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

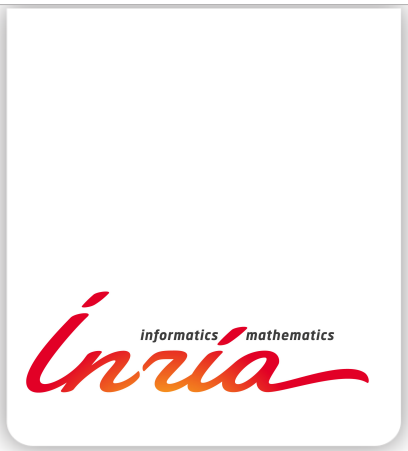
Mickael Rivier, Pietro Marco Congedo

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

Mickaël Rivier , Pietro Marco Congedo

RESEARCH REPORT
N° 9214
September 12, 2019
Project-Teams DEFI

ISSN 0249-6399 ISRN INRIA/RR--9214--FR+ENG



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

Mickaël Rivier *†, Pietro Marco Congedo *

Project-Teams DEFI

Research Report n° 9214 — September 12, 2019 — 37 pages

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 robustness and reliability measures, which can be computed with tunable and refinable fidelity.
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 fidelity of these estimations can be tuned throughout the optimisation so as to reach high accuracy 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 is very useful to reduce the overall computational cost. The adaptive refinement within the Bounding-Box approach is based on the computation of a Pareto Optimal Probability (POP) for 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 two engineering applications: the design of two-bar truss in structural mechanics and the design of a thermal protection system for atmospheric reentry.

Key-words: Multi-objective optimisation, Uncertainty-based optimisation, Error bounding boxes, Imprecise Pareto front, Surrogate-assisting strategy

* DeFI - CMAP - Ecole Polytechnique, Inria Saclay - Ile de France, Polytechnique - X, CNRS
† ArianeGroup, Le Haillan

RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE

1 rue Honoré d'Estienne d'Orves
Bâtiment Alan Turing
Campus de l'École Polytechnique
91120 Palaiseau

Le framework SABBa appliqué à l'optimisation contrainte multi-objectif sous incertitudes

Résumé : Ce travail se concentre sur les problèmes d'optimisation sous incertitudes multi-objectif. Le framework SABBa (Surrogate-Assisted Bounding-Box approach) est proposé pour traiter les mesures de robustesse et de fiabilité, qui peuvent être calculées avec une fidélité adaptable et raffinable.

Une Boîte conservative est définie comme un produit multi-dimensionnel d'intervalles centrés sur les objectifs et contraintes estimés, de manière à contenir leur valeur réelle. La fidélité de ces estimations peut être ajustée au fur et à mesure du processus d'optimisation de manière à atteindre des estimations haute-fidélité uniquement sur les individus les plus prometteurs. Dans SABBa, cette approche est couplée à une stratégie d'assistance par modèle de substitution, qui permet de réduire le coût global du processus d'optimisation. Nous proposons de guider le raffinement adaptatif des Boîtes conservatives par le calcul de la probabilité pour chaque boîte d'être Pareto-optimale.

La méthode proposée est d'abord appliquée à plusieurs cas-tests analytiques et comparée à un approche par modèle de substitution *a priori*. Un indicateur par distance de Hausdorff modifiée probabiliste est proposée pour comparer les différentes stratégies. SABBa est ensuite appliqué sur deux cas ingénierie : une structure à deux poutres et un système de protection thermique ablatif pour réentrée atmosphérique.

Mots-clés : Optimisation multi-objectif, Optimisation sous incertitudes, Boîte d'erreur conservative, Front de Pareto imprécis, Assistance par modèle de substitution

1 Introduction

The topic of optimisation under uncertainty is of particular interest for companies nowadays as (i) the always increasing computational power now allows for multiple runs and reanalysis of simulations and (ii) 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 Optimisation (RBO). One of the central issues in conducting such analysis is the computational cost of attaining a prescribed level of accuracy. Most of the time, these optimisations are performed on numerical simulations, with run times ranging from minutes to weeks. This prohibits direct simulation-based methods such as nested Monte Carlo Simulations (MCS), performing a full Uncertainty Propagation (UP) at each optimisation iteration.

Contributions to reliability-based optimisation mostly rely on the computation of failure probability, such as the First-Order Reliability Method (FORM), and the Performance Measure Approach (PMA) [1, 2]. Other decoupled approaches are based on local approximations of the failure probability and the associated gradient to perform line search optimisation, as in [3], based on Subset Simulation (SS) computations, or in [4], supplemented by a stabilisation strategy. The idea of using a surrogate model to tackle the computational cost limitation has been extensively used with Neural Networks [5], Polynomial Chaos Expansion (PCE) [6] and PCE-Kriging metamodels [7]. Finally, in [8], a Stepwise Uncertainty Reduction (SUR) strategy for failure probability estimation is proposed, based on the volume of the excursion set. Other approaches to mention are Ref. [9] and [10], based on "safety factor" constraints and Extreme Value Distribution (EVD) respectively.

In literature, there are also several contributions to the topic of Robust Optimisation, in particular applied to structural optimisation and Topological Optimisation (TO). Taguchi's robustness paradigm provides a typical RO formulation. It aims at maximising the performance (or mean performance) whilst minimising the associated variance. Refs. [11, 12, 13, 14] are examples of this paradigm for RTO problems. To perform mean performance optimisation, Ref. [15] introduces the Stochastic Subset Optimisation (SSO), which is then supplemented by a local optimisation through Simultaneous Perturbation Stochastic Approximation (SPSA). SPSA was also used in [16], coupled with an adaptive Importance Sampling (IS) strategy. Worst-case optimisation has been performed with monotonic assumptions in [17] and [18]. Surrogate-based methodologies are also massively exploited in the area of Robust Optimisation, for example with PCE [6], and kriging [19]. A general discussion over the use of metamodels in the context of uncertainty-based optimisation is proposed in [20]. It shows that such methods yield promising results, notably the kriging or Gaussian Process (GP) surrogate models [21, 22, 23]. Some works take full advantage of the nature of Gaussian processes. In [24], the projection of a GP surrogate model is conducted analytically to perform mean performance optimisation. Optimisation in the design space is then performed through Bayesian Optimisation (BO). In [25], they exploit the ability of Gaussian process modelling to deal with heterogeneous noises in the observations. Finally, [26] uses conditional simulations in the context of multi-objective optimisation to sample possible Pareto optima. More generally, there have been extensions of GP-based optimisation (BO or Efficient Global Optimisation, EGO) to stochastic or noisy black-box evaluations, for example, with Sequential Kriging Optimisation (SKO) [27], and with the concept of Noisy Expected Improvement [28].

In several contributions [29, 30, 31], the authors introduced the concept of Bounding-Boxes and revisited the Pareto dominance rules. These concepts have been generalised in [32] to tackle noisy estimations with tunable fidelity, and supplemented by a Surrogate-Assisting strategy to obtain the parsimonious Surrogate-Assisted Bounding-Box approach (SABBa) framework. The

main contribution of the present paper is the extension of SABBa to the problem of optimisation under uncertainty, where both probabilistic objectives (RO) and probabilistic constraints (RBO) must be tackled. To this extent, we propose a *Constrained Boxed Pareto Dominance* rule to deal with estimated constraints. Parsimony is improved through the choice of GP surrogate models, allowing for natural computations of conservative errors and refinement criteria. In addition, the Pareto Optimal Probability (POP) is computed for each box to focus the refinements on boxes that are non-dominated with high probability. All numerical details for practical implementation and quantitative comparisons are given in the following. We show that the direct Double Loop approach, where a full Uncertainty Propagation (UP) is performed at each optimisation iteration, is extremely expensive and not competitive with respect to the so-called A Priori MetaModel (APMM) strategy. Hence, we systematically compare SABBa to the latest.

We depict a general uncertainty-based optimisation problem in Section 2, with some classical robustness and reliability measures. Then, we introduce the concept of Bounding-Boxes and Pareto Optimal Probability (POP) in Section 3. We extend the existing concepts from [29, 30, 31, 32] 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 Uncertainty-based optimisation: 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 RBO).

A deterministic constrained multi-objective optimisation problem can be described as follows:

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

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 $\boldsymbol{x} \in \mathcal{X}$ are the n design variables included in the design domain $\mathcal{X} \subset \mathbb{R}^n$.

In the presence of uncertain parameters $\boldsymbol{\xi}$ (*e.g.* environmental, material or geometrical parameters), the objective and constraint functions depend on both $\boldsymbol{x}$ and $\boldsymbol{\xi}$. Hence, $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi})$ must be replaced with robustness and reliability measures, $\boldsymbol{\rho_f}(\boldsymbol{x})$ and $\boldsymbol{\rho_g}(\boldsymbol{x})$ respectively. These allow the formulation of a general uncertainty-based constrained multi-objective optimisation problem, as follows:

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

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

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

$$\begin{aligned} \text{Expectation:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})], \\ \text{Variance:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \mathbb{V}_{\boldsymbol{\xi}}[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})], \\ \text{Min/Max:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \min_{\boldsymbol{\xi}}[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})] \text{ or } \max_{\boldsymbol{\xi}}[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})], \\ \text{Quantile:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \mathrm{q}^p_{\boldsymbol{\xi}}[\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})], \ p \in [0, 1], \end{aligned}$$

and the same for $\boldsymbol{\rho_g}$. Note that any other measure could also be treated similarly in the following.

Remark For conciseness, $\boldsymbol{\rho_f}$ and $\boldsymbol{\rho_g}$ are gathered in a vector $\boldsymbol{\rho}$,

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

The main issue here is the accurate computation of these quantities. In this work, we approximate statistical measures through Monte Carlo Simulations (MCS) on surrogate models. To improve the accuracy of these approximations, one can increase the training set used for building the surrogate.

Hence, the uncertainty-based optimisation problem reduces to a constrained multi-objective optimisation problem with tunable fidelity computations for all objectives and constraints. The concept of tunable fidelity in the unconstrained case has been introduced in [32] for multi-objective optimisation problems. Next section illustrates the concept of Bounding-Boxes and their applicability to constrained problems.

3 Bounding-Boxes

In this section, we introduce the concept of Bounding-Box. Notations and the *Boxed Pareto dominance* (introduced in [32]) are recalled. The extension to constrained optimisation problems is then illustrated in Section 3.1.2. Finally, the concept of Pareto Optimal Probability is presented in Section 4.1 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, the unknown $\boldsymbol{\rho}$ is approximated by $\widetilde{\boldsymbol{\rho}}^l$, l being the fidelity of the approximation. The exact approximation error $\boldsymbol{\varepsilon}^l = \boldsymbol{\rho} - \widetilde{\boldsymbol{\rho}}^l$ is conservatively approximated by $\overline{\boldsymbol{\varepsilon}}^l \geq |\boldsymbol{\varepsilon}^l|$. Hence it comes that $\boldsymbol{\rho} \in \mathcal{B}(\widetilde{\boldsymbol{\rho}}^l, \overline{\boldsymbol{\varepsilon}}^l)$ (see Fig. 1).

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 $\left(\mathcal{B}(\boldsymbol{a}, \boldsymbol{r}), \mathcal{B}(\boldsymbol{b}, \boldsymbol{r'})\right) \in \wp(\mathbb{R}^m)^2$, the *Boxed Pareto dominance* is defined as follows:

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

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

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

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

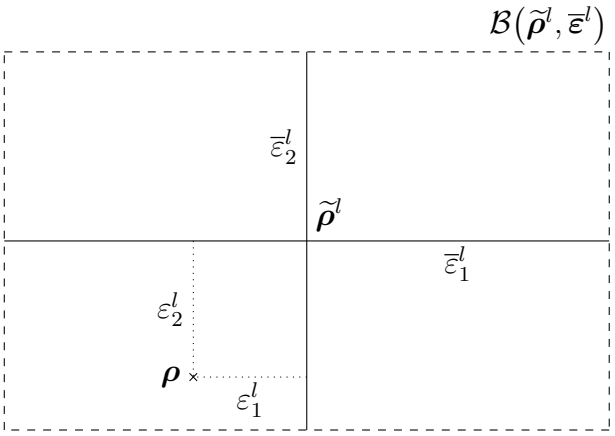


Figure 1: Bounding-Box approximation

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

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

$\forall j \in [\![1, m]\!]$, the symbol $\pm$ (implicitly $\pm_j$) indicates the goal in the j^{th} dimension:

$$\pm = \begin{cases} + & \text{for minimisation} \\ - & \text{for maximisation.} \end{cases}$$

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. In the case of bi-minimisation, the dominance are the following: $\mathcal{B}_1 \underset{\mathcal{B}}{\sim} \mathcal{B}_2$, $\mathcal{B}_1 \underset{\mathcal{B}}{\sim} \mathcal{B}_3$, $\mathcal{B}_1 \underset{\mathcal{B}}{\sim} \mathcal{B}_4$, $\mathcal{B}_2 \underset{\mathcal{B}}{\sim} \mathcal{B}_3$, $\mathcal{B}_2 \underset{\mathcal{B}}{\succ\!\!\succ} \mathcal{B}_4$, $\mathcal{B}_3 \underset{\mathcal{B}}{\sim} \mathcal{B}_4$. Hence, only $\mathcal{B}_1$, $\mathcal{B}_2$ and $\mathcal{B}_3$ are non-dominated.

