarly in the following as we choose to approximate statistical measures through Monte Carlo Simulations (MCS) on surrogate models. We consider the unitary evaluation of $\boldsymbol{q}$ to be expensive, thus a compromise must be chosen between the accuracy and the computational cost associated with the evaluation of $\boldsymbol{\rho}$. This compromise is guided here by the number of evaluations of $\boldsymbol{q}$, which are the training data permitting to construct the surrogate model for MCS numerical integration.

The uncertainty-based optimisation problem hence reduces to a constrained multi-objective optimisation problem with tunable accuracy computations for all objectives and constraints. The concept of tunable accuracy in the unconstrained case has been introduced in [37] for multi-objective optimisation problems (≤ 3 objectives). Next section illustrates the concept of Bounding-Boxes and their applicability to constrained problems.

3. Bounding-Boxes

In this section, we first recall the definition of Bounding-Boxes. Their extension to constrained optimisation problems is then illustrated in Section 3.1.2. Finally, the concept of Pareto Optimal Probability is presented in Section 3.2 alongside computational insights.

3.1. Bounding-Box definition

A m-dimensional box is defined as follows, with $\boldsymbol{a} \in \mathbb{R}^m$ the center and $\boldsymbol{r} \in \mathbb{R}^m_+$ the positive half-width vector:

$$\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) = \left\{ \boldsymbol{b} \in \mathbb{R}^m \mid \boldsymbol{b} \in [\boldsymbol{a} - \boldsymbol{r}, \boldsymbol{a} + \boldsymbol{r}] \right\} \in \wp(\mathbb{R}^m),$$

where $\wp(\mathbb{R}^m)$ is the power set of $\mathbb{R}^m$.

The Bounding-Box approach consists in approximating an unknown value, here statistical measures $\boldsymbol{\rho}$, by a conservative box containing the true values. More precisely, an approximation $\widetilde{\boldsymbol{\rho}}$ of the unknown value $\boldsymbol{\rho}$ is considered and the exact approximation error $\boldsymbol{\varepsilon} = \boldsymbol{\rho} - \widetilde{\boldsymbol{\rho}}$ is conservatively approximated by $\overline{\boldsymbol{\varepsilon}} \geq |\boldsymbol{\varepsilon}|$. Hence it comes that $\boldsymbol{\rho} \in \mathcal{B}(\widetilde{\boldsymbol{\rho}}, \overline{\boldsymbol{\varepsilon}})$ (see Fig. 1).

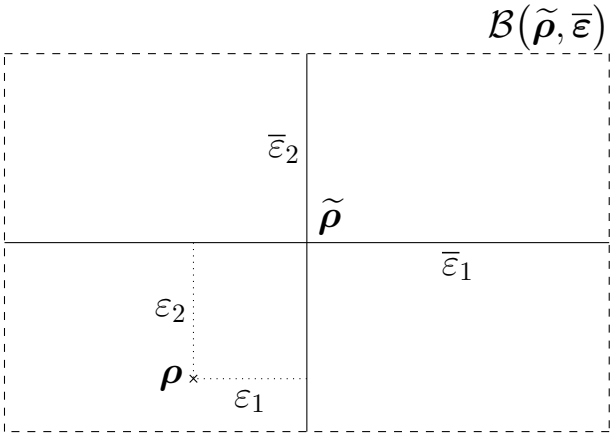


Figure 1: Bounding-Box approximation

3.1.1. Boxed Pareto Dominance

To perform multi-objective optimisation, the Pareto dominance rules are usually exploited to assess dominance ($\succ$), strict dominance ($\succcurlyeq$) or indifference ($\sim$) between designs. In the Bounding-Box context, for comparing two boxes $\big(\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}), \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}')\big) \in \wp(\mathbb{R}^m)^2$, the *Boxed Pareto dominance* in the multi-minimisation case is defined as follows:

- $\mathcal{B}(\boldsymbol{a}, \boldsymbol{r})$ dominates $\mathcal{B}(\boldsymbol{b}, \boldsymbol{r}')$ in the *Boxed Pareto dominance* sense:

$$\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \underset{\mathcal{B}}{\succ} \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \iff \forall j \in [\![1, m]\!],\ a_j + r_j \leq b_j - r'_j \quad \text{and} \quad \exists j \in [\![1, m]\!],\ a_j + r_j < b_j - r'_j;$$

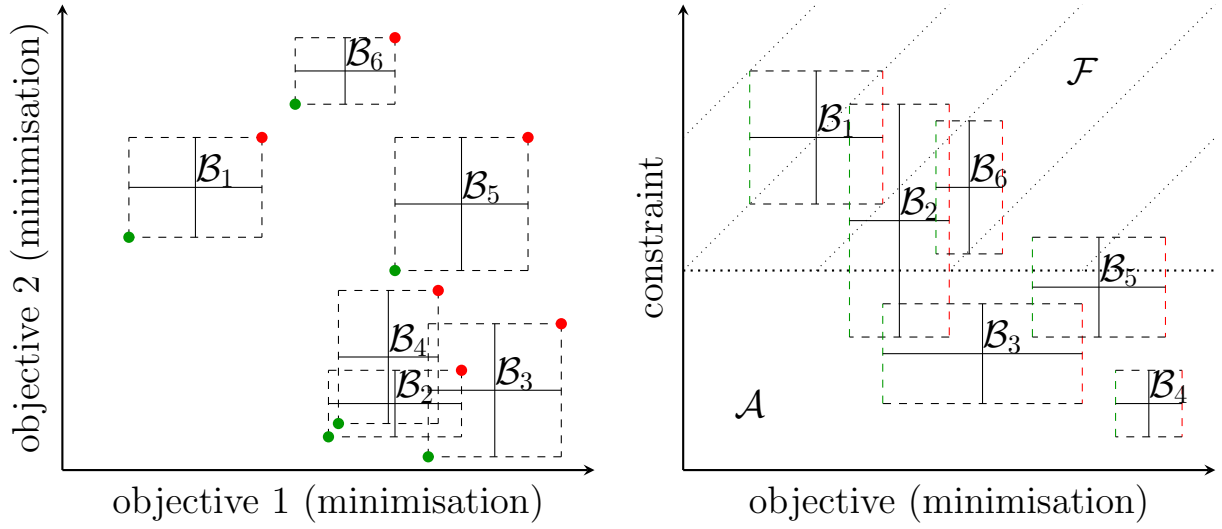
- $\mathcal{B}(\boldsymbol{a}, \boldsymbol{r})$ strictly dominates $\mathcal{B}(\boldsymbol{b}, \boldsymbol{r}')$ in the *Boxed Pareto dominance* sense:

$$\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \underset{\mathcal{B}}{\succcurlyeq} \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \iff \forall j \in [\![1, m]\!],\ a_j + r_j < b_j - r'_j;$$

- $\mathcal{B}(\boldsymbol{a}, \boldsymbol{r})$ and $\mathcal{B}(\boldsymbol{b}, \boldsymbol{r}')$ are indifferent in the *Boxed Pareto dominance* sense:

$$\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \underset{\mathcal{B}}{\sim} \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \iff \mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \underset{\mathcal{B}}{\not\succ} \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \ \text{ and } \ \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \underset{\mathcal{B}}{\not\succ} \mathcal{B}(\boldsymbol{a}, \boldsymbol{r}).$$

Intuitively, $\mathcal{B}(\boldsymbol{a}, \boldsymbol{r})$ dominates $\mathcal{B}(\boldsymbol{b}, \boldsymbol{r}')$ if the worst outcome of $\mathcal{B}(\boldsymbol{a}, \boldsymbol{r})$ dominates in the classical sense the best outcome of $\mathcal{B}(\boldsymbol{b}, \boldsymbol{r}')$. An example is given hereafter in Fig. 2(a). In the case of bi-minimisation, $\mathcal{B}_1 \underset{\mathcal{B}}{\succ\!\succ} \mathcal{B}_6$, and the other boxes are indifferent with each other. Hence, among these six boxes, only $\mathcal{B}_6$ is dominated.



(a) (b)

Figure 2: (a) *Boxed Pareto Dominance*, (b) *Constrained Boxed Pareto Dominance*. Comparison of six boxes, best and worst outcomes in green and red, respectively.

3.1.2. Constrained Boxed Pareto dominance

The *Boxed Pareto dominance* allows comparing boxes in a multi-objective optimisation context. For dealing with robust objectives as well as reliability-based constraints, the Pareto dominance should take into account constraint violation. To this extent, a *Constrained Pareto dominance* $\succ_c$ is proposed, where again $\succ$ refers to the classical Pareto dominance.

In the multi-objective constrained problem, let us denote by $\boldsymbol{a_f}$ the objective values and $\boldsymbol{a_g}$ the constraint values of design $\boldsymbol{a}$. Without loss of generality, these m_2 constraints can be considered of the form:

$$a_{\boldsymbol{g}} \leq 0.$$

The admissible and failure sets are then defined as follows:

$$\mathcal{A} = \left\{\boldsymbol{a} \in \mathbb{R}^m \mid a_{\boldsymbol{g}} \leq 0\right\},$$
$$\mathcal{F} = \left\{\boldsymbol{a} \in \mathbb{R}^m \mid a_{\boldsymbol{g}} \nleq 0\right\}.$$

Contrarily to the classical Pareto dominance, the *Constrained Pareto dominance* compares boxes from the coupled objective/constraint spaces. We propose to use the following

dominance rule in a classical constrained environment:

$$\begin{aligned}
\boldsymbol{a} \succ_c \boldsymbol{b} &\iff \boldsymbol{a} \in \mathcal{A} \text{ and } \boldsymbol{a_f} \succ \boldsymbol{b_f} \quad \text{or} \quad \boldsymbol{b} \in \mathcal{F} \\
\boldsymbol{a} \succcurlyeq_c \boldsymbol{b} &\iff \boldsymbol{a} \in \mathcal{A} \text{ and } \boldsymbol{a_f} \succcurlyeq \boldsymbol{b_f} \quad \text{or} \quad \boldsymbol{b} \in \mathcal{F} \\
\boldsymbol{a} \sim_c \boldsymbol{b} &\iff \boldsymbol{a} \not\succ_c \boldsymbol{b} \text{ and } \boldsymbol{b} \not\succ_c \boldsymbol{a}.
\end{aligned}$$

Note that in this case, $\mathcal{A} = \mathcal{F}^{\complement}$, where the superscript $\cdot^{\complement}$ stands for complement. This relation would not hold in the context of boxed objectives and constraints, as shown in the following.

When comparing boxes of $\mathbb{R}^m$, the admissible $\mathcal{A}_{\mathcal{B}}$ and failure $\mathcal{F}_{\mathcal{B}}$ sets contain the boxes that lie entirely in $\mathcal{A}$ and $\mathcal{F}$ respectively:

$$\begin{aligned}
\mathcal{A}_{\mathcal{B}} &= \left\{\mathcal{B} \in \wp(\mathbb{R}^m) \mid \forall \boldsymbol{a} \in \mathcal{B},\ \boldsymbol{a} \in \mathcal{A}\right\} \\
\mathcal{F}_{\mathcal{B}} &= \left\{\mathcal{B} \in \wp(\mathbb{R}^m) \mid \forall \boldsymbol{a} \in \mathcal{B},\ \boldsymbol{a} \in \mathcal{F}\right\}
\end{aligned}$$

Hence, in the general case, $\mathcal{A}_{\mathcal{B}} \neq \mathcal{F}_{\mathcal{B}}^{\complement}$. A more computationally friendly formulation is the following, using again the best and worst outcomes:

$$\begin{aligned}
\mathcal{A}_{\mathcal{B}} &= \left\{\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \in \wp(\mathbb{R}^m) \mid \boldsymbol{a} + \boldsymbol{r} \in \mathcal{A}\right\}, \\
\mathcal{F}_{\mathcal{B}} &= \left\{\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \in \wp(\mathbb{R}^m) \mid \boldsymbol{a} - \boldsymbol{r} \in \mathcal{F}\right\}.
\end{aligned}$$

The *Boxed Constrained Pareto dominance* is then defined as follows:

$$\begin{aligned}
\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \underset{\mathcal{B}}{\succ_c} \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') &\iff \mathcal{B}(\boldsymbol{a_f}, \boldsymbol{r_f}) \underset{\mathcal{B}}{\succ} \mathcal{B}(\boldsymbol{b_f}, \boldsymbol{r'_f}) \text{ and } \mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \in \mathcal{A}_{\mathcal{B}} \quad \text{or} \\
&\qquad \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \in \mathcal{F}_{\mathcal{B}}, \\
\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \underset{\mathcal{B}}{\succcurlyeq_c} \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') &\iff \mathcal{B}(\boldsymbol{a_f}, \boldsymbol{r_f}) \underset{\mathcal{B}}{\succcurlyeq} \mathcal{B}(\boldsymbol{b_f}, \boldsymbol{r'_f}) \text{ and } \mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \in \mathcal{A}_{\mathcal{B}} \quad \text{or} \\
&\qquad \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \in \mathcal{F}_{\mathcal{B}}, \\
\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \underset{\mathcal{B}}{\sim_c} \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') &\iff \mathcal{B}(\boldsymbol{a}, \boldsymbol{r}) \underset{\mathcal{B}}{\not\succ_c} \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \text{ and } \mathcal{B}(\boldsymbol{b}, \boldsymbol{r}') \underset{\mathcal{B}}{\not\succ_c} \mathcal{B}(\boldsymbol{a}, \boldsymbol{r}).
\end{aligned}$$

With this Pareto dominance rule, two boxes in $\mathcal{F}_{\mathcal{B}}$ will mutually dominate each other. Such a behaviour is desired to make sure that any box of the boxed failure set is completely dominated, regardless of the performance of the other boxes. This will be exploited in the next section, where the Pareto Optimal Probability of each box is computed.

In the general case, for a box to be dominated, the non-constrained *Boxed Pareto Dominance* must be fulfilled in the objective dimensions and the dominant box must lie entirely in the admissible set. An example is given in Fig. 2(b) with a constrained mono-objective minimisation. Here, $\{\mathcal{B}_1, \mathcal{B}_6\} \subset \mathcal{F}_{\mathcal{B}}$, $\{\mathcal{B}_3, \mathcal{B}_4\} \subset \mathcal{A}_{\mathcal{B}}$ and $\{\mathcal{B}_2, \mathcal{B}_5\} \subset \mathcal{F}_{\mathcal{B}}^{\complement} \cap \mathcal{A}_{\mathcal{B}}^{\complement}$. In terms of dominance, $\mathcal{B}_1$ and $\mathcal{B}_6$ are dominated by all other boxes for being entirely in the failure set. $\mathcal{B}_4$ is also dominated by $\mathcal{B}_3$, which is entirely in the admissible set.

3.2. Pareto Optimal Probability

In the above, dominance rules allow to discriminate dominated boxes from non-dominated ones among a finite set of boxes. This rule is exploited in [35, 37] to refine only non-dominated

boxes. To discriminate in a more rigorous way between non-dominated boxes, we propose to compute for each Bounding-Box its probability of being non-dominated. Such a computation is based on the assumption that the true robustness and reliability measures $\boldsymbol{\rho}$ in Figure 1 can be modelled with an aleatory variable following a uniform distribution within the Bounding Box $\mathcal{B}(\widetilde{\boldsymbol{\rho}}^l, \overline{\boldsymbol{\varepsilon}}^l)$. The problem relapses to computing the Pareto Optimal Probability of aleatory variables with known uniform distributions.

With $\boldsymbol{\mathcal{B}}_{all}$ a given set of boxes $\left\{\mathcal{B}_i\right\}_i$, the exact POP computation follows the formula below:

$$POP_{true}(\mathcal{B}_i) = \mathbb{P}_{\{\boldsymbol{Z}_k\}_k}\left[\bigcap_{\substack{\mathcal{B}_j \in \boldsymbol{\mathcal{B}}_{all} \\ j \neq i}} \boldsymbol{Z}_j \not\prec_c \boldsymbol{Z}_i\right], \tag{4}$$

where for all k, $\boldsymbol{Z}_k \sim \mathcal{U}\big(\mathcal{B}_k\big)$. As assumed earlier, the realisations $\boldsymbol{Z}_k$ are drawn uniformly within the set $\mathcal{B}_k$.