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 first proposed, where again $\succ$ refers to the classical Pareto dominance.

Let us assume that elements to compare are in $\mathbb{R}^m$ and $m = m_1 + m_2$ with dimensions 1 to m_1 being the objective dimensions (subject to minimisation or maximisation) and the m_2 latest being the constraint dimensions (assumed as lower or upper bounds). We also introduce the vector $\boldsymbol{a}_{\mathcal{D}}$, with $\mathcal{D}$ a set of indices, which is the projection of $\boldsymbol{a}$ on the associated dimensions. Formally, $\boldsymbol{a}_{\mathcal{D}}$ is defined as:

$$\boldsymbol{a}_{\mathcal{D}} = \sum_{d \in \mathcal{D}} a_d \boldsymbol{e}_d,$$

with $\boldsymbol{e}_k$ the unit vector in the k-th direction.

In the multi-objective constrained problem, vector $\boldsymbol{a}_{[\![1,m_1]\!]}$ represents the objective values and $\boldsymbol{a}_{[\![m_1+1,m]\!]}$ the constraint values. Without loss of generality, these m_2 constraints can be considered of the form:

$$\forall j \in [\![m_1 + 1, m]\!], \ \pm a_j \leq 0,$$

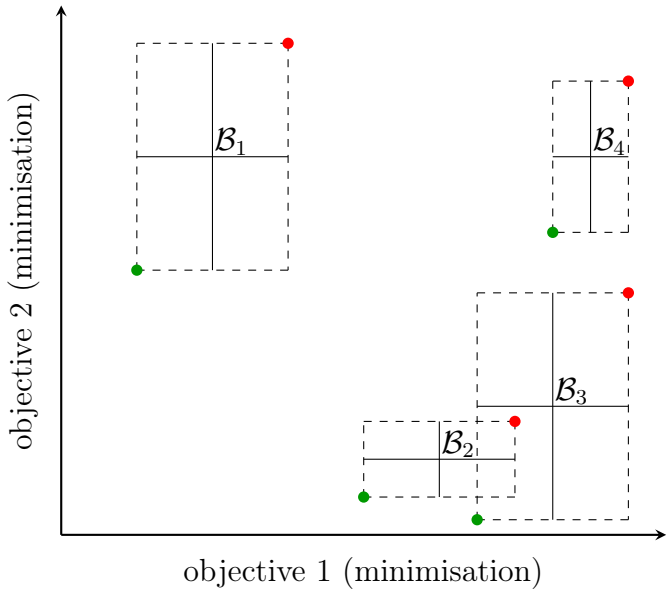


Figure 2: *Boxed Pareto Dominance*: Comparison of 4 boxes, best and worst outcomes in green and red, respectively.

where

$$\pm = \begin{cases} + & \text{for upper bounds} \\ - & \text{for lower bounds.} \end{cases}$$

The admissible and failure sets are then defined as follows:

$$\mathcal{A} = \left\{ \boldsymbol{a} \in \mathbb{R}^m \mid \forall j \in [\![m_1+1, m]\!],\ \pm a_j \leq 0 \right\},$$
$$\mathcal{F} = \left\{ \boldsymbol{a} \in \mathbb{R}^m \mid \exists j \in [\![m_1+1, m]\!],\ \pm a_j > 0 \right\}.$$

Contrarily to the classical Pareto dominance, the *Constrained Pareto dominance* compares boxes from the coupled objective/constraint spaces. The dominance rule when comparing two designs classically (without boxes) is defined as follows

$$\begin{aligned} \boldsymbol{a} \succ_c \boldsymbol{b} &\iff \boldsymbol{a}_{[\![1,m_1]\!]} \succ \boldsymbol{b}_{[\![1,m_1]\!]} \text{ and } \boldsymbol{a} \in \mathcal{A} \quad \text{or} \quad \boldsymbol{b} \in \mathcal{F} \\ \boldsymbol{a} \succcurlyeq_c \boldsymbol{b} &\iff \boldsymbol{a}_{[\![1,m_1]\!]} \succcurlyeq \boldsymbol{b}_{[\![1,m_1]\!]} \text{ and } \boldsymbol{a} \in \mathcal{A} \quad \text{or} \quad \boldsymbol{b} \in \mathcal{F} \\ \boldsymbol{a} \sim_c \boldsymbol{b} &\iff \boldsymbol{a} \not\succ_c \boldsymbol{b} \text{ and } \boldsymbol{b} \not\succ_c \boldsymbol{a} \end{aligned}$$

where the admissible set $\mathcal{A}$ contains the points satisfying all constraints while any point of the failure set $\mathcal{F}$ disrespects at least one constraint. Note that in the deterministic 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:

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

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:

$$\mathcal{A}_\mathcal{B} = \big\{\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \in \wp(\mathbb{R}^m) \mid \forall j \in [\![m_1+1, m]\!],\ \pm a_j + r_j \leq 0\big\},$$
$$\mathcal{F}_\mathcal{B} = \big\{\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \in \wp(\mathbb{R}^m) \mid \exists j \in [\![m_1+1, m]\!],\ \pm a_j - r_j > 0\big\}.$$

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

$$\begin{aligned}
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{\mathcal{B}}{\succ_c} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \iff & \mathcal{B}(\boldsymbol{a},\boldsymbol{r})_{[\![1,m_1]\!]} \underset{\mathcal{B}}{\succ} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}')_{[\![1,m_1]\!]} \text{ and } \mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \in \mathcal{A}_\mathcal{B} \quad \text{or} \\
& \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \in \mathcal{F}_\mathcal{B}, \\
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{\mathcal{B}}{\succcurlyeq_c} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \iff & \mathcal{B}(\boldsymbol{a},\boldsymbol{r})_{[\![1,m_1]\!]} \underset{\mathcal{B}}{\succcurlyeq} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}')_{[\![1,m_1]\!]} \text{ and } \mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \in \mathcal{A}_\mathcal{B} \quad \text{or} \\
& \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \in \mathcal{F}_\mathcal{B}, \\
\mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{\mathcal{B}}{\sim_c} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \iff & \mathcal{B}(\boldsymbol{a},\boldsymbol{r}) \underset{\mathcal{B}}{\not\succ_c} \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \text{ and } \mathcal{B}(\boldsymbol{b},\boldsymbol{r}') \underset{\mathcal{B}}{\not\succ_c} \mathcal{B}(\boldsymbol{a},\boldsymbol{r}),
\end{aligned}$$

with the notation $\mathcal{B}(\boldsymbol{a},\boldsymbol{r})_{[\![1,m_1]\!]} = \mathcal{B}(\boldsymbol{a}_{[\![1,m_1]\!]}, \boldsymbol{r}_{[\![1,m_1]\!]})$.

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. 3 with a constrained mono-objective minimisation. Here, $\mathcal{B}_1 \in \mathcal{F}_\mathcal{B}$, $\mathcal{B}_2 \in \mathcal{F}_\mathcal{B}^\complement \cap \mathcal{A}_\mathcal{B}^\complement$, $\mathcal{B}_3 \in \mathcal{A}_\mathcal{B}$ and $\mathcal{B}_4 \in \mathcal{A}_\mathcal{B}$. The dominance are then as follows: $\mathcal{B}_1 \underset{\mathcal{B}}{\prec\prec_c} \mathcal{B}_2$, $\mathcal{B}_1 \underset{\mathcal{B}}{\prec\prec_c} \mathcal{B}_3$, $\mathcal{B}_1 \underset{\mathcal{B}}{\prec\prec_c} \mathcal{B}_4$, $\mathcal{B}_2 \underset{\mathcal{B}}{\sim_c} \mathcal{B}_3$, $\mathcal{B}_2 \underset{\mathcal{B}}{\sim_c} \mathcal{B}_4$, $\mathcal{B}_3 \underset{\mathcal{B}}{\succcurlyeq_c} \mathcal{B}_4$. Only $\mathcal{B}_2$ and $\mathcal{B}_3$ are non-dominated. $\mathcal{B}_1$ is dominated by being entirely in the failure set and $\mathcal{B}_4$ is dominated by $\mathcal{B}_3 \in \mathcal{A}_\mathcal{B}$ in the classical *Boxed Pareto Dominance* sense in 1D.

3.2 Pareto Optimal Probability

In the above, dominance rules allow to discriminate dominated boxes from non-dominated ones. This rule is exploited in [31, 32] 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 $\big\{\mathcal{B}_i\big\}_i$, the exact POP computation follows the formula below:

$$POP_{true}(\mathcal{B}_i) = \mathbb{P}_{\{\boldsymbol{Z}_k\}_k}\left[\bigcap_{\substack{\mathcal{B}_j \in \boldsymbol{\mathcal{B}}_{all} \\ j \neq i}} \boldsymbol{Z}_j \not\succ_c \boldsymbol{Z}_i\right], \tag{1}$$

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, the returned score is sometimes not consistent in the presence of clusters of boxes in the objective

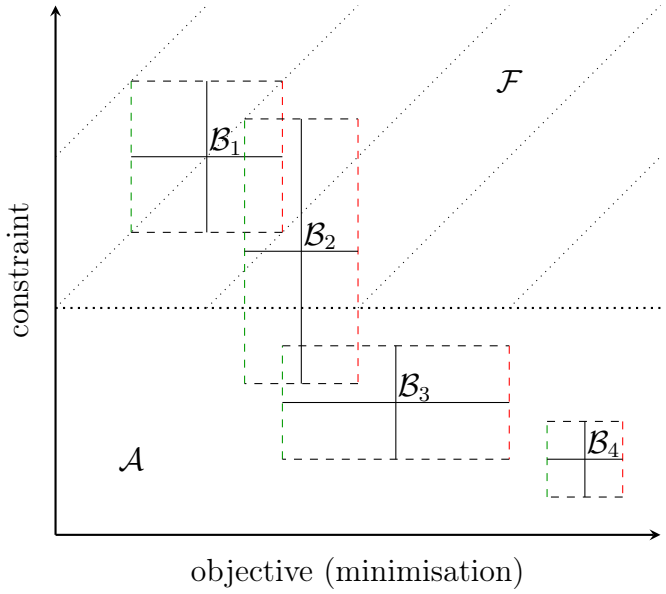


Figure 3: *Constrained Boxed Pareto Dominance*: Comparison of 4 boxes, best and worst outcomes in green and red, respectively.

and constraint space. For this reason, we propose two approximations of the POP, that will be quantitatively compared.

Ref. [29] 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. Such an idea is used here, replacing the domination probability with the POP, as follows:

$$POP_{av}(\mathcal{B}_i) = \frac{1}{N-1} \sum_{\substack{\mathcal{B}_j \in \boldsymbol{\mathcal{B}}_{all} \\ j \neq i}} \mathbb{P}_{\boldsymbol{Z}_j, \boldsymbol{Z}_i}\left[\boldsymbol{Z}_j \nprec_c \boldsymbol{Z}_i\right], \tag{2}$$

with the same definition of $\boldsymbol{Z}_k$ as above and where N is the total number of boxes in $\boldsymbol{\mathcal{B}}_{all}$.

The explicit details for computing the above expression are given in A. Since we compute each probability between only two boxes, the computational burden is very low. However, clustered boxes still yield inconsistent scores, as will be shown in the following.

A second metric, denoted as POP_{min}, is defined as follows, with $\boldsymbol{\mathcal{B}}_{nd}$ the set of non-dominated designs (using the *Constrained Boxed Pareto dominance*):

$$POP_{min}(\mathcal{B}_i) = \min_{\substack{\mathcal{B}_j \in \boldsymbol{\mathcal{B}}_{nd} \\ j \neq i}} \left(\mathbb{P}_{\boldsymbol{Z}_j, \boldsymbol{Z}_i}\left[\boldsymbol{Z}_j \nprec_c \boldsymbol{Z}_i\right]\right). \tag{3}$$

This approximation, yields a good relative ranking between boxes and is efficiently computable. It also shows robust behaviour when dealing with clustered boxes.