Note however that this POP computation yields a combinatorial complexity that profoundly limits its calculation in closed form and would require Monte-Carlo approximation. Moreover, clustered boxes tend to dominate each other and result in low POP values, even when they are close to the Pareto front, which can be counter-intuitive. For this reason, we propose two approximations of the POP, that will be quantitatively compared.

Ref. [33] proposes the use of a box probabilistic ranking as the fitness function for ESPEA (Estimated Strengh Pareto Evolutionary Algorithm). The score of a given box is computed by averaging the one-to-one domination probability with respect to all other boxes. Since we compute each probability between only two boxes, the computational burden is very low. However, clustered boxes still yield unintuitive scores.

In this work, we propose a new metric denoted as POP_{min}. It is defined as follows, with $\boldsymbol{\mathcal{B}}_{nd}$ the set of non-dominated designs (using the *Constrained Boxed Pareto dominance*):

$$POP_{min}(\mathcal{B}_i) = \min_{\substack{\mathcal{B}_j \in \boldsymbol{\mathcal{B}}_{nd} \\ j \neq i}} \Big(\mathbb{P}_{\boldsymbol{Z}_j, \boldsymbol{Z}_i}\big[\boldsymbol{Z}_j \not\prec_c \boldsymbol{Z}_i\big]\Big). \tag{5}$$

This metric yields a good relative ranking between boxes and is efficiently computable. It also shows a more intuitive behaviour when dealing with clustered boxes. POP_{min} and POP_{true} metrics are quantitatively compared in Table 1 on the two small examples depicted in Figure 2, and show very similar scoring.

Thus, for its simplicity, interpretability and very low computational burden, POP_{min} will be used in the following.

4. SABBa framework

The objective of SABBa is to lower the computational cost of estimating the Pareto front by adaptively refining the approximations $\widetilde{\boldsymbol{\rho}}$ only at the most promising designs. The unconstrained version of this framework is developed in [37] and relies on the coupling between the Bounding-Box approach and a Surrogate-Assisting (SA) model built directly on the objective values. SABBa can be coupled to generic optimisation algorithm and any

	Figure 2(a) example		Figure 2(b) example	
	POP_{true}	POP_{min}	POP_{true}	POP_{min}
$\mathcal{B}_1$	1.0	1.0	0.0	0.0
$\mathcal{B}_2$	0.902	0.912	0.254	0.254
$\mathcal{B}_3$	0.39	0.419	0.724	0.746
$\mathcal{B}_4$	0.621	0.622	0.0	0.0
$\mathcal{B}_5$	0.017	0.07	0.022	0.031
$\mathcal{B}_6$	0.0	0.0	0.0	0.0

Table 1: POPs comparison on both examples from Figure 2

surrogate model can be employed for the SA strategy. This feature allows the approach to be readily applicable and to benefit from new optimisation or metamodelling techniques.

In this paper, we make use of the NOMAD [38] optimisation technique for its reliable management of multiple objectives and constraints within a derivative-free framework (version 3.6.2). All surrogate models are Gaussian Processes, constructed using the python Gaussian Process package GPy [39] (version 1.8.5).

In the following, we provide the global algorithm and necessary numerical ingredients for the implementation of SABBa. Specifically, Section 4.1 presents the structure and algorithm of the framework in details, Section 4.2 deals with the computation and refinement of the Bounding-Boxes and Section 4.3 proposes a quality indicator for quantitative comparison on analytical test-cases.

4.1. Algorithm

The framework follows the structure depicted in Figure 3. The optimiser manages the design space exploration with the aim of covering the whole Pareto front.

We illustrate now the typical sequence of operations within an iteration of the algorithm. Let us assume to have already computed measures $\widetilde{\boldsymbol{\rho}}$ at a set $\mathcal{X}_c$ of design so far in the optimisation process. The current SA model $\boldsymbol{\rho}_{SA}$ is constructed on these $\widetilde{\boldsymbol{\rho}}$ estimations. At each optimisation iteration, and for each new design $\boldsymbol{x}$:

- If the chosen SA model yields accurate prediction, *i.e.* $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1$, the SA-based measures estimations $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ are returned to the optimiser. Gaussian process-based computations of $\boldsymbol{\rho}_{SA}$ and $\overline{\boldsymbol{\varepsilon}}_{SA}$ are proposed in Section 4.2.2.

- Else, measures must be estimated. Design $\boldsymbol{x}$ is thus added to the set $\mathcal{X}_c$. Outputs $\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})$ are computed for some values of $\boldsymbol{\xi}$ in order to build the surrogate model $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$ on which the robustness and reliability measures $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ can be estimated for free. The associated accuracy mainly depends on the number of $\boldsymbol{\xi}$ training samples. The Gaussian Process surrogate models also permits to compute an approximation $\overline{\boldsymbol{\varepsilon}}(\boldsymbol{x})$ of the estimation error. Accuracy can then be improved by increasing the size of the training data $\left\{\boldsymbol{\xi}_i, \boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi}_i)\right\}$ on which the surrogate $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$ is built.

In the case of a set of designs $\mathcal{X}_{new}$ estimated simultaneously, gathered, such as for the initial Design of Experiments (DoE) or for a generation in evolutionary algorithms, we propose to guide the refinements with the Pareto Optimal Probability (POP) of the estimated boxes, in decreasing order. Intuition behind the choice of decreasing order is that the most promising boxes, once highly refined, might dominate most of the remaining boxes, thus rendering their refinement unnecessary. This refinement is performed until each box of $\mathcal{X}_{new}$ is either dominated, or has reached an estimated error $\overline{\boldsymbol{\varepsilon}}$ below a user-defined threshold $\boldsymbol{s}_2$. POP computation is presented in section 3.2.

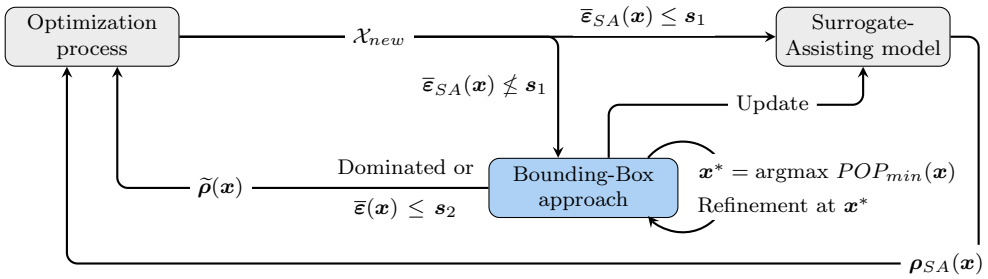


Figure 3: Structure of the SABBa framework

The strategy presented above and pictured in Figure 3 is made explicit in Algorithm 1. The tunable parameters are listed below and the Bounding-Box computational details are given in the following section.

Algorithm 1 Algorithm overview

Loop over values of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$
Initialise $\mathcal{X}_c$ empty
while Optimisation running **do**
    Read new designs $\mathcal{X}_{new}$
    for each $\boldsymbol{x} \in \mathcal{X}_{new}$ **do**
        if $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1$ **then**
            Return $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ to the optimiser
        else
            Add $\boldsymbol{x}$ in $\mathcal{X}_c$
            Compute a first approximation $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ of $\boldsymbol{\rho}(\boldsymbol{x})$
        end if
    end for
    while $\mathcal{X}_r = \left\{\boldsymbol{x} \in \mathcal{X}_{\widetilde{\mathcal{P}}_{\mathcal{B}}} \cap \ \mathcal{X}_c \mid \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}) > \boldsymbol{s}_2\right\}$ is non-empty **do**
        Find $\boldsymbol{x}^* = \underset{\boldsymbol{x} \in \mathcal{X}_r}{\text{argmax}} POP_{min}(\boldsymbol{x})$
        Compute $\boldsymbol{q}(\boldsymbol{x}^*, \boldsymbol{\xi})$ at some $\boldsymbol{\xi}$
        Update surrogate $\widehat{\boldsymbol{q}}(\boldsymbol{x}^*, \cdot)$
        Compute new estimates $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x}^*)$ and $\overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}^*)$
    end while
    Update $\boldsymbol{\rho}_{SA}$ and $\overline{\boldsymbol{\varepsilon}}_{SA}$ with the new $\widetilde{\boldsymbol{\rho}}$ and $\overline{\boldsymbol{\varepsilon}}$ values
    Return $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{X}_{new}$ to the optimiser
end while

$\mathcal{X}_{\widetilde{\mathcal{P}}_{\mathcal{B}}}$ contains all designs which associated boxes are non-dominated and $\mathcal{X}_c$ refers to the designs which measures have not been returned by the SA model. Hence, the boxes in $\mathcal{X}_r$ are all estimated without the SA model and both non-dominated and not refined up to $\boldsymbol{s}_2$.

Remark. We consider to loop over finer and finer values of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$. This allows more accurate intermediate results and slightly cheaper overall convergence. Practically, vectors of thresholds can be specified by the user.

The algorithm relies on a quite small set of parameters, namely:

- The predefined sequence of pair of thresholds $(\boldsymbol{s}_1, \boldsymbol{s}_2)$;
- The number of function evaluations N_{first} for the first approximations $\widetilde{\boldsymbol{\rho}}$ and N_{new} the number of additional evaluations for refinement;
- The number of designs N at each optimisation iteration and N_{init} at the first iteration. The optimiser may impose these parameters.

4.2. Bounding-Boxes computation and refinement

To compute the boxes, robustness and reliability measures must be estimated, and the associated error must be quantified. In particular, we focus in this paper on the following

measures: Expectation, variance, minimum, maximum and quantile. The values returned to the optimiser are denoted by $\widehat{\boldsymbol{\rho}}(\boldsymbol{x})$ and $\widehat{\boldsymbol{\varepsilon}}(\boldsymbol{x})$. As shown in Fig. 3 and Alg. 1, they can be computed from the $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$ surrogate with numerical integration or read from the SA model. These two computations are presented in details in the following sections.

Note that we make use of Gaussian Processes as surrogate models, giving both a predictive value and an associated predictive variance. Box widths are then computed based on this variance, relaxing the conservative assumption to a $\pm 3\sigma$ paradigm. Practically, with $\boldsymbol{\sigma}^2(\boldsymbol{z})$ the predictive variance of a GP surrogate model, an error can be computed as:

$$\overline{\boldsymbol{\varepsilon}}(\boldsymbol{z}) = 3\boldsymbol{\sigma}(\boldsymbol{z}).$$

Although this does not imply $\overline{\boldsymbol{\varepsilon}}(\boldsymbol{z}) \geq |\boldsymbol{\varepsilon}(\boldsymbol{z})|$, with $\boldsymbol{\varepsilon}$ the true error of the surrogate, the probability of dissatisfying the conservative assumption is very low.

Other surrogate modelling techniques could be employed, as long as a specific strategy is given for computing $\overline{\boldsymbol{\varepsilon}}$.

4.2.1. Computed boxes

For each design $\boldsymbol{x}$ where the SA model is not exploited, the boxes must be directly computed from samples of $\boldsymbol{q}$ in the uncertain space. This step requires: (i) constructing a surrogate model $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$ of the quantities of interest, (ii) estimating the statistical measures $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ and associated error $\overline{\boldsymbol{\varepsilon}}(\boldsymbol{x})$ and (iii) refining these estimations until the prescribed accuracy is reached.

Surrogate model. Here, the underlying functions $\boldsymbol{q}(\boldsymbol{x}, \cdot)$ are approximated by GP models. Two approaches are considered:

- *Separated Spaces:* At a given $\boldsymbol{x}$, a GP is constructed only in Ξ on a set of samples $\left\{\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi}_i)\right\}_i$. This GP directly corresponds to $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$. In this setting, a set of N_{first} evaluations must be computed for getting the first approximation $\widetilde{\boldsymbol{\rho}}$ at line 10 of Alg. 1.

- *Coupled Space:* Here, a GP is built in $\mathcal{X} \times \Xi$ using all previous samples $\left\{\boldsymbol{q}(\boldsymbol{x}_i, \boldsymbol{\xi}_i)\right\}_i$. From this approximation $\widehat{\boldsymbol{q}}$ of $\boldsymbol{q}$, the cut $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$ can be returned for measure estimation. Except at the very first loop, N_{first} can thus be set to 0.

Note that while the Separated Spaces (SS) approach builds a GP in a space of lower dimension ($Card(\Xi)$) than the Coupled Space (CS) approach ($Card(\mathcal{X} \times \Xi)$), thus easing the surrogate modelling task, it does not use surrounding samples in the design space and restarts from scratch at each new design point $\boldsymbol{x}$. These strategies will be compared on analytical test-cases in Section 5.

Measure and error estimation. Both SS and CS approaches allow having a predictive model $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$, that will be written $\widehat{\boldsymbol{q}}_{\boldsymbol{x}}$, and the associated predictive variance $\boldsymbol{\sigma}_{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$. The $\pm 3\sigma$ paradigm provides the box widths, denoted as $\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}} = 3\boldsymbol{\sigma}_{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$. For each output measure, indexed by k, the classical empirical estimators are exploited on the surrogate model $\widehat{\boldsymbol{q}}_x$ for

the computation of $\widetilde{\rho}_k(\boldsymbol{x})$, such as the sample mean $\frac{1}{N}\sum_i \widehat{\boldsymbol{q}}_x(\boldsymbol{\xi}_i)$ with *i.i.d.* samples $\boldsymbol{\xi}_i$. The associated errors $\overline{\varepsilon}_k(\boldsymbol{x})$ are then estimated using Eq. (6):

$$\overline{\varepsilon}_k(\boldsymbol{x}) = \begin{cases} \mathbb{E}_{\boldsymbol{\xi}}\big[\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})\big] & \text{for expectation} \\ \mathbb{E}_{\boldsymbol{\xi}}\big[\big(\overline{\varepsilon}_\mu(\boldsymbol{x}) + \overline{\varepsilon}_{\boldsymbol{q_x}}(\boldsymbol{\xi})\big)^2 + 2\big|\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\mu}}(\boldsymbol{x})\big|\big(\overline{\varepsilon}_\mu(\boldsymbol{x}) + \overline{\varepsilon}_{\boldsymbol{q_x}}(\boldsymbol{\xi})\big)\big] & \text{for variance} \\ \max\big(\big|\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|, \big|\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^+_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|\big) & \text{for minimum} \\ \max\big(\big|\widetilde{\boldsymbol{M}}(\boldsymbol{x}) - \max_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|, \big|\widetilde{\boldsymbol{M}}(\boldsymbol{x}) - \max_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^+_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|\big) & \text{for maximum} \\ \max\big(\big|\widetilde{\mathbf{q}}^p(\boldsymbol{x}) - \mathrm{q}^p_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|, \big|\widetilde{\mathbf{q}}^p(\boldsymbol{x}) - \mathrm{q}^p_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^+_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|\big) & \text{for quantile} \end{cases} \tag{6}$$

where $\widehat{\boldsymbol{q}}^+_{\boldsymbol{x}}(\boldsymbol{\xi}) = \widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})$ and $\widehat{\boldsymbol{q}}^-_{\boldsymbol{x}}(\boldsymbol{\xi}) = \widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})$. In practice, the quantiles and expected values above are approximated by means of Monte Carlo Sampling (MCS) on surrogate model at very low cost. A formal justification of these box sizes is given in Appendix B.

Refinement. We also provide guidance for refining the surrogate model $\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi})$ that permits low-cost MCS. While one could rely on a space-filling paradigm, we choose here to use GP-based refinement criteria. In practice, the proposed criterion should explore areas of high probability featuring significant uncertainty. In addition, this criterion should prioritise refinement of the least well-estimated measures. To this extent, we propose partial criteria associated with each statistical measure that shall then be combined within a weighted sum to obtain the global criterion. These partial criteria are given hereafter:

$$c_k(\boldsymbol{\xi}) = \begin{cases} \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})\phi(\boldsymbol{\xi}) & \text{for expectation} \\ \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})\phi(\boldsymbol{\xi}) & \text{for variance} \\ \left[\dfrac{\widetilde{\boldsymbol{m}} - \widehat{\boldsymbol{q}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})}{2\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})}\right]_+ & \text{for minimum} \\ \left[\dfrac{\widehat{\boldsymbol{q}}^+_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{M}}}{2\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})}\right]_+ & \text{for maximum} \\ \left[\dfrac{\widetilde{\mathbf{q}}^p - \widehat{\boldsymbol{q}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})}{2\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})}\right]_+ \left[\dfrac{\widehat{\boldsymbol{q}}^+_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\mathbf{q}}^p}{2\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q_x}}(\boldsymbol{\xi})}\right]_+ \phi(\boldsymbol{\xi}) & \text{for quantile} \end{cases} \tag{7}$$

with the same definition of $\widehat{\boldsymbol{q}}^+_{\boldsymbol{x}}(\boldsymbol{\xi})$ and $\widehat{\boldsymbol{q}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})$ as before and where $\big[\,\cdot\,\big]_+ = \max(0, \cdot)$. To promote areas of high probability, we choose here to multiply the criteria for the mean, variance and quantile measures by the input Probability Density Function (PDF) $\phi(\boldsymbol{\xi})$ in order to put more weight on the most likely area. Justifications for these formulas are provided in Appendix C.