These three metrics *i.e.* POP_{true}, POP_{av} and POP_{min} are compared on three examples to illustrate the challenges associated with the POP computation. Each example is constituted of a given set of boxes and an optimisation problem: (1) a bi-minimization problem with highly clustered boxes (Figures 4), (2) a bi-minimization problem (Figure 5 (a)) and (3) a constrained

mono-objective minimization problem (Figure 5 (b)). The computed values of POP are reported in Table 1.

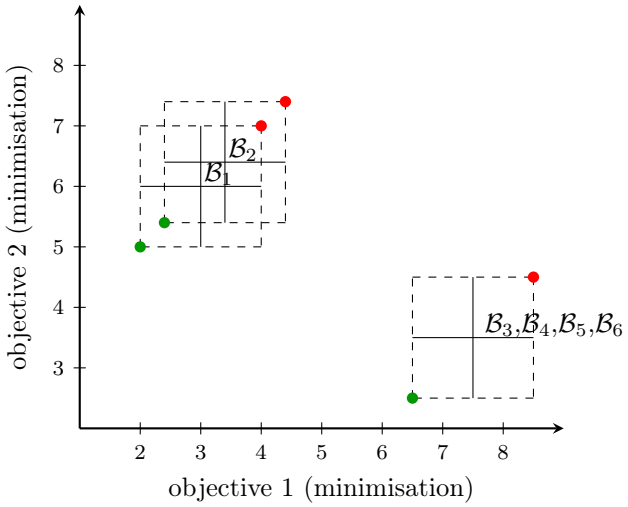


Figure 4: Example 1, with boxes $\mathcal{B}_3$ to $\mathcal{B}_6$ superimposed. Best and worst outcomes in green and red respectively. Associated POPs in Table 1.

Figure 4 illustrates the example 1 with clustered boxes ($\mathcal{B}_3$ to $\mathcal{B}_6$). Geometrically, all these boxes are equally close to the Pareto front, whereas box B is sensibly less efficient than $\mathcal{B}_1$, hence further from the Pareto front. However, one can see in Table 1 that POP_{true} and POP_{av} yield higher values of POP for box $\mathcal{B}_2$ than for boxes $\mathcal{B}_3$ to $\mathcal{B}_6$. POP_{av} and POP_{min} are easily computable, but the former can lead to unintuitive probabilities for strictly dominated boxes. This is notably the case for box $\mathcal{B}_6$ in Example 2 and box $\mathcal{B}_4$ in Example 3, as shown again in Table 1.

	Example 1			Example 2			Example 3		
	POP_{true}	POP_{av}	POP_{min}	POP_{true}	POP_{av}	POP_{min}	POP_{true}	POP_{av}	POP_{min}
$\mathcal{B}_1$	0.898	0.98	0.898	1.0	1.0	1.0	0.0	0.0	0.0
$\mathcal{B}_2$	0.538	0.908	0.538	0.902	0.98	0.912	0.254	0.279	0.254
$\mathcal{B}_3$	0.521	0.85	0.75	0.39	0.828	0.419	0.724	0.943	0.746
$\mathcal{B}_4$	0.521	0.85	0.75	0.621	0.924	0.622	0.0	0.628	0.0
$\mathcal{B}_5$	0.521	0.85	0.75	0.017	0.508	0.07	0.022	0.483	0.031
$\mathcal{B}_6$	0.521	0.85	0.75	0.0	0.735	0.0	0.0	0.0	0.0

Table 1: POPs comparison on examples 1 to 3 (Figs. 6 and 7)

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 allocating the fidelity $l(\boldsymbol{x})$ for computing the robustness and reliability measures $\tilde{\boldsymbol{\rho}}^l$

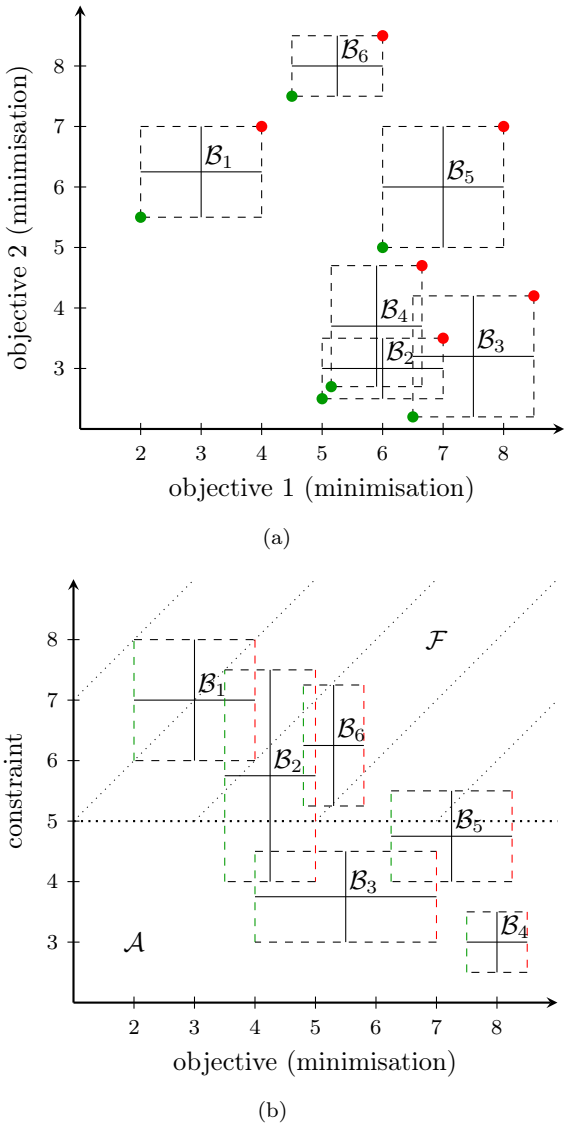


Figure 5: (a) Example 2 with a bi-minimization problem ; (b) Example 3 with a constrained mono-objective minimization. Best and worst outcomes in green and red respectively. Associated POPs in Table 1 .

associated to a design $\boldsymbol{x}$. The unconstrained version of this framework is developed in [32] and relies on the coupling between the Bounding-Box approach presented above and a Surrogate-Assisting (SA) model built on the objectives in order to bypass function evaluations, notably toward the end of the optimisation. SABBa can be coupled to any optimisation algorithm and surrogate model for the SA strategy. This feature allows the approach to be very readily applicable and to benefit from new optimisation or metamodelling techniques. In this paper, the chosen optimisation algorithm is NOMAD [33] for its reliable management of multiple objectives and constraints within a derivative-free framework (version 3.6.2). All surrogate models are

handled with Gaussian Processes (using the python Gaussian Process package GPy [34] version 1.8.5).

In the following, we provide the global algorithm and the necessary numerical ingredients for the implementation. Specifically, Section 4.1 presents extensively the structure and algorithm of the framework, 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 6. The optimiser manages the design space exploration with the aim of computing an accurate approximation of the Pareto front. We illustrate now the typical sequence of operations within an iteration of the algorithm. Let us assume to have evaluated so far $\left\{\boldsymbol{x}_i, \boldsymbol{\rho}^l(\boldsymbol{x}_i)\right\}_i$ for all the designs belonging to a set called $\mathcal{X}_c$ (which includes each design evaluated by means of the Bounding-Box approach) and that the SA model $\boldsymbol{\rho}^t_{SA}$ and associate error $\overline{\boldsymbol{\varepsilon}}^t_{SA}$ have been computed on these estimations. The algorithm features the following operation for each optimisation iteration, *i.e.* for a given design $\boldsymbol{x}$:

- If the SA model yields accurate prediction on this design, meaning $\overline{\boldsymbol{\varepsilon}}^t_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1$, the robustness and reliability measures returned by the surrogate $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ are given to the optimiser;

- Else, the measures must be computed and the design $\boldsymbol{x}$ is added to the set $\mathcal{X}_c$. To this extent, $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi})$ are computed for some values of $\boldsymbol{\xi}$ in order to build the surrogate models $\widehat{\boldsymbol{f}}(\boldsymbol{x}, \cdot)$ and $\widehat{\boldsymbol{g}}(\boldsymbol{x}, \cdot)$. This allows to estimate empirically, with a fidelity l mainly determined by the number of $\boldsymbol{\xi}$ samples, the robustness and reliability measures $\boldsymbol{\rho}^l(\boldsymbol{x}) = \left(\boldsymbol{\rho}^l_{\boldsymbol{f}}(\boldsymbol{x}), \boldsymbol{\rho}^l_{\boldsymbol{g}}(\boldsymbol{x})\right)$ at very low cost. The Gaussian Process surrogate models also permit to compute an approximation $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})$ of the associated error. The fidelity l at $\boldsymbol{x}$ can then be improved by increasing the size of the training data $\left\{\boldsymbol{\xi}_i, \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}_i), \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\xi}_i)\right\}$ on which the surrogates $\widehat{\boldsymbol{f}}(\boldsymbol{x}, \cdot)$ and $\widehat{\boldsymbol{g}}(\boldsymbol{x}, \cdot)$ are built.

In the case of designs estimated simultaneously, 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, which is the probability of being not dominated. This refinement is performed until each box is either dominated, or has reached an estimated error $\overline{\boldsymbol{\varepsilon}}^l$ below a user-defined threshold $\boldsymbol{s}_2$. POP computation is presented in section 3.2.

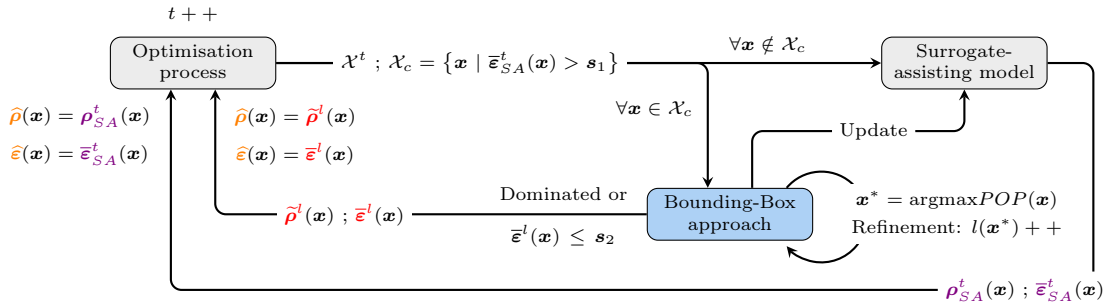


Figure 6: Structure of the SABBa framework

The strategy presented above and pictured in Figure 6 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 $\boldsymbol{s}_2$
$t = 0$
Initialise $\mathcal{X}_c$ empty
while Optimisation running **do**
  $t++$
  Read new designs $\mathcal{X}^t$
  for each $\boldsymbol{x} \in \mathcal{X}^t$ **do**
    if $\overline{\boldsymbol{\varepsilon}}^t_{SA}(\boldsymbol{x}) \leq \boldsymbol{s}_1$ **then**
      Return $\boldsymbol{\rho}^t_{SA}(\boldsymbol{x})$ to the optimiser
    else
      Add $\boldsymbol{x}$ in $\mathcal{X}_c$
      Compute a first approximation $\widetilde{\boldsymbol{\rho}}^0(\boldsymbol{x})$ of $\boldsymbol{\rho}$
    end if
  end for
  while $\mathcal{X}_r = \left\{\boldsymbol{x} \in \mathcal{X}_{\widetilde{\mathcal{P}}_{\mathcal{B}}} \cap \mathcal{X}_c \mid \overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x}) > \boldsymbol{s}_2\right\}$ is non-empty **do**
    Find $\boldsymbol{x}^* = \underset{\boldsymbol{x} \in \mathcal{X}_r}{\operatorname{argmax}} POP(\boldsymbol{x})$
    $l(\boldsymbol{x}^*)++$
    Compute $\boldsymbol{f}(\boldsymbol{x}^*, \boldsymbol{\xi})$ and $\boldsymbol{g}(\boldsymbol{x}^*, \boldsymbol{\xi})$ at some $\boldsymbol{\xi}$
    Update $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x}^*)$ and $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x}^*)$
  end while
  Update $\boldsymbol{\rho}^t_{SA}$ and $\overline{\boldsymbol{\varepsilon}}^t_{SA}$ with the new $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})$ values
  Return $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ 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 $\widetilde{\boldsymbol{\rho}}^0$ 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. 6 and Alg. 1, they can be computed with tunable fidelity using the Bounding-Box approach or can be computed on 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 $\mathbb{P}\big[\overline{\boldsymbol{\varepsilon}}(\boldsymbol{z}) < |\boldsymbol{\varepsilon}(\boldsymbol{z})|\big] < 0.3\%$.