These partial criteria are then combined into the final refinement criterion through the following weighted sum:

$$c(\boldsymbol{\xi}) = \sum_{k=1}^{m} w_k \bar{c}_k(\boldsymbol{\xi}) \tag{8}$$

with $\bar{c}_k(\boldsymbol{\xi})$ the normalised partial criteria, m the number of measures and w_k the weights. The normalised partial criteria are computed as follows:

$$\bar{c}_k(\boldsymbol{\xi}) = \frac{c_k(\boldsymbol{\xi}) - \min\limits_{\boldsymbol{\xi}}[c_k(\boldsymbol{\xi})]}{\max\limits_{\boldsymbol{\xi}}[c_k(\boldsymbol{\xi})] - \min\limits_{\boldsymbol{\xi}}[c_k(\boldsymbol{\xi})]} \in [0,1].$$

Finally, we propose here to compute the weights as the ratio between the conservative error $\overline{\boldsymbol{\varepsilon}}_k$ and the target accuracy. In this manner, any partial criterion associated with high error compared to the target accuracy will heavily influence the final criterion. Practically, to emphasise this behaviour, the dependence to the conservative error is chosen quadratic:

$$w_k = \left(\frac{\overline{\varepsilon}_k}{s_{2_k}}\right)^2.$$

Note that in the case of multi-point refinement, we conduct a greedy sequential approach by assuming that previous refinements are performed, fixing the predictive values of the GP surrogate model and recomputing the predictive variance to obtain the updated refinement criterion. We can then perform Black-box evaluations in parallel on these points. This approach is usually called the Kriging Believer strategy and allows performing multi-point refinement efficiently without the need of any clustering or local penalisation heuristics. However, this strategy makes a lot of assumptions and is often sub-optimal when performing many refinements in parallel.

4.2.2. SA-based boxes

When the SA model error is low enough, below $\boldsymbol{s}_1$ in all output dimension, the predictive value $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ is directly returned to the optimiser. As stated previously, we propose here to use Gaussian Processes as SA model, which gives access to the predictive variance in order to compute box widths.

The SA model is built on the previously computed boxes to return an approximation $\boldsymbol{\rho}_{SA}$ of $\boldsymbol{\rho}$ in $\mathcal{X}$. In practice, the available data for constructing this model are boxes of various sizes. These boxes can be considered as noisy evaluations of $\boldsymbol{\rho}$ at different $\boldsymbol{x}$, with heterogeneous noises. Using again the $\pm 3\sigma$ paradigm, these noises can be translated into heterogeneous variances $\sigma_i^2 = \left(\frac{\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x}_i)}{3}\right)^2$. This allows to make use of heteroscedastic Gaussian Processes, which naturally take into account heterogeneous noise variances, under a gaussianity assumption. The measure estimations $\boldsymbol{\rho}_{SA}$ and associated error $\overline{\boldsymbol{\varepsilon}}_{SA}$ at a new design $\boldsymbol{x}$ are computed as follows:

$$\begin{aligned} \boldsymbol{\rho}_{SA}(\boldsymbol{x}) &= \boldsymbol{k}_*^T (K + \Delta)^{-1} \widetilde{\boldsymbol{\rho}} \\ \overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) &= 3\boldsymbol{\sigma}_{SA}(\boldsymbol{x}) = 3\big(k_{**} - \boldsymbol{k}_*^T (K + \Delta)^{-1} \boldsymbol{k}_*\big). \end{aligned} \tag{9}$$

The matrix $K = K\big(\{\boldsymbol{x}_i\}_i, \{\boldsymbol{x}_i\}_i\big)$ represents the autocovariance matrix between training points, $k_{**} = k\big(\boldsymbol{x}, \boldsymbol{x}\big)$ at the new design and $\boldsymbol{k}_* = \boldsymbol{k}\big(\boldsymbol{x}, \{\boldsymbol{x}_i\}_i\big)$ the covariance vector between $\boldsymbol{x}$ and training points. The diagonal matrix $\Delta = diag\big(\{\sigma_i^2\}_i\big)$ represents the heterogeneous gaussian noises presented previously. More details on GP surrogate models can be found in [40].

4.3. Quality indicator

Assessing and comparing the performance of several methods requires the computation of a quantitative quality indicator. To this extent, the Hausdorff distance d_H is a classical choice for computing the closeness of the found optimal set to the true one.

However, in practice, the modified Hausdorff distance d'_H proposed in [41] captures the similarities more efficiently by replacing a maximum with an expectation.

In the Bounding-Box context, there is no Pareto Optimal set but rather a set of non-dominated designs, each of them having a Pareto Optimal Probability. The approximated Pareto front $\widetilde{\mathcal{P}}$ and Pareto optima $\mathcal{X}_{\widetilde{\mathcal{P}}}$ are hence aleatory. We propose to compute the expected value of the modified Hausdorff distance with respect to realisations of $\widetilde{\mathcal{X}}_{\mathcal{P}}$.

The deterministic modified Hausdorff distance d'_H writes follows:

$$\begin{aligned} d'_H(\mathcal{A},\mathcal{B}) &= \max\left(d'_1(\mathcal{A},\mathcal{B}), d'_1(\mathcal{B},\mathcal{A})\right), \\ d'_1 &= \frac{1}{N}\sum_{a\in\mathcal{A}} d_2(a,\mathcal{B}), \\ d_2(a,\mathcal{B}) &= \min_{b\in\mathcal{B}}[\|a-b\|_2]. \end{aligned}$$

Under the box independence and uniformity assumptions, as in Section 3.2 we have $\forall \boldsymbol{x}$,

$$\boldsymbol{\rho}(\boldsymbol{x}) \sim \mathcal{U}\Big(\mathcal{B}\big(\widehat{\boldsymbol{\rho}}(\boldsymbol{x}), \widehat{\boldsymbol{\varepsilon}}(\boldsymbol{x})\big)\Big),$$

where $\widehat{\boldsymbol{\rho}}(\boldsymbol{x})$ and $\widehat{\boldsymbol{\varepsilon}}(\boldsymbol{x})$ are the values returned to the optimiser, either from the BB approach or the SA strategy. The proposed expected modified Hausdorff distance reads:

$$Q_{\mathcal{B}} = \mathbb{E}_{\widetilde{\mathcal{X}}_{\mathcal{P}}}\left[d'_H(\mathcal{X}_{\mathcal{P}}, \widetilde{\mathcal{X}}_{\mathcal{P}})\right]. \tag{10}$$

where a realisation of $\widetilde{\mathcal{X}}_{\mathcal{P}}$ corresponds to a realisation of $\boldsymbol{\rho}(\boldsymbol{x})$ for all $\boldsymbol{x}$ and the computation of the classical Pareto optima.

Intuitively, $Q_{\mathcal{B}}$ can be seen as an averaged distance between the real Pareto-optimal area and the preimage of a Pareto front realization. In a normalized input space such as the $[0,1]^2$ square, a value of $Q_{\mathcal{B}} = 1$ corresponds to a terrible score, with approximated and real Pareto optima on opposite sides of the input space. On the contrary, a score of $Q_{\mathcal{B}} = 10^{-2}$ reveals an excellent agreement between the approximated and the real optimal area.

4.4. Summary of SABBa variants

Based on the SABBa algorithm (Alg. 1), we explore potential variants according to some choices. First, as denoted at the beginning of Section 4.2.1, boxes centers and widths can either be computed using a *Coupled Space* (CS) surrogate model, or *Separated Spaces* (SS) surrogate models. Secondly, the user can use (or not) the Surrogate-Assisting (SA) strategy. These algorithmic variants are made explicit in Algorithm 2, where the use of the SA strategy is toggled at line 6, and the SS or CS variants are selected lines 16 to 19.

Algorithm 2 SABBa variants

Loop over values of $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$
Initialise $\mathcal{X}_c$ empty
while Optimisation running **do**
    Read new designs $\mathcal{X}_{new}$
    for each $\boldsymbol{x} \in \mathcal{X}_{new}$ **do**
        if using SA and $\overline{\boldsymbol{\varepsilon}}_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1$ **then**
            Return $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ to the optimiser
        else
            Add $\boldsymbol{x}$ in $\mathcal{X}_c$
            Compute a first approximation $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ of $\boldsymbol{\rho}(\boldsymbol{x})$
        end if
    end for
    while $\mathcal{X}_r = \left\{\boldsymbol{x} \in \mathcal{X}_{\widetilde{\mathcal{P}}_{\mathcal{B}}} \cap \ \mathcal{X}_c \mid \overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}) > \boldsymbol{s}_2\right\}$ is non-empty **do**
        Find $\boldsymbol{x}^* = \underset{\boldsymbol{x} \in \mathcal{X}_r}{\operatorname{argmax}} POP_{min}(\boldsymbol{x})$
        Compute $\boldsymbol{q}(\boldsymbol{x}^*, \boldsymbol{\xi})$ at some $\boldsymbol{\xi}$
        if using SS surrogate model **then**
            Update $\widehat{\boldsymbol{q}}(\boldsymbol{x}^*, \cdot)$ in Ξ
        else if using CS surrogate model **then**
            Update $\widehat{\boldsymbol{q}}(\cdot, \cdot)$ in $\mathcal{X} \times \Xi$
        end if
        Compute new estimates $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x}^*)$ and $\overline{\boldsymbol{\varepsilon}}(\boldsymbol{x}^*)$
    end while
    Update $\boldsymbol{\rho}_{SA}$ and $\overline{\boldsymbol{\varepsilon}}_{SA}$ with the new $\widetilde{\boldsymbol{\rho}}$ and $\overline{\boldsymbol{\varepsilon}}$ values
    Return $\widetilde{\boldsymbol{\rho}}(\boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{X}_{new}$ to the optimiser
end while

These combinations lead to four main variants, denoted SA-CS, SA-SS, CS and SS, where the two latest do not use the SA strategy. Note that the SS variant is the only one that does not build any surrogate in the design space, making it very computationally expensive. For this reason, it will not be further studied in the following of this paper. Parallels can be made between these strategies and formulations 3 and 4 from [24]. Among the three remaining variants, one can expect SA-CS to give better results than SA-SS and CS, taking advantage of both the low-dimensional Surrogate-Assisting model and the coupled space correlation to speed up box refinement.

In the following, we compare the performance of the proposed variants to an A Priori MetaModel (APMM) strategy. A surrogate model $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \boldsymbol{\xi})$ is constructed in the coupled space based on a given Design of Experiments (DoE). The robustness and reliability measures at different locations $\boldsymbol{x}$ are then numerically approximated using this surrogate throughout the optimisation process. The overall computational cost of this strategy is thus entirely dependent on the initial DoE.

In the next section, the variants described above are applied to analytical test-cases to

assess their performance. The quality of the optimisation outputs is quantified using the quantitative indicator presented above.

5. Analytical comparisons

We apply here SABBa to two analytical test-cases. The first one is low-dimensional and deals with a Taguchi multi-objective robustness formulation (mean optimisation and variance minimisation). The second test-case features a higher number of dimensions and consists in a mean optimisation under quantile constraint.

Ten runs are performed for each strategy to capture both the mean convergence curve and the associated variability, represented as a translucent band around the mean. For plotting purposes, the log distance is assumed to show a Gaussian distribution over the repeated runs, implying a log-normal distribution over the actual distance. Note that a high variability of the convergence curves reveals a lack of reliability of the associated approaches.

5.1. Test-case 1: Unconstrained Taguchi optimisation

This problem is a bi-objective robust optimisation proposed in [35]. There are two design variables $\boldsymbol{x}$ and one uncertain parameter ξ, and the problem reads:

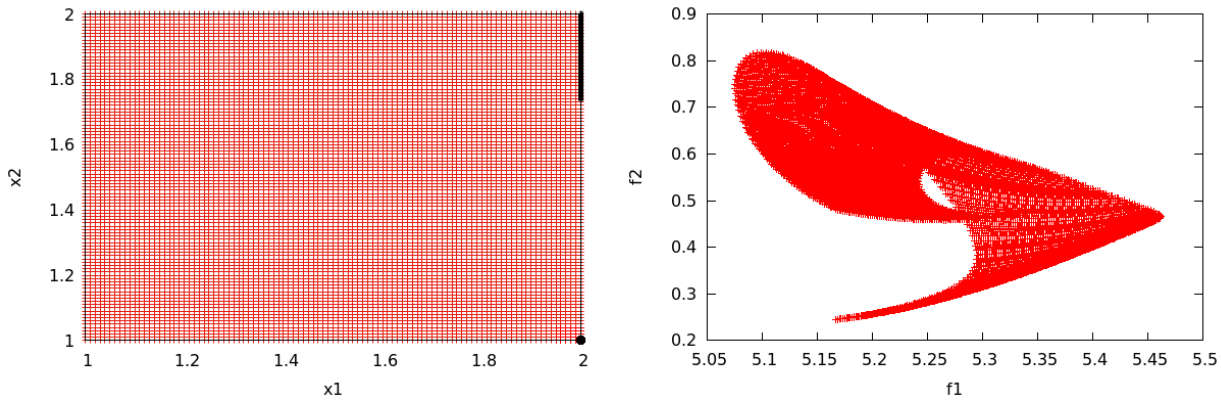
$$\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \begin{pmatrix} \mu(\boldsymbol{x}) \\ \sigma^2(\boldsymbol{x}) \end{pmatrix} \\
\text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_\xi[q(\boldsymbol{x}, \xi)] \\
& \sigma^2(\boldsymbol{x}) = \mathbb{V}_\xi[q(\boldsymbol{x}, \xi)] \\
\text{with:} \quad & q(\boldsymbol{x}, \xi) = \xi - x_1\xi^5 + \cos(2\pi x_2 \xi) + 5 \\
& \xi \sim \mathcal{U}([0,1]) \\
\text{by changing:} \quad & (x_1, x_2) \in [1,2]^2
\end{aligned} \tag{11}$$

The Pareto front associated with this problem is discontinuous, and the optimal set in the design space is the union of a segment and a point (see Fig. 4).

For this first test-case, we consider two different surrogate-modelling capabilities. The aim is to illustrate SABBa performance both when the surrogate is able to represent the underlying functions accurately and when it shows poor efficiency. This highly impacts the quality of the APMM $\widehat{\boldsymbol{q}}$, the Surrogate-Assisting model $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ and the separated or Coupled-Space models $\widehat{\boldsymbol{q}}(\boldsymbol{x}, \cdot)$.

5.1.1. With high-quality surrogate model

The unconstrained Taguchi optimisation problem is first solved using high-quality meta-modelling approaches. Practically, we build a heterogeneous GP surrogate model with an anisotropic RBF kernel (also called squared-exponential, exponential quadratic or Gaussian),



(a) (b)

Figure 4: Test-case 1: a) Discretisation of the design space in red and Pareto optimal sets in black. b) Image of the discretised points in the objective (μ,σ^2) space.

and Automatic Relevance Determination (ARD):

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left(-\frac{r_{\boldsymbol{l}}^2}{2}\right)$$
$$\text{with } r_{\boldsymbol{l}} = \sqrt{\sum_{i=1}^{m}\left(\frac{x_i - x_i'}{l_i}\right)^2}$$

We recall that the kernel function gives the covariance matrix in Equation (9).

This model requires $m+1$ hyperparameters $\{\sigma^2, l_1, \ldots, l_m\}$ to be optimised but captures the characteristic lengthscale associated with each input dimension.