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 $\boldsymbol{f}$ and $\boldsymbol{g}$ in the uncertain space. This corresponds to lines 9 to 15 in Algorithm 1. Gaussian Processes are used to approximate $\boldsymbol{f}(\boldsymbol{x},\cdot)$ and $\boldsymbol{g}(\boldsymbol{x},\cdot)$. Two approaches are considered:

- *Separated Spaces:* A GP is built in Ξ at a given $\boldsymbol{x}$ using a set of samples $\big\{\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\xi}_i), \boldsymbol{g}(\boldsymbol{x},\boldsymbol{\xi}_i)\big\}_i$. This directly corresponds to $\widehat{\boldsymbol{f}}(\boldsymbol{x},\cdot)$ and $\widehat{\boldsymbol{g}}(\boldsymbol{x},\cdot)$.
- *Coupled Space:* A GP is built in $\mathcal{X}\times\Xi$ using a set of samples $\big\{\boldsymbol{f}(\boldsymbol{x}_i,\boldsymbol{\xi}_i), \boldsymbol{g}(\boldsymbol{x}_i,\boldsymbol{\xi}_i)\big\}_i$. This gives approximations $\widehat{\boldsymbol{f}}$ and $\widehat{\boldsymbol{g}}$ of $\boldsymbol{f}$ and $\boldsymbol{g}$. At a given $\boldsymbol{x}$, the cuts $\widehat{\boldsymbol{f}}(\boldsymbol{x},\cdot)$ and $\widehat{\boldsymbol{g}}(\boldsymbol{x},\cdot)$ can be returned.

Remark 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.

Both SS and CS approaches allow having predictive models $\widehat{\boldsymbol{f}}(\boldsymbol{x},\cdot)$ and $\widehat{\boldsymbol{g}}(\boldsymbol{x},\cdot)$, that will be written $\widehat{\boldsymbol{f}}_{\boldsymbol{x}}$ and $\widehat{\boldsymbol{g}}_{\boldsymbol{x}}$, and their predictive variance $\boldsymbol{\sigma}_{\boldsymbol{f}}(\boldsymbol{x},\cdot)$ and $\boldsymbol{\sigma}_{\boldsymbol{g}}(\boldsymbol{x},\cdot)$. The fidelity l of these models, used to compute $\widetilde{\boldsymbol{\rho}}^l(\boldsymbol{x})$ and $\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x})$, refers to the number of training data. This level of fidelity is implicit in the following equations. The $\pm 3\sigma$ paradigm provides the box widths, denoted as $\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{f}_{\boldsymbol{x}}} = 3\boldsymbol{\sigma}_{\boldsymbol{f}}(\boldsymbol{x},\cdot)$ and $\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{g}_{\boldsymbol{x}}} = 3\boldsymbol{\sigma}_{\boldsymbol{g}}(\boldsymbol{x},\cdot)$. For each output measure, indexed by k, the classical empirical estimators are exploited for the computation of $\widehat{\rho}_k(\boldsymbol{x}) = \widetilde{\rho}^l_k(\boldsymbol{x})$ and the associated width $\widehat{\varepsilon}_k(\boldsymbol{x})$ is given in Eq. (4) for $\boldsymbol{f}_{\boldsymbol{x}}$ (the same can be written for $\boldsymbol{g}_{\boldsymbol{x}}$):

$$\widehat{\varepsilon}_k(\boldsymbol{x}) = \overline{\varepsilon}^l_k(\boldsymbol{x}) = \begin{cases} \mathbb{E}_{\boldsymbol{\xi}}\big[\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})\big] & \text{for expectation} \\ \mathbb{E}_{\boldsymbol{\xi}}\big[\big(\overline{\boldsymbol{\varepsilon}}_{\mu}(\boldsymbol{x}) + \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})\big)^2 + 2\big|\widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{\mu}}(\boldsymbol{x})\big|\big(\overline{\boldsymbol{\varepsilon}}_{\mu}(\boldsymbol{x}) + \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})\big)\big] & \text{for variance} \\ \max\big(\big|\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|, \big|\widetilde{\boldsymbol{m}}(\boldsymbol{x}) - \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}^+_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|\big) & \text{for minimum} \\ \max\big(\big|\widetilde{\boldsymbol{M}}(\boldsymbol{x}) - \max_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|, \big|\widetilde{\boldsymbol{M}}(\boldsymbol{x}) - \max_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}^+_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|\big) & \text{for maximum} \\ \max\big(\big|\widetilde{\mathbf{q}^p}(\boldsymbol{x}) - \mathrm{q}^p_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}^-_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|, \big|\widetilde{\mathbf{q}^p}(\boldsymbol{x}) - \mathrm{q}^p_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}^+_{\boldsymbol{x}}(\boldsymbol{\xi})]\big|\big) & \text{for quantile} \end{cases} \tag{4}$$

where $\widehat{\boldsymbol{f}}^+_{\boldsymbol{x}}(\boldsymbol{\xi}) = \widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})$ and $\widehat{\boldsymbol{f}}^-_{\boldsymbol{x}}(\boldsymbol{\xi}) = \widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})$. In practice, the 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 B.

We also provide guidance for refining the surrogate model $\widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi})$. While one could rely on a space-filling paradigm, geometrically filling the widest hole in the sampling, we choose here to use GP-based refinement criteria. A partial criterion is first computed for each statistical measure, with the following formulas:

$$c_k(\boldsymbol{\xi}) = \begin{cases} \overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi}) p(\boldsymbol{\xi}) & \text{for expectation} \\ \overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi}) p(\boldsymbol{\xi}) & \text{for variance} \\ \left[\dfrac{\widetilde{\boldsymbol{m}} - \widehat{\boldsymbol{f}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi})}{2\overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})} \right]_+ & \text{for minimum} \\ \left[\dfrac{\widehat{\boldsymbol{f}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi}) - \widetilde{\boldsymbol{M}}}{2\overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})} \right]_+ & \text{for maximum} \\ \left[\dfrac{\widetilde{\mathbf{q}^p} - \widehat{\boldsymbol{f}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi})}{2\overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})} \right]_+ \left[\dfrac{\widehat{\boldsymbol{f}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi}) - \widetilde{\mathbf{q}^p}}{2\overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})} \right]_+ p(\boldsymbol{\xi}) & \text{for quantile} \end{cases} \tag{5}$$

with the same definition of $\widehat{\boldsymbol{f}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi})$ and $\widehat{\boldsymbol{f}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi})$ as before and where $\left[\cdot\right]_+ = \max(0, \cdot)$. Note that we heuristically choose here to multiply the criteria for the mean, variance and quantile measures by the input Probability Density Function (PDF) $p(\boldsymbol{\xi})$ in order to put more weight on the most likely area. Justifications for these formulas are provided in C. These partial criteria are then scalarised into the final refinement criterion through the following weighted sum:

$$c(\boldsymbol{\xi}) = \sum_{k=1}^{m} w_k \bar{c}_k(\boldsymbol{\xi}) \tag{6}$$

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:

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

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

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

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 allows performing multi-point refinement efficiently without the need of any clustering heuristic. However, this strategy makes a lot of assumptions and may not yield optimal results 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 of the SA model is directly returned to the optimiser. As stated previously, the SA model is a Gaussian Process, 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 $\mathcal{X}$. Because this model is built on the previously computed boxes and evolves throughout the optimisation steps, the SA model is referred as $\boldsymbol{\rho}^t_{SA}$, with t the optimisation step.

In practice, the available data for constructing this model are boxes of different 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 different noises can be translated into heterogeneous variances $\sigma_i^2 = \left(\frac{\overline{\boldsymbol{\varepsilon}}^l(\boldsymbol{x}_i)}{3}\right)^2$ with $\boldsymbol{x}_i$ the designs at which boxes were computed using the direct approach described in 4.2.1. This allows to make use of heteroscedastic Gaussian Processes, which naturally take into account heterogeneous noise variances, under the gaussianity assumption. The measure estimations $\widehat{\boldsymbol{\rho}}$ and associated error $\widehat{\boldsymbol{\varepsilon}}$ at a new design $\boldsymbol{x}$ are computed as follows:

$$\begin{aligned}\widehat{\boldsymbol{\rho}}(\boldsymbol{x}) &= \boldsymbol{\rho}_{SA}(\boldsymbol{x}) = \boldsymbol{k}_*^T (K+\Delta)^{-1} \widetilde{\boldsymbol{\rho}}^l \\ \widehat{\boldsymbol{\varepsilon}}(\boldsymbol{x}) &= 3\boldsymbol{\sigma}_{SA}(\boldsymbol{x}) = 3\big(k_{**} - \boldsymbol{k}_*^T (K+\Delta)^{-1} \boldsymbol{k}_*\big).\end{aligned} \tag{7}$$

The superscript t is not explicitly written as the above predictor is updated at each optimisation step. Here, $K = K\big(\{\boldsymbol{x}_i\}_i, \{\boldsymbol{x}_i\}_i\big)$ 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 [35].

4.3 Quality indicator

Assessing and comparing the performances of the method requires the computation of a quantitative quality indicator. To this extent, the Hausdorff distance is a classical choice for computing the closeness of the found optimal set to the true one. In practice, the modified Hausdorff distance proposed in [36] captures the similarities more efficiently. With Bounding-Boxes, 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 the realisations of $\mathcal{X}_{\widetilde{\mathcal{P}}}$.

The deterministic modified Hausdorff distance d'_H is given as follows:

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

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 in Figure 6. The proposed expected modified Hausdorff distance reads:

$$Q_{\mathcal{B}} = \mathbb{E}_{\mathcal{X}_{\widetilde{\mathcal{P}}}}\big[d'_H(\mathcal{X}_{\mathcal{P}}, \mathcal{X}_{\widetilde{\mathcal{P}}})\big]. \tag{8}$$

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

In the next section, the SABBa framework for optimisation under uncertainty problems is applied to analytical test-cases to assess its performance against classical approaches. 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 analytical test-case 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.

We systematically compare the performance of the proposed framework to an A Priori Meta-Model (APMM) strategy. In the latest, we build a surrogate model of the output in the *Coupled Space* (CS) before the optimisation process. A Double Loop, performing an Uncertainty Propagation at each optimisation iteration, can then be conducted on the surrogate at a very low cost. The overall computational cost of building this surrogate model is here chosen *a priori* and can be adapted to the available budget, with a substantial impact on the accuracy of the results.

We show in the first test-case the performance of the direct Double Loop approach with respect to the APMM strategy. The former performs N_{UP} evaluations for measure computation at N_{opt} designs, explored by the optimiser. It is shown to be extremely inefficient.

We compare here three variants of SABBa, denoted as follows:

- SA-SS: when using the Surrogate-Assisting strategy and Separated-Spaces models for measures computation;
- SA-CS: when using the Surrogate-Assisting strategy and a Coupled-Space model for measures computation.
- CS: when using the Coupled-Space model without the Surrogate-Assisting strategy;

Among the above 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.

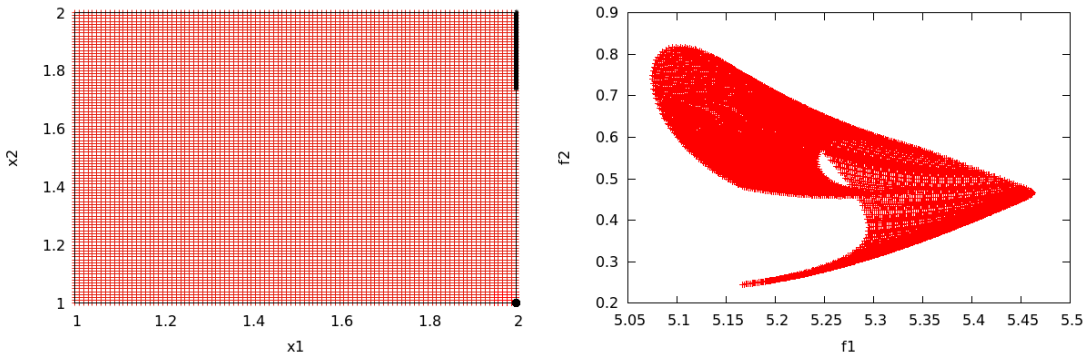
Ten runs are performed for each strategy to capture both the mean convergence curve and the associated variability, represented as a translucent band around the mean. The log distance is assumed to show a Gaussian distribution over the repeated runs, implying a log-normal distribution over the actual 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 [31]. There are two design variables $\boldsymbol{x}$ and one uncertain parameter ξ, and the problem reads:

$$\begin{aligned} \text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \begin{pmatrix} \mu(\boldsymbol{x}) \\ \sigma^2(\boldsymbol{x}) \end{pmatrix} \\ \text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_\xi[f(\boldsymbol{x},\xi)] \\ & \sigma^2(\boldsymbol{x}) = \mathbb{V}_\xi[f(\boldsymbol{x},\xi)] \\ \text{with:} \quad & f(\boldsymbol{x},\xi) = \xi - x_1\xi^5 + \cos(2\pi x_2 \xi) + 5 \\ & \xi \sim \mathcal{U}([0,1]) \\ \text{by changing:} \quad & (x_1, x_2) \in [1,2]^2 \end{aligned} \tag{9}$$

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. 7).



(a) (b)

Figure 7: 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.

The Double Loop and A Priori MetaModel strategies are quantitatively compared in Figure 8. Each Uncertainty Propagation within the Double Loop is performed up to a relative accuracy of 5%. A very slow convergence of the Double Loop approach is observed.

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 $\widehat{\boldsymbol{f}}$, the Surrogate-Assisting model $\boldsymbol{\rho}_{SA}(\boldsymbol{x})$ and the separated or Coupled-Space models $\widehat{\boldsymbol{f}}(\boldsymbol{x},\cdot)$.

5.1.1 With high-quality surrogate model

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

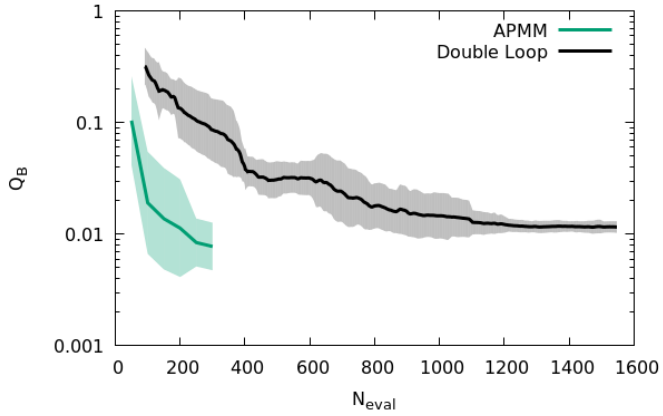


Figure 8: Test-case 1: Cost comparison between the A Priori MetaModel and Double loop approach.

matic 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 (7).

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.

The convergence curve of the indicator defined in Equation (8) is plotted in Figure 9. Figures 9(a) and 9(b) show a very reduced variability of SABBa compared to the APMM strategy. However, only SA-CS (purple curve in Fig. 9(b)) shows a significant improvement for the mean convergence.

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 9 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.

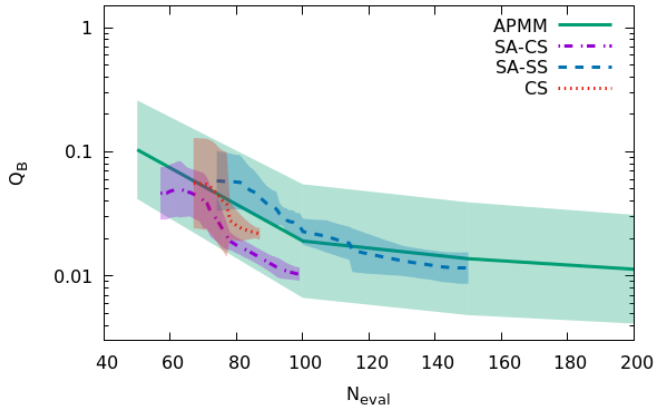
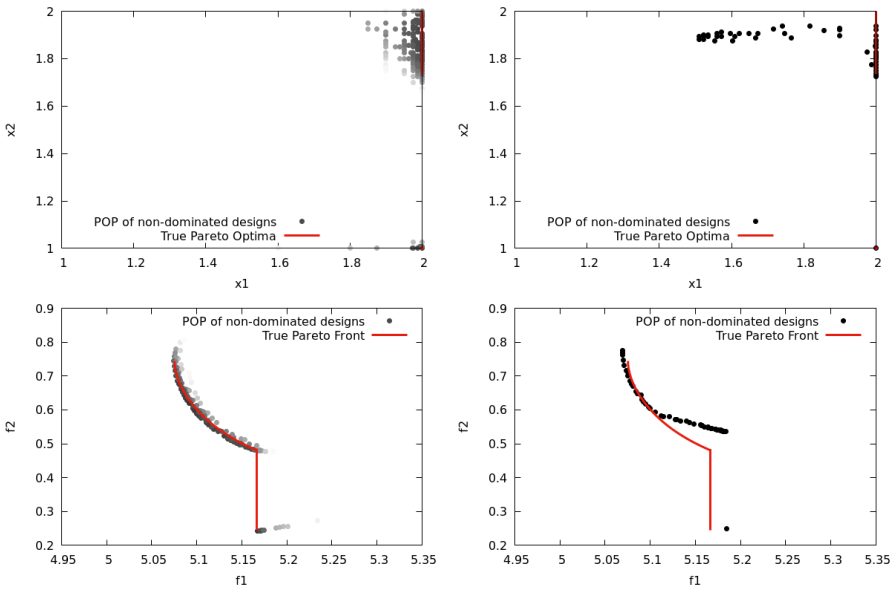


Figure 9: Test-case 1-a: Cost comparison between APMM and three SABBa variants.

We picture this risk in Figure 10, where the worst result out of ten repeated runs of SA-CS and the APMM strategy are plotted. These outputs correspond to $N_{eval} \approx 100$.

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).



(a) (b)

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

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 a homogeneous 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 optimized. 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.

Such improvement is indeed revealed in Figure 11. 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.

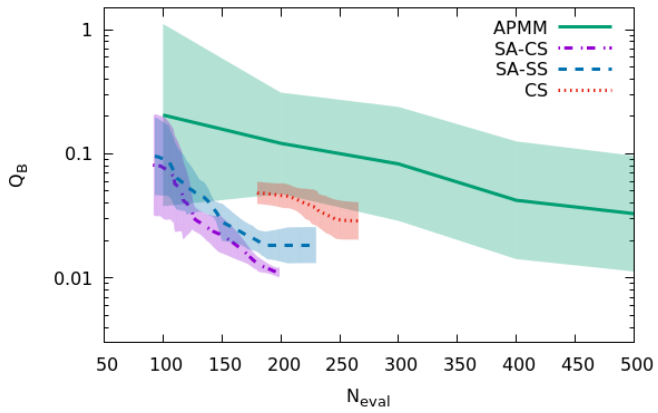
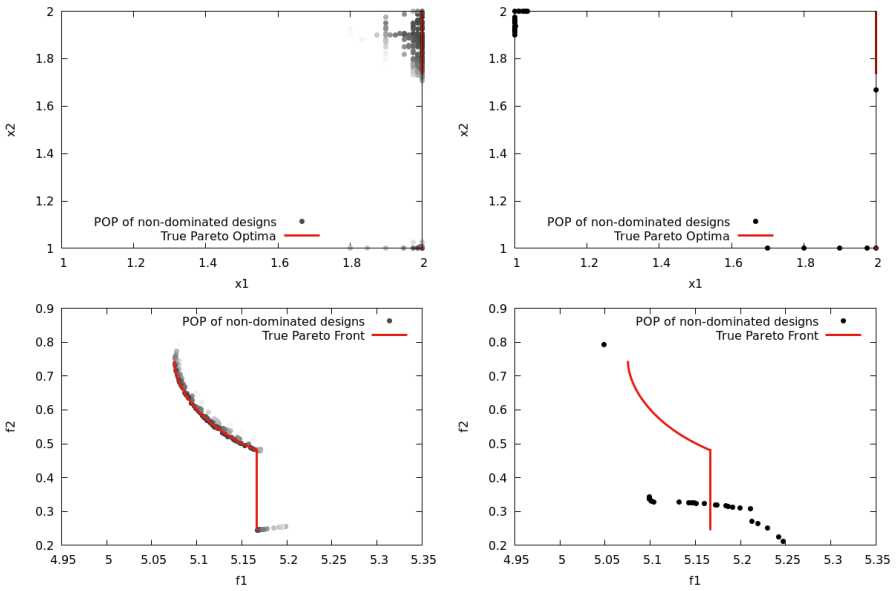


Figure 11: Test-case 1-b: Cost comparison between APMM and three SABBa variants.

As previously, the worst optimisation outputs are plotted for SABBa SA-CS and the APMM approach. In Figure 12, SABBa shows much higher consistency and accuracy compared to APMM.

The use of coupled space surrogate models and Surrogate-Assisting strategy have shown to bring an 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.



(a) (b)

Figure 12: Worst optimisation results: (a) SABBa CS-REU-G, 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:

$$
\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \mu(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = \mathrm{q}^{95\%}(\boldsymbol{x}) \leq 1 \\
\text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[f(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \mathrm{q}^{95\%}(\boldsymbol{x}) = \mathrm{q}^{95\%}_{\boldsymbol{\xi}}[g(\boldsymbol{x}, \boldsymbol{\xi})] \\
\text{with:} \quad & f(\boldsymbol{x}, \boldsymbol{\xi}) = \sum_{i=1}^{3} \left[(1 - x_i) + 3 \left(1 + \frac{\arctan\big(5(\xi_i - 0.5)\big)}{2} \right) (x_{i+1} - x_i^2)^2 \right] \\
& g(\boldsymbol{x}, \boldsymbol{\xi}) = \left(4 - 2.1 x_1^2 + \frac{x_1^4}{3} \right) x_1^2 + x_1 x_2 + \big(-4 + 4 x_2^2 \big) x_2^2 \\
& \qquad + \frac{\cos(2\pi\xi_1) - \sin(\frac{\pi}{2}\xi_1) - \xi_1 - \big(\cos(2\pi 0.05) - \sin(\frac{\pi}{2} 0.05) - 0.05\big)}{5} \\
& \boldsymbol{\xi} \sim \mathcal{U}([0,1]^3) \\
\text{by changing:} \quad & \boldsymbol{x} \in [-0.2, 1.2]^4
\end{aligned}
\tag{10}
$$

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:

$$\mu(\boldsymbol{x}) = \sum_{i=1}^{3} \left[(1 - x_i) + 3\big(x_{i+1} - x_i^2\big)^2 \right],$$
$$\mathrm{q}^{95\%}(\boldsymbol{x}) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + \big(-4 + 4x_2^2\big)x_2^2. \tag{11}$$

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 13 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.

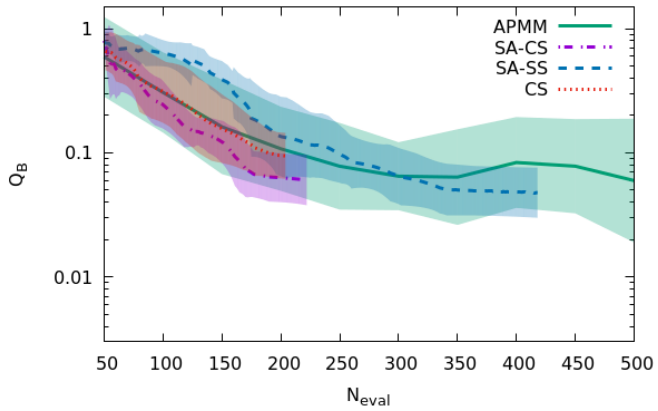


Figure 13: Test-case 2: Cost comparison between APMM and three SABBa variants.

As done before, we plot the worst optimisation outputs in Figures 14. 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.