Note that here, the thresholds $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are sequentially refined five times. To alleviate the tuning of these thresholds, SABBa can deal with normalised thresholds $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$. At each iteration, the range h_i covered in the i^{th}-dimension is updated, namely,

$$\forall i, h_i = \max_{\boldsymbol{x}}[\rho_i(\boldsymbol{x})] - \min_{\boldsymbol{x}}[\rho_i(\boldsymbol{x})],$$

and the thresholds $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ are given in percentage of $\boldsymbol{h}$. Here, $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ are both sequentially taken as 50%, 40%, 30%, 20%, 10% and 5% in all dimensions. The chosen parameters of SABBa are as follows: $N_{init} = 10$, $N = 1$ (sequential optimiser), $N_{first} = 5$ and $N_{new} = 1$ (sequential refinement).

Figure 5 pictures the convergence curves of the APMM strategy and the three studied SABBa variants (SA-CS, SA-SS and CS). The mean curves are mostly comparable, with a slight advantage for SA-CS. However, the associated variability is much smaller when using any SABBa variant with respect to the APMM strategy. The indicator is assumed to follow a normal distribution in log scale, hence to be log-normal. The high variance associated to the APMM strategy coupled with the heavy tail of the log-normal distribution makes the approach very unreliable. We picture this risk in Figure 6, where the worst result out of ten

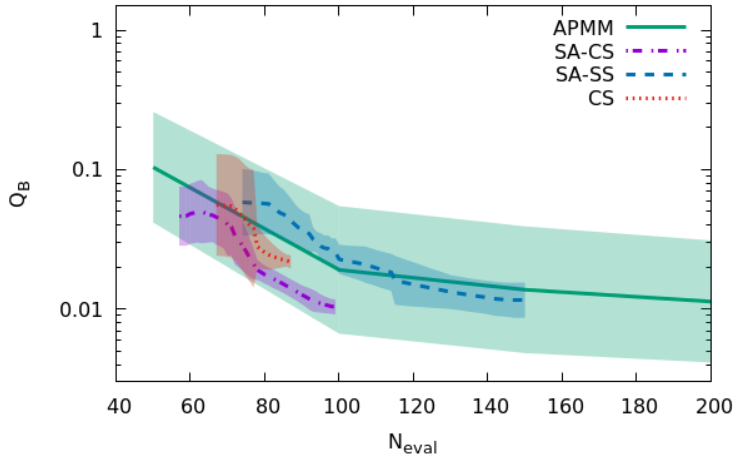
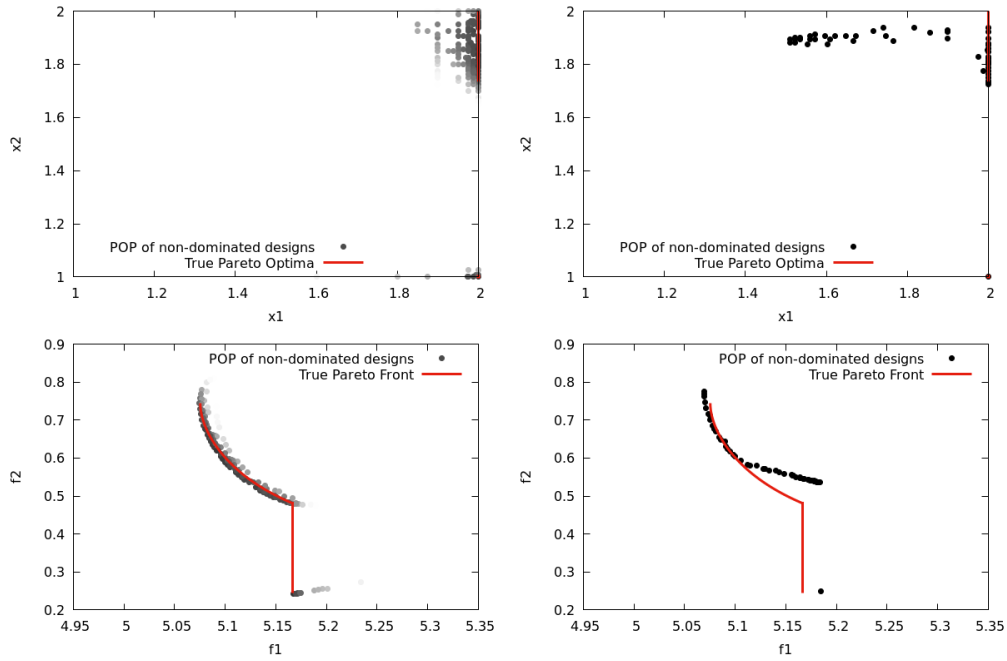


Figure 5: Test-case 1-a: Cost comparison between APMM and three SABBa variants.

repeated runs of SA-CS and the APMM strategy are plotted. These outputs correspond to $N_{eval} \approx 100$.



(a) (b)

Figure 6: Worst optimisation results: (a) SABBa SA-CS, 93 evaluations , $Q_{\mathcal{B}} = 1.29 \times 10^{-2}$. (b) APMM, 100 evaluations , $Q_{\mathcal{B}} = 1.91 \times 10^{-1}$.

In these figures, optimal designs returned by SABBa are plotted in greyscale. This refers to the Pareto Optimal Probability (POP) of each design, ranging continuously from 0 (white)

to 1 (black).

In line with Figure 5, Figure 6 reveals that the APMM strategy may return very inaccurate results when the DoE that was used for constructing the coupled-space model is not well sampled in the optimal area. On the contrary, the adaptive surrogate model construction permits SABBa to return very stable results.

5.1.2. With a low-quality surrogate model

In the following, the same optimisation problem is solved using a low-quality surrogate model in both SABBa and the APMM strategy. Contrarily to the previous formulation, we build an isotropic GP surrogate model with only one lengthscale, that has to account for all dimensions. Practically,

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left(-\frac{r_l^2}{2}\right)$$
$$\text{with } r_l = \frac{1}{l}\sqrt{\sum_{i=1}^{m}\left(x_i - x_i'\right)^2}$$

Only two hyperparameters $\{\sigma^2, l\}$ must be optimised. However, this model will notably fail when the characteristic lengths of the function in the different dimensions are very disparate. This test-case aims at simulating problems where the coupled space behaviour is hard to model. This would naturally arise when the number of dimensions is significant. Hence, one can expect the SA strategy to yield a significant cost improvement through low-dimensional measures surrogate modelling.

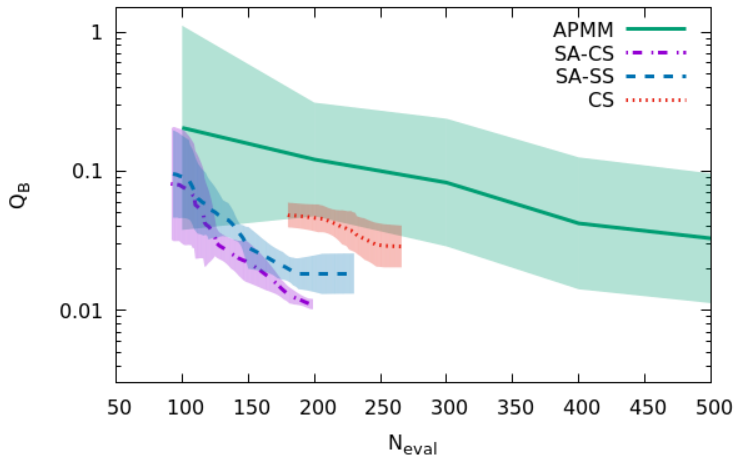


Figure 7: Test-case 1-b: Cost comparison between APMM and three SABBa variants.

Such improvement is indeed revealed in Figure 7. The use of the SA strategy has two consequences: (i) non-Surrogate-Assisted approaches (APMM and SABBa CS) show poor performance compared to the other strategies and (ii) the gap between SA-CS and SA-SS is

much narrower. Indeed, here, they both rely for the most part on the low-dimensional SA model.

As previously, the worst optimisation outputs are plotted for SABBa SA-CS and the APMM approach. In Figure 8, SABBa shows again much higher consistency and accuracy compared to APMM.

The use of coupled space surrogate models and Surrogate-Assisting strategy have shown to bring a significant cost reduction for the SABBa SA-CS framework. It performs better in average than the APMM strategy and shows far greater consistency and robustness.

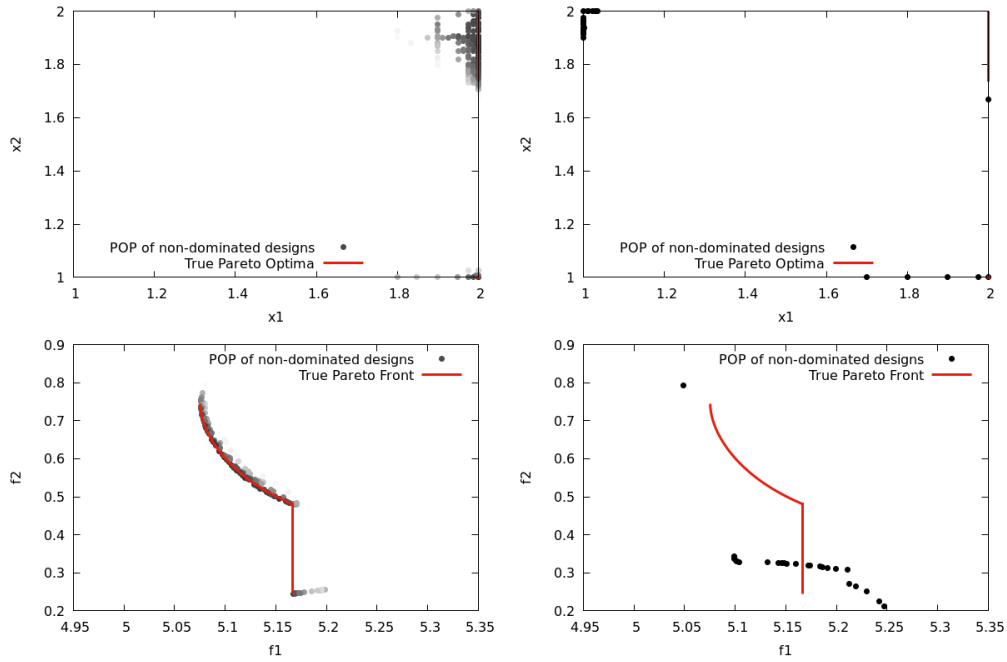


Figure 8: Worst optimisation results: (a) SABBa SA-CS, 196 evaluations , $Q_{\mathcal{B}} = 1.23 \times 10^{-2}$. (b) APMM, 200 evaluations , $Q_{\mathcal{B}} = 7.94 \times 10^{-1}$.

5.2. Test-case 2: Quantile-constrained mean performance optimisation

We propose this second test-case to assess the performance of SABBa in a higher dimensional case and in the presence of a reliability-based constraint (derived from the Six-Hump Camel function). The objective is a robustness measure derived from a simplified Rosenbrock function. We consider here four design variables and three uncertain parameters. The

problem is stated as follows:

$$
\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \mu(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = \mathrm{q}^{95\%}(\boldsymbol{x}) \leq 1 \\
\text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[q_1(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \mathrm{q}^{95\%}(\boldsymbol{x}) = \mathrm{q}^{95\%}_{\boldsymbol{\xi}}[q_2(\boldsymbol{x}, \boldsymbol{\xi})] \\
\text{with:} \quad & q_1(\boldsymbol{x}, \boldsymbol{\xi}) = \sum_{i=1}^{3} \left[\big(1 - x_i\big) + 3 \left(1 + \frac{\arctan\big(5(\xi_i - 0.5)\big)}{2} \right) \big(x_{i+1} - x_i^2\big)^2 \right] \\
& q_2(\boldsymbol{x}, \boldsymbol{\xi}) = \left(4 - 2.1 x_1^2 + \frac{x_1^4}{3} \right) x_1^2 + x_1 x_2 + \big(-4 + 4 x_2^2 \big) x_2^2 \\
& \qquad\qquad + \frac{\cos(2\pi\xi_1) - \sin(\frac{\pi}{2}\xi_1) - \xi_1 - \big(\cos(2\pi 0.05) - \sin(\frac{\pi}{2} 0.05) - 0.05 \big)}{5} \\
& \boldsymbol{\xi} \sim \mathcal{U}([0,1]^3) \\
\text{by changing:} \quad & \boldsymbol{x} \in [-0.2, 1.2]^4
\end{aligned}
\tag{12}
$$

One can note that the robustness measure reduces to the classical formulation of the 4D Rosenbrock function and the reliability measures to the Six-Hump Camel function. Analytically, it holds that:

$$
\begin{aligned}
\mu(\boldsymbol{x}) &= \sum_{i=1}^{3} \left[\big(1 - x_i\big) + 3 \big(x_{i+1} - x_i^2\big)^2 \right], \\
\mathrm{q}^{95\%}(\boldsymbol{x}) &= \left(4 - 2.1 x_1^2 + \frac{x_1^4}{3} \right) x_1^2 + x_1 x_2 + \big(-4 + 4 x_2^2 \big) x_2^2 .
\end{aligned}
\tag{13}
$$

The optimum of this deterministic problem is found at $\boldsymbol{x}_* \approx (0.7033, 0.7035, 0.6212, 0.3859)$ with $\mathrm{q}^{95\%}(\boldsymbol{x}_*) = 1$ and $\mu(\boldsymbol{x}_*) \approx 0.4981$.

For this test case, both $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ are sequentially taken as 50%, 40%, 30%, 20%, 10%, 5%, 3%, 2%, 1% and 0.5%. As for the other parameters, $N_{init} = 10$, $N = 1$ (sequential optimiser), $N_{first} = 5$ and $N_{new} = 1$ (sequential refinement).

Figure 9 shows a slight mean improvement when using SA-CS compared to the APMM strategy. The variability of the output is also halved with all SABBa variants, which is critical for real-world applications. Note that the plateau that is reached by the SA-CS curve is actually a plotting artefact. All runs have different final number of evaluations, and we considered the indicator $Q_{\mathcal{B}}$ constant when the optimum is reached. Therefore, the average of these curves tends to flatten at the end.

Again, we plot the worst optimisation outputs in Figures 10. They are depicted in parallel coordinates plots, with a $POP_{\min}$ greyscale. Each efficient individual is represented by a grey curve and the true optimum is in red. This comparison reveals again the high unreliability of the APMM strategy with respect to the SABBa framework.

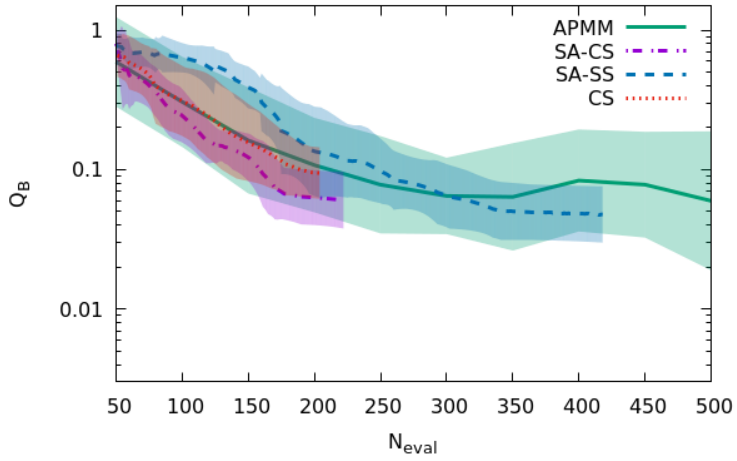


Figure 9: Test-case 2: Cost comparison between APMM and three SABBa variants.