6 Physical applications

6.1 Two-bar truss

The two-bar truss optimization problem is notably illustrated in Refs. [37] and [20]. The uncertainty-based optimization problem is formulated as follows (with a schematic representation in Figure 15(a)):

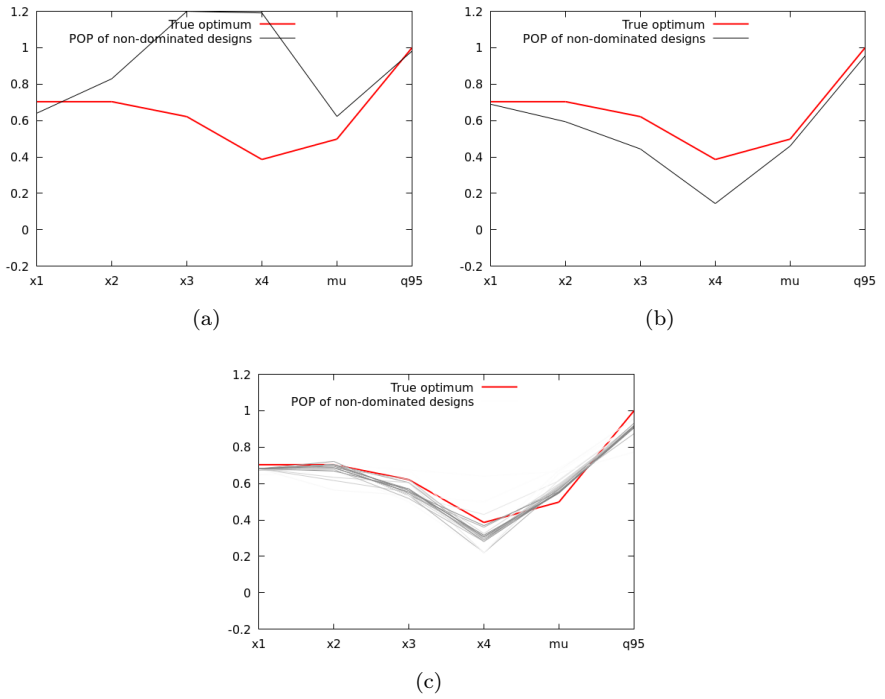


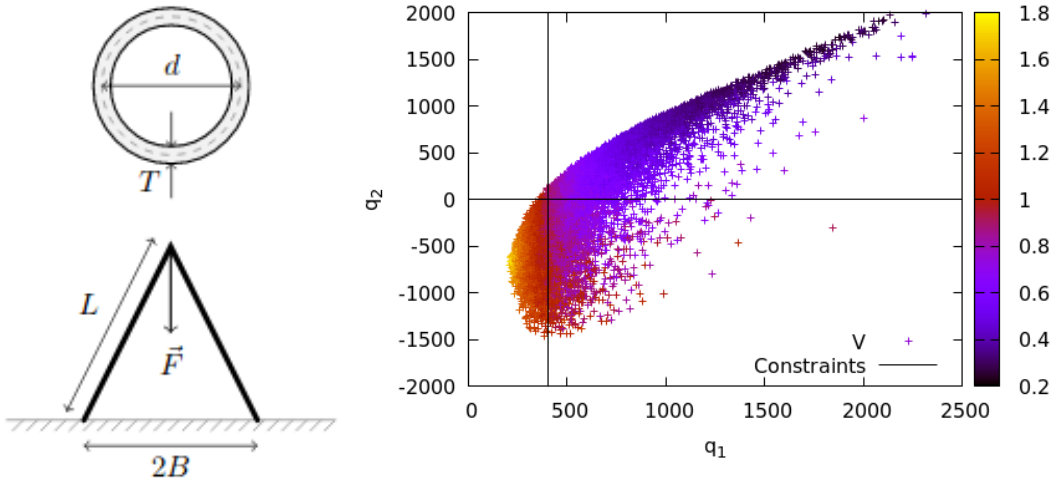
Figure 14: Worst optimisation results: (a) APMM, 150 evaluations, $Q_{\mathcal{B}} = 1.0 \times 10^{0}$. (b) SABBa CS-REU-G, 161 evaluations, $Q_{\mathcal{B}} = 1.12 \times 10^{-1}$. (c) APMM, 200 evaluations, $Q_{\mathcal{B}} = 3.24 \times 10^{-1}$.

$$
\begin{aligned}
\text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = V(\boldsymbol{x}) \\
\text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = \begin{pmatrix} \mathrm{q}_1(\boldsymbol{x}) \\ \mathrm{q}_2(\boldsymbol{x}) \end{pmatrix} \leq \begin{pmatrix} s_{\max} \\ 0 \end{pmatrix} \\
\text{where:} \quad & \mathrm{q}_1(\boldsymbol{x}) = \mathrm{q}^{0.999}[s(\boldsymbol{x}, \boldsymbol{\xi})] \\
& \mathrm{q}_2(\boldsymbol{x}) = \mathrm{q}^{0.999}[s(\boldsymbol{x}, \boldsymbol{\xi}) - s_{crit}(\boldsymbol{x}, \boldsymbol{\xi})] \\
\text{with:} \quad & \boldsymbol{\xi} \sim \mathcal{N}\left(\begin{pmatrix} 150000 \\ 210000 \end{pmatrix}, \begin{pmatrix} 30000^2 & 0 \\ 0 & 21000^2 \end{pmatrix} \right) \\
\text{by changing:} \quad & \boldsymbol{x} \in [20, 80] \times [800, 1200] \times [700, 800] \times [2, 3]
\end{aligned} \tag{12}
$$

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. 15(b), where the objective value ($V(\boldsymbol{x})$, to be minimized)

is plotted in the constraints space with a color scale. The abscissa and ordinate directions refer to the constraints $q_1(\boldsymbol{x})$ and $q_2(\boldsymbol{x})$; the associated thresholds are drawn with red lines. The admissible set is in the lower left quadrangle ($q_1 < 400$ N.mm^{-2} and $q_2 < 0$ N.mm^{-2}) and one may note that the objective is nearly constant on the limit $q_1 = 400$ N.mm^{-2}.



(a) (b)

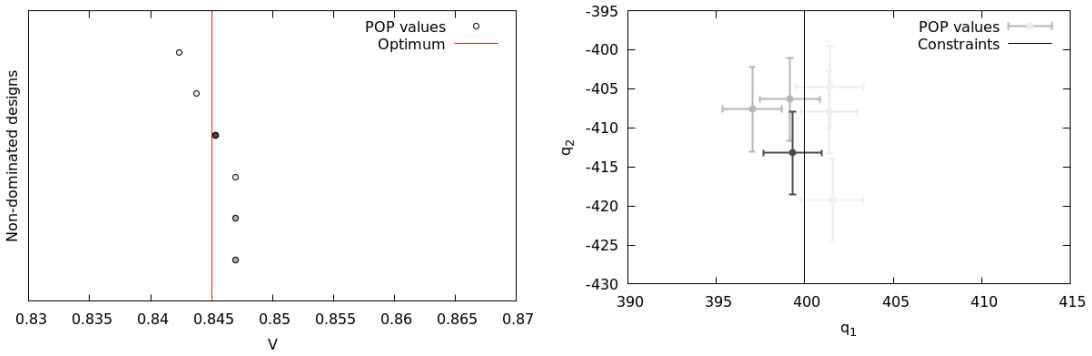
Figure 15: (a) Schematic representation of the two bars, with bar section above (from [37]). (b) Representation in the objective/constraints space, constraints in abscissa and ordinate, objective in color.

This optimisation has been solved with the SA-CS variant. The thresholds $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are sequentially reduced up to 0.5% of the total ranges. A set of 6 non-dominated designs is returned, with different Pareto Optimal Probabilities (POP). This is reported in Figure 16, where the optimal designs are drawn in the output space with POP greyscale (black: 1; white: 0). Figure 16(a) shows the deterministic objective values associated to each non-dominated design ; the ordinate only avoids overlapping points for better readability. Figure 16(b) shows the boxes in the constraints space q_1 and q_2. One can note that the boxes associated to the highest POP are located mostly in the admissible space ($q_1 \leq 400$ N.mm^{-2}). This optimisation was performed with **65** function evaluations, which is very inexpensive with respect to the accuracy of the output.

The design associated to the highest POP is $\boldsymbol{x}^* = (63.292, 1021.4, 701.25, 2.0812)$. We can verify that $P_{\boldsymbol{\xi}}[s(\boldsymbol{x}^*, \boldsymbol{\xi}) \geq s_{\max}] \simeq 1.2 \times 10^{-3}$ and $P_{\boldsymbol{\xi}}[s(\boldsymbol{x}^*, \boldsymbol{\xi}) \geq s_{crit}(\boldsymbol{x}^*, \boldsymbol{\xi})] \leq 1.0 \times 10^{-4}$. The first quantile constraint is not perfectly satisfied. This is due to (i) the associated box not being entirely in the admissible space, (ii) the final accuracy not being very sharp and (iii) the quantile computation being performed with simple Monte Carlo, hence with high variance. A dedicated technique for accurate quantile computation would be required to avoid adding estimation noise, that is not taken into account here.

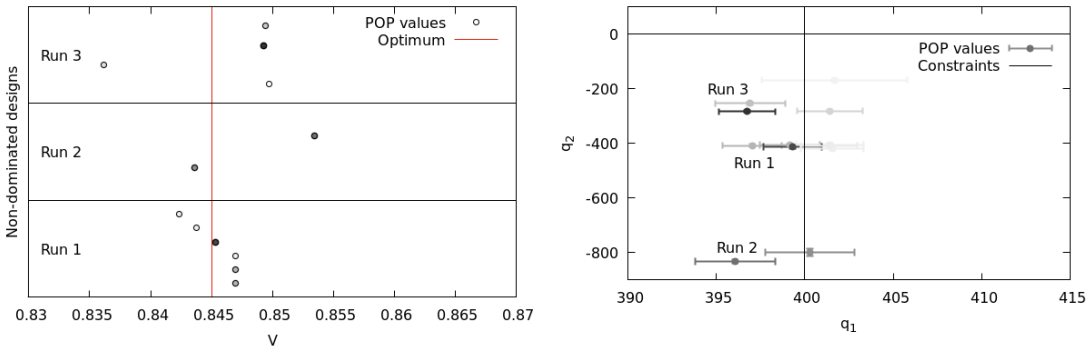
While the objective and q_1 values are well captured (optimal and close to constraint limit), the q_2 seems not to have a significant impact. To verify this, the optimisation is solved two more times. The results are plotted in Figure 17.

As expected, while similar optimal values for V and q_1 are found, the associated values of q_2



(a) (b)

Figure 16: Optimisation output with POP greyscale in (a) objective space (optimum in red) and (b) constraints space (constraint limit in black).



(a) (b)

Figure 17: Optimisation output with POP greyscale in (a) objective space (optimum in red) and (b) constraints space (constraint limit in black).

vary dramatically. Specifically, the optimal V (in L) is in the $\pm 10^{-2}$ L range around the true optimum while the total spanned range is of 1.6 L. Similarly, optimal q_1 varies of less than 10 N.mm^{-2} over the total range of more than 2500 N.mm^{-2}. However, q_2 ranges approximately from -850 N.mm^{-2} to -150 N.mm^{-2}. This is far from negligible, as it can be seen in Figure 18 and validates the observation that V seemed nearly constant on the q_1 constraint limit.

Note that in all three runs, designs associated to an objective value below the optimum have boxes that lay partially in the failure region $q_1 > 400$ N.mm^{-2}. Such an objective value cannot be reached while satisfying the constraints, which reveals that the true q_1 value are indeed over 400 N.mm^{-2}.

These two additional runs were performed with **72** and **65** function evaluations. Finer thresholds and additional evaluations would be required in order to find a unique global optimum on the $q_1 = 400$ N.mm^{-2} line. However, up to the chosen 0.5% threshold, the optimum has been

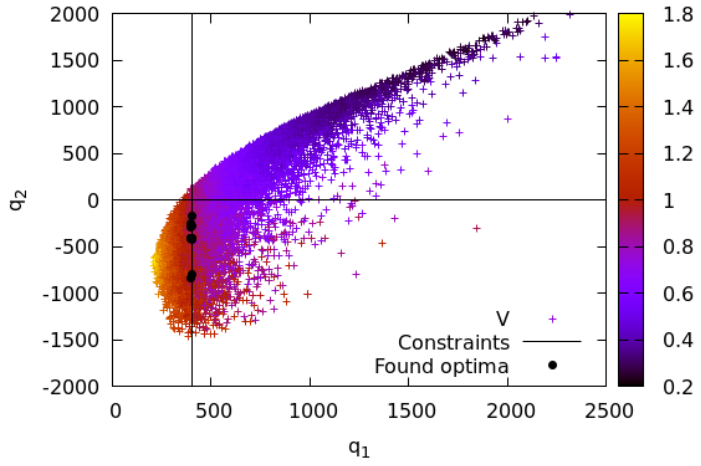


Figure 18: Found optima from the three runs (in black) in the constraints space, with objective value in color.

well captured at a very low cost.

6.2 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. 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). Its composition and properties are comparable to PICA ; 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 [38]. 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 19 illustrates the typical evolution of the temperature inside the material and at the heated surface.

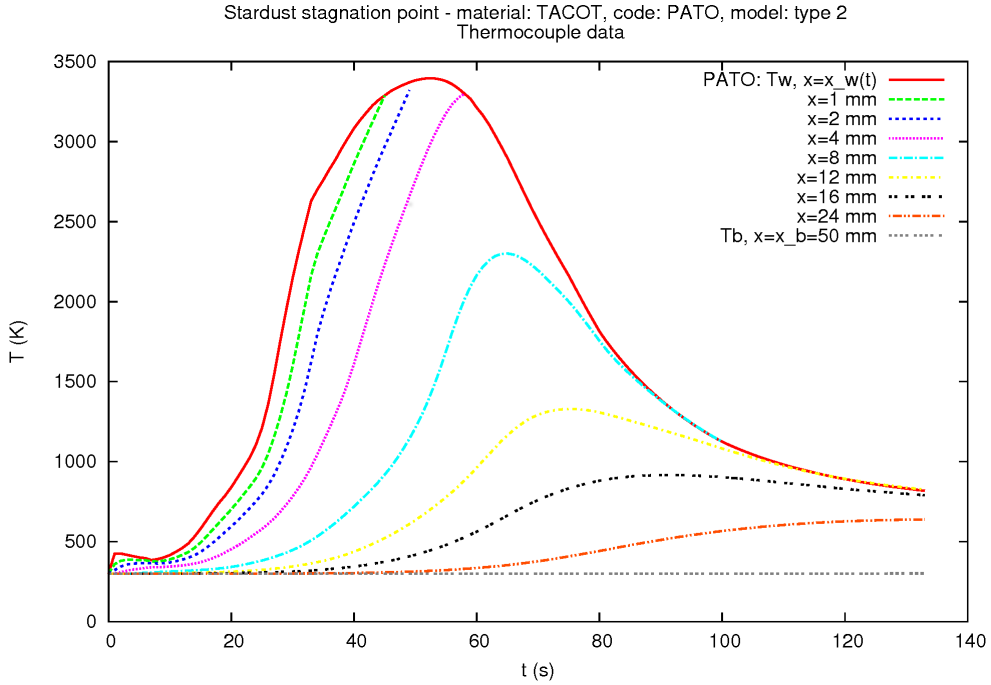


Figure 19: Surface and in-depth temperatures obtained with nominal parameters.

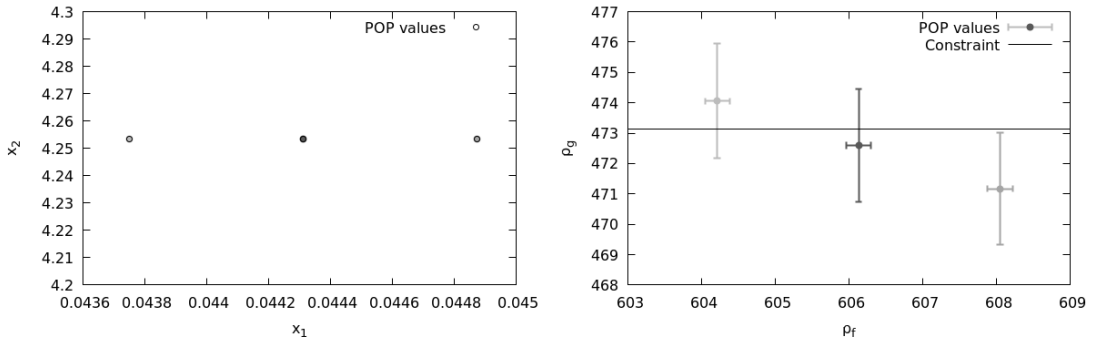
Formally, the optimisation problems reads as follows:

$$\begin{aligned} \text{minimise:} \quad & \boldsymbol{\rho_f}(\boldsymbol{x}) = \mu(\boldsymbol{x}) \\ \text{satisfying:} \quad & \boldsymbol{\rho_g}(\boldsymbol{x}) = M(\boldsymbol{x}) \leq 473.15 \\ \text{where:} \quad & \mu(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\sigma(\boldsymbol{x}, \boldsymbol{\xi})] \\ & M(\boldsymbol{x}) = \max_{\boldsymbol{\xi}}[T_b(\boldsymbol{x}, \boldsymbol{\xi})] \\ \text{with:} \quad & \boldsymbol{\xi} \sim \mathcal{U}(\Xi) \\ \text{by changing:} \quad & \boldsymbol{x} \in [0.01, 0.1] \times [3.5, 7] \end{aligned} \tag{13}$$

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 19 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.

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 [39]. Here, retained uncertain parameters are: 1) Density and volume fraction of the fibrous preform (5% uncertainty each); 2) Density and volume fraction of the phenolic resin (5% uncertainty each); 3) Thermal properties of the charred material: heat capacity, conductivity and emissivity (5% uncertainty each); 4) Oxygen fraction in the pyrolysis gases (10% uncertainty); 5) Pyrolysis reaction activation energy (10% uncertainty); 6) Overall width of the system (0.1 cm uncertainty). Both design parameters are also affected by an uncertainty (± 0.005 for x_1 and ± 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.



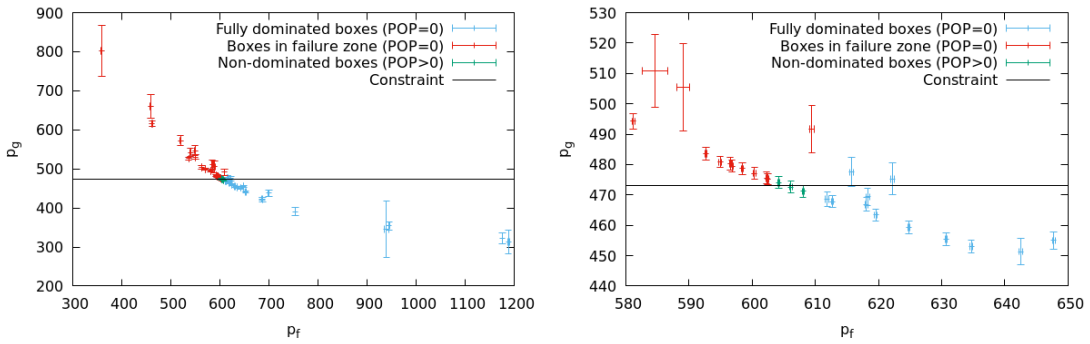
(a) (b)

Figure 20: Outputs of the SABBa framework in (a) the input space, (b) the objective/constraint space.

The three Pareto optimal designs found by SABBa are plotted in Figure 20. 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. 20(b), where $\boldsymbol{\rho_f}$ must be minimised and the admissible set is below the constraint threshold $\boldsymbol{\rho_g} = 473.15$ K. These optima were found using again the SABBa framework with Coupled-space model (SA-CS) for a computational cost of only **40** function evaluations. In Figure 21, we have reported the performances of all the designs evaluated during the optimization. They are plotted in blue when dominated, in red when entirely in failure zone and in green when non-dominated.

Among the three optimal designs depicted in Figure 20, only the right one lies entirely in the admissible set. Using the notations from Section 3.1.2, this box belongs to $\mathcal{A}_{\mathcal{B}}$. Hence, contrarily to the two other optimal designs, it can dominate other boxes. In Figure 21, 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 behavior explains why only three designs are kept, as shown in Figure 20. One can note that the tunable fidelity has been well exploited by the Bounding-Box approach in Figure 21, where extreme boxes (on the very left or very right) are estimated with very wide uncertainty.

Figure 22 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 20 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.



(a) (b)

Figure 21: 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.

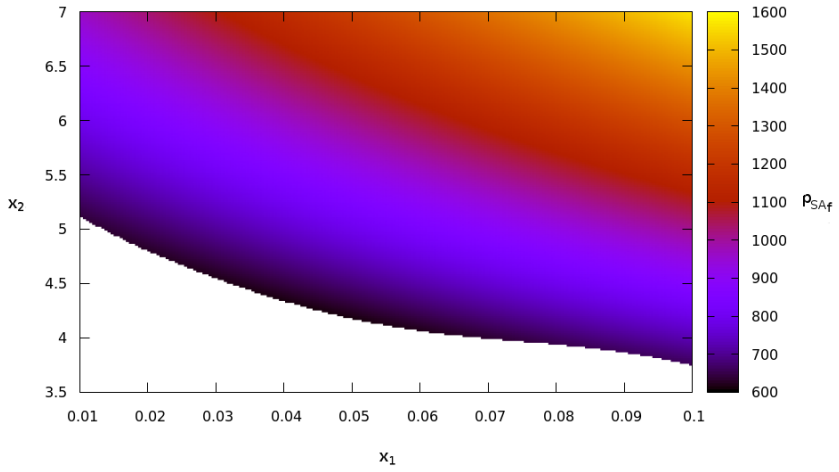


Figure 22: Final SA model, representing $\boldsymbol{\rho}_{SA_f}(\boldsymbol{x})$ where the constraint $\boldsymbol{\rho}_{SA_g}(\boldsymbol{x}) \leq 473.15$ K is satisfied.

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 measures approximations is used to compare designs in a tunable fidelity 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 on both analytical and applicative test cases. We have systematically compared SABBa to the A Priori MetaModel strategy. 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.

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 less than 70 evaluations. A 12-dimensional Thermal Protection System (TPS) design with worst-case constraints has also been performed with 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 easy coupling to any existing deterministic process.

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. Following actions could be the application of SABBa to other costly engineering cases, the inclusion of Value at Risk and Conditional Value at Risk (VaR and CVaR) measures, the definition of non-uniform boxes and the development of a specific Bayesian Optimisation technique for SABBa.

A POP computational details

The set of boxes $\boldsymbol{\mathcal{B}} = \{\mathcal{B}_i\}_i$ is given, with $\mathcal{B}_i = \mathcal{B}(\boldsymbol{a}_i, \boldsymbol{r}_i)$. Every $\boldsymbol{Z}_i \sim \mathcal{U}(\mathcal{B}_i)$ is of dimension $m = m_1 + m_2$ where m_1 is the number of objectives and m_2 the number of constraints. Hence, $\boldsymbol{Z}_{i_{[\![1,m_1]\!]}} = \boldsymbol{Z}_{i_{\boldsymbol{f}}}$ are the objectives values and $\boldsymbol{Z}_{i_{[\![m_1+1,m]\!]}} = \boldsymbol{Z}_{i_{\boldsymbol{g}}}$ the constraints values. The probability in Equation (2) is computed as follows, using the independence assumptions between boxes and between dimensions:

$$\begin{aligned}
\mathbb{P}_{\boldsymbol{Z}_j,\boldsymbol{Z}_i}\left[\boldsymbol{Z}_j \not\succ_c \boldsymbol{Z}_i\right] &= 1 - \mathbb{P}_{\boldsymbol{Z}_j,\boldsymbol{Z}_i}\left[\boldsymbol{Z}_j \succ_c \boldsymbol{Z}_i\right] \\
&= 1 - \Big(\mathbb{P}_{\boldsymbol{Z}_{i_{\boldsymbol{g}}}}\left[\boldsymbol{Z}_{i_{\boldsymbol{g}}} \not\leq \boldsymbol{0}\right] \\
&\quad + \mathbb{P}_{\boldsymbol{Z}_{i_{\boldsymbol{g}}}}\left[\boldsymbol{Z}_{i_{\boldsymbol{g}}} \leq \boldsymbol{0}\right]\mathbb{P}_{\boldsymbol{Z}_{j_{\boldsymbol{g}}}}\left[\boldsymbol{Z}_{j_{\boldsymbol{g}}} \leq \boldsymbol{0}\right]\mathbb{P}_{\boldsymbol{Z}_{j_{\boldsymbol{f}}},\boldsymbol{Z}_{i_{\boldsymbol{f}}}}\left[\boldsymbol{Z}_{j_{\boldsymbol{f}}} \succ \boldsymbol{Z}_{i_{\boldsymbol{f}}}\right]\Big) \\
&= \mathbb{P}_{\boldsymbol{Z}_{i_{\boldsymbol{g}}}}\left[\boldsymbol{Z}_{i_{\boldsymbol{g}}} \leq \boldsymbol{0}\right]\Big(1 - \mathbb{P}_{\boldsymbol{Z}_{j_{\boldsymbol{g}}}}\left[\boldsymbol{Z}_{j_{\boldsymbol{g}}} \leq \boldsymbol{0}\right]\mathbb{P}_{\boldsymbol{Z}_{j_{\boldsymbol{f}}},\boldsymbol{Z}_{i_{\boldsymbol{f}}}}\left[\boldsymbol{Z}_{j_{\boldsymbol{f}}} \succ \boldsymbol{Z}_{i_{\boldsymbol{f}}}\right]\Big)
\end{aligned} \tag{14}$$

where

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

and $\left[\,\cdot\,\right]_0^1$ 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 (14) is computed

as follows:

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

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. 23. Note that these values can be negative and that the computation is not symmetric between $\mathcal{B}_j$ and $\mathcal{B}_i$.