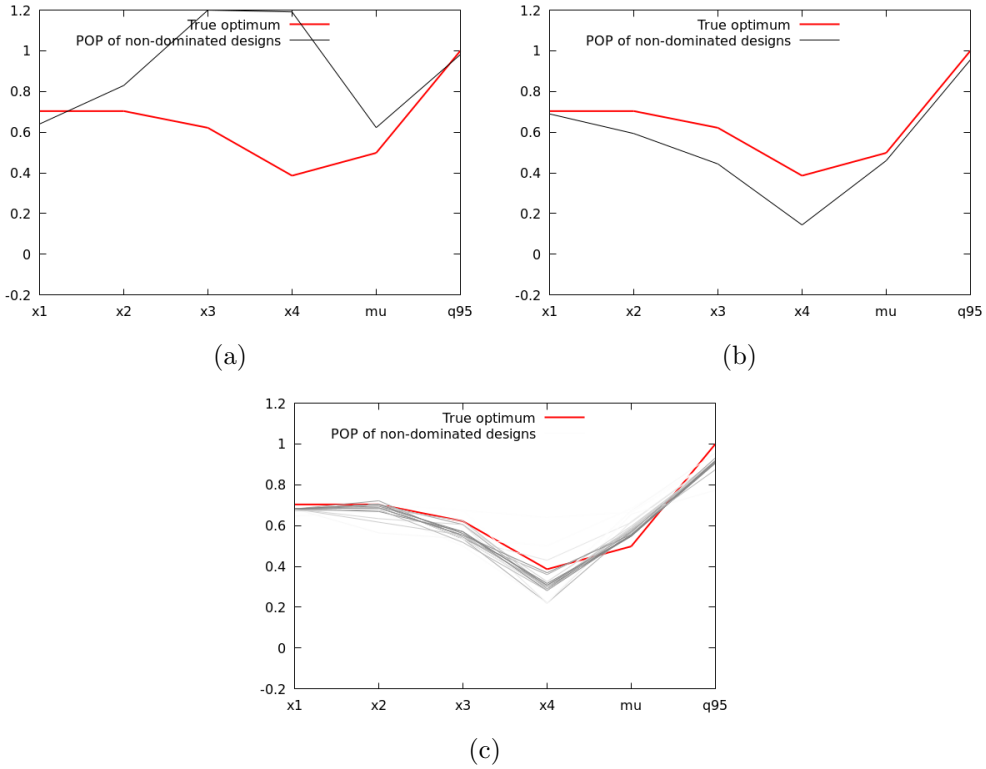


Figure 10: Worst optimisation results: (a) APMM, 150 evaluations, $Q_{\mathcal{B}} = 1.0 \times 10^{0}$. (b) SABBa SA-CS, 161 evaluations, $Q_{\mathcal{B}} = 1.12 \times 10^{-1}$. (c) APMM, 200 evaluations, $Q_{\mathcal{B}} = 3.24 \times 10^{-1}$.

6. Physical applications

Three real-world test cases are tackled hereafter. The first one, Section 6.1, is the classical two-bar truss structural test case, in which the external force and the Young modulus are

considered uncertain. This application features four design parameters and two uncertain variables. Then, shape optimization of an Organic Ranking Cycle (ORC) turbine is performed in Section 6.2, with nine design parameters and three uncertain parameters. Finally, the design of a Thermal Protection System (TPS) is optimized in Section 6.3, where the design space is of dimension two and the uncertain space of dimension twelve.

6.1. Two-bar truss

The two-bar truss optimization problem is notably illustrated in Refs. [42] and [20]. The uncertainty-based optimization problem is formulated as follows (with a schematic representation in Figure 11(a)):

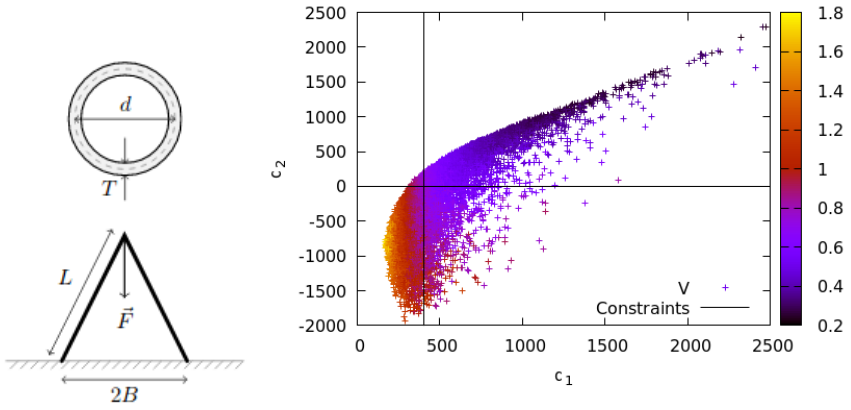
$$\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = V(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = \begin{pmatrix} c_1(\boldsymbol{x}) \\ c_2(\boldsymbol{x}) \end{pmatrix} \leq \begin{pmatrix} s_{\max} \\ 0 \end{pmatrix} \\
\text{where:} \quad & c_1(\boldsymbol{x}) = \mathrm{q}^{0.999}[s(\boldsymbol{x}, \boldsymbol{\xi})] \\
& c_2(\boldsymbol{x}) = \mathrm{q}^{0.999}[s(\boldsymbol{x}, \boldsymbol{\xi}) - s_{crit}(\boldsymbol{x}, \boldsymbol{\xi})] \\
\text{with:} \quad & \boldsymbol{\xi} \sim \mathcal{N}\left(\begin{pmatrix} 150000 \\ 210000 \end{pmatrix}, \begin{pmatrix} 30000^2 & 0 \\ 0 & 21000^2 \end{pmatrix} \right) \\
\text{by changing:} \quad & \boldsymbol{x} \in [20, 80] \times [800, 1200] \times [700, 800] \times [2, 3]
\end{aligned} \tag{14}$$

In the above, the design variables are: (x_1) the diameter of the cross section d, (x_2) the bar length L, (x_3) the structure half-width B and (x_4) the thickness of the cross section T. The uncertain parameters refer to (ξ_1) the external force F and (ξ_2) the elastic modulus E. The objective is to minimize the total volume $V(\boldsymbol{x}) = 2\pi x_1 x_2 x_4 \times 10^{-6}$ while verifying that the probability of $s(\boldsymbol{x}, \xi) = \dfrac{x_2 \xi_1}{2\pi x_1 x_4 \sqrt{x_2^2 - x_3^2}}$ exceeding $s_{\max} = 400$ N.mm^{-2} and $s_{crit}(\boldsymbol{x}, \xi) = \dfrac{\pi^2 \xi_2 (x_1^2 + x_4^2)}{8 x_2^2}$ are both below 0.001.

The problem is represented in Fig. 11(b), where the objective value ($V(\boldsymbol{x})$, to be minimised) is plotted in the constraints space with a colour scale. The abscissa and ordinate directions refer to the constraints $c_1(\boldsymbol{x})$ and $c_2(\boldsymbol{x})$; the associated thresholds are drawn with red lines. The admissible set is in the lower left quadrangle ($c_1 < 400$ N.mm^{-2} and $c_2 < 0$ N.mm^{-2}) and one may note that the objective is nearly constant on the limit $c_1 = 400$ N.mm^{-2}, with optimum value at $(c_1, c_2) = (400, 0)$.

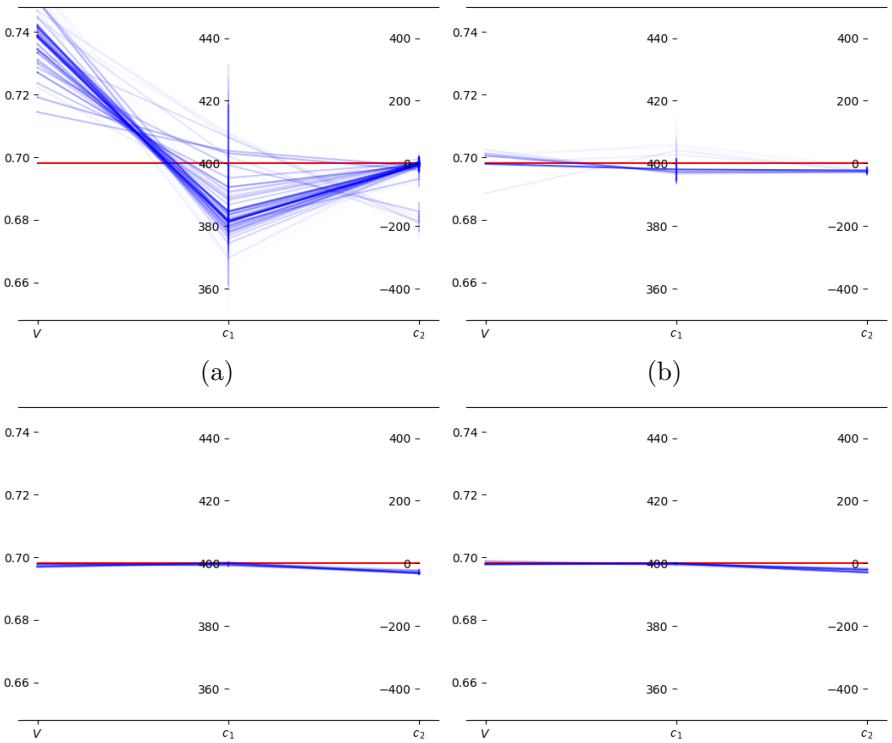
This optimisation has been solved with the SA-CS variant. Thresholds $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are sequentially reduced up to 0.1% of the total ranges, with four steps 3%, 1%, 0.3% and 0.1%.

Results are depicted in Figure 12 with parallel coordinates plots. The four thresholds are sequentially reached with a total number of **42**, **44**, **46** and **52** function evaluations, and an optimal objective value of roughly 0.698 is found. Uniform approximation errors from SABBa are represented as vertical intervals for each design with non-zero POP. Note that



(a) (b)

Figure 11: (a) Schematic representation of the two bars, with bar section above (from [42]). (b) Representation in the objective/constraints space, constraints in abscissa and ordinate, objective in color.



(a) (b)

(c) (d)

Figure 12: Objective and constraints values with (a) **42**, (b) **44**, (c) **46** and (d) **52** evaluations. Transparency driven by POP value and optimal value in red.

the second constraint c_2 is slightly off its optimal value 0 because the objective value V is nearly constant w.r.t. c_2, as can be observed in Figure 11(b).

On this analytical engineering test-case, SABBa reveals very parsimonious. Uncertainty-based optimisation with 95% quantiles has been solved with only 40 to 50 evaluations.

6.2. ORC turbine shape optimisation

This test case aims to show the potential of SABBa in solving a problem formulation that includes the concepts of robustness and reliability. In particular, we focus on multi-objective shape optimization to reduce the mean and variance of a given quantity of interest with a statistical constraint.

Here, we consider the design of a 2D Organic Rankine Cycles (ORCs) turbine, which is a very relevant problem to exploit renewable energy sources. The studied blade profile is the Biere, which results in a convergent-divergent cascade passage to accelerate the fluid up to supersonic speed. Across the cascade, the fluid is expanded from superheated conditions. However, shocks appear past stator vanes, as depicted in Figure 13, and may induce significant losses in the turbine efficiency. The robust design of the trailing edge is thus critical. SABBa is here exploited for optimising the turbine blade profile under uncertain inlet and outlet conditions. To estimate the aerodynamic performances of the turbine, the 2D Non-Ideal Compressible-Fluid Dynamics solver included in the SU2 suite is used [43]. In particular, for computational reasons, the in-house reduced model built in Ref. [43] is used here, allowing for a quick evaluation. Furthermore, an approximate Pareto front which is already known for the tackled OUU problem, is used to assess the performances of SABBa.

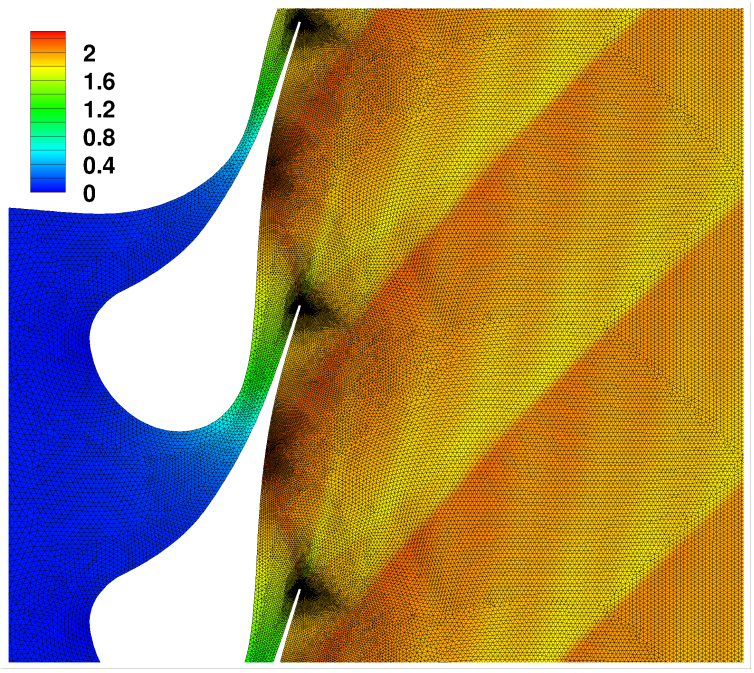


Figure 13: Mach contours at nominal conditions for the baseline profile and computational grid of 36k cells.

A spline-based geometry parameterisation is proposed. A B-spline curve of degree 3 is defined over a total number of 30 Control Points (CP) from which a subset of nine constitute

the design vector $\boldsymbol{x}$, denoted in red in Figure 14. These CPs are allowed to move in the direction normal to the baseline geometry.

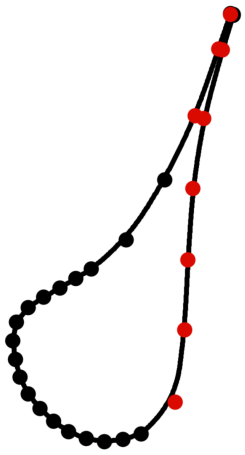


Figure 14: B-splines parameterisation. Fixed CP in black, moving CP in red.

The objective function ΔP is defined as the standard deviation of the azimuthal distribution of static pressure downstream of the blade. A low standard deviation should induce lower shocks in the outlet flow. A constraint is also imposed on the mass-flow rate per unit span $\dot{m}$, to remain close from the nominal value.

The tackled OUU problem reads then as follows:

$$\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \begin{pmatrix} \mu_1(\boldsymbol{x}) \\ \sigma^2(\boldsymbol{x}) \end{pmatrix} \\
\text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = \mu_2(\boldsymbol{x}) \in [0.98\dot{m}_0, 1.02\dot{m}_0] \\
\text{where:} \quad & \mu_1(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\Delta P(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \sigma^2(\boldsymbol{x}) = \mathbb{V}_{\boldsymbol{\xi}}[\Delta P(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \mu_2(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\dot{m}(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \dot{m}_0 = 344.843 \\
\text{with:} \quad & \boldsymbol{\xi} \sim \mathcal{U}(\Xi) \\
\text{by changing:} \quad & \boldsymbol{x} \in \mathcal{X}
\end{aligned} \tag{15}$$

where $\Xi = [7.95, 8.05] \times [541.15, 549.15] \times [1, 2]$. Note that the uncertain parameters correspond to the inlet and outlet conditions: $\boldsymbol{\xi} = [P^t_{in}, T^t_{in}, P^s_{out}]$. Overall, the problem features nine design parameters and three uncertainties.

SABBa is ran with parameters $N_{init} = 5$, $N_{new} = 1$ (sequential optimizer), $N_{first} = 4$ and $N_{ref} = 1$ (sequential refinement). The normalised thresholds $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$ are both sequentially taken as 10%, 5%, 2% and 1% in all dimensions. The Pareto fronts associated with each of these thresholds are depicted hereafter in Figure 15. All boxes with non-zero POP are drawn with a transparency corresponding to the POP value. One can qualitatively see that these

promising boxes get closer and closer to the known Pareto front, depicted in red. SABBa SA-CS is able to get a very good approximation of the robust front at the cost of only **600** function evaluations, although missing the very left part of the front. This test case clearly illustrates the good performances of SABBa within a formulation combining robustness and reliability metrics.