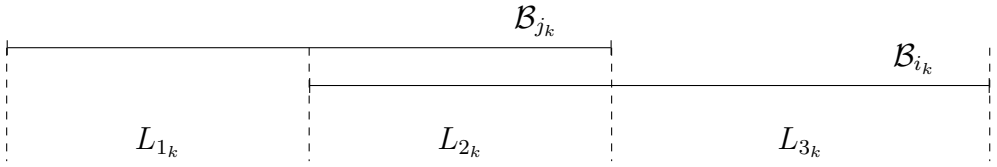


Figure 23: Computational details

B Justification for box sizes

Some developments are given here to explain the choice of the conservative error in Equation (4) associated with measure approximations.

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

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

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

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

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

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 $\boldsymbol{f}(\boldsymbol{x},\cdot) \in [\widehat{\boldsymbol{f}}_{\boldsymbol{x}} - \overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}, \widehat{\boldsymbol{f}}_{\boldsymbol{x}} + \overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}]$. The monotonicity of the expectation operator then gives $\mathbb{E}_{\boldsymbol{\xi}}[\boldsymbol{f}(\boldsymbol{x},\boldsymbol{\xi})] \in \left[\mathbb{E}_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})], \mathbb{E}_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})]\right]$, or equivalently $\boldsymbol{\mu}(\boldsymbol{x}) \in \left[\widetilde{\boldsymbol{\mu}}(\boldsymbol{x}) - \mathbb{E}_{\boldsymbol{\xi}}[\overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})], \widetilde{\boldsymbol{\mu}}(\boldsymbol{x}) + \mathbb{E}_{\boldsymbol{\xi}}[\overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})]\right]$, thus giving the conservative error found in the above $\overline{\varepsilon}_{\mu}(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\xi}}[\overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi})]$.

By defining $\widehat{\boldsymbol{f}}^{+}_{\boldsymbol{x}}(\boldsymbol{\xi}) = \widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \overline{\varepsilon}_{\boldsymbol{f}\boldsymbol{x}}(\boldsymbol{\xi})$ and $\widehat{\boldsymbol{f}}^{-}_{\boldsymbol{x}}(\boldsymbol{\xi}) = \widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \overline{\varepsilon}_{\boldsymbol{f}\boldsymbol{x}}(\boldsymbol{\xi})$, 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 $\widetilde{\boldsymbol{m}} = \min_{\boldsymbol{\xi}}[\widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi})]$,

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

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

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

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

C Justification for refinement criteria

The chosen partial criteria for GP model refinement given in Equation (5) 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:

$$\begin{aligned}
\boldsymbol{c}_{\mu}(\boldsymbol{\xi}) &= \overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi}) p(\boldsymbol{\xi}) \\
\boldsymbol{c}_{\sigma^2}(\boldsymbol{\xi}) &= \overline{\varepsilon}_{\boldsymbol{f}_{\boldsymbol{x}}}(\boldsymbol{\xi}) p(\boldsymbol{\xi})
\end{aligned}$$

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:

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

where $\widehat{\boldsymbol{f}}_{\boldsymbol{x}}^{+}(\boldsymbol{\xi}) = \widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) + \overline{\varepsilon}_{\boldsymbol{f}\boldsymbol{x}}(\boldsymbol{\xi})$ and $\widehat{\boldsymbol{f}}_{\boldsymbol{x}}^{-}(\boldsymbol{\xi}) = \widehat{\boldsymbol{f}}_{\boldsymbol{x}}(\boldsymbol{\xi}) - \overline{\varepsilon}_{\boldsymbol{f}\boldsymbol{x}}(\boldsymbol{\xi})$ 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:

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

This product is maximised on the hyperplane $\widehat{\boldsymbol{f}}_{\boldsymbol{x}} = \mathrm{q}^p$. Multiplying the criteria by $p(\boldsymbol{\xi})$ 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.

References

[1] Jae-Ohk Lee, Young-Soon Yang, and Won-Sun Ruy. A comparative study on reliability-index and target-performance-based probabilistic structural design optimization. *Computers & Structures*, 80(3):257 – 269, 2002.

[2] Qinghai Zhao, Xiaokai Chen, Zhengdong Ma, and Yi Lin. A comparison of deterministic, reliability-based topology optimization under uncertainties. *Acta Mechanica Solida Sinica*, 29(1):31 – 45, 2016.

[3] H.A. Jensen, M.A. Valdebenito, G.I. Schuëller, and D.S. Kusanovic. Reliability-based optimization of stochastic systems using line search. *Computer Methods in Applied Mechanics and Engineering*, 198(49):3915 – 3924, 2009.

[4] M.A. Valdebenito and G.I. Schuëller. Efficient strategies for reliability-based optimization involving non-linear, dynamical structures. *Computers & Structures*, 89(19):1797 – 1811, 2011. Civil-Comp.

[5] Manolis Papadrakakis and Nikos D Lagaros. Reliability-based structural optimization using neural networks and monte carlo simulation. *Computer Methods in Applied Mechanics and Engineering*, 191(32):3491 – 3507, 2002.

[6] Vahid Keshavarzzadeh, Felipe Fernandez, and Daniel A. Tortorelli. Topology optimization under uncertainty via non-intrusive polynomial chaos expansion. *Computer Methods in Applied Mechanics and Engineering*, 318:120 – 147, 2017.

[7] R. Schöbi, B. Sudret, and 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):D4016002, 2017.

[8] Julien Bect, David Ginsbourger, Ling Li, Victor Picheny, and Emmanuel Vazquez. Sequential design of computer experiments for the estimation of a probability of failure. *Statistics and Computing*, 22(3):773–793, May 2012.

[9] Jianye Ching and Wei-Chih Hsu. Approximate optimization of systems with high-dimensional uncertainties and multiple reliability constraints. *Computer Methods in Applied Mechanics and Engineering*, 198(1):52 – 71, 2008. Computational Methods in Optimization Considering Uncertainties.

[10] Jun Xu, Wangxi Zhang, and Rui Sun. Efficient reliability assessment of structural dynamic systems with unequal weighted quasi-monte carlo simulation. *Computers & Structures*, 175:37 – 51, 2016.

[11] Oded Amir, Ole Sigmund, Boyan S. Lazarov, and Mattias Schevenels. Efficient reanalysis techniques for robust topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 245-246:217 – 231, 2012.

[12] Ning Chen, Dejie Yu, Baizhan Xia, and Zhengdong Ma. Topology optimization of structures with interval random parameters. *Computer Methods in Applied Mechanics and Engineering*, 307:300 – 315, 2016.

[13] Ioannis Doltsinis, Zhan Kang, and Gengdong Cheng. Robust design of non-linear structures using optimization methods. *Computer Methods in Applied Mechanics and Engineering*, 194(12):1779 – 1795, 2005. Special Issue on Computational Methods in Stochastic Mechanics and Reliability Analysis.

[14] Zhili Tang and Jacques Périaux. Uncertainty based robust optimization method for drag minimization problems in aerodynamics. *Computer Methods in Applied Mechanics and Engineering*, 217-220:12 – 24, 2012.

[15] Alexandros A. Taflanidis and James L. Beck. An efficient framework for optimal robust stochastic system design using stochastic simulation. *Computer Methods in Applied Mechanics and Engineering*, 198 (1):88 – 101, 2008.

[16] Juan Camilo Medina and Alexandros A. Taflanidis. Adaptive importance sampling for optimization under uncertainty problems. *Computer Methods in Applied Mechanics and Engineering*, 279:133 – 162, 2014.

[17] I. Elishakoff, R.T. Haftka, and J. Fang. Structural design under bounded uncertainty—optimization with anti-optimization. *Computers & Structures*, 53(6):1401 – 1405, 1994.

[18] Stewart McWilliam. Anti-optimisation of uncertain structures using interval analysis. *Computers & Structures*, 79(4):421 – 430, 2001.

[19] J. Zhang, A.A. Taflanidis, and J.C. Medina. Sequential approximate optimization for design under uncertainty problems utilizing kriging metamodeling in augmented input space. *Computer Methods in Applied Mechanics and Engineering*, 315:369 – 395, 2017.

[20] R. Jin, X. Du, and W. Chen. The use of metamodeling techniques for optimization under uncertainty. *Structural and Multidisciplinary Optimization*, 25(2):99–116, Jul 2003.

[21] Kwon-Hee Lee and Gyung-Jin Park. A global robust optimization using kriging based approximation model. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 49(3):779–788, 2006.

[22] Gabriella Dellino, Jack P. C. Kleijnen, and Carlo Meloni. Robust optimization in simulation: Taguchi and krige combined. *INFORMS Journal on Computing*, 24(3):471–484, 2012.

[23] Michael Eldred, Anthony Giunta, Steven Wojtkiewicz, and Timothy 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, sep 2002.

[24] Janis Janusevskis and Rodolphe Le Riche. Simultaneous kriging-based estimation and optimization of mean response. *Journal of Global Optimization*, 55(2):313–336, 2013.

[25] Rodolphe Le Riche, Victor Picheny, Andre Meyer, Nam-Ho Kim, and David 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*, page 2257, 2009.

[26] Mickaël Binois, David Ginsbourger, and Olivier Roustant. Quantifying uncertainty on pareto fronts with gaussian process conditional simulations. *European journal of operational research*, 243(2):386–394, 2015.

[27] D Huang, T T Allen, W I Notz, and N Zeng. Global Optimization of Stochastic Black-Box Systems via Sequential Kriging Meta-Models. *Journal of Global Optimization*, 34(3):441–466, mar 2006.

[28] Victor Picheny, David Ginsbourger, and Yann Richet. Noisy Expected Improvement and on-line computation time allocation for the optimization of simulators with tunable fidelity. working paper or preprint, June 2010.

[29] Jürgen Teich. *Pareto-Front Exploration with Uncertain Objectives*, pages 314–328. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.

[30] Miha Mlakar, Tea Tusar, and Bogdan Filipic. Comparing solutions under uncertainty in multiobjective optimization. *Mathematical Problems in Engineering*, 2014:1–10, 2014.

[31] Francesca Fusi and Pietro Marco Congedo. An adaptive strategy on the error of the objective functions for uncertainty-based derivative-free optimization. *Journal of Computational Physics*, 309:241–266, February 2016.

[32] Mickael Rivier and Pietro Marco Congedo. Surrogate-Assisted Bounding-Box Approach for Optimization Problems with Approximated Objectives. Research Report RR-9155, Inria, February 2018.

[33] Sébastien Le Digabel. Nomad: Nonlinear optimization with the mads algorithm. 37:44, 01 2011.

[34] GPy. GPy: A gaussian process framework in python. `http://github.com/SheffieldML/GPy`, since 2012.

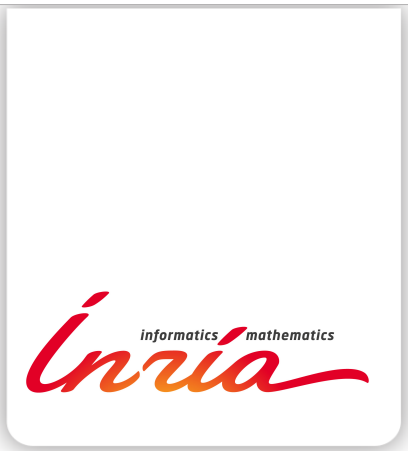
[35] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.

[36] M. P. Dubuisson and A. K. Jain. A modified hausdorff distance for object matching. In *Proceedings of 12th International Conference on Pattern Recognition*, volume 1, pages 566–568 vol.1, Oct 1994.

[37] Vincent Baudoui. *Optimisation robuste multiobjectifs par modèles de substitution*. PhD thesis, 2012. Thèse de doctorat dirigée par Hiriart-Urruty, Jean-Baptiste et Harran-Klotz, Patricia Mathématiques appliquées et énergétique et transferts Toulouse, ISAE 2012.

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

[39] M. Rivier, J. Lachaud, and P.M. Congedo. Ablative thermal protection system under uncertainties including pyrolysis gas composition. *Aerospace Science and Technology*, 84:1059 – 1069, 2019.



RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE

1 rue Honoré d'Estienne d'Orves
Bâtiment Alan Turing
Campus de l'École Polytechnique
91120 Palaiseau

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399