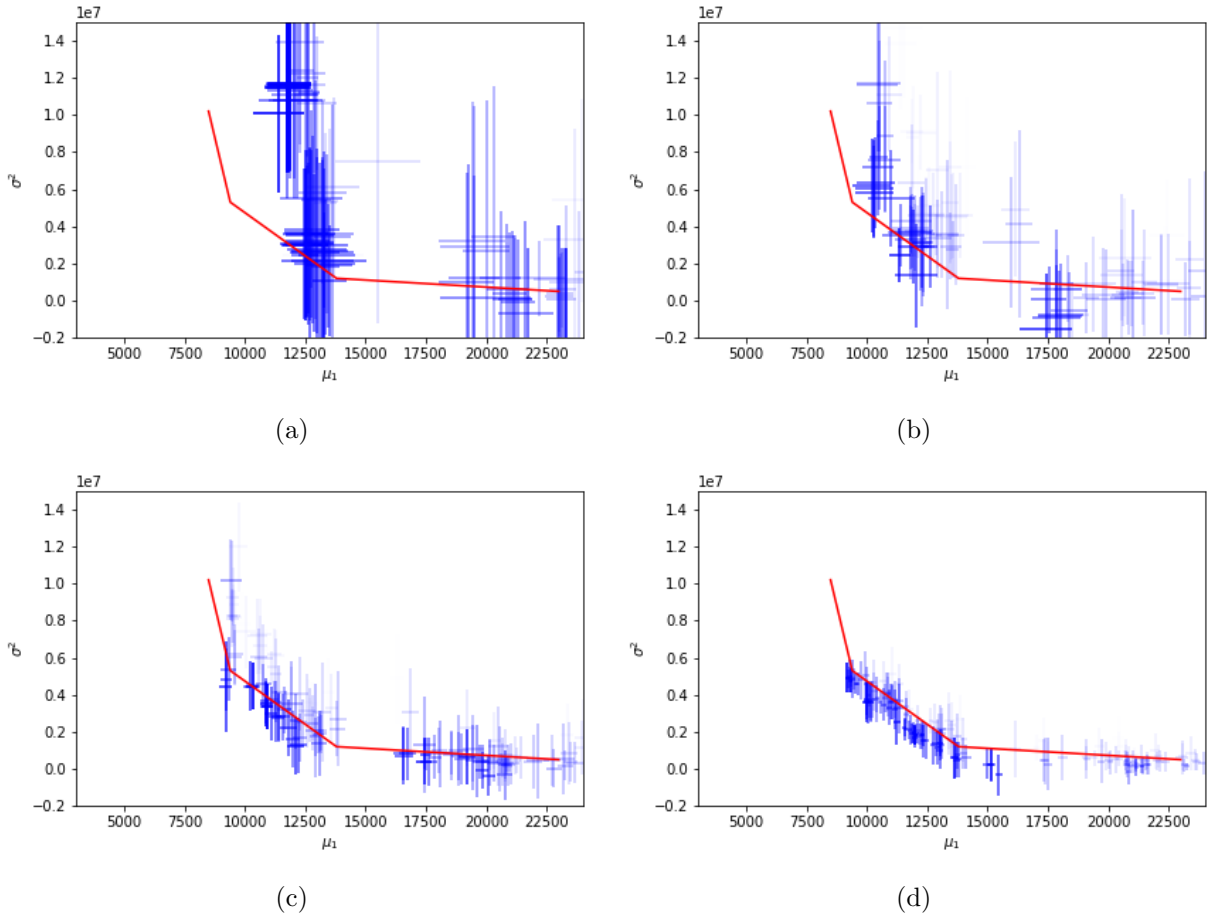


Figure 15: Outputs of the SABBa framework with (a) **156**, (b) **241**, (b) **418**, (b) **598** function evaluations. Transparency of the boxes driven by POP value and known front in red.

6.3. Design of a Thermal Protection System

The SABBa framework is then applied to the design of a thermal protection system for a re-entry vehicle. This test-case deals with 12 dimensions and aims at minimising the mean mass density under worst-case temperature constraint.

We study the re-entry of Stardust, that was the first mission using a low-density carbon-phenolic ablator in 2006. Stardust was the fastest man-made object re-entering the earth atmosphere, at a velocity of 12.7 km/s. A generic heat-shield for atmospheric reentry is depicted in Figure 16.

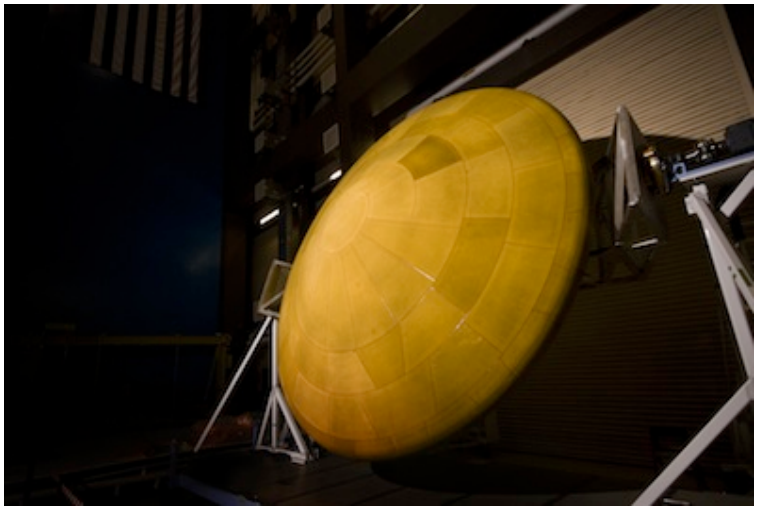


Figure 16: Thermal Protection System (TPS) of the Mars Science Laboratory (MSL) for atmospheric reentry.

Surface total pressure and heat flux were computed from hypersonic computational fluid dynamics (CFD) simulations. In accordance with the state-of-the-art design approach, we assume that the problem is locally mono-dimensional. The analysis is performed using the properties of the Theoretical Ablative Composite for Open Testing (TACOT). Nominal TACOT properties are available in the open literature. Volume-wise, TACOT is made of 10% of carbon fibers, 10% of phenolic resin, and is 80% porous. The thickness of the ablative material is two inches and adiabatic conditions are used at the bondline.

The physical model used here is a generic heat and mass transfer model for porous media that is presented in [44]. The model is implemented in the Porous material Analysis Toolbox (PATO), distributed Open Source. First-order implicit finite-volume schemes in time and space were used for the simulations. The mono-dimensional problem was meshed with 300 finite-volume cells with a logarithmic refinement of parameter 0.2 towards the surface. In this study, we used an equilibrium chemistry model. We study the material response at the stagnation point, which reaches the highest temperature during the reentry. Figure 17 illustrates the typical evolution of the temperature inside the material (dashed lines) and at the heated surface (plain red line).

Formally, the optimisation problems reads as follows:

$$\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \mu(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = M(\boldsymbol{x}) \leq 473.15 \\
\text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\sigma(\boldsymbol{x}, \boldsymbol{\xi})] \\
& M(\boldsymbol{x}) = \max_{\boldsymbol{\xi}}[T_b(\boldsymbol{x}, \boldsymbol{\xi})] \\
\text{with:} \quad & \boldsymbol{\xi} \sim \mathcal{U}(\Xi) \\
\text{by changing:} \quad & \boldsymbol{x} \in [0.01, 0.1] \times [3.5, 7]
\end{aligned} \tag{16}$$

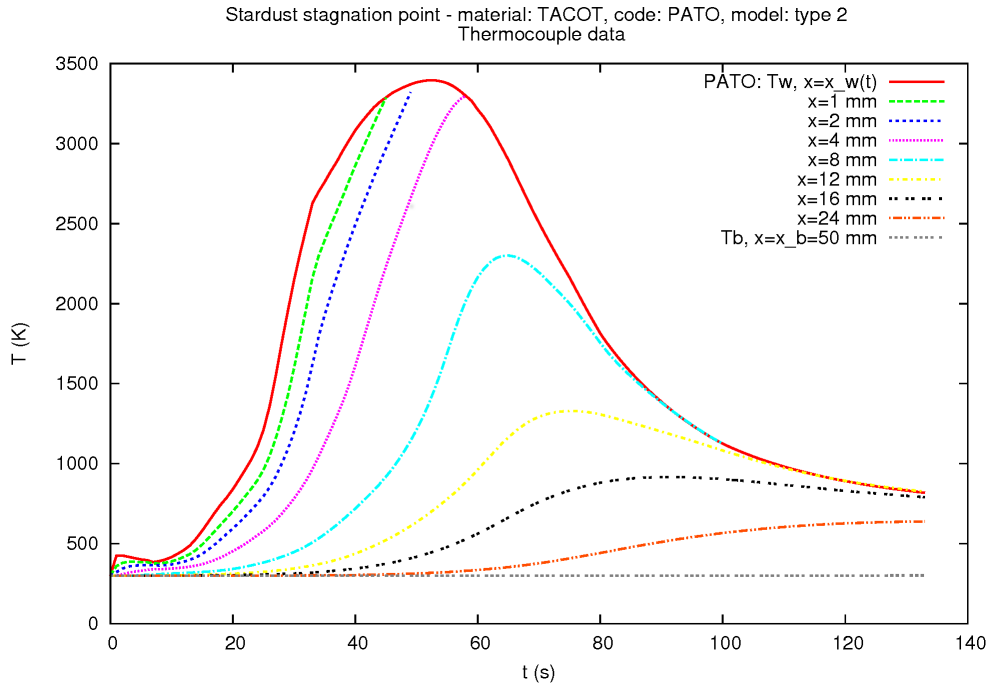


Figure 17: Surface and in-depth temperatures obtained with nominal parameters.

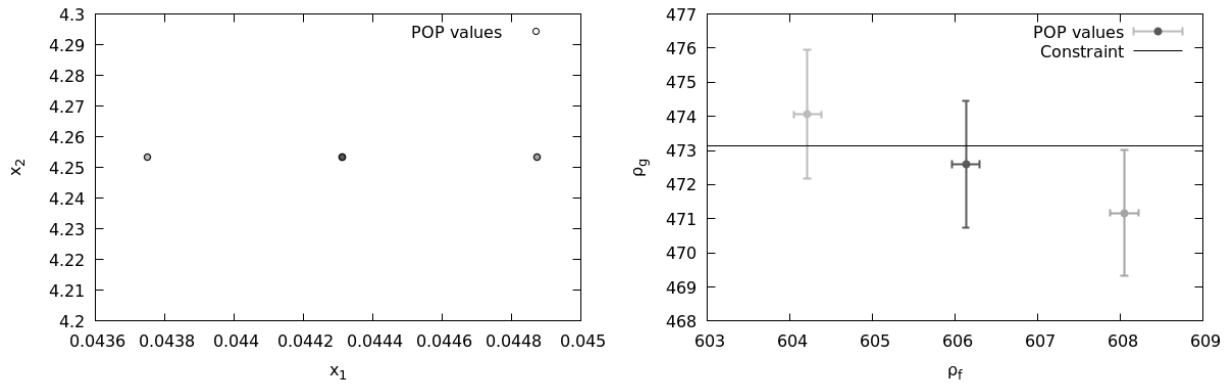
with x_1 the resin volume fraction (originally 10%) an x_2 the overall width of the system (originally 7.21 cm). We observe in Figure 17 that with initial parameters, the bottom temperature stays merely constant during the whole reentry. Hence, the search space is centred on lower x_1 and x_2 values, which tends to increase the bottom temperature.

In this test-case, Ξ is of dimension 12 and all uncertainties are assumed uniform. The choice of the uncertain parameters relies on a sensitivity analysis performed on this test-case in a previous paper [45]. Here, retained uncertain parameters are: 1) Density and volume fraction of the fibrous preform ($\pm$ 5% uncertainty each); 2) Density and volume fraction of the phenolic resin ($\pm$ 5% uncertainty each); 3) Thermal properties of the charred material: heat capacity, conductivity and emissivity ($\pm$ 5% uncertainty each); 4) Oxygen fraction in the pyrolysis gases ($\pm$ 10% uncertainty); 5) Pyrolysis reaction activation energy ($\pm$ 10% uncertainty); 6) Overall width of the system ($\pm$ 0.1 cm uncertainty). Both design parameters are also affected by an uncertainty ($\pm$0.005 for x_1 and $\pm$0.1 for x_2).

Each function evaluation requires the 1D simulation of heat transfer and shield ablation. This takes approximately 10 minutes to compute on a 2.90 GHz processor. An acceptable global cost should remain within a day, or equivalently below 100 to 150 evaluations.

SABBa is ran with parameters $N_{init} = 5$, $N_{new} = 1$ (sequential optimizer), $N_{first} = 4$ and $N_{ref} = 1$ (sequential refinement). As for the normalised thresholds $\bar{\boldsymbol{s}}_1$ and $\bar{\boldsymbol{s}}_2$, they are both sequentially taken as 50%, 40%, 30%, 20%, 10%, 5%, 3%, 2% and 1% in all dimensions.

Three optima are found using SABBa with coupled-space surrogate model (SA-CS) for a

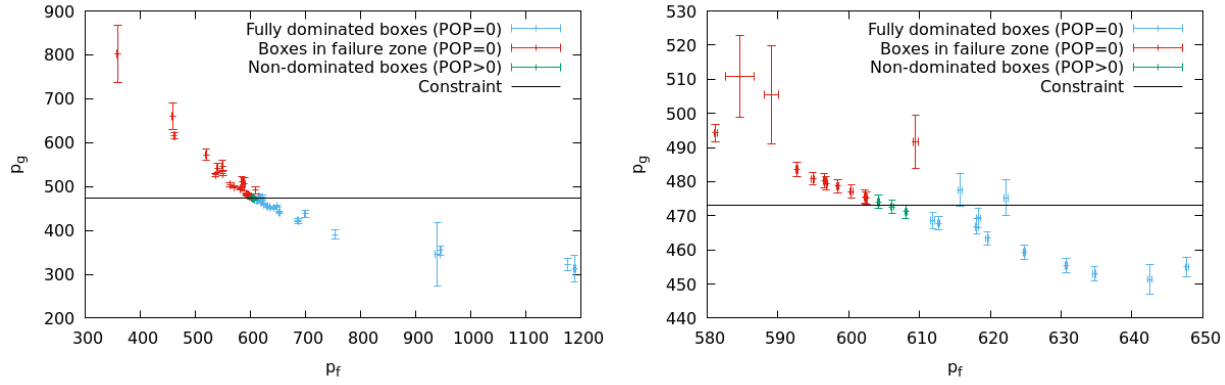


(a) (b)

Figure 18: Outputs of the SABBa framework in (a) the input space, (b) the objective/constraint space.

computational cost of only **40** function evaluations. These Pareto optimal designs are plotted in Figure 18. The design $\boldsymbol{x}^* \approx (4.43 \times 10^{-2}, 4.25)$ has the highest Pareto Optimal Probability (POP), of approximately 65%. It corresponds to the middle box in Fig. 18(b), where $\boldsymbol{\rho_f}$ must be minimised and the admissible set is below the constraint threshold $\boldsymbol{\rho_g} = 473.15$ K.

In Figure 19, we have reported the performance of all the designs evaluated during the optimisation. They are plotted in blue when dominated, red when entirely in the failure zone and green when non-dominated.



(a) (b)

Figure 19: All constructed boxes, depicted in blue when dominated, in red when entirely in failure zone and in green when non-dominated. (a) Full view, (b) zoomed-in view.

Among the three optimal designs depicted in Figure 18, only the right one lies entirely in the admissible set, and thus belongs to $\mathcal{A}_{\mathcal{B}}$. Contrarily to the two other optimal designs, it can dominate other boxes. In Figure 19, it can be observed that this design dominates all the blue boxes, that are not entirely in the failure zone but are strictly worse in the objective dimension. Contrarily, the red boxes belong to $\mathcal{F}_{\mathcal{B}}$ and lie entirely in the failure zone. Hence,

they are all considered dominated. This behaviour explains why only three designs are kept, as shown in Figure 18. It can be observed in Figure 19 that the Bounding-Box approach permits to exploit the tunable accuracy since extreme boxes (on the very left or very right) are estimated with very wide uncertainty.

Figure 20 provides a visualisation of the final SA model. In particular, we plot the SA mean surface density $\boldsymbol{\rho}_{SA_f}(\boldsymbol{x})$, wherever the constraint $\boldsymbol{\rho}_{SA_g}(\boldsymbol{x}) \leq 473.15$ k is satisfied. One can see that (i) the optima plotted in Figure 18 are coherent with this SA model, and (ii) the objective value $\boldsymbol{\rho}_{SA_f}(\boldsymbol{x})$ seems merely constant on the constraint limit. The final accuracy is imposed at 1%, which may not be enough to discriminate designs on the constraint limit. The above found $\boldsymbol{x}^* \approx (4.43 \times 10^{-2}, 4.25)$ is returned by SABBa but the whole frontier from $(0.01, 5.1)$ to $(0.07, 4)$ is of high interest.

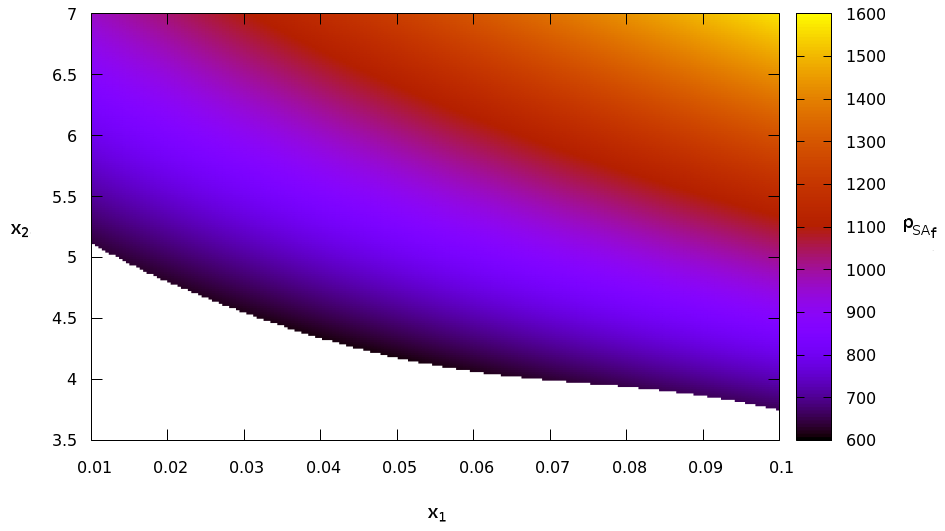


Figure 20: Final SA model, representing $\boldsymbol{\rho}_{SA_f}(\boldsymbol{x})$ where the constraint $\boldsymbol{\rho}_{SA_g}(\boldsymbol{x}) \leq 473.15$ K is satisfied.

7. Conclusions

In this work, we present an efficient framework, SABBa, for constrained optimisation under uncertainty problems. It handles most classical robustness and reliability measures, such as the Taguchi (mean and variance) robust optimisation or the quantile constrained optimisation problems. The parsimony of SABBa relies on several features presented in this paper. The Bounding-Box measure approximation is used to compare designs in a tunable accuracy context. It permits to compute the Pareto Optimal Probability (POP) associated to each design to perform a rigorous ranking among them. Gaussian Process (GP) surrogate models are exploited both for measure estimation and Surrogate-Assisting strategy, and allows to tract the estimation variability throughout the framework.

We have assessed the performance of the framework against a more classical A Priori MetaModel (APMM) strategy on several analytical test cases. To this extent, we have formulated a specific indicator based on a modified Hausdorff distance. Overall, SABBa shows both a faster convergence rate and a dramatic increase in the robustness of the optimisation process.

The proposed method has then been employed on three engineering test-cases to demonstrate its applicability for solving real-world uncertainty-based optimization problems with different reliability and robustness metrics. It has been successfully applied to a low-quantile reliability-based structural optimisation with six dimensions in the coupled space. The accuracy is very satisfactory, with an associated computational cost of 40 to 50 evaluations. It has also been employed for obtaining a bi-objective Pareto front approximation with the aim of minimising shocks past ORC turbine blades, that may induce significant losses. This constrained bi-objective Pareto front approximation only required around 600 function evaluations. Finally, a 12-dimensional Thermal Protection System (TPS) design with worst-case constraints has been performed, with a high impact of the tunable accuracy setting, at the cost of only 40 evaluations.

This approach remains very general and broadly applicable to any optimisation process with statistics-based objectives and constraints. It shows very good parsimony, which is of primary importance for real world applications, where black-box evaluations can take several days to run. Most importantly, the optimiser is not imposed by the approach, allowing for an easy coupling with any existing method, given that it is able to deal with noised estimations of the cost functions.

Several steps in this work remain improvable. We used local refinement criteria, which are cheaper but less efficient than integral criteria. The use of GP surrogate models also limits the number of manageable input dimensions. Surrogate modelling on over few dozens of dimensions would require specific techniques such as feature selection or kernel adaptation. Likewise, in order to efficiently deal with very low probability constraints (*e.g.* 1×10^{-7}), one should couple SABBa to a dedicated tool for low quantile computation. Finally, the proposed method could benefit from multi-output GP models for taking advantage of dependencies between quantities of interest or statistical measures.

On-going work aims at proposing non-uniform error estimations so as to more closely represent the measure uncertainties.

Appendix A. POP computational details

The set of boxes $\boldsymbol{\mathcal{B}} = \left\{\mathcal{B}_i\right\}_i$ is given, with $\mathcal{B}_i = \mathcal{B}\left(\boldsymbol{a}_i, \boldsymbol{r}_i\right)$. Every $\boldsymbol{Z}_i \sim \mathcal{U}\left(\mathcal{B}_i\right)$ is of dimension $m = m_1 + m_2$ where m_1 is the number of objectives and m_2 the number of constraints. We write $\boldsymbol{Z}_{i_{\boldsymbol{f}}}$ the objectives values and $\boldsymbol{Z}_{i_{\boldsymbol{g}}}$ the constraints values. The probability in Equation (5) is computed as follows, using the independence assumptions between boxes

and between dimensions:

$$\begin{aligned}\mathbb{P}_{\boldsymbol{Z}_j,\boldsymbol{Z}_i}\big[\boldsymbol{Z}_j \not\succ_c \boldsymbol{Z}_i\big] &= 1 - \mathbb{P}_{\boldsymbol{Z}_j,\boldsymbol{Z}_i}\big[\boldsymbol{Z}_j \succ_c \boldsymbol{Z}_i\big] \\ &= 1 - \Big(\mathbb{P}_{\boldsymbol{Z}_{i_g}}\big[\boldsymbol{Z}_{i_g} \not\leq \boldsymbol{0}\big] \\ &\quad + \mathbb{P}_{\boldsymbol{Z}_{i_g}}\big[\boldsymbol{Z}_{i_g} \leq \boldsymbol{0}\big]\mathbb{P}_{\boldsymbol{Z}_{j_g}}\big[\boldsymbol{Z}_{j_g} \leq \boldsymbol{0}\big]\mathbb{P}_{\boldsymbol{Z}_{j_f},\boldsymbol{Z}_{i_f}}\big[\boldsymbol{Z}_{j_f} \succ \boldsymbol{Z}_{i_f}\big]\Big) \\ &= \mathbb{P}_{\boldsymbol{Z}_{i_g}}\big[\boldsymbol{Z}_{i_g} \leq \boldsymbol{0}\big]\Big(1 - \mathbb{P}_{\boldsymbol{Z}_{j_g}}\big[\boldsymbol{Z}_{j_g} \leq \boldsymbol{0}\big]\mathbb{P}_{\boldsymbol{Z}_{j_f},\boldsymbol{Z}_{i_f}}\big[\boldsymbol{Z}_{j_f} \succ \boldsymbol{Z}_{i_f}\big]\Big)\end{aligned} \tag{A.1}$$

where

$$\mathbb{P}_{\boldsymbol{Z}_{i_g}}\big[\boldsymbol{Z}_{i_g} \leq \boldsymbol{0}\big] = \prod_{k=m_1+1}^{m} \max\Big(0, \min\Big(1, \frac{-\mathcal{B}_{i_k}^-}{2r_{i_k}}\Big)\Big) = \prod_{k=m_1+1}^{m} \Big[\frac{-\mathcal{B}_{i_k}^-}{2r_{i_k}}\Big]_0^1,$$

and $\big[\,\cdot\,\big]_0^1$ means that values are taken between 0 and 1. The lower bound of the box is written as $\mathcal{B}_i^- = \boldsymbol{a}_i - \boldsymbol{r}_i$. The second probability involved in the last line of Equation (A.1) is computed as follows:

$$\begin{aligned}\mathbb{P}_{\boldsymbol{Z}_{j_f},\boldsymbol{Z}_{i_f}}\big[\boldsymbol{Z}_{j_f} \succ \boldsymbol{Z}_{i_f}\big] &= \prod_{k=1}^{m_1} \mathbb{P}_{Z_{j_k},Z_{i_k}}\big[Z_{j_k} \leq Z_{i_k}\big] \\ &= \prod_{k=1}^{m_1} \left(\Big[\frac{L_{1_k}}{2r_{j_k}}\Big]_0^1 + \Big[\frac{L_{2_k}}{2r_{j_k}}\Big]_0^1 \left(\frac{1}{2}\Big[\frac{L_{2_k}}{2r_{j_k}}\Big]_0^1 + \Big[\frac{L_{3_k}}{2r_{j_k}}\Big]_0^1\right)\right),\end{aligned}$$

where

$$\begin{aligned}L_{1_k} &= \mathcal{B}_{i_k}^- - \mathcal{B}_{j_k}^-, \\ L_{2_k} &= \min(\mathcal{B}_{i_k}^+, \mathcal{B}_{j_k}^+) - \max(\mathcal{B}_{i_k}^-, \mathcal{B}_{j_k}^-), \\ L_{3_k} &= \mathcal{B}_{i_k}^+ - \mathcal{B}_{j_k}^+.\end{aligned}$$

In the above, $\mathcal{B}_i^+ = \boldsymbol{a}_i + \boldsymbol{r}_i$ is the upper bound of the box.

In each dimension, L_{1_k} can be interpreted as the portion of $\mathcal{B}_j$ dominating $\mathcal{B}_i$, L_{3_k} the portion of $\mathcal{B}_i$ dominated by $\mathcal{B}_j$ and L_{2_k} the overlapping area. We depict these lengths in Fig. A.21. Note that these values can be negative and that the computation is not symmetric between $\mathcal{B}_j$ and $\mathcal{B}_i$.

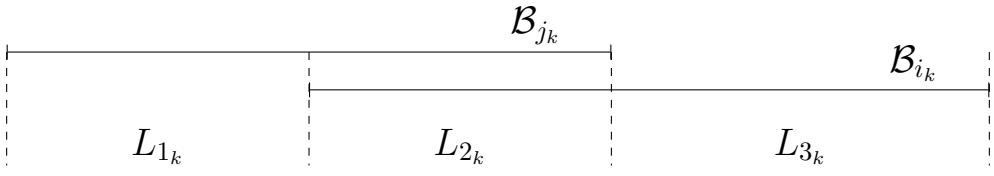


Figure A.21: Computational details

Appendix B. Justification for box sizes

Some developments are given here to explain the choice of the conservative error in Equation (6) associated with measure approximations.

Note that in the following, empirical estimators are considered well converged, so that all impreciseness comes from the surrogate model $\widehat{q}_{\boldsymbol{x}}$ of $q(\boldsymbol{x}, \cdot)$. For example, $\widetilde{\mu}(\boldsymbol{x})$ is considered equal to $\mathbb{E}_{\boldsymbol{\xi}}[\widehat{q}_{\boldsymbol{x}}(\boldsymbol{\xi})]$ in the following formula although only the empirical value is used.

The expectation approximation conservative error $\overline{\varepsilon}_{\mu}$ is quite straightforward:

$$\begin{aligned}
\left|\varepsilon_{\mu}(\boldsymbol{x})\right| = \left|\boldsymbol{\mu}(\boldsymbol{x}) - \widetilde{\boldsymbol{\mu}}(\boldsymbol{x})\right| &= \left|\mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})] - \mathbb{E}_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi})]\right| \\
&= \left|\mathbb{E}_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \varepsilon_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})] - \mathbb{E}_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi})]\right| \\
&= \left|\mathbb{E}_{\boldsymbol{\xi}}[\varepsilon_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})]\right| \\
&\leq \mathbb{E}_{\boldsymbol{\xi}}[\overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})] \\
&= \overline{\varepsilon}_{\mu}(\boldsymbol{x})
\end{aligned}$$

with $\boldsymbol{\mu}(\boldsymbol{x})$ the true statistical moment on $\boldsymbol{q}$ at a given $\boldsymbol{x}$ and $\widetilde{\boldsymbol{\mu}}(\boldsymbol{x})$ the approximated one, computed on the surrogate model $\widehat{\boldsymbol{q}}_{\boldsymbol{x}}$. The true error $\varepsilon_{\boldsymbol{q}_{\boldsymbol{x}}}$ is defined from $\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi}) = \widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \varepsilon_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})$ and is conservatively approximated by $\overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi}) \geq |\varepsilon_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})|$.

The same can be conducted for the variance measure, with a bit more risks of overestimation:

$$\begin{aligned}
\left|\varepsilon_{\sigma^2}(\boldsymbol{x})\right| = \left|\boldsymbol{\sigma}^{\boldsymbol{2}}(\boldsymbol{x}) - \widetilde{\boldsymbol{\sigma}^{\boldsymbol{2}}}(\boldsymbol{x})\right| &= \left|\mathbb{E}_{\boldsymbol{\xi}}[(\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi}) - \boldsymbol{\mu}(\boldsymbol{x}))^2] - \mathbb{E}_{\boldsymbol{\xi}}[(\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\mu}}(\boldsymbol{x}))^2]\right| \\
&= \left|\mathbb{E}_{\boldsymbol{\xi}}[(\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\mu}}(\boldsymbol{x}) + \varepsilon_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi}) - \varepsilon_{\mu}(\boldsymbol{x}))^2] - \mathbb{E}_{\boldsymbol{\xi}}[(\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\mu}}(\boldsymbol{x}))^2]\right| \\
&= \left|\mathbb{E}_{\boldsymbol{\xi}}[(\varepsilon_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi}) - \varepsilon_{\mu}(\boldsymbol{x}))^2 + 2(\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\mu}}(\boldsymbol{x}))(\varepsilon_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi}) - \varepsilon_{\mu}(\boldsymbol{x}))]\right| \\
&\leq \mathbb{E}_{\boldsymbol{\xi}}[(\overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi}) + \overline{\varepsilon}_{\mu}(\boldsymbol{x}))^2 + 2|\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\mu}}(\boldsymbol{x})|(\overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi}) + \overline{\varepsilon}_{\mu}(\boldsymbol{x}))] \\
&= \overline{\varepsilon}_{\sigma^2}(\boldsymbol{x})
\end{aligned}$$

The conservative error associated with the expectation approximation could also be explained based on the monotonicity of the expectation operator. We write $f \geq g \iff \forall \boldsymbol{x}, f(\boldsymbol{x}) \geq g(\boldsymbol{x})$. By definition $\boldsymbol{q}(\boldsymbol{x}, \cdot) \in [\widehat{\boldsymbol{q}}_{\boldsymbol{x}} - \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}}, \widehat{\boldsymbol{q}}_{\boldsymbol{x}} + \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}}]$. The monotonicity of the expectation operator then gives $\mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})] \in \left[\mathbb{E}_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})], \mathbb{E}_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})]\right]$, or equivalently $\boldsymbol{\mu}(\boldsymbol{x}) \in \left[\widetilde{\boldsymbol{\mu}}(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\xi}}[\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})], \widetilde{\boldsymbol{\mu}}(\boldsymbol{x}) + \mathbb{E}_{\boldsymbol{\xi}}[\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})]\right]$, thus giving the conservative error found in the above $\overline{\boldsymbol{\varepsilon}}_{\mu}(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})]$.

By defining $\widehat{\boldsymbol{q}}^{+}_{\boldsymbol{x}}(\boldsymbol{\xi}) = \widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}\boldsymbol{x}}(\boldsymbol{\xi})$ and $\widehat{\boldsymbol{q}}^{-}_{\boldsymbol{x}}(\boldsymbol{\xi}) = \widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{q}\boldsymbol{x}}(\boldsymbol{\xi})$, the monotonicity of the minimum, maximum and quantile operators can also be exploited to obtain conservative error approximations. For the case of the minimum operator, with $\widetilde{\boldsymbol{m}} = \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi})]$,

$$\begin{aligned}
\boldsymbol{m}(\boldsymbol{x}) = \min_{\boldsymbol{\xi}}[\boldsymbol{q}(\boldsymbol{x}, \boldsymbol{\xi})] &\in \left[\min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^{-}_{\boldsymbol{x}}(\boldsymbol{\xi})], \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^{+}_{\boldsymbol{x}}(\boldsymbol{\xi})]\right] \\
&= \left[\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \left|\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^{-}_{\boldsymbol{x}}(\boldsymbol{\xi})]\right|, \widetilde{\boldsymbol{m}}(\boldsymbol{x}) + \left|\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}^{+}_{\boldsymbol{x}}(\boldsymbol{\xi})]\right|\right] \\
&\in \left[\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \overline{\varepsilon}_{\min}(\boldsymbol{x}), \widetilde{\boldsymbol{m}}(\boldsymbol{x}) + \overline{\varepsilon}_{\min}(\boldsymbol{x})\right]
\end{aligned}$$

where $\overline{\varepsilon}_{\min}(\boldsymbol{x}) = \max\left(\left|\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi})]\right|, \left|\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi})]\right|\right)$.

The same idea can be followed for the maximum and quantile approximations, resulting in the following errors:

$$\overline{\varepsilon}_{\max}(\boldsymbol{x}) = \max\left(\left|\widetilde{\boldsymbol{M}}(\boldsymbol{x}) - \max_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi})]\right|, \left|\widetilde{\boldsymbol{M}}(\boldsymbol{x}) - \max_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi})]\right|\right)$$
$$\overline{\varepsilon}_{\mathrm{q}^p}(\boldsymbol{x}) = \max\left(\left|\widetilde{\mathrm{q}^p}(\boldsymbol{x}) - \mathrm{q}^p_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi})]\right|, \left|\widetilde{\mathrm{q}^p}(\boldsymbol{x}) - \mathrm{q}^p_{\boldsymbol{\xi}}[\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi})]\right|\right)$$

Appendix C. Justification for refinement criteria

The chosen partial criteria for GP model refinement given in Equation (7) are also explained in the following.

Both the expectation and variance are global measures over the whole domain. For this reason, it has been chosen to iteratively add a point at the maximum predictive conservative error, to converge the model on the entire space. This strategy is usually called Maximum Mean Square Predictive Error or MMSPE. Note that the partial criteria are multiplied by the input pdf to weight the predictive error according to the probability of occurrence. Thus, the final partial criteria are:

$$\boldsymbol{c}_{\mu}(\boldsymbol{\xi}) = \overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})\phi(\boldsymbol{\xi})$$
$$\boldsymbol{c}_{\sigma^2}(\boldsymbol{\xi}) = \overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})\phi(\boldsymbol{\xi})$$

with ϕ the Probability Density Function (PDF) of the uncertain variables.

The minimum (resp maximum) measure partial criteria is simply the probability of exceeding the current minimal (resp. maximal) value $\widetilde{\boldsymbol{m}}$ (resp. $\widetilde{\boldsymbol{M}}$). This is performed with an assumption of uniform distribution with the conservative error box. Hence, the criteria can be written as such:

$$\boldsymbol{c}_{\min}(\boldsymbol{\xi}) = \left[\frac{\widetilde{\boldsymbol{m}} - \widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi})}{2\overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})}\right]_+$$
$$\boldsymbol{c}_{\max}(\boldsymbol{\xi}) = \left[\frac{\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{M}}}{2\overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})}\right]_+$$

where $\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi}) = \widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \overline{\varepsilon}_{\boldsymbol{q}\boldsymbol{x}}(\boldsymbol{\xi})$ and $\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi}) = \widehat{\boldsymbol{q}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \overline{\varepsilon}_{\boldsymbol{q}\boldsymbol{x}}(\boldsymbol{\xi})$ and with $\left[\,\cdot\,\right]_+ = \max(0, \cdot)$ referring to the value if positive, 0 either.

Finally, for the case of the quantile measure, it has been chosen to compute the product of the aforementioned probability of exceeding the quantile value, multiplied by the input density:

$$\boldsymbol{c}_{\mathrm{q}^p}(\boldsymbol{\xi}) = \left[\frac{\widetilde{\mathrm{q}^p} - \widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi})}{2\overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})}\right]_+ \left[\frac{\widehat{\boldsymbol{q}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi}) - \widetilde{\mathrm{q}^p}}{2\overline{\varepsilon}_{\boldsymbol{q}_{\boldsymbol{x}}}(\boldsymbol{\xi})}\right]_+ \phi(\boldsymbol{\xi})$$

This product is maximised on the hyperplane $\widehat{\boldsymbol{q}}_{\boldsymbol{x}} = \mathrm{q}^p$. Multiplying the criteria by $\phi(\boldsymbol{\xi})$ puts more weight according to the probability of occurrence. The spread may not be optimal,

but the optimisation of the criteria being performed on a fixed sampling, the tightening of the conservative error is very likely to decrease the value of the criterion in the surrounding area because samples are very unlikely to be localised exactly on the isoline of the function.

We can note that the chosen criteria are far from being optimal. However, these choices give a fast determination of the refinement point, and the number of samples is usually low enough so that the choice of the training point is not of significant importance compared to the quality of the metamodelling strategy.

Bibliography

References

[1] S. Jamshed, Chapter 4 - high reynolds number flows, in: S. Jamshed (Ed.), Using HPC for Computational Fluid Dynamics, Academic Press, Oxford, 2015, pp. 81–100. doi:https://doi.org/10.1016/B978-0-12-801567-4.00004-0.
URL https://www.sciencedirect.com/science/article/pii/B9780128015674000040

[2] J.-O. Lee, Y.-S. Yang, W.-S. Ruy, A comparative study on reliability-index and target-performance-based probabilistic structural design optimization, Computers & Structures 80 (3) (2002) 257 – 269. doi:https://doi.org/10.1016/S0045-7949(02)00006-8.
URL http://www.sciencedirect.com/science/article/pii/S0045794902000068

[3] Q. Zhao, X. Chen, Z. Ma, Y. Lin, A comparison of deterministic, reliability-based topology optimization under uncertainties, Acta Mechanica Solida Sinica 29 (1) (2016) 31 – 45. doi:https://doi.org/10.1016/S0894-9166(16)60005-8.
URL http://www.sciencedirect.com/science/article/pii/S0894916616600058

[4] A. Chaudhuri, B. Kramer, K. E. Willcox, Information reuse for importance sampling in reliability-based design optimization, Reliability Engineering & System Safety 201 (2020) 106853.

[5] W.-S. Liu, S. H. Cheung, Reliability based design optimization with approximate failure probability function in partitioned design space, Reliability Engineering & System Safety 167 (2017) 602–611.

[6] H. Jensen, M. Valdebenito, G. Schuëller, D. Kusanovic, Reliability-based optimization of stochastic systems using line search, Computer Methods in Applied Mechanics and Engineering 198 (49) (2009) 3915 – 3924. doi:https://doi.org/10.1016/j.cma.2009.08.016.
URL http://www.sciencedirect.com/science/article/pii/S0045782509002710

[7] M. Valdebenito, G. Schuëller, Efficient strategies for reliability-based optimization involving non-linear, dynamical structures, Computers & Structures 89 (19) (2011) 1797 – 1811, civil-Comp. doi:https://doi.org/10.1016/j.compstruc.2010.10.014.
URL http://www.sciencedirect.com/science/article/pii/S0045794910002464

[8] M. Papadrakakis, N. D. Lagaros, Reliability-based structural optimization using neural networks and monte carlo simulation, Computer Methods in Applied Mechanics and Engineering 191 (32) (2002) 3491 – 3507. doi:https://doi.org/10.1016/S0045-7825(02)00287-6.
URL http://www.sciencedirect.com/science/article/pii/S0045782502002876

[9] V. Keshavarzzadeh, F. Fernandez, D. A. Tortorelli, Topology optimization under uncertainty via non-intrusive polynomial chaos expansion, Computer Methods in Applied Mechanics and Engineering 318 (2017) 120 – 147. doi:https://doi.org/10.1016/j.cma.2017.01.019.
URL http://www.sciencedirect.com/science/article/pii/S0045782516313019

[10] A. J. Torii, R. H. Lopez, L. F. F. Miguel, A second order sap algorithm for risk and reliability based design optimization, Reliability Engineering & System Safety 190 (2019) 106499.

[11] R. Schöbi, B. Sudret, S. Marelli, Rare event estimation using polynomial-chaos kriging, ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering 3 (2) (2017) D4016002. arXiv:https://ascelibrary.org/doi/pdf/10.1061/AJRUA6.0000870, doi:10.1061/AJRUA6.0000870.
URL https://ascelibrary.org/doi/abs/10.1061/AJRUA6.0000870

[12] P. Ni, J. Li, H. Hao, W. Yan, X. Du, H. Zhou, Reliability analysis and design optimization of nonlinear structures, Reliability Engineering & System Safety 198 (2020) 106860.

[13] M. Li, Z. Wang, Surrogate model uncertainty quantification for reliability-based design optimization, Reliability Engineering & System Safety 192 (2019) 106432.

[14] J. Bect, D. Ginsbourger, L. Li, V. Picheny, E. Vazquez, Sequential design of computer experiments for the estimation of a probability of failure, Statistics and Computing 22 (3) (2012) 773–793. doi:10.1007/s11222-011-9241-4.
URL https://doi.org/10.1007/s11222-011-9241-4

[15] H. Kroetz, M. Moustapha, A. T. Beck, B. Sudret, A two-level kriging-based approach with active learning for solving time-variant risk optimization problems, Reliability Engineering & System Safety 203 (2020) 107033.

[16] M. Moustapha, B. Sudret, J.-M. Bourinet, B. Guillaume, Quantile-based optimization under uncertainties using adaptive kriging surrogate models, Structural and multidisciplinary optimization 54 (6) (2016) 1403–1421.

[17] J.-M. Bourinet, Reliability assessment by adaptive kernel-based surrogate models-approximation of non-smooth limit-state functions (2019).

[18] A. A. Taflanidis, J. L. Beck, An efficient framework for optimal robust stochastic system design using stochastic simulation, Computer Methods in Applied Mechanics and Engineering 198 (1) (2008) 88 – 101.
URL http://resolver.caltech.edu/CaltechAUTHORS:TAFcmame08

[19] J. C. Medina, A. A. Taflanidis, Adaptive importance sampling for optimization under uncertainty problems, Computer Methods in Applied Mechanics and Engineering 279 (2014) 133 – 162. doi:https://doi.org/10.1016/j.cma.2014.06.025.
URL http://www.sciencedirect.com/science/article/pii/S0045782514002114

[20] R. Jin, X. Du, W. Chen, The use of metamodeling techniques for optimization under uncertainty, Structural and Multidisciplinary Optimization 25 (2) (2003) 99–116. doi:10.1007/s00158-002-0277-0.
URL https://doi.org/10.1007/s00158-002-0277-0

[21] J. Zhang, A. Taflanidis, J. Medina, Sequential approximate optimization for design under uncertainty problems utilizing kriging metamodeling in augmented input space, Computer Methods in Applied Mechanics and Engineering 315 (2017) 369 – 395. doi:https://doi.org/10.1016/j.cma.2016.10.042.
URL http://www.sciencedirect.com/science/article/pii/S0045782516314529

[22] K.-H. Lee, G.-J. Park, A global robust optimization using kriging based approximation model, JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing 49 (3) (2006) 779–788. doi:10.1299/jsmec.49.779.

[23] G. Dellino, J. P. C. Kleijnen, C. Meloni, Robust optimization in simulation: Taguchi and krige combined, INFORMS Journal on Computing 24 (3) (2012) 471–484. arXiv:https://doi.org/10.1287/ijoc.1110.0465, doi:10.1287/ijoc.1110.0465.
URL https://doi.org/10.1287/ijoc.1110.0465

[24] M. Eldred, A. Giunta, S. Wojtkiewicz, T. Trucano, Formulations for Surrogate-Based Optimization Under Uncertainty, in: 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Multidisciplinary Analysis Optimization Conferences, American Institute of Aeronautics and Astronautics, 2002. doi:10.2514/6.2002-5585.
URL https://doi.org/10.2514/6.2002-5585

[25] J. Janusevskis, R. Le Riche, Simultaneous kriging-based estimation and optimization of mean response, Journal of Global Optimization 55 (2) (2013) 313–336. doi:10.1007/s10898-011-9836-5.
URL https://doi.org/10.1007/s10898-011-9836-5

[26] R. Le Riche, V. Picheny, A. Meyer, N.-H. Kim, D. Ginsbourger, Gears design with shape uncertainties using controlled monte carlo simulations and kriging, in: 50th

AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 17th AIAA/ASME/AHS Adaptive Structures Conference 11th AIAA No, 2009, p. 2257.

[27] J. Fang, Y. Gao, G. Sun, C. Xu, Q. Li, Multiobjective robust design optimization of fatigue life for a truck cab, Reliability Engineering & System Safety 135 (2015) 1–8.

[28] M. Ribaud, C. Blanchet-Scalliet, C. Helbert, F. Gillot, Robust optimization: a kriging-based multi-objective optimization approach, Reliability Engineering & System Safety 200 (2020) 106913.

[29] M. Binois, D. Ginsbourger, O. Roustant, Quantifying uncertainty on pareto fronts with gaussian process conditional simulations, European journal of operational research 243 (2) (2015) 386–394.

[30] M. Menz, S. Dubreuil, J. Morio, C. Gogu, N. Bartoli, M. Chiron, Variance based sensitivity analysis for monte carlo and importance sampling reliability assessment with gaussian processes, arXiv preprint arXiv:2011.15001 (2020).

[31] S. Ferson, P. Van Den Brink, T. Estes, K. Gallagher, R. O'Connor, F. Verdonck, Bounding uncertainty analyses, Application of uncertainty analysis to ecological risks of pesticides (2010).

[32] T. J. Barth, On the calculation of uncertainty statistics with error bounds for cfd calculations containing random parameters and fields (2016).

[33] J. Teich, Pareto-Front Exploration with Uncertain Objectives, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001, pp. 314–328. doi:10.1007/3-540-44719-9_22.
URL http://dx.doi.org/10.1007/3-540-44719-9_22

[34] M. Mlakar, T. Tusar, B. Filipic, Comparing solutions under uncertainty in multiobjective optimization, Mathematical Problems in Engineering 2014 (2014) 1–10. doi:10.1155/2014/817964.
URL http://dx.doi.org/10.1155/2014/817964

[35] F. Fusi, P. M. Congedo, An adaptive strategy on the error of the objective functions for uncertainty-based derivative-free optimization, Journal of Computational Physics 309 (2016) 241–266. doi:10.1016/j.jcp.2016.01.004.
URL https://hal.inria.fr/hal-01251604

[36] F. Khosravi, A. Raß, J. Teich, Efficient computation of probabilistic dominance in robust multi-objective optimization, arXiv preprint arXiv:1910.08413 (2019).

[37] M. Rivier, P. M. Congedo, Surrogate-assisted bounding-box approach for optimization problems with tunable objectives fidelity, Journal of Global Optimization. (Aug 2019). doi:10.1007/s10898-019-00823-9.
URL https://doi.org/10.1007/s10898-019-00823-9

[38] S. Le Digabel, Nomad: Nonlinear optimization with the mads algorithm 37 (2011) 44.

[39] GPy, GPy: A gaussian process framework in python, http://github.com/SheffieldML/GPy (since 2012).

[40] C. E. Rasmussen, Gaussian processes in machine learning, in: Advanced lectures on machine learning, Springer, 2004, pp. 63–71.

[41] M. P. Dubuisson, A. K. Jain, A modified hausdorff distance for object matching, in: Proceedings of 12th International Conference on Pattern Recognition, Vol. 1, 1994, pp. 566–568 vol.1. doi:10.1109/ICPR.1994.576361.

[42] V. Baudoui, Optimisation robuste multiobjectifs par modèles de substitution, Ph.D. thesis, thèse de doctorat dirigée par Hiriart-Urruty, Jean-Baptiste et Harran-Klotz, Patricia Mathématiques appliquées et énergétique et transferts Toulouse, ISAE 2012 (2012).
URL http://www.theses.fr/2012ESAE0007

[43] N. Razaaly, G. Persico, G. Gori, P. M. Congedo, Quantile-based robust optimization of a supersonic nozzle for organic rankine cycle turbines, Applied Mathematical Modelling 82 (2020) 802–824. doi:https://doi.org/10.1016/j.apm.2020.01.048.
URL https://www.sciencedirect.com/science/article/pii/S0307904X20300482

[44] J. Lachaud, N. N. Mansour, Porous material analysis toolbox based on openfoam and applications, Journal of Thermophysics and Heat Transfer 28 (2) (2014) 191–202, doi: 10.2514/1.T4262.

[45] M. Rivier, J. Lachaud, P. Congedo, Ablative thermal protection system under uncertainties including pyrolysis gas composition, Aerospace Science and Technology 84 (2019) 1059 – 1069. doi:https://doi.org/10.1016/j.ast.2018.11.048.
URL http://www.sciencedirect.com/science/article/pii/S127096381831472X